Improving correlation-based DEMs by image warping and facade correlation

Christophe Vestri
ISTAR & INRIA/Robotvis
06560 Sophia-Antipolis, France
vestri@istar.fr

Frédéric Devernay
I3S/IMAGES, Université de Nice
Sophia-Antipolis, France
devernay@unice.fr

Abstract

We present in this article a system which improves a DEM. The method reconstructs all the facades of the building and then, corrects the initial DEM by deleting the points of the roofs which pass over the boundaries defined by the facades. We correct the shapes of the buildings with the initial photographs to have sharp contours. The proposed approach does not use any a priori information about the orientation of the facades and the shape of the buildings. We present results with synthetic and real images.

1 Introduction and related work

This paper presents a system to improve a Urban DEM (Digital Elevation Model) made by correlation-based stereo from multiple aerial photographs. This system uses the same images to correct the shape of the building *a posteriori*. In aerial photography, a correlation algorithm will usually succeed in the reconstruction of the horizontal features of the scene (roofs and ground) because they are fronto-parallel surfaces. But the same algorithm will fail with vertical features (facades) which are very steep surfaces of this scene. Without any assumption on the shape of the building, we propose to reconstruct all its facades by using warped images. Then, we improve the DEM by removing the points of the initial building which are over the facade boundaries.

We first compare our technique with some of others methods and particularly those which use images of facades to verify, construct or refine a building model. A first class of methods is designed for large-scale applications as cartography or telecom. These methods mainly use top views of the scene as aerial or satellite images, or aerial laser-scanning. Feature-based methods use hierarchical grouping and matching to reconstruct the model by using features, extracted from the images, of increasing complexity [1, 5]. Nevatia uses walls and shadows for verification [6]. Adaptive window correlation methods [4] work well on extracting roofs and ground, but they assume that depth discontinuities are materialized by image contours. As the area-based methods, they cannot use facade images.

The second class of methods is mainly designed for computer graphics, virtual reality or architecture. These methods aim at reconstructing an accurate building model (both geometrically and visually) and use mainly ground photographs. Coorg and Teller use the space-sweep algorithm in [2] to recover horizontal lines from the facades, then they reconstruct the model of the building and extract the median texture. In [3], Debevec *et al*. construct a coarse model by hand, then they warp the initial images with respect to the coarse model and refine the model by applying a simple correlation-based stereo algorithm. These two methods are closer to our system than the first class of methods that uses aerial images as we do. We warp images so that facades appear similar in a stereo pair, but we use an incidence counting technique to avoid the use of an initial building model.

We describe our system in Section 2, and explain in Section 3 the important parts of our system. Results are presented with synthetic views and with aerial photographs over Berlin.

2 Motivation and overall strategy

Our global strategy to model building or a set of buildings is composed of three main stages. The first stage is the construction of a dense and reliable DEM using a classical correlation-based stereo method. The second stage is the segmentation of this DEM into locally planar surfaces. The objective of this second stage is to describe the scene with surface patches which correspond to the various facets of the buildings. The third and last stage is the vectorization of the boundaries of each surface patch to obtain the final model of the buildings. Errors in the DEM will have a large impact on the segmentation and vectorization stages. The system we propose in this article corrects boundaries of the buildings on the DEM to make easier the two last stages of our global strategy.

Facades represent an important surface in the different image, especially near the image boundaries. Because they

are steep surfaces of the scene, they appear very distorted in the images and a simple correlation-based stereo algorithm cannot recover these vertical surfaces. Besides, because of the difference in illumination between the roofs and the ground (especially in shadows), the correlation-based stereo will enlarge the roofs in these areas. This paper presents a system that uses the facade images of a building, taken into the aerial photographs, to improve the initial correlation-based DEM. The framework of our system is: first, we construct new images of facades wherein a correlation-based algorithm can easily match facades and reconstruct them with a good geometric accuracy. Then, we correct the erroneous points of the DEM which pass over the facade boundaries.

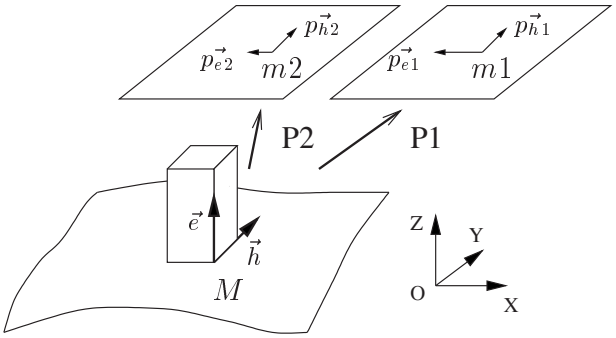


Figure 1. Choice of images: $\vec{e}$ is the vertical direction and $\vec{h}$ is the orientation of the facade in the horizontal plane OXY. $\vec{p_{ei}}$ and $\vec{p_{hi}}$ are the projections of $\vec{e}$ and $\vec{h}$ at M in image i.

Let us suppose first that we know the orientation in the scene of the facades we want to extract (figure 1). The different stages of our system are the followings:

1. *Choosing two views of the facade*. Because facades are not viewed in the same way in all images, we need to choose two images using some criterion (cf. 3.1).

2. *Constructing the images of facades*. These images are warped images of the chosen couple of images. The facades have the same visual aspect on these images (cf. 3.2).

3. *Matching by correlation and reconstruction*. We apply a simple correlation-based algorithm to match and reconstruct the 3D points of the facades.

4. *Filtering out the erroneous 3D points*. (cf. 3.4).

5. *Improving the DEM*. Finally, we use the 3D reconstructed points (for the defined orientation of facade) to improve the DEM. We simply dig in the DEM where we reconstruct a point (cf. 3.5).

To treat all the facades of a building, we use this algorithm on all the possible facade orientations in the scene, thus making no assumption on the building shape. A subsampling of the orientation space by 20^o is enough to recover all the facades. This system is not dependent on the number of available images, the orientation of the facades, or the shape of the buildings.

3 Implementation

All the views of the scene are supposed to be calibrated. The initial DEM is made by fusion of several DEMs, each of them corresponding to the matching of two adjacent views. The list of image couples will be later used to conduct the choice of images for a given orientation.

To keep the computational time low, the whole process is only applied to the neighborhood of a building: each building or group of adjacent buildings can be automatically extracted from the raw DEM. We now present the implementation of each stage of the algorithm. This algorithm works on a given facade orientation: if there is a true facade for this orientation, it extracts the points of the facade, filters outliers, and improves the DEM. The whole system uses this algorithm with all possible facade orientations, sampled by 20^o and improves the DEM incrementally.

3.1 Choosing the images for a given facade orientation

First, we select two images from the list of image couples available, where the facades are best viewed. We tried two criteria: (1) the images where the projected surface of the facade is of maximum size, (2) the angle between the normal direction to the surface of the building and the direction toward the nadir of the images (vertical projection of the optical center on the ground) is minimum (the view must be in the axis of the facade).

We define two 3D unit vectors, the elevation vector $\vec{e}$ (orientation of the vertical) and the orientation vector $\vec{h}$ (orientation of the facade in the horizontal plane), as shown in figure 1. These two vectors are defined on a 3D point M taken on the ground and in the middle of the DEM of the building. This couple of vectors corresponds to a unit facade in the 3D scene. We compute the two 2D vectors $\vec{p_{ei}}$ and $\vec{p_{hi}}$ in each image i which respectively correspond to the projections of our couple of 3D vectors $\vec{e}$ and $\vec{h}$. $\vec{p_{ei}} = (x_{ei}, y_{ei})$ represents the orientation of the vertical at m_i in image i and $\vec{p_{hi}} = (x_{hi}, y_{hi})$ represents the orientation of the facade at m_i in the same image.

We define our two criteria for an image i as

$$A_i = \|\vec{p_{ei}} \times \vec{p_{hi}}\|, \quad C_i = \frac{\vec{p_{ei}} \cdot \vec{p_{hi}}}{\|\vec{p_{ei}}\| \|\vec{p_{hi}}\|},$$

where $\times$ is the cross-product between two vectors. The A_i measure must be maximum and C_i, which corresponds to the cosine of the angle of the two vectors $\vec{p_{ei}}$ and $\vec{p_{hi}}$, must be minimum. We tried each of these criteria and the combination of both to select the image couple. Best results were found using only the A_i criterion. We first select the image i corresponding to the maximum of A_i. Among all the image couples which contain this image, we select the one with the maximum A_j for the other image j. This ensures that the two images are adjacent.

3.2 Construction of the new images

In this step, we want to construct a pair of rectified images where the facades of a given orientation have the same aspect for simplifying their match. The area of interest (our building) in each image is defined as the projection of a rough envelope of our building, computed from the raw DEM. The deformation of the facade was induced by the projection, each projection inducing a different deformation of the same facade. Constructing images where facades have the same aspect can be achieved by compensating this deformation, as shown in the figure 2.

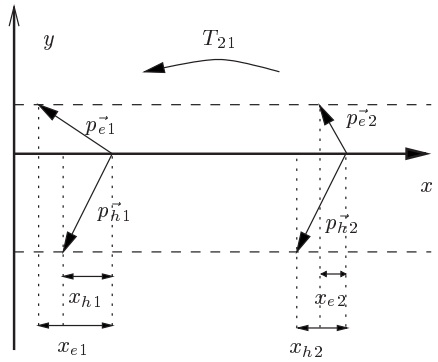


Figure 2. Facade transformation: $\vec{p_{e1}}$ and $\vec{p_{h1}}$ are the projections of $\vec{e}$ and $\vec{h}$ at M in the rectified image 1. $\vec{p_{e2}}$ and $\vec{p_{h2}}$ correspond to their projections in the rectified image 2. The facade transformation T_{21} from image 2 to image 1 consists in warping image 2 so that $(\vec{p_{e2}}, \vec{p_{h2}})$ maps to $(\vec{p_{e1}}, \vec{p_{h1}})$.

Because the images are rectified, the projections of 3D vectors $\vec{e}$ and $\vec{h}$ in each image have the same components in the y-axis. We consider these vector projections to be locally constant, since we are working at the scale of a building in an aerial photograph (the projections are locally linear). To get the same visual aspects for facades with this orientation in both images, we apply the affine transform T_{21}, called *facade transformation*, which maps $(\vec{p_{e2}}, \vec{p_{h2}})$ to $(\vec{p_{e1}}, \vec{p_{h1}})$ (it is uniquely defined this way). Since both images are rectified, this transformation only modifies the x coordinate: the epipolar constraint is respected.

We also studied the possibility to compute a transformation that would apply to the whole image instead of the linearized version described before. Unfortunately, the research of such a transformation is equivalent to solve a PDE (Partial Differential Equation) which is not integrable. Working locally is equivalent to linearizing this function.

To avoid resampling twice each image, we do not construct rectified images. From initial views, we apply successively the rectification and the transformation of facade. The second transformation is only applied to one image (fig. 2). We choose the other image as the one that has the maximum A_i (area criterion) score, so that the facade transformation expands the image rather than stretching it.

3.3 Reconstruction of the facades

Figure 6 presents results of the matching process. Images A, B and C show the matching process with images that are rectified in the classical way. Images A, D and E present the results of the same matching process after facade transformation. In the second case, the disparity surface which corresponds to the right facade is almost constant. The reconstructed 3D points of the facade will be more dense and accurate. We see in figure 6 that we are able to match the points of the facade, but many matches in the disparity map are erroneous data and must be rejected.

3.4 Filtering the reconstructed points

The filtering stage is composed of three passes: a filter on each disparity map using the normal vector to the 3D surface, a 2D filter using an accumulation map, and a 3D filter which takes into account the shape of the surface.

First, in each disparity map, we compute the normal vector to the surface for each reconstructed 3D point by plane fitting on its neighborhood (the neighborhood is considered in the disparity image space). If the normal vector orientation is too far from the chosen facade orientation, the corresponding 3D point is discarded (the threshold is 15^o). The normal vector is then reprojected in the horizontal plane of the 3D space (we constrain facades to be vertical).

The next filter uses an accumulation map, which is referenced in the same cartographic coordinates as the DEM. The reconstructed points are projected onto this map, and 3D points which correspond to the same facade will accumulate at the same location in the map. The grid size is about twice the resolution of initial images, and points with an accumulation lower than 2 are discarded.

The last filter derives from the ideas of the perceptual grouping. An interesting approach for surface reconstruc-

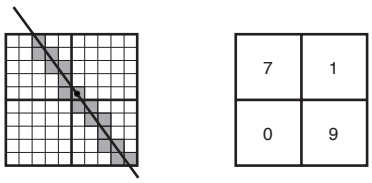


Figure 3. Filtering the points: The left array presents the voxels (large traits) and sub-cells (thin traits). The segment represents an hypothesis of facade. Grey cells are the selected cells by the Bresenham's algorithm. The right array presents the count of vote in each voxel.

tion from sparse data is described in [7]. Our 3D filter is a more simple system based on a vote filtering which gives priority to the vertical surfaces. We construct a 3D array in cartographic coordinates. The voxel size is 1m in X and Y and 3 meters in Z.

Each of the 3D points is linked to its corresponding voxel, and the point votes for its facade neighborhood. The facade neighborhood is defined as a 3D rectangular patch centered at the 3D point and oriented by its normal vector. The voting system uses a subsampling of the voxel space at a 20cm resolution, because the resolution of the 3D array is not enough for computing a representative vote: we want a facade which passes through a corner of the voxel to have a less important vote than a facade which passes in the middle of the voxel.

For voting along the facade, we first create an 6-meter long horizontal segment centered at the 3D point and vote for the voxels where it passes through. We use the Bresenham's algorithm to find all the sub-cells of the voxels where the segment passes over. Then, we replicate the vote along the segment to upper and lower voxels in the 3D array (2 upper and 2 lower voxels). The weight of the vote (fig. 3) is simply the number of sub-cells marked by Bresenham algorithm which are inside the cell. We keep the 3D points for which the corresponding voxel has three 3D points inside, and which have a minimum number of 300 votes.

3.5 Improving the DEM

To improve the initial DEM, we use the fact that no 3D point can physically be located between a reconstructed point and the optical centers of the views that we use to reconstruct this point. We compute optic rays for each point and "drill" the DEM from the reconstructed point to the optical center (fig. 4). This method offers the advantage to preserve the independence with respect to the surface model or the building model. This system can use several "drilling tools": we used a prism with a 1.5m-diameter circular vertical cross-section. The DEM improvement is done incrementally for each facade orientation.

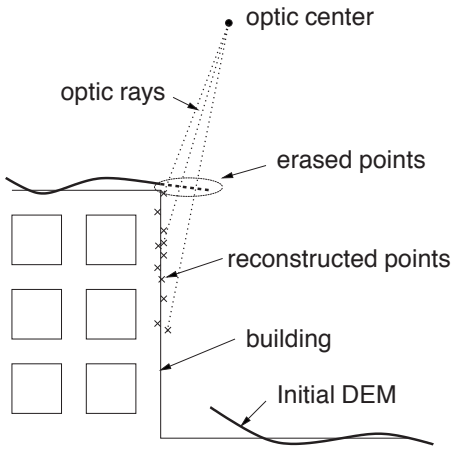


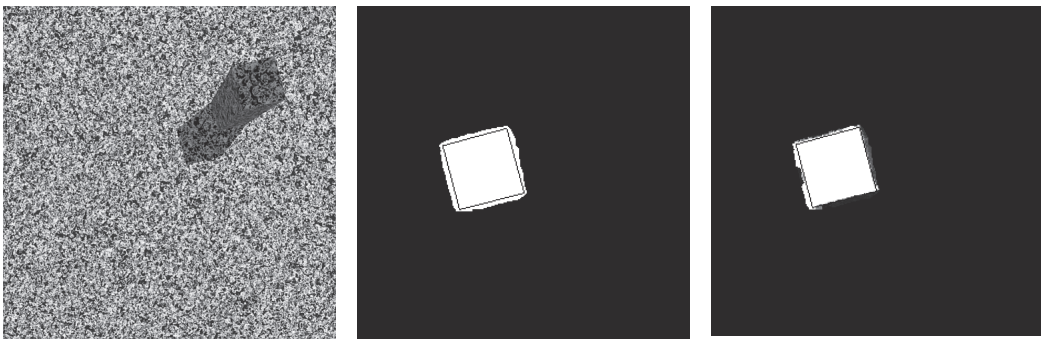
Figure 4. Improvement of the DEM.

4 Results

First, We present results with synthetic images in the figure 5. Image A is one of the nine views of a simple textured scene: (1) an horizontal plane to represent the ground, (2) a box to represent a building and (3) a light to represent the sun. Image B is the DEM automatically built using ISTAR's high-resolution DEM production line: correlation-based stereo was performed on each pair of consecutive images, and the resulting DEMs were merged using a robust method. We see in the image that the building is out the real boundaries. These errors appear in each stereo pair, and thus cannot be removed by the robust DEM merging. We present the improve DEM in image C. Because of the shadow, the algorithm failed in recovering the west facade. The correlation-based stereo algorithm need a surface with a sufficient textured for the matching. The three other facades are recovered and corrected.

For the real case, the results were obtained using 1:15000 scale aerial images of Berlin, Germany, with a 80/80% overlap and a focal length of 15/23. A given point is seen at least on 20 images. The images were scanned at $25\mu m$, which makes a ground resolution of 37.5cm. The 50cm DEM was built using ISTAR's high-resolution DEM production line.

Because of the difference in contrast between the roof and the ground (especially in shadows), we can see on this DEM (A in figure 7) that the buildings appear bigger than they are. We present the improved DEM in the image C, the algorithm is able to efficiently recover the main parts of the vertical surfaces. In spite of an important amount of noise,



A B C

Figure 5. Results on a synthetic building: A is one of the nine synthetic views of the scene. B is the DEM built using high-resolution DEM production line. The roof is out of the real boundaries of the building (thin traits). We present the results after improvement in the image C, grey areas were removed from the DEM. We recover three facades of the building.

the filter is good enough to keep all the relevant data, but fails to detect an outlier in the middle of the building on top of the image. In both cases, with synthetic and real views, we showed that our system is able to correct the shape of the building, and to recover the real boundaries of the building.

5 Conclusions and Perspectives

We presented a system which improves a correlation-based DEM by correcting the boundaries of the buildings to have sharp contours. The strategy consists in reconstructing 3D points on the building facades, and erasing the points of the roofs which pass over the facade boundaries by carving the DEM. We use warped images and a standard correlation-based stereo method to reconstruct the 3D points on the facades. These warped images have the same visual aspect for a given orientation, so the matching process will give accurate and dense 3D points on the facades. This is done for every facade orientation, and the DEM is then improved using the reconstructed and filtered points.

This method is mainly inspired by work on accurate building reconstruction from ground images [2, 3], but the method only uses the aerial images the DEM was built from. The method succeed in the dense reconstruction of the vertical features (facades) of the scene by using warped images of facades. The system offers the advantage to use no a priori information about the orientation of the facades and the shape of buildings.

We showed that our system is able to recover the boundaries and the shape of the building with accuracy on a synthetic and a real case. Nevertheless, there are still some errors which could be removed by further improving the filtering stage. We are also working on using facade locations and facade orientations extracted from the initial DEM to apply our system only on these orientations in these locations, in order to decrease complexity and computational time.

References

[1] R. Collins, C. Jaynes, Y.-Q. Cheng, X. Wang, F. Stolle, H. Schultz, A. Hanson, and E. Riseman. The umass ascender system for 3d site model construction. In *RADIUS: Image Understanding for Imagery Intelligence*, pages 209–222, 1996.

[2] S. Coorg and S. Teller. Automatic extraction of textured vertical facades from pose imagery. Technical Report 729, M.I.T. Laboratory for Computer Science, 1998.

[3] P. Debevec, C. Taylor, and J. Malik. Modeling and rendering architecture from photographs: a hybrid geometry- and image-based approach. In *SIGGRAPH*, pages 11–20, New Orleans, Aug. 1996.

[4] T. Kanade and M. Okutomi. A stereo matching algorithm with an adaptive window: Theory and experiment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(9):920–932, Sept. 1994.

[5] D. M. McKeown, G. E. Bulwinkle, S. D. Cochran, W. A. Harvey, C. McGlone, J. McMahill, M. F. Polis, and J. A. Shufelt. Research in image understanding and automated cartography: 1997-1998. Technical Report 98-158, Carnegie Mellon University, Computer Science Department, 1998.

[6] R. Nevatia and A. Huertas. Knowledge-based building detection and description: 1997-1998. In *Proceedings of the ARPA Image Understanding Workshop*, Monterey, California, 1998.

[7] C.-K. Tang and G. Medioni. Inference of integrated surface, curve, and junction descriptions from sparse 3d data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1206–1223, Nov. 1998.

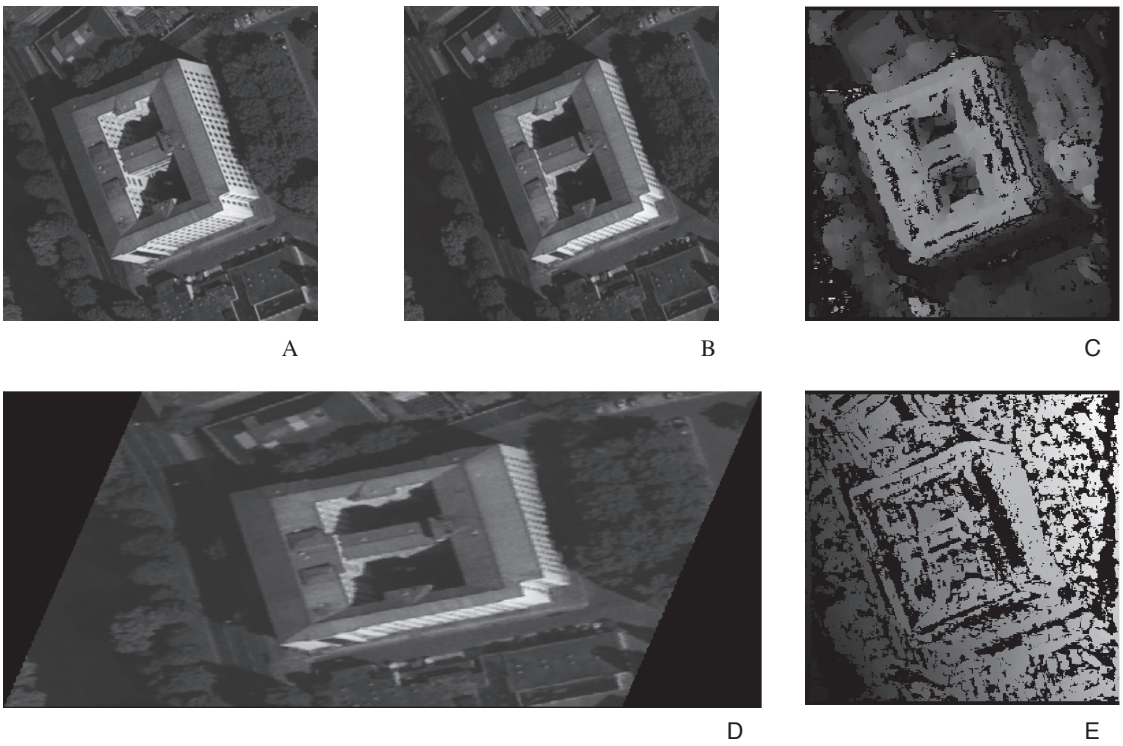


Figure 6. Correlation of facades: A **and** B **are rectified images of the building.** C **is the disparity map, with** A **as reference, obtained by a correlation-based matching process. The roofs and ground are recovered correctly, whereas the right facade has a stair shape (it should be planar).** D **is** B **warped by the facade transformation. The right facade in** D **has the same visual aspect as in** A**. In the corresponding disparity map (**E**), the disparity on the facade is almost constant. Image resolution is 37.5cm.**

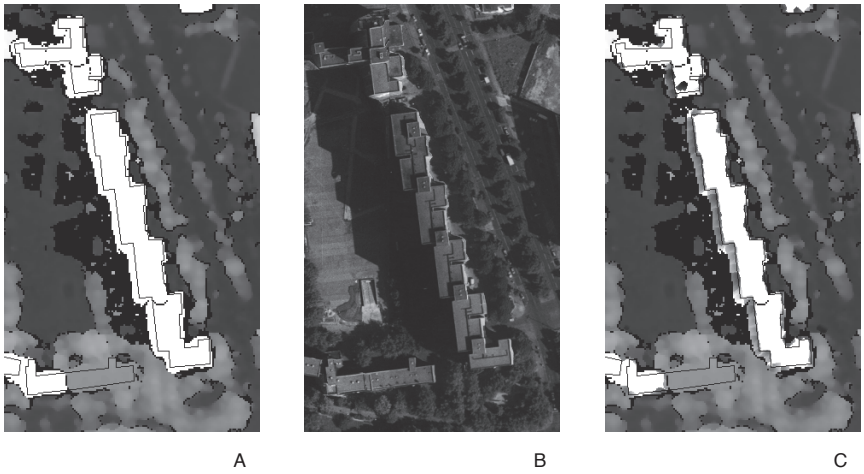


Figure 7. Results on real buildings: A **is the initial DEM of the building.** B **is the corresponding ortho-image. In** A**, the roof passes over the boundaries of the building (thin traits).** C **shows the result after improvement, grey areas were removed from the DEM, and we recover the real boundaries of the building.**