Eurographics Conference on Visualization (EuroVis) (2014)
R. Borgo, R. Maciejewski, and I. Viola (Editors)

STAR – State of The Art Report

Bridge and terrain reconstruction in LiDAR point clouds

Timotej Kovač

Abstract

When LiDAR data of the terrain is being acquired some object might interfere with the surroundings in such a way that a lot of detail is lost. One type of these interference objects are bridges. Details of them are lost and they are in the way of capturing on the terrain bellow. In this paper we propose an algorithm to reconstruct the lost terrain bellow them and also their reconstruction using additional data. We discuss the results that this kind of an approach brings and what can be done to further improve it.

1. Introduction

In this paper we discuss a possible approach to reconstruct bridges and terrain undernthem. We also discuss how the additional data in the form of SHP files which were available to us from the e-Geodetski podatki web site proved effective in our algorithm. [shp20, ege20]. These contain geometries and some basic information about infrastructure such as bridges and natural resources such as rivers. We also show the results that we produced using some a simple interpolation approach, our own approach and the final result of the reconstruction process.

2. Implementation

For our tests we have chosen a very complex bridge example in order to make our approach as robust as possible. Our example consisted of two adjecent bridges that extended over a moving body of water at an angle that wasn't perpendicular to it. Both also extended over a large portion of the terrain.

Our implementation was done in Java with the help of libraries LASzip and GeoTools libraries which were used for reading the LAZ and SHP files. [las20, geo20].

2.1. Bridge reconstruction

Because of the nature of how the LiDAR data is gathered the surface bellow the bridge is missing. To reconstruct this part of the bridge we used the additional data about the bridges location and width to determine 4 keypoints that defined that bridge. In reality bridges can be more complicated and composed of multiple bridge segments so our approach was to determine the keypoints on the particular bridge segment and at the end these would be groupped together to form a final result. Using this approach we gained the x and y values of each of the defining vertices of the bridge and so had to interpolate the z coordinate. This was done using either finding enough low value points (based on the z component) that were classified as being part of the bridge. These also had to be in some determined range to be deemed valid. After that the z-component was calculated as an interpolation with weight in regards to distance from the point we wanted to interpolate and other points that matched the above criteria. Then we simply interpolated withing these four points the ones that defined the bridge bottom.

2.2. Terrain reconstruction

The majority of our effort was aimed towards reconstructing the surface underneath the bridge.

At the start we tried some basic approaches towards terrain reconstruction and got some insight into problems that we will have to solve. These were:

- how to generate points on this area to best reconstruct the terrain and
- how to deal with outliers, vegetation, power lines and object that are adjecent to bridges and don't represent the terrain.

Our approach can be seen on figure 1 and is described bellow.

1. Define the bridge polygon from the SHP file and temporary remove every point that is determined to belong to a bridge.
2. Generate values x and y along the bridge in such a way that they run parallel to the valley bellow the bridge.
3. Sample points from both sides of this line (terrain adjecent to the bridge marked with A and B). If there are too few points sampled then continue searching in the

© The Eurographics Association 2014.

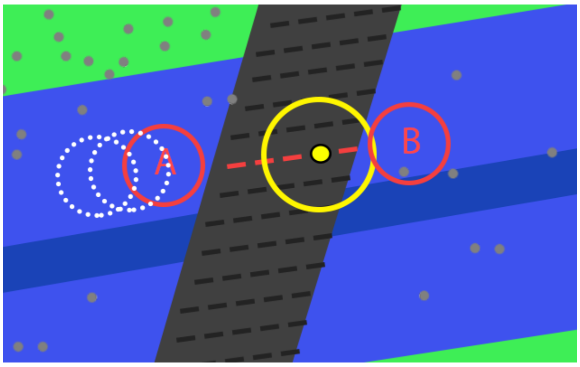

Figure 1:

same direction until some threshold (white dotted circle). If there are still not enough points to be found sample the surrounging area of the terrain that has already been completed (yellow circle).

4. Process the sampled points to remove any objects that aren't terrain.
5. Interpolate the z coordinate with distance as a weight on sampled points.

3. Results

Our first attempt at solving this issue can be seen as the red part of the figure 2.

Here we used a simple approach of a weighted nearest neigbour interpolation. As it can be seen reconstructed points do not follow the reality of the terrain well. The majority of points are interpolated to roughly the right height but the transitions between them are very harsh.

Our approach on the other hand gave us very good results in reconstructing the terrain under a bridge and can be seen drawn in blue in the same figure 2. The whole reconstruction process of the bridge and terrain on an example of two highway bridges produced the results in under half a minute, but this may vary depending on the system used and number and complexity of bridges. The cloud points before and after the reconstruction process can be seen in figures 3 and 4. In them the added points of the bottom bridge and the reconstruction of the bottom of the river under the bridge and its banks can be clearly seen.

4. Further work

One of the issues we didn't solve is that of saving the reconstructed data in LAZ format. We weren't sucessful in finding any free library that we could use to save the reconstructed data in the same LAZ format as the input file. That is why the system outputs a simple OBJ file which contains the reconstructed bridge points alongside with the original points inside of a certain circumference of the bridge.

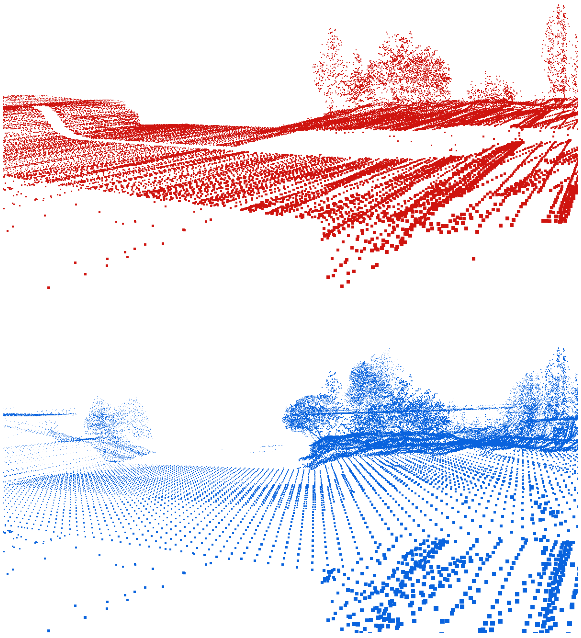

Figure 2: *Results of terrain reconstruction. Image in red shows the closest neighbour interpolation while the blue image shows selected group interpolation.*

Another issue we had and didn't manage to solve was that of bridge, road and river intersections. For our approach it was crucial that we detected which bridges, roads and rivers intersected where, so that we could paired them and calculated the normals used for terrain and bridge reconstruction. Roads were needed to get the width of the bridge, as that wasn't specified in the bridges SHP file. Because of that we had to manually open 3 different SHP files and gathered the required information out of each one. The road that is on the bridge and the bridge itself were in two different SHP files and even though they supposedly all use the D48/GK koordinate system the values didn't match up [d4820]. When trying to solve this by converting objects in their coordinate systems into a common one the problem still persisted. We approached this problem by manually extracting the IDs of bridges, roads and rivers from the different SHP files and then only searching among those. This isn't the best solution as manually extracting IDs is a tideous task but neverthelesss for our purpouses we went with this approach. If this was fixed however one would just need to uncomment the code for intesections and this problem would be solved, as we did implement the correct way of merging these various objects.

More of the infrastructure of terrain bellow the bridge could also be supported. For now only rivers have been sup-

© The Eurographics Association 2014.

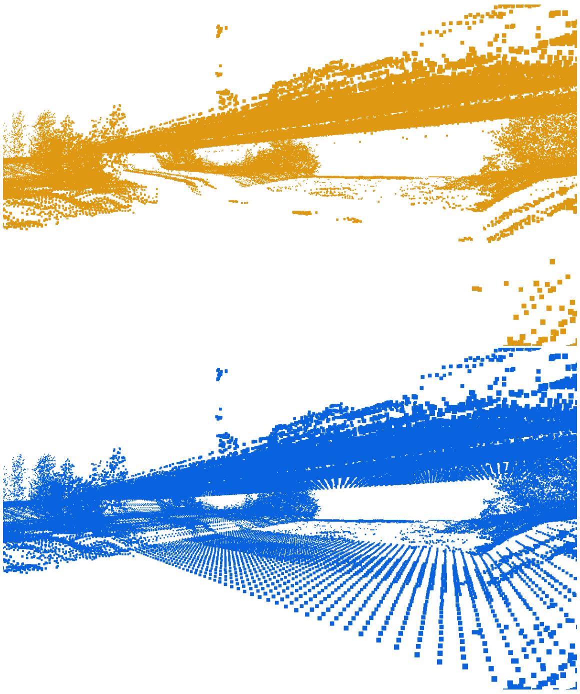

Figure 3: *Results of terrain and bridge reconstruction. Image in yellow shows the original data while the blue image shows the added reconstructed data.*

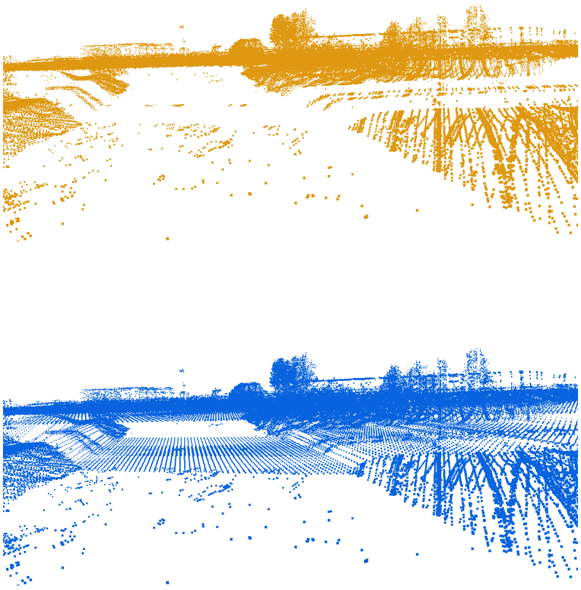

Figure 4: *Results of terrain and bridge reconstruction. Image in yellow shows the original data while the blue image shows the added reconstructed data.*

ported but also roads, railways or any other kind of infrastructure could be supported as well with the right SHP files and filtering attributes.

5. Conclusion

We were sucessful in implementing a rather robust approach for reconstructing bridges and terrain underneath. The reconstruction algorhitm is aware of the natural flow of the terrain and the points that are important for interpolation. Compared to simple interpolation methods our approach provided better results while still giving the result in an acceptable amount of time. Further work could be done improving this approach especially on reconstructing the bridge.

References

[d4820] E-prostor - Državni koordinatni sistem (D48/GK). Dosegljivo: https://www.e-prostor.gov.si/zbirke-prostorskih-podatkov/drzavni-prostorski-koordinatni-sistem/horizontalna-sestavina/drzavni-koordinatni-sistem-d48gk/, 2020. [Dostopano: 13. 5. 2020]. 2

[ege20] e-Geodetski podatki. Dosegljivo: https://egp.gu.gov.si/egp/index.html, 2020. [Dostopano: 13. 5. 2020]. 1

[geo20] GeoTools The Open Source Java GIS Toolkit. Dosegljivo: https://geotools.org/, 2020. [Dostopano: 13. 5. 2020]. 1

[las20] mreutegg/laszip4j: The LASzip library ported to Java. Dosegljivo: https://github.com/mreutegg/laszip4j, 2020. [Dostopano: 13. 5. 2020]. 1

[shp20] Shapefile - Wikipedia. Dosegljivo: https://en.wikipedia.org/wiki/Shapefile, 2020. [Dostopano: 13. 5. 2020]. 1

© The Eurographics Association 2014.