Large scale semantic 3D modeling of the urban landscape
Esteban Lopez, I.

Citation for published version (APA):
Esteban Lopez, I. (2012). Large scale semantic 3D modeling of the urban landscape
List of Figures

1.1 [LEFT] Monocular vision where backtracking a ray is not enough to identify the position of the object in space. [RIGHT] Stereo vision where backtracking two rays intersect in the object in space. 

2.1 LEFT: Pinhole camera model. Rays of light reflected from the real object pass through a small opening in the camera and intersect the imaging sensor. The image is therefore recorded flipped both vertically and horizontally. RIGHT: Camera as a central projection. Rays of light reflected from the real object are projected through the center of projection onto the imaging sensor.

2.2 LEFT: One axis of the world coordinate system intersects perpendicularly with the image plane at the Principal Point. In this setup, the origin of the image coordinate system coincides with the intersection point. RIGHT: In this setup, the Principal Point and the origin of the image coordinate system do not coincide.

2.3 LEFT: Barrel distortion. RIGHT: pincushion distortion.

2.4 World, image and camera reference system of our choosing.

4.1 3D reconstruction and modeling pipeline. The methods discussed in each of the steps are indicated.

4.2 Original (LEFT) and undistorted (RIGHT) images after performing camera calibration.
4.3 LEFT: Reconstruction through intersection of rays between camera center and matching image points. RIGHT: Rays spanning between the camera center and matching image points generally do not intersect in space. It is common to assume that the solution will lie on the shortest line connecting the two rays. 48

4.4 Matching of polygonal planes to a raw point cloud. 57

4.5 Textured set of polygonal planar patches. 57

5.1 The scale problem in monocular vision. Estimating the camera pose $C_1$ is intrinsically scale free. By sliding the camera over an unknown amount along the translation direction $t_{[0,1]}$, both structures $S$ become indistinguishable reconstructions, only the scale is different. If three cameras are considered, the local scale accounts for the ratio between the scales of the two pairs of cameras (there are three pairs of cameras, though only 2 of the scale ratios are linearly independent). The global scale accounts for the scale of real life objects. 62

5.2 Error on the estimation of the scale using the 4 proposed methods (2000 runs). Method 2 is the worst given that the motion on the $y$ direction does not contribute to the computation of $s$. Methods 1 and 3 offer similar performance, with a slight preference for method 1. Method 4 is slightly better than the rest. 71

5.3 Error on the estimation of rotation and translation of a third camera pose given 2 previous camera poses and the reconstructed structure using linear triangulation (2000 runs). TOP: error in the rotation calculated as absolute angle of $\Delta R$ in 5.7. The 1-point coincides with 1.5 because the rotation is calculated with the 8-point algorithm. MIDDLE: error in the translation vector calculated as the distance of the difference vector between ground truth and estimation. BOTTOM: error in the angle estimation of the translation direction. NOTE that 1-point (Least Squares) and 1-point (Optimal) are the same for translation and rotation accuracy. Moreover 1-point and P1.5P are the same for rotation accuracy. 78

5.4 Sample images of first dataset. 80
5.5 Top: Estimated trajectory (red circles) and recovered point cloud for first set. Bottom: reconstructed 3D model. No global Bundle Adjustment was necessary.

5.6 Top view of the reconstructed scene (in black) and camera positions (red circles).

5.7 Sample images of second dataset.

5.8 Estimated trajectory (red circles) and recovered point cloud for second set. The camera started at the right-bottom corner moving towards the right side and ending in the same position. This was ensured using the same image for first and last position. No global Bundle Adjustment was applied.

5.9 P11 dataset consisting of 11 images.

5.10 3D reconstruction of P11 benchmark dataset. No bundle adjustment was used and the scale between consecutive images was computed using our optimal algorithm.

6.1 Pipelines for Semantic Modeling of tightly packed cities using GIS, point clouds and aerial images.

6.2 Colored GIS data obtained from OpenStreetMap.org.

6.3 Palais de Rumine. BLUE: OpenStreetMap data. RED: Manually enhanced polygons.

6.4 Point cloud obtained from aerial images.

6.5 Point cloud obtained from ground images.

6.6 Aerial Image obtained from Google Maps.

6.7 Average error over 10 runs in the estimation of gravity as a function of the number of planar patches used.

6.8 LEFT: facade points project near the building outline. RIGHT: roof points project inside the building outline.

6.9 Representation of the error function for optimization.

6.10 Ground surface estimation process: TOP-LEFT. Height-map initialization. TOP-RIGHT. Large declination smoothing. BOTTOM-LEFT. Isolated patch removal. BOTTOM-RIGHT. Final smoothed ground height.

6.11 Validation of the estimated ground surface.
6.12 Three stage procedure for obtaining the height of facades. TOP-LEFT: distribution of point-facade distances over height and distance. TOP-RIGHT: exponential function used to grade points based on their distance to the facades. BOTTOM: resulting graded distribution. ........................... 105
6.13 Summed up distribution of the height of a single building (BLUE) and derivative (RED). .......................................................... 106
6.14 Estimation of the height of facades for 3 different buildings. ............ 106
6.15 Comparison between estimated facade height and real building. ........ 107
6.16 LEFT: The aerial image and the outline of the polygon of a building is shown. Red crosses represent the Harris corners found in the image. Blue circles are the corners that lie inside the polygon. RIGHT: triangulated vertices. ...................................................... 109
6.17 Top view of the registered sources of information. Green shows the aerial point cloud. Red shows the ground based point cloud. Blue shows the GIS data. .............................................................. 111
6.18 Top view of the textured ground plane and the GIS outlines of buildings. 112
6.19 Google image used for texturing the estimated ground surface. ........... 113
6.20 Views of the model of the city center of Lausanne where the height of the buildings has been estimated. ................................. 114
6.21 Area: Model of the city of Lausanne, color is assigned to buildings based on the area they occupy, from smaller (blue) to bigger (red). ....... 115
6.22 Height: Model of the city of Lausanne, color is assigned to buildings based on their height, from smaller (blue) to bigger (red). ............... 115
6.23 Volume: Model of the city of Lausanne, color is assigned to buildings based on the volume they occupy, from smaller (blue) to bigger (red). .. 116
6.24 Two views of the complete model of the city of Lausanne. Texture has been applied to the rooftops and the ground surface. .................... 117

7.1 3D modeling pipeline. Some well known alternatives for each step are pointed out. ................................................................. 120
7.2 LEFT: distorted image of a street where lines that are straight in real life can be seen. Three of those straight lines consisting of three points are selected and can be seen in red. RIGHT: the selected lines are used to compute the radial distortion parameters which are then used to undistort the original image resulting in the undistorted image of the street.

7.3 TOP: 4 input images of a mountain view in Switzerland. BOTTOM: stitched images after computation of the image-to-image homography. Vertical lines visible on the right side of the image are caused by two factors: slight differences in the automatic camera setup due to changing conditions when measuring the light in every image; the camera translating slightly on every image instead of only rotation.

7.4 TOP: 3 views of a street out of 57 images taken the camera roughly pointing towards the buildings. BOTTOM: Top view of the camera positions (in red) and reconstruction of the SIFT features used to compute the poses.

7.5 Black pyramids: representation of the position of cameras 1, 2 and 3 before scale adjustment. Grey pyramid: camera 3 after scale adjustment. Firstly, the relative displacement between cameras 1 and 2 is computed using, for instance, the 8-point algorithm. This yields a scale free motion of camera 2 with respect to camera 2. Then the displacement of camera 3 with respect to camera 2 is calculated. This yields a scale free displacement represented as the black pyramid in camera 3. Those two camera displacements are on a different scale. We set the reference scale as 1 for the first camera displacement and the scale of the second camera displacement between cameras 2 and 3 is computed, resulting in the scale adjusted position of camera 3 represented as the grey pyramid.

7.6 Reprojection error (as the sum of squared distances between the reprojected points and the original image features) during local refinement. After only two iterations the error is close to a minimum, making the procedure very efficient.
7.7 TOP: Images used to obtain a scaled consistent motion. BOTTOM-LEFT: resulting 3D structure before optimization. BOTTOM-RIGHT: resulting structure after global optimization. The originally estimated position of the cameras (before optimization) was altered to produce a noisy pointcloud. .................................................. 129

7.8 TOP-LEFT: Colored point cloud. TOP-RIGHT: Set of planar patches found with the RANSAC approach. BOTTOM-LEFT: Colored set of planar patches. BOTTOM-RIGHT: Dense reconstruction using PMVS2 130

7.9 A few undistorted views of a street. .................................................. 130

7.10 Triangulated SIFT features and scale consistent camera trajectory. The images are shown in the computed position. ................................. 131

7.11 TOP: Colored planar patches. BOTTOM: Textured patches. ........... 132