

## Integration of dynamic KML format and pocket GPS system

Mr. Vipinkumar R. Pawar<sup>1</sup>, Mr. Swapnil Thorat<sup>2</sup>, Miss. Antara Indulkar<sup>3</sup>,  
Miss. Komal Nandgaye<sup>4</sup>

<sup>1</sup> PhD Scholar, EXTC Department,

Terna College of Engineering, Navi Mumbai

<sup>2</sup> Research Assistant, Department of Instrumentation, IIRSG, Nashik.

<sup>3</sup> Student, Dhole Patil College of Engineering, Pune

<sup>4</sup> Student, Dhole Patil College of Engineering, Pune

**Abstract** - GPS receivers provide information which is difficult to understand by users, we propose an application which converts raw location data to KML format file that is applicable to navigational maps. We extract the coordinates and time from GPS receivers and analyze this geological data. Then a new KML file will be created. We load the file to a specific geographic application to evaluate conversion results. Now a day's GPS receivers has been integrated into devices such as navigation, smart phone, black-box etc to support location based services. These services provide useful information around the users such as navigational information and other facilities. For easy understanding this information should be represented onto commercial maps. We propose to develop an application to convert navigational data received from GPS receiver to KML file which is used for commercial maps.

**Keywords:** - GPS, KML

### I. Introduction

#### 1.1. NMEA Code Description

NMEA stands for the National Marine Electronics Association. Today in the world of GPS, NMEA is a standard data format supported by all GPS manufacturers.

The NMEA standards purpose is to give users the ability to mix and match hardware and software. NMEA-formatted GPS data is easier to write software for a wide variety of GPS receivers. Without a standard such as NMEA, it would be time-consuming and expensive to write and maintain such software.

There are many different types of NMEA messages with different capabilities. Furthermore, NMEA data can be transmitted via different types of communications interfaces such as RS-232, USB, and Bluetooth, Wi-Fi, UHF, and many others.

U-centre software provides users with a simple way to interface with U-blox GNSS chipsets, modules and boards. It enables easy evaluation, performance testing, development and debugging of GNSS positioning chips and modules. U-centre allows easy connection to U-blox products and provides features to view, log, and analyze performance.

#### 1.2. Features of U-Centre:

1. Support for receivers using u-blox positioning technology. U-centre can communicate with these receivers using either the UBX protocol or the NMEA-0183 standard protocol.
2. Support for receivers that utilize standard NMEA strings.
3. U-centre presents all the information collected during the operation of the GNSS device. All aspects of GNSS data (position, velocity, time, satellite tracking, etc.) can be monitored and logged under various test scenarios for the evaluation of a receiver. U-centre software allows analysis of the collected data in order to investigate performance issues such as accuracy, road test position and trajectory, satellite tracking, time to first fix, etc. All processed data can be captured in ASCII format and ported into spreadsheets for creating additional plots and statistics.
4. Camera View: photographic data can be stored in the log file together with the navigation data and later be replayed in the application.
5. Export data files to Maps.
6. Data recording and playback function.
7. Structural and graphical data visualization in real-time.
8. Export functionality to standard PC applications.

### 1.3. General information about displayed values in U-Centre:

Longitude and latitude are displayed according to the datum selected in the GNSS device (either the default WGS-84 or based on user-defined parameters). This option can be polled and set using the UBX-CFG-DAT message.

Time is displayed with reference to UTC.

Elevation is displayed with reference to either MSL (Height above Mean Sea Level or Orthometric Height) or to HAE (Height above WGS-84-Ellipsoid). The reference is controlled by the GNSS configuration.

### 1.4. NMEA Message Structure

To understand the NMEA message structure, let's examine the popular \$GPGGA message:

`$GPGGA,181908.00,3404.7041778,N,07044.3966270,W,4,13,1.00,495.144,M,29.200,M,0.10,0000*40`

All NMEA messages start with the \$ character, and each data field is separated by a comma.

1. **GP** represent that it is a GPS position (GL would denote GLONASS).
2. **181908.00** is the time stamp: UTC time in hours, minutes and seconds.
3. **3404.7041778** is the latitude in the DDMM.MMMMM format. Decimal places are variable.
4. **N** denotes north latitude.
5. **07044.3966270** is the longitude in the DDDMM.MMMMM format. Decimal places are variable.
6. **W** denotes west longitude.
7. **4** denote the Quality Indicator:
8. **13** denote number of satellites used in the coordinate.
9. **1.0** denotes the HDOP (horizontal dilution of precision).
10. **495.144** denote altitude of the antenna.
11. **M** denotes units of altitude (eg. Meters or Feet)
12. **29.200** denotes the geoidal separation (subtract this from the altitude of the antenna to arrive at the Height above Ellipsoid (HAE)).
13. **M** denotes the units used by the geoidal separation.
14. **1.0** denotes the age of the correction (if any).
15. **0000** denotes the correction station ID (if any).
16. **\*40** denote the checksum.
17. The \$GPGGA is a basic GPS NMEA message. There are alternative and companion NMEA messages that provide similar or additional information.

## III. Methodology

### 3.1 SYSTEM DESIGN

The block diagram of the proposed system as shown in Fig. 1 consists of GPS module Neo-6M, Arduino Nano, and Transmitter device such as Bluetooth, LoRa or ZigBee. Neo-6M is the GPS receiver which receives data such as longitude, latitude and altitude as well as time. The Arduino Nano microcontroller processes these received data from GPS receiver process that data in user defined format. The transmitter device sends that formatted data to PC for further processing / communication.

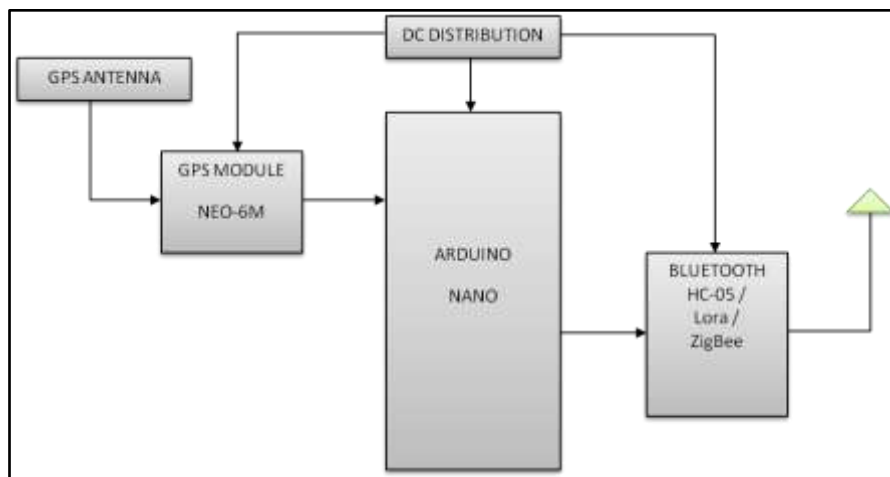


Figure 1: System Design

### 3.2. Flow Chart

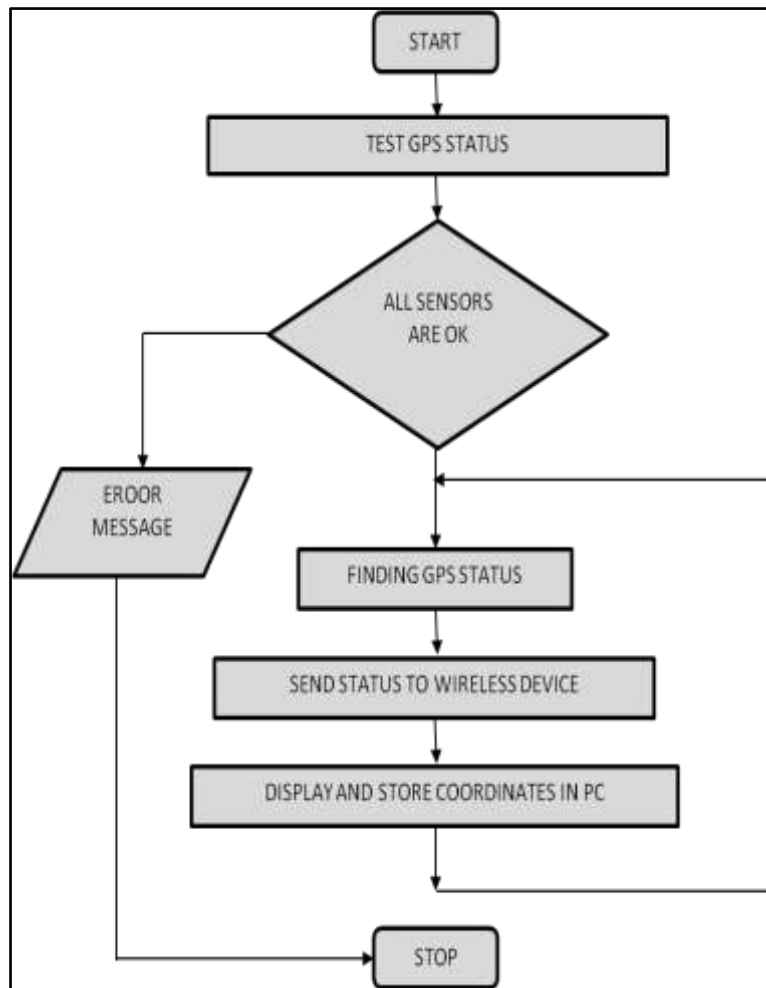


Figure 2: Flow Chart

### 3.3. Circuit Diagram:

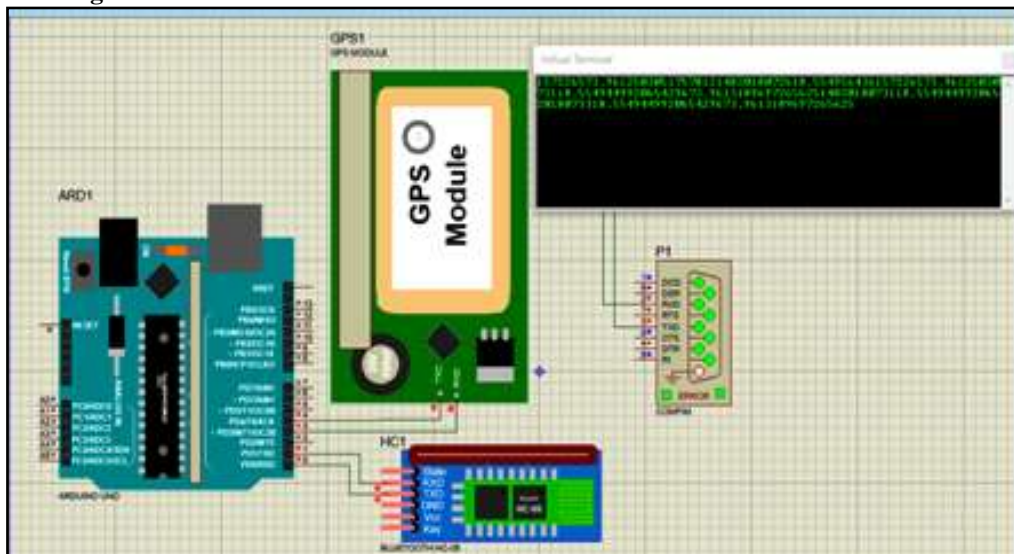


Figure 3: Circuit Diagram

## IV. Achieved Results

### 4.1. Satellite Signal

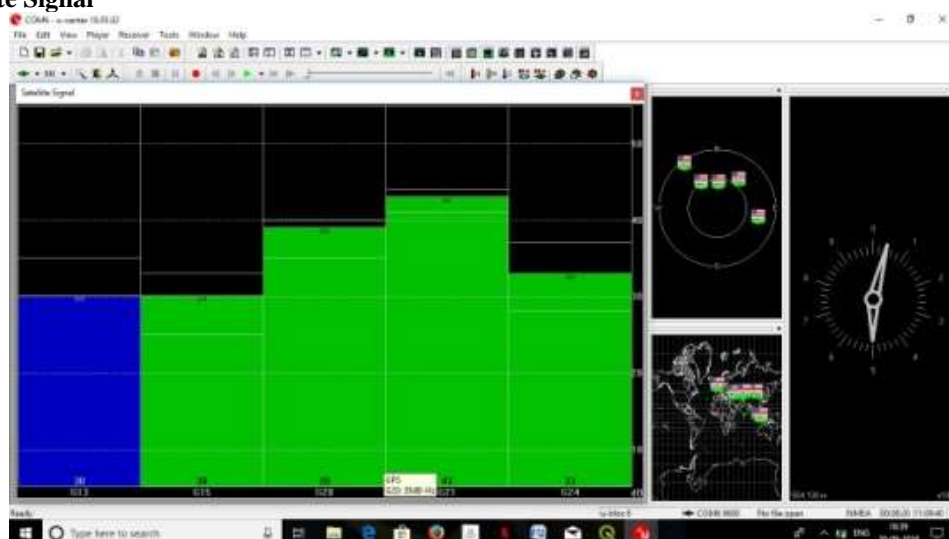


Figure 4: Satellite Signal

Shows an available satellite signal strength.

### 4.2. Satellite Position

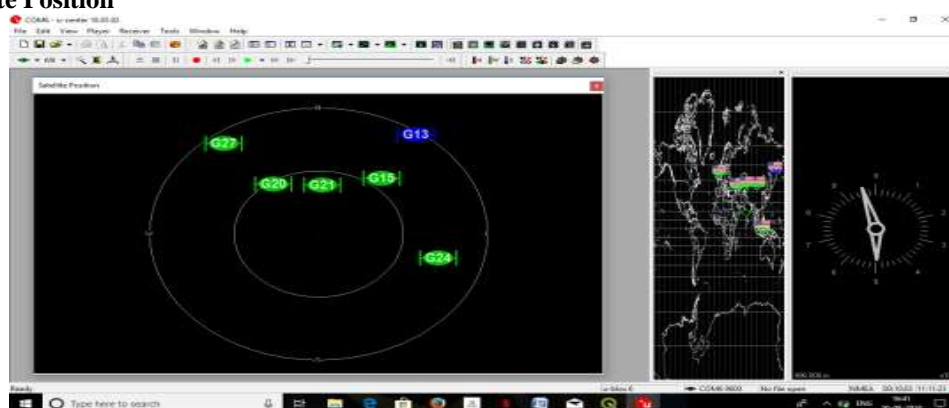


Figure 5: Satellite Position

Shows or hide the current satellite positions.

### 4.3. Ground Track

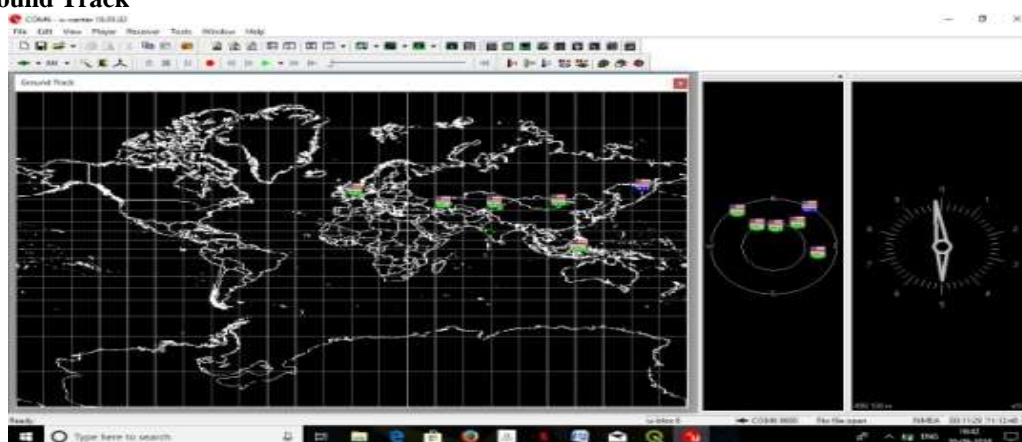


Figure 6: Ground Track

Shows or hide the current position on the world map.

#### 4.4 Map View



Figure 7: Map View

To use a map in u-center, three calibration points are needed. For these points you have to know the pixel coordinates and the according WGS84 coordinates in the latitude/longitude format in degrees (Longitude: -180.0° to 180.0°, latitude: -90.0° to 90.0°). These points are stored in the map calibration file. The calibration file must be stored at the same location as the bitmap itself. It has the same name but a different extension (\*.mcf). The format of the calibration file is very simple and can be edited in a simple editor like notepad.

#### 4.5. Packet Console

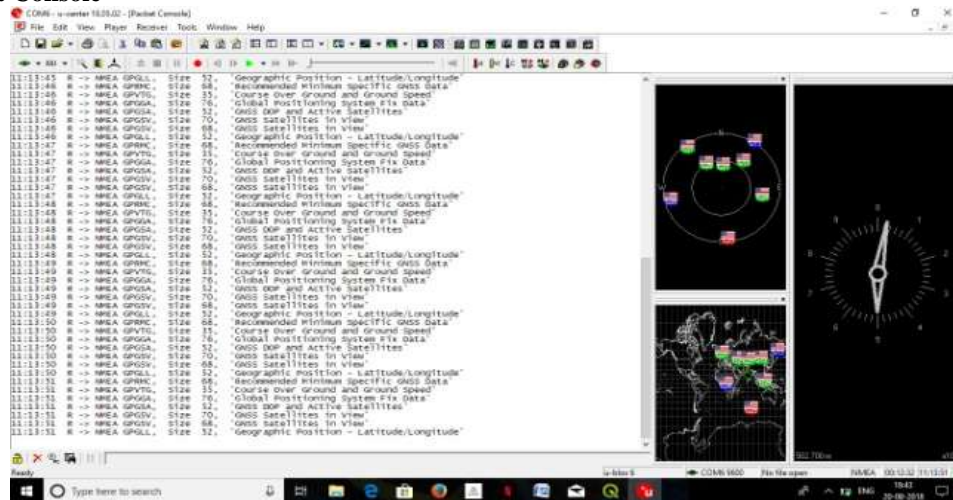


Figure 8: Packet Console

The packet console lists all incoming and outgoing messages, and provides information about message length and type. The direction of the message is shown in the following way:

R <- Indicates the message was sent to the receiver

R -> Indicates the message came from the receiver

L -> Indicates the message came from the log file





#### 4.8. Chart View

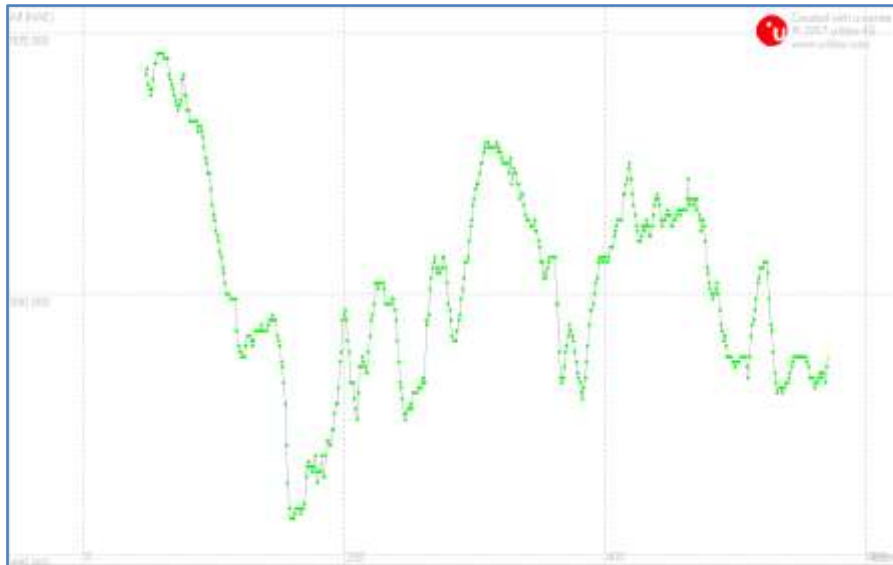


Figure 11: Chart View

Chart view allows the user to conveniently view GNSS data records in graphical form. The data can be scaled in many different ways and formats. It's even possible to print the entire chart.

#### 4.9 Histogram View

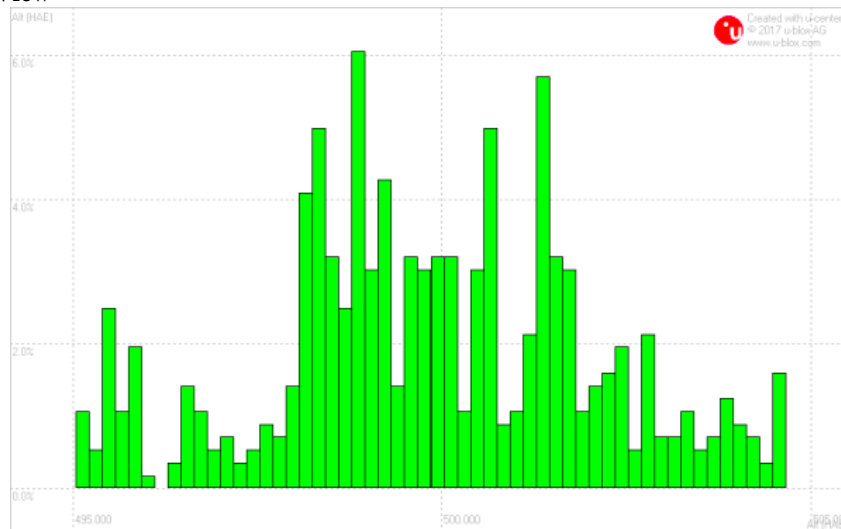


Figure 12: Histogram View

Histogram views allow the user to view GNSS data and probability distributions and print the entire histogram if desired. The number of bins (storage containers) can be set by the user.

#### 4.10. Deviation Map

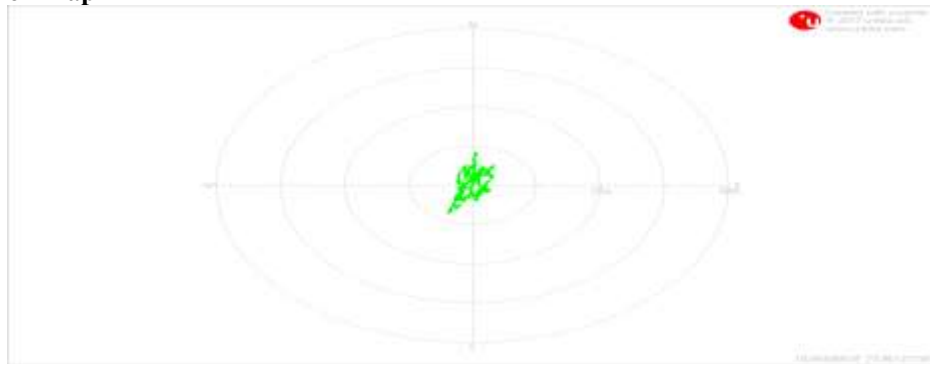


Figure 13: Deviation Map

The deviation map displays positions in longitude and latitude relative to a defined reference position.

#### References

- [1]. Casey Brennan, 2007, Introduction to the Global Positioning System (GPS) Geodesy & GPS for Dummies, Session III, National Science Teachers Association 2007 Annual Conference St. Louis, MO March 31, 2007.
- [2]. Mohd Shoab, Kamal Jain, M Anulhaq, M Shashi, Development and Implementation of NMEA Interpreter for Real Time GPS Data Logging, 3rd IEEE International Advance Computing Conference (IACC), 2013
- [3]. Dariusz Frejlichowski, Andrzej Lisaj, ANALYSIS OF LOSSLESS RADARIMAGES COMPRESSION FOR NAVIGATION IN MARINETRAFFIC AND REMOTE TRANSMISSION, IEEE 2008
- [4]. G.S.Prasanth Ganesh, B.Balaji , T.A.Srinivasa Varadhan, Anti-Theft Tracking System for Automobiles (AutoGSM), IEEE 2011
- [5]. Chuanmin Hu, Brock Murch, Alina A. Corcoran, Lianyuan Zheng, Brian B. Barnes, Robert H. Weisberg, Karen Atwood, and Jason M. Lenes, Developing a Smart Semantic Web With Linked Data and Models for Near-Real-Time Monitoring of Red Tides in the Eastern Gulf of Mexico, IEEE 2015
- [6]. Aldilano Bella Marlintha, Budhi Irawan, Roswan Latuconsina, DESIGN AND IMPLEMENTATION OF SMART VILLAGE MAPPING GEOGRAPHIC INFORMATION SYSTEM BASED WEB IN THE CINUNUK VILLAGE, IEEE 2017
- [7]. Jihn-Fa Jan, Digital Heritage Inventory using Open Source Geospatial Software, IEEE 2016
- [8]. Yili Chan\*, M. Mori\*\*, WEB-BASED FLOOD MONITORING SYSTEM USING GOOGLE EARTH AND 3D GIS, IEEE 2011
- [9]. J. Toman and J. I. Olszewska, Algorithm for Graph Building Based on Google Maps and Google Earth, IEEE 2014
- [10]. T. F. Abidin1, M. Subiallo1, T. A. Gani, R. J'erdhiana3, Periodic Update and Automatic Extraction of Web Data for Creating a Google Earth Based Tool, IEEE 2015
- [11]. Dar-HsiungWang, Han-Ching Hsieh, Chin-Shien Wu, Visualization with Google Earth and Gaming Engine, IEEE 2012
- [12]. DU Ying-jun YU Chong-chong Liu Jie, A Study of GIS development based on KML and Google Earth, IEEE 2009
- [13]. Aijun Chen, Gregory G. Leptoukh, Member, IEEE, and Steven J. Kempler, Using KML and Virtual Globes to Access and Visualize Heterogeneous Datasets and Explore Their Relationships Along the A-Train Tracks, IEEE 2010
- [14]. Soonyoung Song, Kunmin Yeo, Yong-Tae Lee, Won Ryu, Youngil Kim, An Application to Generate Keyhole Markup Language File Format From GPS Location Data
- [15]. Gang Mei, John C.Tipper, Nengxiong Xu, Discrete Surface Modeling Based on Google Earth: A Case Study,