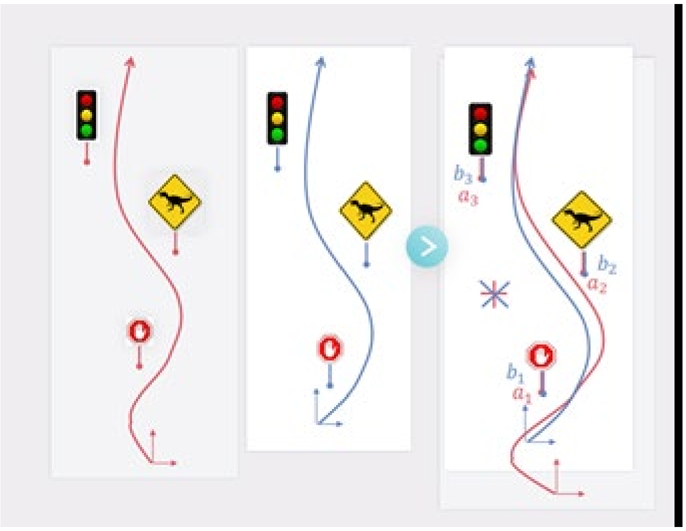
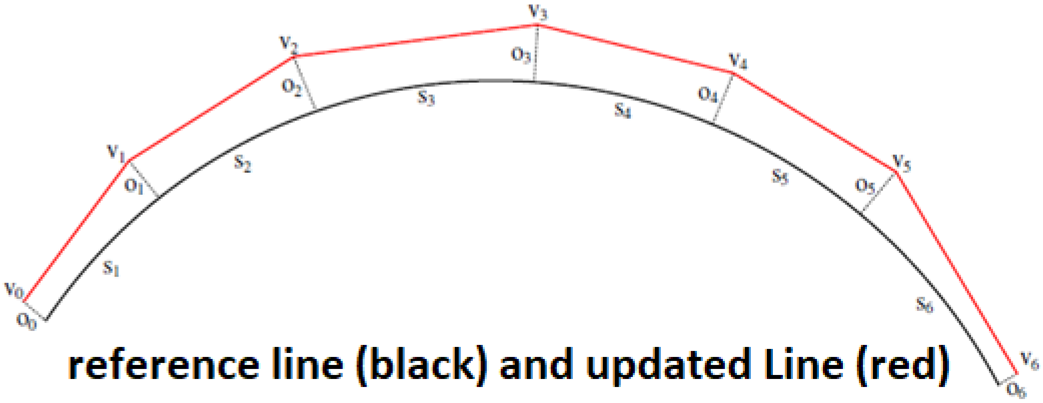
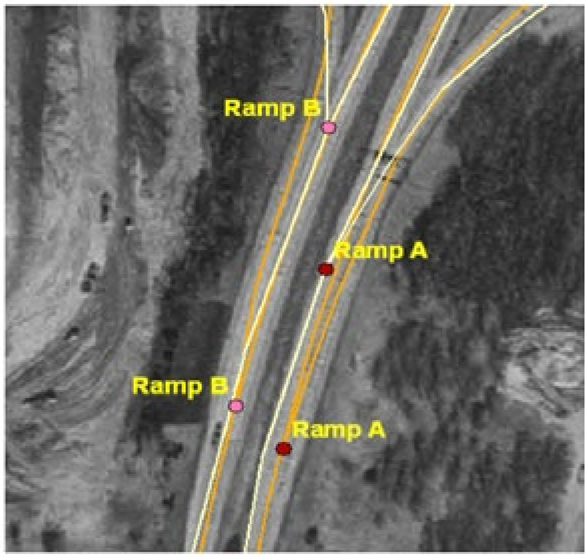
Tasks:

1. Update records
2. Show change on map
3. find Alignment Error, Misalignment between OSM OD and Update map





reference line (black) and updated Line (red)



Error types and sources in roadway centerline maps.

Requirements for update:

- A way to combine GPS trajectories from multiple files which can overlap roads, give new road, road linking function, when there is a brand new road one doesn't exist in OSM
- A way to insert, delete, update planView, objects, tags, attributes, link and positions
- A way to update single and batch records using csv, geojson
- Record version history
- A way to measure accuracy – delta new data (ground truth) vs old data for same area

Input files: crosswalk.csv, intersection.csv, lane.csv, pole.csv, railroad_crossing.csv, stopline.csv, signal.csv, line.csv, roadedge.csv, whiteline.csv, yellowline.csv etc

Input Data:

File: junction.geojson or junction.csv

Dummy Data: "properties": {"id": 1}, "geometry": {"type": "Polygon", "coordinates": [-84.400229853678411, 33.754140778654254],

File: solid_white_line.geojson or solid_white_line.csv

Dummy Data: "properties": { "id": 1 }, "geometry": { "type": "LineString", coordinates": [[-84.400187832991421, 33.754170493111644], [-84.400122222972414, 33.754215323073645], [-84.400088823317574, 33.75423876661155], [-84.400057559533693, 33.754261102635446], …………

File: stop_line.geojson or stop_line.csv

Dummy Data: "properties": { "id": 1 }, "geometry": { "type": "LineString", "coordinates": [[-84.400187954116888, 33.754170429286823], [-84.400141707895102, 33.754122990467856], [-84.400141707895102, 33.754122990467856]] } }

File: road_boundary.geojson or road_boundary.csv

Dummy Data: "properties": {"id": 1}, "geometry": {“type": "LineString", "coordinates": [-84.400206577930533, 33.754166839120323], …………………………..

File: pedestrian_crossing.geojson or pedestrian_crossing.csv

Dummy Data: "properties": {"id": 1}, "geometry": {“type": "Polygon", "coordinates": [-84.400229853678411, 33.754140778654254],

File: bike_crossing.geojson or bike_crossing.csv

Dummy Data: "properties": {"id": 1}, "geometry": {“type": "Polygon", "coordinates": [[-84.400259410999382, 33.754122483964842] …

File: broken_white_lines.geojson or broken_white_lines.csv

Sample Data: "properties": {“id": 1}, "geometry": {“type": "LineString", "coordinates": [[-84.400100368502962, 33.754189027713061],

File: double_yellow.geojson or double_yellow.csv

Dummy Data: "properties": {"id": 1}, "geometry": {“type": "LineString", "coordinates": [[-84.400141885585924, 33.754123063150729], ………………………..

File: laneid+1.csv

Dummy Data: "properties": {"id": 1}, "geometry": {“type": "Polygon", "coordinates": [[-84.400259410999382, 33.754122483964842] …

File: laneid+2.csv

Dummy Data: "properties": {"id": 1}, "geometry": {“type": "Polygon", "coordinates": [[-84.400259410999382, 33.754122483964842] …

File: laneid+3.csv

Dummy Data: "properties": {"id": 1}, "geometry": {“type": "Polygon", "coordinates": [[-84.400259410999382, 33.754122483964842] …

File: laneid+3.csv

Dummy Data: "properties": {"id": 1}, "geometry": {“type": "Polygon", "coordinates": [[-84.400259410999382, 33.754122483964842] …

File: Guardrail.csv

Dummy Data: "properties": {"id": 1}, "geometry": {“type": "LineString", "coordinates": [[-84.400141885585924, 33.754123063150729], ………………………..

File: Incident.geojson or. csv

Dummy Data: "properties": {“Laneid": 1}, "geometry": {“type": "point", "coordinates": [[-84.400259410999382, 33.754122483964842] …

File: Pothole.geojson or. csv

Dummy Data: "properties": {“Laneid": 1}, "geometry": {“type": "point", "coordinates": [[-84.400259410999382, 33.754122483964842] …

File: Overpass.geojson or. csv

Dummy Data: "properties": {“id": 1}, "geometry": {“type": "LineString", "coordinates": [[-84.400141885585924, 33.754123063150729], “height”: {…}]]

File: TrafficSigns.geojson or. csv

Dummy Data: "properties": {“id": 1}, {“type: }, "geometry": {“type": "LineString", "coordinates": [[-84.400141885585924, 33.754123063150729], “height”: {…}]]

File: Speedlimit.geojson or. csv

Dummy Data: "properties": {“id": 1}, {“type: }, "geometry": {“type": "LineString", "coordinates": [[-84.400141885585924, 33.754123063150729], “height”: {…}]]

Insert Update Rules:

1) road
   tags/attrs
  1.1) planView
   pos
  1.2) lane
   pos, tags, attrs
  1.3) objects
   tags/attrs, pos
  1.4) signals
   tags/attrs, pos

2) junction
  tags/attrs, pos

- Load road csv data
  Find nearest relevant tags/attributes
  If found
  Update their sxy position
  If not found
  Insert attribute and calculate sxy position

- Find planView for the GPS position
  Update lane sxy position and relevant geometry
  Update lane Ids and Road Ids
  If not found, insert

- Find Junction
  Update tags/attrs, pos, link, name
  If not found, insert

- Find Signal
  Update tags/attrs, pos, link
  If not found, insert