Parking Helper IOS Application

- Predicting the possibility of being vacant for each parking space in parking lots

By Bo'Ao Xia for CPLN 680: Advanced Topics in GIS
May 2nd, 2015

- ## Introduction:

The United States is called the country on the wheels because of the huge number of vehicles. At the same time, the highly developed highway system makes the long trips a pleasure.

But there is a serious problem coming with this pleasure, parking! It's a real problem for the drivers to find a parking space in a crowded city or a large parking lot of a plaza.

After one week-long working, many people prefer to go to a shopping mall with their families than stay at home. When they arrive at the shopping mall, they have to spend tons of time wandering in the parking lot to find a parking space.

Nowadays there are many websites and applications which can help people reserve or find a street parking space. But few of them are willing to solve the parking problem of a shopping mall because it's more complicated to study the rules of parking in a large parking lot. There are plenty of parking spaces and people can park any where they want which leads to the real problem: where do people want to park? At the same time, the opposite of this question will solve the problem: what kind of spaces are more likely to be vacant?

What people need is finding an empty parking space in real time when they are driving. So there is a clear need for this analysis and guiding application on drivers' mobile devices such as cellphones which can meet their need.

- ## Review of Existing Tools:

BestParking (Website&App: http://www.bestparking.com/):

This application is developed to help users to reserve parking spots in a specific parking place in the city. In this application user interface, the labels on the map convey the information we need. The numbers in labels are the parking cost per hour and the color of labels tells the estimated rate of cost. All the information is estimated by some algorithms.
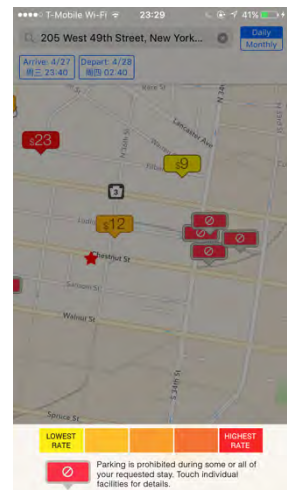
Parknav (Website&App: http://parknav.com/):

This application is developed to estimate the parking chance of each
street. The color of each street demonstrate the parking chances from
low to high. Users can easily tell from the map that which places are
more likely to have vacant parking spaces. The strength of this
application is that they built a efficient model to estimate the parking
chances but one weakness of it is the application is only available in
cities which means that users can't use it when they want to park in
suburban or rural areas.
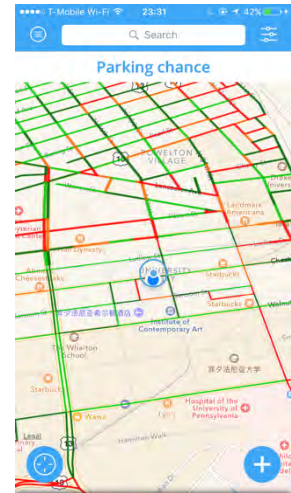
• Problem Statement:

According to the existing need of reality and those precedents, the purpose of this
project is to develop an IOS application prototype to predicting the possibility of being
vacant for each parking space in suburban parking lots or parking lots of shopping malls.
The core component of this application is to build a prediction model. With this model,
the possibility of being vacant will be estimated. For the reason that this is a prototype,
the data is analog data with some assumptions. The prediction model is supposed to be
a regression model based on some real data. But only with analog data, the parameters
in the model are simulated. Users hope that they can get all the information from the
map. The labels of parking spaces will tell them the parking chance and a route button
clicking will lead them to their destinations. The aim of this prototype application is to
be more efficient and accurate. At the same time, the users of this application will also
provide some information for the server and database to complete the prediction
analysis.

Project Setup:

Target Deployment Vision	iOS 9.3
Developing Toolkit	ArcGIS SDK API for iOS

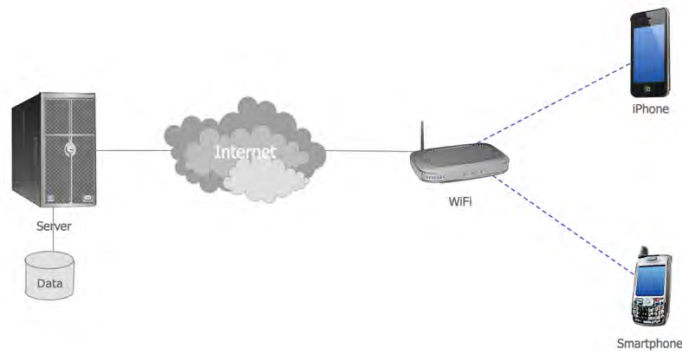| Developing Environment	Xcode
Programming Language	Objective-C
Server	ArcGIS server

- **Methodology:**

1. **Demand Analysis**

The basic demands are as following:

   a) Estimate the probability.

   b) Get users current location.

   c) Calculate the routes to the destination.

2. **Application Workflow**

Graph. 1. Data Stream Workflow

The graph 1 above demonstrates how the data stream in this project works between mobile devices and servers. It's a typical workflow that mobile devices will send and retrieve data from the local server through internet. In this project, the application will be installed in iPhone and the deployment target is 9.3. The stimulated data is a shapefile which contains hundreds of polygons representing the parking spaces and

several polylines representing the entrances and exits of this parking lot. Data will be published by ArcGIS server and stored in Geodatabase. Because this is a prototype, the dataset doesn't require many spaces and processing. So ArcGIS server and its geodatabase is enough for this project. In future works, if more data are included in this application, the server has to be enlarged with better processing ability.

3. Prediction Model

In this project, the core component of this application is developing a prediction model. Basically, this prediction model is a regression model. In order to make the results more precise, there will be several steps and selection criterion. Firstly, select some predictors. Secondly, analyze the relationship between the predictors and the probability of being vacant. Because of the lack of real data, the prediction model is stimulated in this project. The parameter of each predictor will be replaced.

The formulation of this prediction model:

$$Probability = x_1 * Predictor_1 + x_2 * Predictor_2 + x_3 * Predictor_3 + x_4 * Predictor_4 + \cdots$$

Probability ranges from 0 to 1, representing the possibility of being vacant.

The required attributes fields and explanations are in the following table.

For Each Parking Space:

Factor	Attribute	Explaination
Location Type	Exit	The Location of the parking space, e.g. exit of the parking lot
	Entrance	
	...	
Parking Type	Reserved for staff	The parking space type
	Reserved for Disabled	
	Normal	
NumPassBy	15	The number of cars passing by this spot within 5 minutes
Time Period	Morning	Which time period the current time belongs to
	Noon	
	Afternoon	
	Evening	
Date Period	Weekday	It's weekday or weekend
	Weekend	
Weather	Rainy	What's the weather like today
	Snowy	
	...	
Temperature	70	degree Fahrenheit
Occupancy Rate	0.65	Estimated occupancy rate (occupied and estimated prob < 0.4)
TimeSUpdating	300	Time since last updating (second)
TimeSPhChanging	3	Time since last phase changing (minute)
DirectedNum	68	How many cars directed to this mall for the last two hours

In this table, some data are based on data mining and some are from application users. For example, "NumPassBy" is calculated by counting how many users using this application drive passed by this spot. The "Occupancy Rate" is an estimated value calculated by counting the occupied spaces and the spaces in phase three with a probability of being vacant less than 40%. The "TimeSUpdating" and "TimeSPhChanging" should be valued based on the users of this application. The "DirectedNum" is mined from this application guiding database or google map.

4. Data Preparation

In this project, because there is no existing parking space data, the stimulated data has to be drawn and georeferenced manually. The data processing steps are as following:

a. Capture high resolution aerial image from website.

b. Draw outlines of parking spaces according to this image in Adobe Illustrator, use polygons to represent parking spaces and export as one DWG file.

c. Import this file into ArcGIS, georeference the data and project it, then export as one shapefile.

d. Add those predictor attributes to these polygons.

e. Publish this data through ArcGIS server.

5. User Interface Design

In order to simplify the User Interface and make it more friendly to users, the principle of UI design is simple and efficient. There will be few buttons. At right, there is a screen capture of this application user interface. At the top of the interface, users can search their destinations and direct them there. At left side of the bottom, the button "Tab To Locate" can help users find their current location. At right side of the bottom, the "Plus" button and "Minus" button can help users zoom out and in. At the very bottom, the selection bar provides users different kinds of base maps. In future work, there might be a sign-in component for users so that they can save their preferences and private information.
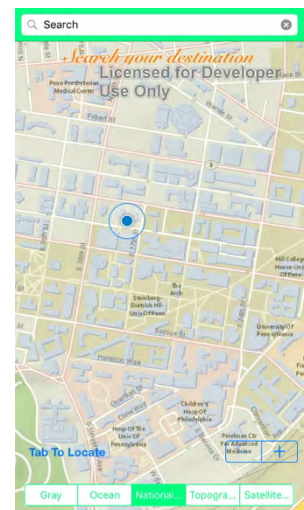
6. Algorithm Design

In this project, the programming language I used is Objective-C. Since the data is a shapefile published through ArcGIS server, so ArcGIS Runtime SDK or iOS will be perfect for this project with providing lots of APIs.

According to the user interface, there are functions for each button or component. In Xcode (Developing environment for iOS), it's easy to add function for each button or view component.

The whole algorithm can be regarded as two parts. The first part is designed based on the users operating processes. The second part is designed based on the the Parking Space Phase Changing process.
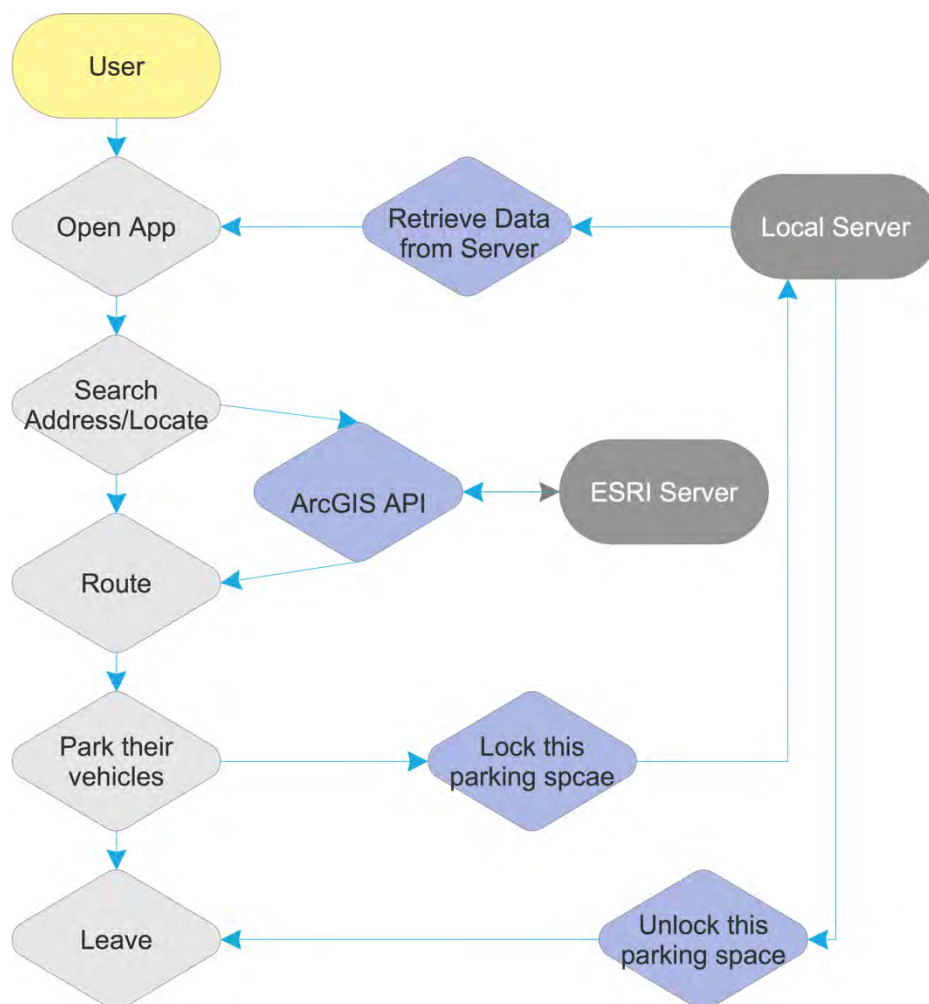
Graph. 2.User Operating Process

In Graph 2, gray polygons represent different status or operation of users and purple polygons represents the methods or functions used to complete users' operations. For "Retrieving data from server", the corresponding method is getting data with JSON package and codes:

```
self.featureServiceTable = AGSGDBFeatureServiceTable(serviceURL: featureLayerURL,
credential: nil, spatialReference:
AGSSpatialReference.webMercatorSpatialReference())
```

For "ArcGIS API", the corresponding method is routing method from ArcGIS API as following. Actually, there is more than that because we need to use network analysis in this step which means that we have to process the polylines first to make it a network file.

```
let routeTask = AGSRouteTask(URL: url)
```

For "Locking this parking space", in order to lock a parking space, the one thing need to do is changing the status value of this polygon from vacant to occupied.

For "Unlock this parking space", in order to unlock a parking space, one opposite thing needed to be done which is changing status value from occupied to vacant.

For the reason that users will edit the data online, so the method to complete this function is:

```
saveFeature on AGSGDBFeatureTable
```

For "Search" function, the corresponding method is:

```
let address = ["Address": "3600 Chestnut",  "City": "Philadelphia", "State": "PA",
"Zip": "19104"]
let outFields = ["Shape", "Score", "Match_addr", "Side ", "HouseNum",
"StreetName", "StreetType", "City","State", "Zip"]
let outSR = AGSSpatialReference.wgs84SpatialReference()
locator.locationsForAddress(address, returnFields: outFields)
```

This method can geocode the address that users type in.

For "Tab To Locate", the corresponding method is:

```
self.mapView.locationDisplay.startDataSource()
```

At the same time, ArcGIS SDK allows developers to stimulate location manually. It's a useful function in a prototype kind of application.
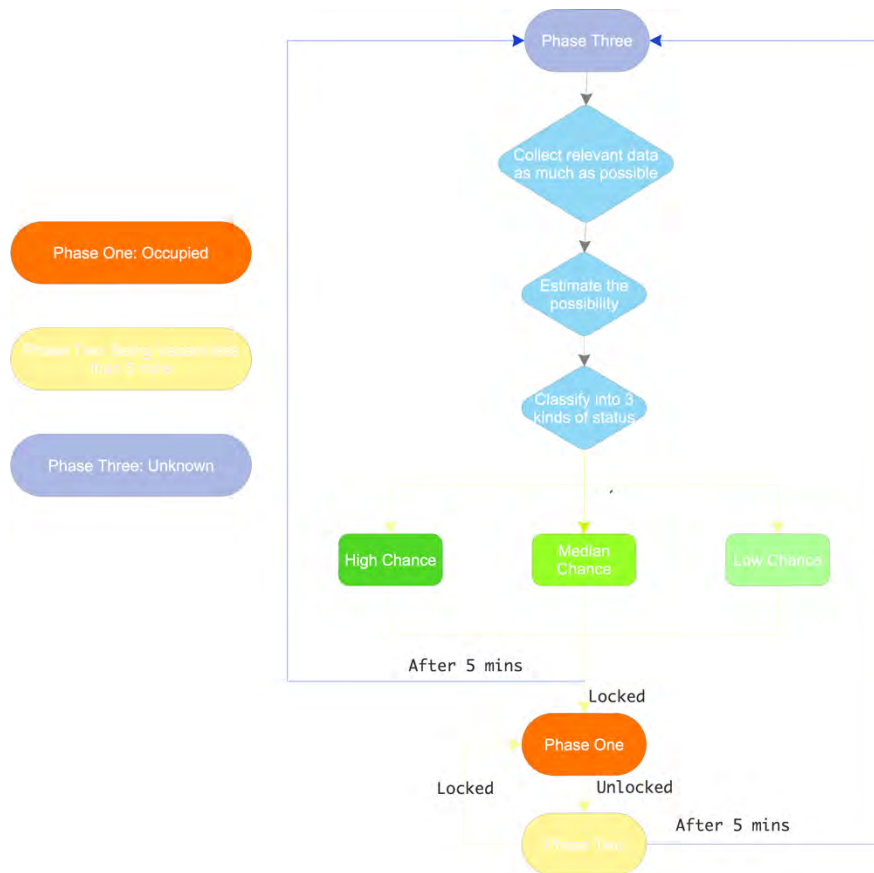

Graph. 3. Parking Phase Changing Process

In Graph 3, it's a simplified Phase Changing Process workflow. At first, there is one assumption that the status of one parking space can be defined as three phases. Phase one: Occupied. Phase Two: Being vacant less than 5 minutes. Phase Three Phase: Unknown. In this assumption, the status of each parking space is the record that updated based on users' behaviors. For example, after locking a parking space, the status will change from vacant to occupied. Then it means that this parking space is in

phase one. When the users start guiding and leave their spots, the current parking space status will turn to vacant. Within the following 5 minutes, if the status is not changing to occupied again, this parking space is in Phase Two. After 5 minuets, the status of this parking space is changing to unknown which is Phase Three for the reason that there might be some one who doesn't use this application parking in this spot. When the parking space is in Phase Three, the prediction model will estimate its possibility of being vacant according to its existing attributes' values.

- **Results and Products:**
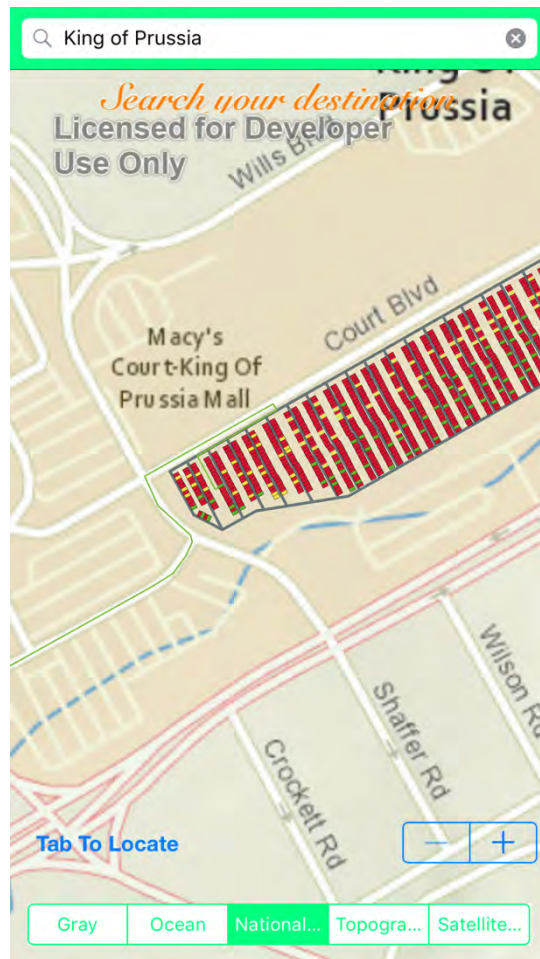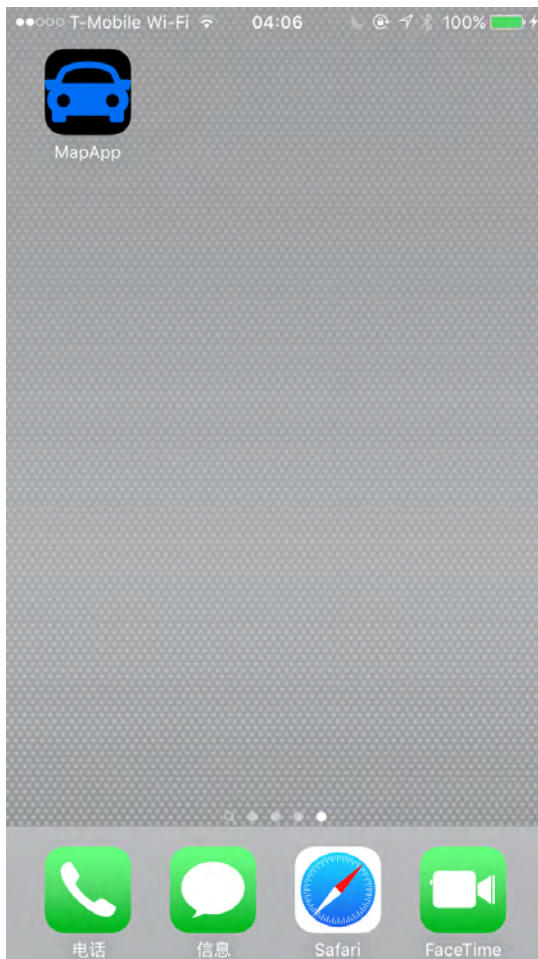
- **Discussion:**

Considering that the completed application is a prototype and its prediction model is based on stimulated data. So the restriction of this application is obvious. The

accuracy of the predicting results still needs feasibility testing. At the same time, there are very few components contained in this application. In the future work, there are several parts can be done to make this application more practical and reliable.

1. Use real data from some kinds of surveys or studies.
2. Revise the prediction model according to the results based on real data.
3. Add sign-in and sign-out components.
4. Add other components according to users' feedback.

- **Reference Websites:**

[1].   http://www.bestparking.com/

[2].   http://parknav.com/

[3].   https://developers.arcgis.com/ios/

[4].   https://www.openstreetmap.org

[5].   http://maps.us.nearmap.com/

[6].   https://aws.amazon.com/education/awseducate/

[7].   http://codewithchris.com/