Supplementary Note to "Detecting Building-level Changes of a City Using Street Images and a 2D City Map"

Daiki Tetsuka Takayuki Okatani
Graduate School of Information Sciences, Tohoku University
okatani@vision.is.tohoku.ac.jp

Preprocessing: SfM and registration

As mentioned above, the estimation of building existences (the thick line box of Fig.4 of the main paper) is preceded by SfM followed by the registration of its result with the map. These preprocessings are summarized here.

We employ a standard approach for SfM. Feature points are first extracted using SURF [1] in each image of the sequence, and then they are matched based on descriptor similarity to obtain putative correspondences between each neighboring pair of images. The outliers are then discarded from them by using RANSAC with the five point algorithm [5], yielding a number of point trajectories and initial estimates of the camera poses. Their 3D positions are then estimated by triangulation, followed by the application of robust bundle adjustment.

It should be noted that we match feature points only between consequtive images in the sequence; thus, once a scene point ceases to be tracked (e.g., when it is out of image area), the same scene point will be treated as a new point if it has appeared in images again. If one uses an appropriate matching scheme, these points will be correctly matched and identified as a single scene point, which enables loop closure and then could improve the accuracy of reconstruction. We do not do so to avoid the resulting increase in computational cost. We perform SfM in an open-loop manner, thereby being able to have a highly sparse structure resulting in a small computational cost. This is important as we are considering the city-scale reconstruction. Moreover, the accuracy of camera trajectories is not particularly a problem in our case, as we have GPS data for each camera position. In brief, when the vehicle run the same place of a city multiple times, the same scene points can be reconstructed as different points. This will be appropriately considered in our method described later.

The SfM reconstruction thus obtained is then registered with the map by finding a similarity transformation such that the transformed camera positions and their GPS data are as close to each other as possible in the least squares sense. As the SfM reconstruction inevitablly suffers from drifts, we divide the whole sequence into a number of subsequences whose length is about 100 images, or equivalently, 200 meters in travel distance, for each of which SfM is performed. This also contribute to the reduction of the total computational cost. Not to overlook buildings at the both ends of each subsequence, the original sequence is divided so that the ends of the neighboring subsequences overlap with each other. An example of the SfM reconstruction after the registration with the map is shown in Fig.1.

In our 2D map of a city, each building is represented as a polygon approximating its ground projection. It is impossible to recover the original 3D shapes of the buildings. Thus, we assume that the polygons depict its outer walls of the first floor and also these walls have be at least five meters high. When comparing the SfM point cloud with the buildings, we will use only the points up to the five meters high above the ground level and neglect other points.

More images for the results shown in the main paper

In the main paper, we report the results of the proposed change detection method for three cities. Additional images omitted in the main paper are shown in Fig.11, 12, and 13. These show how the proposed method iteratively judges the existences of buildings. Note that Fig.11 is an enlarged version of Fig. 8.

Analysis of failure cases

In the experiments reported in the main paper (Table.1), the proposed method did not achieve 100% accuracy, and there are a certain amount of errors in the results. Failure cases are classified into two types, although their boundary is somewhat obscure. One is the case attributable to the limitation with the proposed method, and the other is the case attributable to the fundamental difficulty with labeling existence/non-existence to each building.

The former occurs mostly when the assumptions of the method are violated. Figure 14(a) shows examples. This image is captured in a backlight situation, which leads to a difficulty with the extraction of feature points, resulting in a failure. It is also seen in the same image that a building has lost its walls due to the tsunami or a fire caused by it. This leads to the same difficulty, which is also the case when walls do not have sufficient textures.

There are some buildings that are difficult even for human to judge their existence/non-existence, which could cause the second failure cases. Figure 14(b) shows an example, where only foundations of buildings remain after the tsunami. Figure 14(c) shows another example, a partially demolished building. Because of the nature of the disaster, there are many such buildings in the cities considered in our experiments. Whether they should be labeled as existing or non-existing will depend on application. In our experiments, we label all of these buildings as existing in the ground truths.

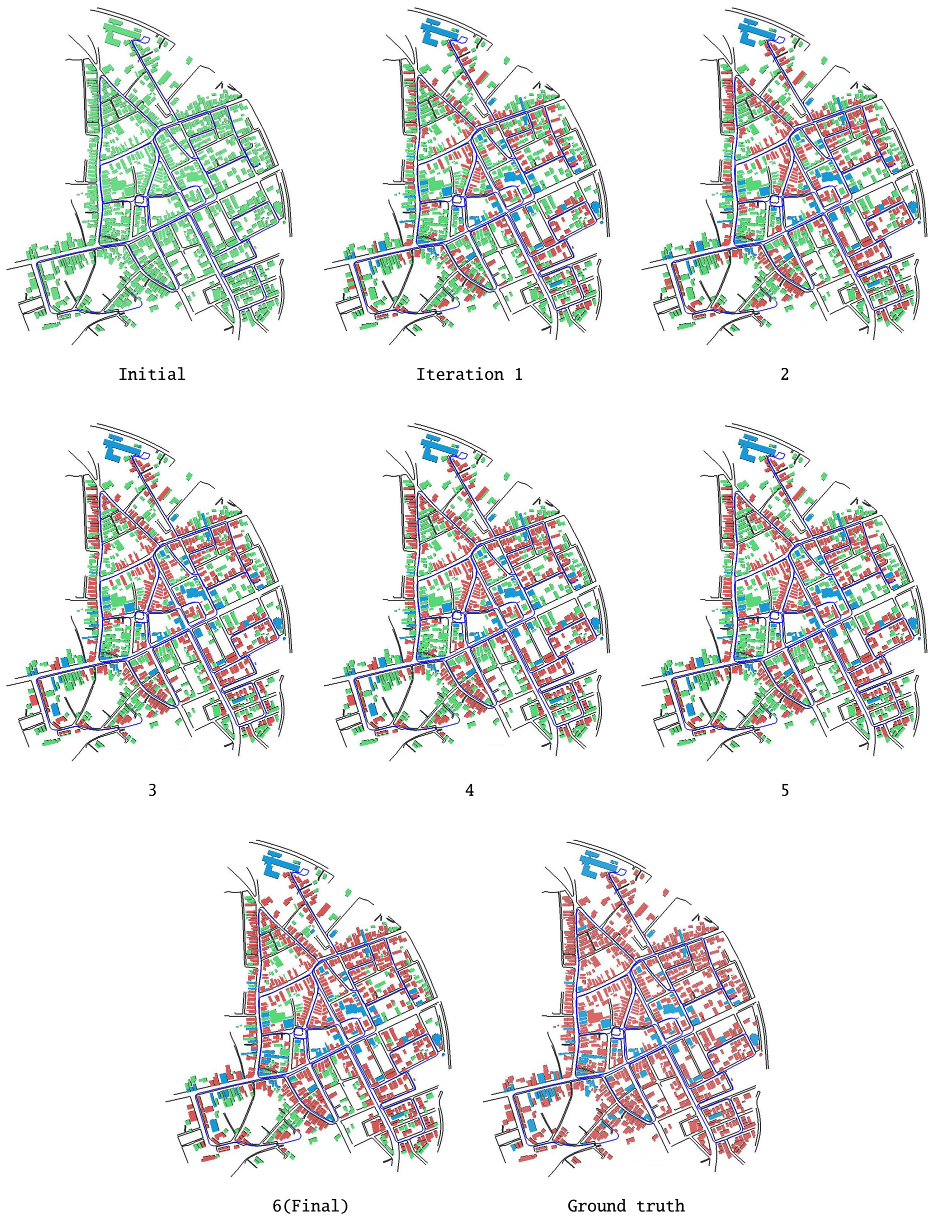

Figure 11. Intermediate results at different iteration counts for Otsuchi Town. (An enlarged version of Fig.8 of the main paper). Best viewed on a color monitor. Green, blue, and red polygons indicate the buildings that have not been judged yet, those judged to be existing, and those judged to be non-existing, respectively. The camera trajectory is displayed in dark blue.

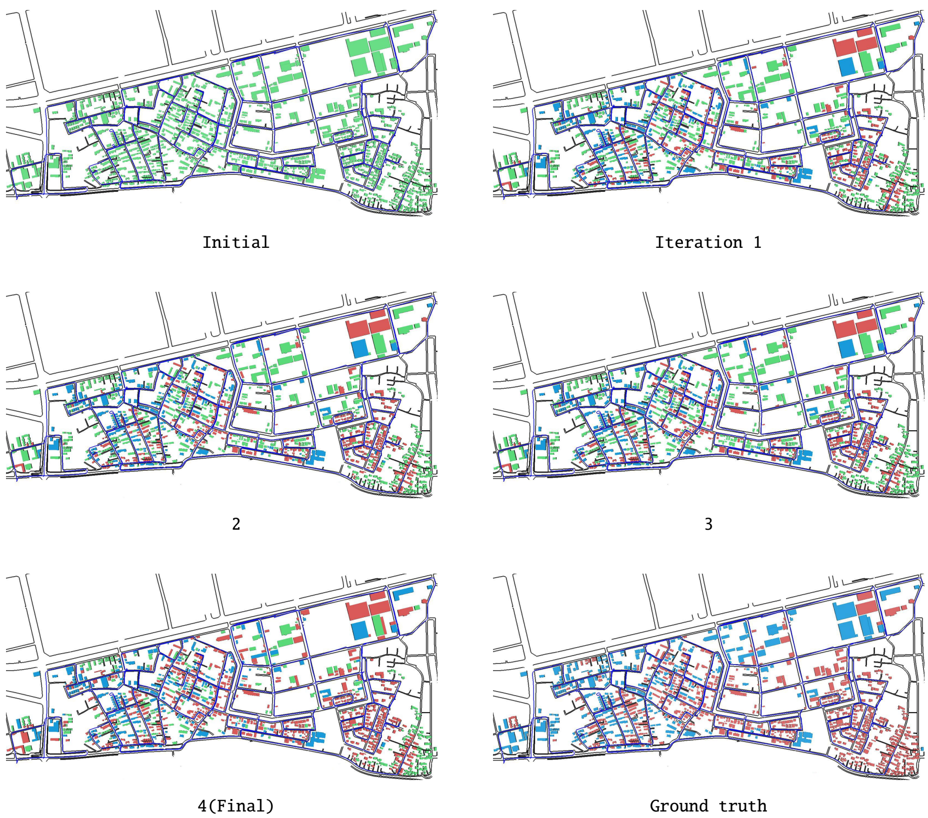

Figure 12. Intermediate results at different iteration counts for Miyagino Ward.

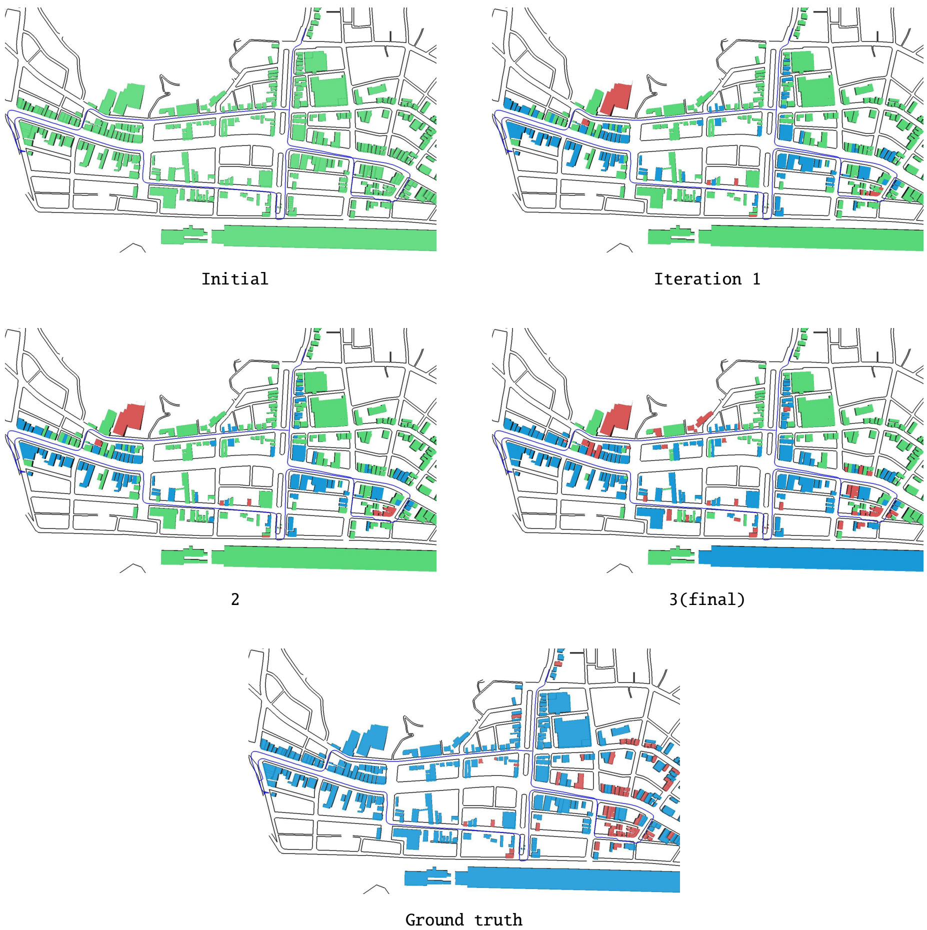

Figure 13. Intermediate results at different iteration counts for Kamaishi City.

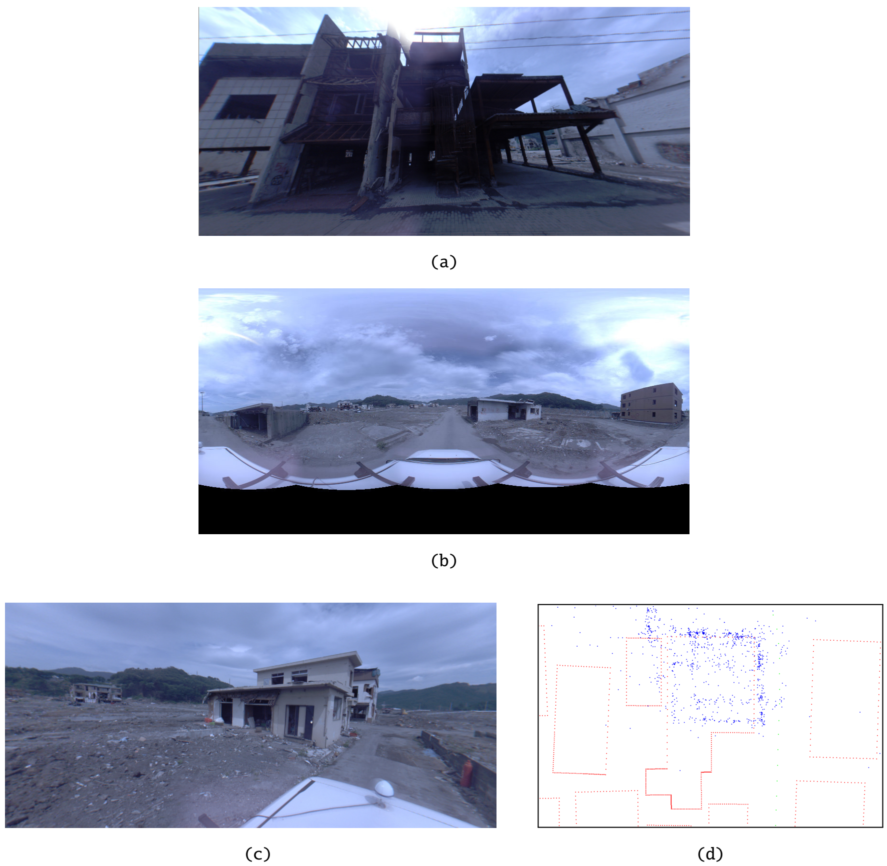

Figure 14. Example failure cases. (a) Backlight and buildings without wall. (b) Only building foundations remain. (c) Partially demolished building. (d) The reconstructed point cloud overlaid on its original buidling shape.