Realignment of Road Network Maps with GPS Tracking Data

Nicola JAMMALIEH and Joshua GREENFELD, Israel

Keywords: Road network, GPS data, Alignment, Digital maps

ABSTRACT

Road network datasets are widely available either for a fee or for free on the Internet. Unfortunately, some of them are not always accurate and up-to-date. These inaccuracies could cause navigation errors and prove costly to users. Therefore, it is important to devise a useful, efficient and cost effective method to make the datasets more accurate. One way to rectify the dataset quality is to use GPS data collected by GPS enabled navigation devices. When the map is not accurate it is reasonable to assume that the GPS data is more accurate than the map. Thus, GPS tracks can be used to realign the traveled street segments. One can view this as the inverse of the map matching problem. Instead of matching GPS positions to the map, we match the map to GPS tracks (or points).

This paper outlines a comprehensive approach for realigning street segments to GPS data collected from moving vehicles. The process includes GPS data filtering, matching GPS points to existing road segments, shifting the road segments to the GPS points and forming new intersections and vertices. The end result of the process is a revised map of the road segments in their corrected positions. For each of these tasks new algorithms or enhanced existing algorithms were developed and employed. The proposed process was successfully implemented on real world data and the results of the realigned road segments are shown, analyzed and verified. The realigned network showed full agreement with high accuracy orthophoto of the test area.

INTRODUCTION

The most effective method for representing a road network in GIS and other transportation applications, is by preparing a digital road map. This map contains not only the geometric properties of the network but also a large number of attributes such as road type, road number, number of lanes, road surface, speed limits, facilities, etc.. Digital maps can be viewed and analyzed in various scales and can be used for various applications such as finding the shortest path for a given weight (shortest distance, minimal traffic volume, a minimum number of intersections, etc.), navigation, fleet management, transportation studies, etc. However, inaccurate digital maps are likely to yield inaccurate results. Therefore, it is desirable to develop an efficient, cost effective way to create accurate maps or improve existing ones.

Digital maps are created and updated using different methods and diverse technologies. The least expensive and one of the fastest methods for creating digital maps for relatively large areas is digitization of existing maps. However, digitized maps are often incomplete and inaccurate. Traditional photogrammetric techniques for creating and updating digital maps are much more accurate but are also much more expensive and require special expertise. Therefore, these methods are less suitable for creating or correcting road datasets.

With the proliferation of GPS navigation devices and the use of smart phone based navigation applications, the road networks are being constantly digitized. Many of these recorded tracks or traces are stored on-line (e.g. OpenStreetMaps [*1*]) and can be used to create or update digital maps of the road network. The accuracy of these GPS traces is in many places better than the accuracy of the available digital map. With the modernization of the GPS system and the availability of additional Global Navigation Satellite Systems (GNSS), this will become even more apparent. To enable the realignment of the road segments to match the GPS tracks it is necessary to develop a method and procedure in which the map is corrected where GPS tracks are available but, at the same time, the integrity of the complete dataset is not disturbed. The method also has to be applicable to a varying number of repeated traces of the roads, not only when there are many of them which makes the solution easier.

There are two approaches for creating digital road map datasets. The first is to create a completely new dataset without any regards to existing maps. The main focus of this approach is to average out the point clouds created by repeated tracing of the network. This approach was taken by Edelkamp and Schrödl [*12*], Schöredl et al. [*11*], Worral and Nebot [*14*], and Cao and Krumm [*8*]. The second approach is to update and improve the locational geometry of the existing datasets. An example of this approach to the problem is presented in He [*6*].The advantage of the second approach is that it preserves the already associated none-geometric attributes of the road segments and we don't have to recreate them. Recreating the attribute information for newly created datasets could become a daunting task.

In this paper we describe a realignment method that is based on the latter approach. First it matches the GPS data points to road segments of the existing digital map. The matching process can be improved if superfluous GPS points are first filtered out. Superfluous GPS points are recorded when the vehicle is idling or moving at a very low speed. These extra points have no contribution to the matching process and usually degrade the solution because of GPS positioning error. Our method also requires some generalization of the map to ensure that enough GPS points are associated with each road segment. Next, new intersection locations are computed and the current map intersections are shifted to the new GPS derived intersections. The final step is to reconstruct the geometry of the existing map to correspond to the newly derived intersections and the traveled GPS path. The description of the methodology and the implementation of these steps are presented next.

THE MAP REALIGNMENT PROCESS

Our proposed realignment method is based on using the existing map of the road network and realigning it with GPS traveled tracks. As mentioned earlier, the advantage of this approach is that it alters only the geometric characteristic of the network but preserves the attributes associated with each road segment. It also maintains the segments which were not travelled so the dataset is more complete. Our method is generally based on the approach offered by He

[*6*]. He [*6*] suggested a Hierarchical Approach for the realignment process which is based on finding intersection and refining their locations. Our approach is also based on relocating the intersections but it is a different, more comprehensive process that works on an entire network with any geometric configuration. Our method includes the following consecutive steps:

1. Preliminary GPS Data Preparation – remove superfluous and outlying data points
2. Preliminary Map Data Preparation – a generalization process to increase the likelihood of associating GPS points with map segments
3. Map matching – associate GPS points with map segments
4. Repositioning of the intersections – compute intersections based on GPS points
5. Restoring the original geometric properties of the network – relocate map segments to correspond to the new intersection location and to reconstruct the original shape of the network
6. Enhancing the geometric properties of the network – revise the shape of the network to match it to the GPS tracks.

If the original map is highly inaccurate, steps 2-6 can be repeated where subsequent iterations use the resultant realigned network as input to the next iteration. These six steps are described herein.

Step 1: Preliminary GPS Data Preparation

The quality of the collected GPS data varies mainly as a function of the geometry of the satellite constellation, the level of unobstructed view of the satellites and the degree of multi-path effects. Therefore, GPS data collected from a moving vehicle has to be examined and prepared for effective use. We used three filtering measures to remove erroneous GPS data points or superfluous data points that do not contribute to the solution. Data points were filtered out or removed based on the following factors:

1. Remove points collected when the vehicle was moving at very low speed (less than 5 Km/hr) or when the vehicle was at a complete stop
2. Remove points with low accuracy indicators
3. Remove blunders

GPS positions are collected at a preset time interval. Thus, if a vehicle is delayed at a traffic signal it will collect multiple data for the same location. Because of GPS positioning errors, this creates a pattern implying that the vehicle is moving in small random vectors, with large variations in the heading azimuth, while the vehicle is really stationary. A similar pattern is created when the vehicle moves at a very slow speed. Therefore, any successive GPS point observed within a selected distance tolerance from the last observed point is ignored. The tolerance can be based on the positioning accuracy of the used GPS receivers or on an arbitrary distance such as 5 or 10 meters.

Because of the inherited error in GPS positioning, especially under difficult urban environment, the computed GPS position can sometimes exhibit spikes like outliers. They have to be removed from the dataset. Consequently, a special strategy was developed to determine if a GPS point is inconsistent with the heading pattern of preceded and successive collected points. If the point is inconsistent, and displays significant departure from the established patterns it is considered to be an outlier and removed from the dataset. The outlier detection strategy is shown in figure 1.

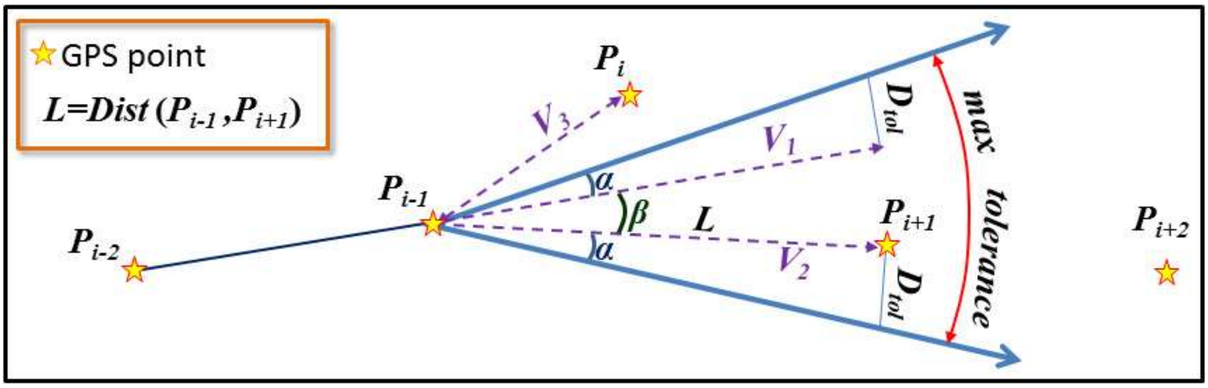

FIGURE 1. Outlier Determination strategy for GPS tracks

Figure 1 shows five consecutive GPS points P_j, $j=i\text{-}2,i+2$. We want to examine if point P_i is an outlier. To do so, we create two vectors, V_1 and V_2. V_1 is the extension of the line P_{i-2}, P_{i-1} and V_2 is the line connecting P_{i-1} and P_{i+1}. The length of V_1 and V_2 is L, the distance between P_{i-1} and P_{i+1}. The angle between V_1 and V_2 is β. The parameter D_{tol} is the displacement error that corresponds to the GPS positioning error and α is the angular displacement caused by D_{tol}. α and D_{tol} can be computed from:

$$\alpha = \arctan\left(\frac{D_{Tol}}{L}\right) \tag{1}$$

and

$$D_{Tol} = \bar{x} + \sigma_{\bar{x}} * Z_{1-\frac{\alpha}{2}} \tag{2}$$

Where $\bar{x}$ is the average GPS positional error, $\sigma_{\bar{x}}$ is the standard deviation of $\bar{x}$ and $Z_{1-\frac{\alpha}{2}}$ is a statistical multiplier for significance level α. The total acceptable angular change of direction is given by:

$$\max\_tolerance = 2 * \alpha + \beta \tag{3}$$

Points that fall inside *max_tolerance* area are considered to be consistent with the pattern of the GPS track and points outside the *max_tolerance* area are flagged as outliers. The use of angular tolerance works better than a simple distance offset tolerance because it is less sensitive to spacing variations of GPS points. It provides a wider *max_tolerance* angle for densely recorded GPS points and a smaller *max_tolerance* angle for more sparsely recorded GPS points.

One has to be careful not to remove too many points because it could undermine the quality of the solution. For example, removing too many points from a road segment could result in having too few data points to reconstruct that road segment. Too few points could also degrade the accuracy with which we can reconstruct the path in which the vehicle was traveling.

Step 2: Preliminary Map Data Preparation

A road network is composed of a collection of straight and curved line segments. In many cases the links between intersections are fragmented into small straight line segments. If these segments are too short only very few or even no GPS points could be associated with them. This is fine when we match the GPS points to the map, but if we want to redraw the map based on GPS points it is essential that several GPS points are matched to any road segments. This way the reconstructed segment becomes more robust and more reliable. To achieve this goal and to improve the map realignment process, the original map is first generalized into longer straight line segments with similar geometric characteristics. It is important to note that this simplification process is performed only on vertices not on the intersections.

A straightforward generalization process based on Douglass Peucker [*4*] is shown in figure 2. In figure 2 we can see that some of the original segments lack enough GPS points to support their reconstruction. At the same time all the generalized segments have sufficient GPS points to support their reconstruction.

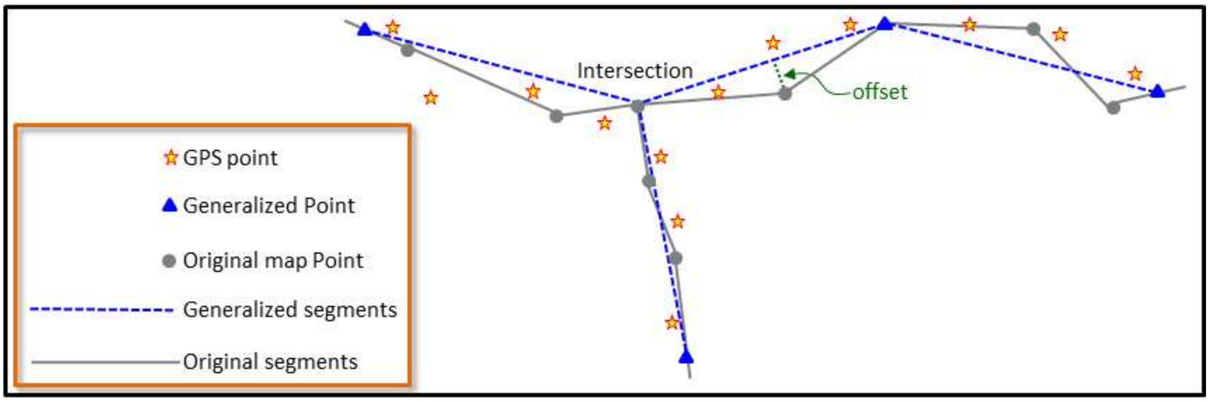

FIGURE 2. Generalization of network road segments to ensure the inclusion of enough GPS points in the realignment process.

Step 3: Map matching

Following the filtering of GPS points and the generalization of the map, it is now necessary to match the GPS points to the generalized map. Map matching has been extensively studied and several methods and approaches have been published on how to match GPS points to a street network. A good summary and comparison of many of these methods can be found in Quddus [*10*]. Since map matching is not the main focus of this paper, it is just one component of the realignment process; we used a slightly revised algorithm presented by Greenfeld [5]. The Greenfeld algorithm uses a similarity cost function to evaluate the likelihood of one (or several successive) GPS points to correspond to a candidate road segment. The algorithm starts with initially matching the first GPS point to the nearest network segment. Then, it uses the direction of vector from the first GPS point to the second point to find a nearby road segment with similar direction. Once the initial match is completed, consecutive GPS points are matched to the most similar network segment in the vicinity of the last matched segment. The similarity cost function used by Greenfeld [2] was:

$$W = W_{AZ} + W_{D} + W_{I} \quad (4)$$

Where:

W – is the total score (or cost)

W_{AZ} - is the weight for similarity in direction (azimuth) between the GPS track and the candidate segment

W_{D} - is the weight for proximity (distance) of the GPS point to the candidate segment

W_{I} - is the weight for intersection, if the GPS track crosses the candidate segment.

To improve the performance of the Greenfeld algorithm, an additional weight (W_{Inside}) was added to the cost function W. The projected GPS point on the candidate segment may fall on (inside) the segment or on one of the extensions of the segment. Intuitively, there is a higher probability that a GPS point will be inside the segment than on its extensions. Thus, W_{Inside} gives an additional score for a segment if the projected GPS point falls inside the segment and no additional score if it falls outside of the segment. The W_{Inside} weight is given by:

$$
\begin{aligned}
&if\ \left(P_a \le D_{Segment}\ and\ P_a \ge 0\right)\ then \\
&\quad W_{Inside} = Percentage * W_D \\
&else \\
&\quad W_{Inside} = 0
\end{aligned}
$$

Where:

W_{Inside}	– the weight for falling inside the candidate segment
P_a	– is the distance from the beginning of the segment to the projected GPS point on the segment
$D_{Segment}$	– is the length of the segment
Percentage	– a constant parameter
W_D	– is the proximity (projection distance) weight from Greenfeld [5]

The value for *Percentage* was empirically determined to be 0.5 (50%). This means that if a point falls inside the candidate segment, it receives and additional weight in the amount of one half of the proximity weight.

Thus, the revised matching cost function is:

$$W = W_{AZ} + W_{D} + W_{Inside} + W_{Intersect} \quad (5)$$

This cost function gave slightly better results especially near intersections where the matching task is more challenging.

Step 4: Repositioning of the intersections

As stated earlier, the proposed realignment process computes new locations for the street intersections of the network. To relocate the original intersection based on the GPS tracks it is necessary to compute the intersection or the convergence point of the GPS tracks. Since the GPS tracks are unlikely to converge at a single point for many reasons (e.g. they are collected on different lanes) the following two stage strategy is proposed to determine the new location of the intersection:

Stage One: Shift and rotate the generalized segment that is connected to the considered intersection. This task has four steps. This is done by:

(a) Connect the first and last matched GPS points to the generalized segment. This provides us with an initial approximation of the line that best fits the GPS points converging at the intersection.

(b) Compute the weighted average azimuth of all successive GPS points matched to the generalized segment. The weight (W_i) applied to each azimuth (AZ_i) is based on the distance (D_i) between the projection of the nearest GPS point (for which the azimuth is computed) to the intersection (see figure 3(*a*)). In our implementation we used $W_i = 1/D_i^4$.

(c) Rotate the generalized segment about its midpoint to the computed average Azimuth

(d) Compute the weighted average distance ($\bar{d}$) of the GPS points (using W_i from (*b*)) to the rotated segment from (*c*), and shift the rotated segment by $\bar{d}$. This will place the rotated segment in a position and orientation that best fits the GPS points.

Repeat (*a*) to (*d*) for all segments connected to the intersection.

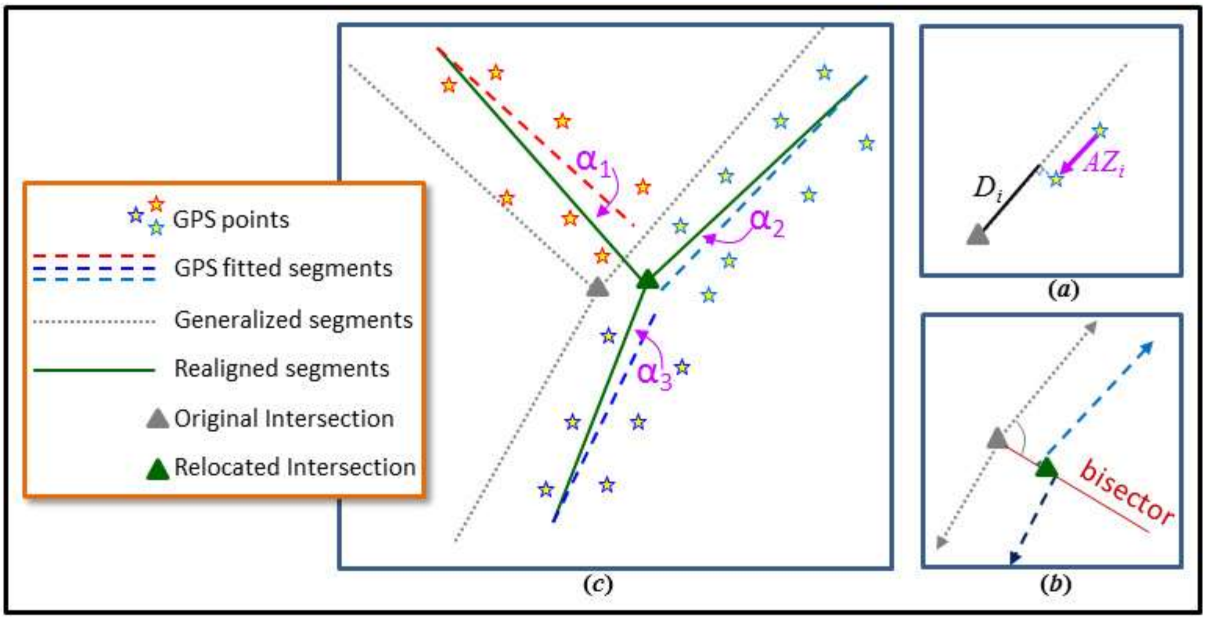

FIGURE 3. Computing the intersection's new location. (*a*) Definition of D_i and AZ_i, (*b*) determining the intersection's location for two nearly parallel segments (*c*) determining the intersection's location for three or more segments.

Stage Two: Find the point of intersection of the GPS fitted segments.

Finding the intersection point can be relatively easy for a "T" or "X" shaped intersection, but rather difficult for two nearly parallel segments. Moreover, it is unlikely that three or more GPS based segments will meet at a single point. To find the most probable location of the intersection of three or more segments, we propose a Least Squares based method that finds the point which minimizes the azimuth change applied to the GPS derived segments. For example we minimize the angles α_1, α_2 and α_3 shown in figure 3(c). The observation equation of the Least Squares solution is:

$$\alpha_i = \operatorname{atan}\left(\frac{(Y_b^i - Y_I)}{(X_b^i - X_I)}\right) - \operatorname{atan}\left(\frac{Y_b^i - Y_a^i}{X_b^i - X_a^i}\right) \tag{6}$$

Where:

(X_b^i, Y_b^i) – Coordinates of the segments' end point away from the intersection
(X_a^i, Y_a^i) – Coordinates of the segments' end point near the intersection
(X_I^i, Y_I^i) – The adjusted coordinates of the relocated intersection

The lengths of the respected segments D_i can be used as weights in the Least Squares process. To connect two GPS derived segments one can either compute their mathematical intersection (if the angle between the segments is acute) or average the coordinates of the segments intersecting the bisector extended from the original intersection of the original adjacent segments (see figure 3(b)). Following an elaborate error propagation analysis it was found that if the angle between the segments is $35°-145°$ one should use a straightforward mathematical intersection. Otherwise, the computation becomes less stable and the bisector method should be used.

Step 5: Restoring the original geometric properties of the network
One of the first steps of this realignment procedure was to generalize the original road network. It was done to facilitate matching of a larger number of GPS points to the segments and make the solution more reliable. However, in this process the original geometric properties of the network were changed and slightly compromised. In the current step the geometric properties of the original network are restored and repositioned to match the newly established intersection.

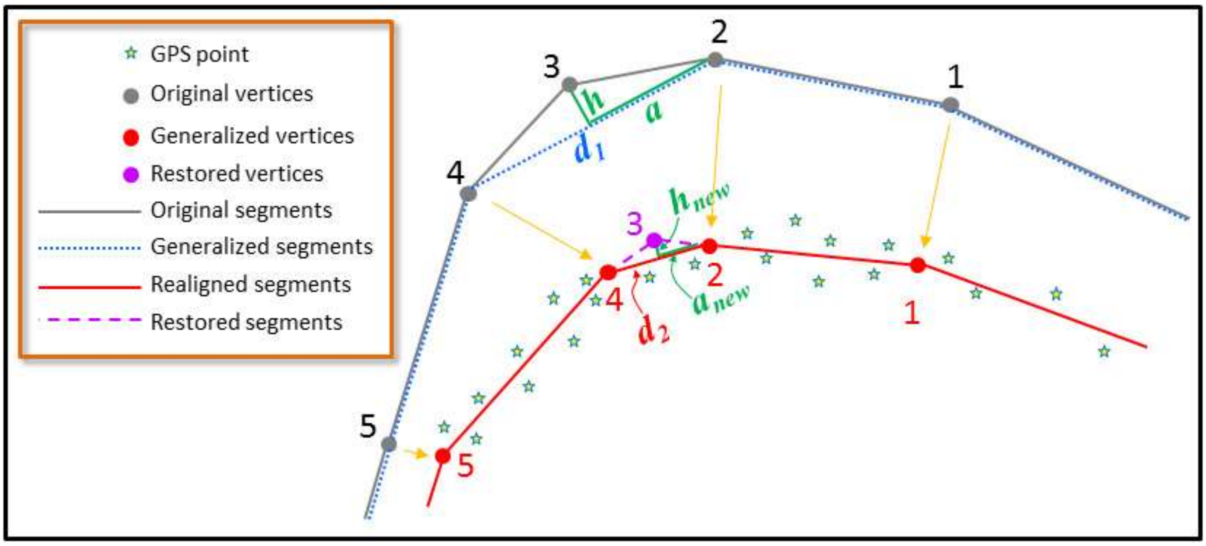

FIGURE 4. Restoring vertices that were eliminated during the generalization process.

The geometric restoration process is shown in figure 4. Vertex 3 in figure 4, was eliminated during the generalization process because it was deemed to be insignificant for depicting the shape of the curving road. But because we want to realign the complete original network, it is now necessary to relocate that point to be consistent with the realigned intersections/vertices. This is done by rescaling the abscissa (a) and offset (h) of the point from the original network to the realigned one. The new abscissa (a_{new}) and offset (h_{new}) is computed from:

$$a_{new} = a \cdot \frac{d_2}{d_1} \tag{7}$$

$$h_{new} = h \cdot \frac{d_2}{d_1} \tag{8}$$

Where: d_1 and d_2 are the distances between the end points of the generalized segment at the original location and at the new realigned location, respectively. See figure 4 for d_1 and d_2.

Step 6: Enhancing the geometric properties of the network

The geometrical representation of the road network in the original map can sometimes be too coarse, general or outdated. Because the GPS tracks capture the more current and factual geometrical characteristics of the travelled segments, we can use this information to add vertices to the realigned network. The most simple and intuitive approach to adding vertices is to compute the distances from each GPS point to the road segment and if that distance is larger than a selected tolerance, it will be added as a new vertex. However, since GPS locations can be spiky at times, this method could result in adding erroneous vertices. Therefore, we suggest that instead of computing the distance of a single GPS point from the road segment, we compute the mean of three consecutive GPS points and use that to determine if a new vertex is warranted. Once it was determined that a new vertex has to be added, its location should take into account GPS tracks observed while traveling both ways. In figure 5 the green dot represents the location of the added vertex if only the top GPS track is used. But, if we consider the lower track as well, the location of the added vertex becomes the purple dot. It can be seen that the purple dot is a better choice for a new vertex.

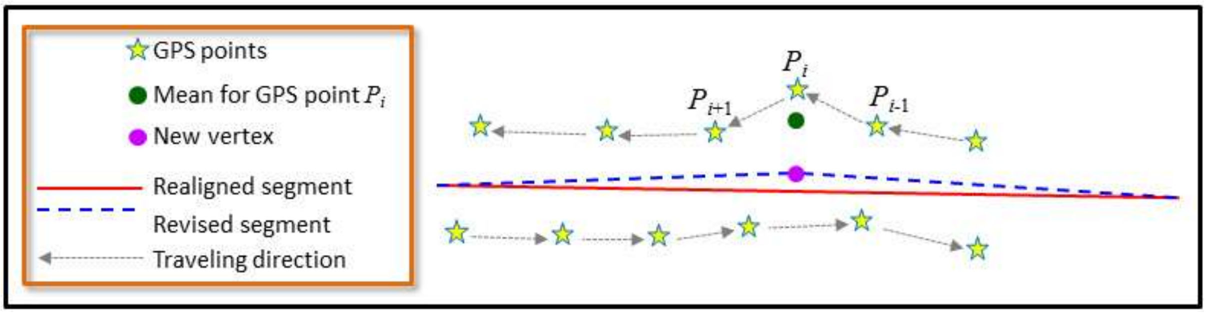

FIGURE 5. Adding a vertex based on GPS tracks.

The implementation of adding vertices to the realigned network requires devising a criterion for when it is appropriate to add a vertex and when to terminate the process. After examining several options we found that the following strategy yields the best results under most circumstances. We define an angle β which is the difference between the azimuth of the segment and the azimuths between each consecutive pair of GPS points. We then compute the average $\bar{\beta}$ and the standard deviation σ_β. It was found that for $\sigma_\beta > 2°$ too few vertices were added and for $\sigma_\beta < 1°$ too many vertices were added. Thus it is recommended to add new vertices as long as $1 < \sigma_\beta < 2°$.

EXPERIMENTS AND RESULTS

To evaluate the performance of the proposed realigning method, it was tested with extensive real world data in an urban area setting. The traveled streets varied from two to four lanes. Some of them were one way streets while others were two way streets. Some streets have residential buildings with up to four stories high and some are open areas. Some streets had large trees with dense canopy while others had no trees. To enable quality assessment of the proposed method, a very accurate (±0.5 meter) orthophoto of the area was used. The accuracy of the orthophoto is much better than the estimated accuracy of the GPS points which was computed to be ±3.2 meter by the GPS post processing software.

To assess the overall quality of our results we compared the coordinates of the intersections in the realigned map with those measured on the orthophoto. The quality assessment was computed from:

$$L1 = \frac{\sum_{i=1}^{n} d_i}{n} \tag{9}$$

Where:
n – is the number of checked intersection and

$$d_i = \sqrt{\left(X_i^{Map} - X_i^{ortho}\right)^2 + \left(Y_i^{Map} - Y_i^{ortho}\right)^2} \tag{10}$$

Another quality indicator, L2, was used as well. L2 is computed similarly to L1 except, it compares the original locations of the intersections with those on the orthophoto. Finally, we calculated the difference between L1 and L2 which is a measure of the improvement of the map correctness. The results of these computations are shown in table 1.

Intersection	L1 [m]	L2 [m]	L2-L1
21	5.43	12.42	7.00
22	1.25	18.90	17.65
33	2.95	9.21	6.25
136	2.54	11.58	9.04
158	6.82	20.54	13.72
200	18.01	29.23	11.22
169	2.05	16.12	14.06
174	4.37	6.27	1.90
176	9.92	31.09	21.17
186	20.28	31.80	11.52
189	1.98	26.26	24.29
201	1.98	11.11	9.14
202	4.90	17.74	12.85
206	7.65	5.27	-2.38
228	5.29	4.44	-0.86
229	0.58	12.79	12.22
230	4.45	16.09	11.64
231	7.22	19.30	12.08
232	3.58	31.16	27.59
233	1.15	17.12	15.97
234	3.63	11.27	7.64
236	6.88	15.30	8.42
270	5.30	2.76	-2.54
276	3.21	0.88	-2.33
291	7.78	17.53	9.75
292	2.86	18.34	15.48
295	2.97	7.57	4.60
296	2.83	19.20	16.37
302	2.00	20.38	18.37
303	2.27	16.69	14.43
309	4.23	11.82	7.60
311	3.83	6.12	2.29
324	5.76	39.83	34.07
	average		
	5.03	**16.25**	**11.22**
		Average improvement	**11.22**
		Std. Deviation [m]	**8.38**
		Std. Deviation of the mean [m]	**1.46**

TABLE 1. Comparison between the intersections location on the orthophoto and the location of the same intersections before (L1) and after (L2) realignment

It can be seen that on average, the location of the intersections improved by more than 11 meters compared to their original position. In general, intersections of segments with more GPS points yielded much better results compared to those with very few GPS points. In addition, at some intersections, GPS points were collected traveling one way but not in the opposite direction. Such uneven data collection pattern could cause a bias shift when fitting the generalized road segment to the GPS points. This is because the average of GPS observation while traveling on both sides of the road creates a better centerline compared to travelling only on one side of the road. Examples of intersections with too few GPS points or uneven GPS tracks are intersections 200, 176, 186, 231 and 236. The results even at these intersections were better compared to the original dataset but less accurate compared to well sampled intersections. In fact, when assessing the results only for intersections where GPS data was collected both ways, the average L1 improved to 3.46 meters while the average difference between L1 and L2 remained the same (11.20 meters). The average value of L1 (3.46 meters) corresponds well to the post processing determined GPS positioning standard deviation of ±3.2 meters. This indicates that the conceivable improvement was met.

Figure 6 shows a section of the original and realigned maps on the background of the high quality orthophoto. The light blue lines depict the original map segments, the orange lines depict the results of the realignment after the first iteration, the red lines are the results after the second iteration and the dark blue lines represent the final realignment results after the third iteration. Iterations are needed especially when the original map is way off compared to where it should be. If the original segments are grossly off, the first iteration will move the realigned intersection and segments closer to their corrected location but not enough. This is because the initial matching of the GPS points to grossly misplaced road segments could produce inexact results. Using iterations overcomes this problem because the second iteration uses the results from the first one which are closer to the GPS points compared to the original map. The impact of the iterations on the process is shown in figure 7(*a*). The first iteration (shown in orange) located the intersection to the right of its correct location. After the second iteration (shown in red) and the third iteration (shown in blue) the intersection was located where it was supposed to be.

As mentioned before, our proposed method (and any other method) requires that there are sufficient GPS points tracked for the road segments and preferably, while traveling in both directions. If too few GPS points are collected on a segment and only for one traveling direction, the resulting realignment could become less accurate as seen in figure 7(*b*). But, in general, figure 6, demonstrates how well our proposed method works. The dark blues lines are perfectly aligned with the street network of the orthophoto. One could also see how the dark blue lines fit the GPS points represented by small dots.

FIGURE 6. Example of the realignment results on the background of a high quality orthophoto.

(*a*) (*b*)

FIGURE 7. Example of the realignment results (*a*) the impact of iterations (*b*) the impact of spares GPS points.

SUMMARY AND CONCLUSIONS

In this paper we outlined a detailed method for realigning street segments of an existed road network using GPS data points. One of the objectives of the method was to realign the network not to recreate it from scratch. While there are several methods to recreate the network independently of the existing one, such an approach would require a tedious process of recreation of the non-geometric attributes of the dataset. Our approach was to keep the original dataset and just correct the geometric elements associated with each road segment. Our approach makes it easier to improve the network piecewise because untraveled segments are still represented from the original dataset.

Our proposed realignment method has six steps that could be iterated to achieve better results. It starts with pruning the GPS points and preparing the map for the realignment process. Next, the GPS points are matched to generalized road segments. Once the GPS points are associated with specific road segments, they are used to derive a new road segment that fits them best. From the GPS fitted segments we determine the new locations of the intersections and finally the geometric characteristics of the original road segments and subsequently those of the GPS tracks are restored.

Visual inspection of the results of our method against high accuracy orthophotos showed that the original network was thoroughly realigned to its real world position. The average improvement of the location of the intersections was over 11 meters. The average accuracy of the realigned map was found to be consistent with the GPS positioning accuracy. The more complete and accurate GPS coverage used, the better the results that can be obtained.

REFERENCES:

1. Bernstein, D., and Kornhauser, A. (1996). "An introduction to map matching for personal navigation assistants." .

2. Chen, W., Li, Z., Yu, M., and Chen, Y. (2005). "Effects of Sensor Errors on the Performance of Map Matching." The Journal of Navigation, 58(02), 273.

3. Douglas D. H. and Peucker T. K. (1973). Algorithms for the reduction of the number of points required to represent a line or its caricature. The Canadian Cartographer, Vol(2):112-122, 1973.

4. GPS tracks from Open Street Maps, http://www.openstreetmap.org/traces

5. Greenfeld, J. S. (2002). "Matching GPS observations to locations on a digital map." 81th Annual Meeting of the Transportation Research Board, Paper 02-3017, January 2002, Washington DC.

6. He, R. R., Kornhauser, A. L., and Ran, B. (2003). "Map Alignment Using Probe Vehicle Data." TRB 2003 Annual Meeting, .

7. Huabei Yin, and Wolfson, O. (2004). "A weight-based map matching method in moving objects databases." Scientific and Statistical Database Management, 2004. Proceedings. 16th International Conference on, 437-438.

8. Lili Cao and John Krumm. 2009. From GPS traces to a routable road map. In Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS '09). ACM, New York, NY, USA, 3-12.

9. Quddus, M. A., Ochieng, W. Y., Zhao, L., and Noland, R. B. (2003). "A general map matching algorithm for transport telematics applications." GPS Solutions, 7(3), 157-167.

10. Quddus, M. A., Ochieng, W. Y., and Noland, R. B. (2007). "Current map-matching algorithms for transport applications: State-of-the art and future research directions." Transportation Research Part C: Emerging Technologies, 15(5), 312-328.

11. Schoredl, S., et al. (2004), Mining GPS Traces for Map Refinement.Data Mining and Knowledge Discovery, 2004. 9(1): p. 59-87.

12. Stefan Edelkamp and Stefan Schrodl. (2003). Route planning and map inference with global positioning traces. In Computer Science in Perspective, Rolf Klein, Hans-Werner Six, and Lutz Wegner (Eds.). Springer-Verlag New York, Inc., New York, NY, USA 128-151.

13. White, C. E., Bernstein, D., and Kornhauser, A. L. (2000). "Some map matching algorithms for personal navigation assistants." Transportation Research Part C: Emerging Technologies, 8(1-6), 91-108.

14. Worrall, S. and E. Nebot (2007), Automated Process for Generating Digitised Maps Through GPS Data Compression, in 2007 Australasian Conference on Robotics & Automation. 2007: Brisbane, Australia.

15. Wu, D., Zhu, T., Lv, W., and Gao, X. (2007). "A heuristic map-matching algorithm by using vector-based recognition." 2007 International Multi-Conference on Computing in the Global Information Technology (ICCGI'07), IEEE, Piscataway, NJ, USA, 7.

BIOGRAPHICAL NOTES

The presenter is a graduate of the Technion Institute of Technology, Faculty of Civil and Environmental Engineering where received the BSc in 2006 and the MSc degree in 2010. He has been working at SOI (Survey of Israel) since 2012 and currently occupies the position of section director of plan quality validation.

The current paper has been presented and published in 2013 at TRB (Transportation Research Board) 92nd annual meeting at Washington DC, United States.

Contacts

Nicola Jammalieh
SOI (Survey of Israel),
Tel aviv, Israel
Email: nicola@mapi.gov.il
Phone number: +972-50-6222-261

Joshua Greenfeld, Ph.D. (**corresponding Author**)
Professor Emeritus New Jersey Institute of Technology, and
Faculty of Civil and Environmental Engineering
Israel Institute of Technology
Haifa, Israel
email: greenfeld@njit.edu
Phone number: +972-54-535-6433
Fax number +972-4-829-5708