CONTEXT-BASED URBAN TERRAIN RECONSTRUCTION FROM UAV-VIDEOS FOR GEOINFORMATION APPLICATIONS

D. Bulatov [a]*, P. Solbrig [a], H. Gross [a], P. Wernerus [a], E. Repasi [a], C. Heipke [b]

[a] Fraunhofer Institute of Optronics, System Technologies and Image Exploitation (IOSB)
(dimitri.bulatov@iosb.fraunhofer.de)
[b] Institute of Photogrammetry and GeoInformation, Leibniz Universität Hannover
(heipke@ipi.uni-hannover.de)

KEY WORDS: UAVs, Photogrammetry, Sensor Orientation, Urban Terrain Reconstruction

ABSTRACT:

Urban terrain reconstruction has many applications in areas of civil engineering, urban planning, surveillance and defense research. Therefore the needs of covering ad-hoc demand and performing a close-range urban terrain reconstruction with miniaturized and relatively inexpensive sensor platforms are constantly growing. Using (miniaturized) unmanned aerial vehicles, (M)UAVs, represents one of the most attractive alternatives to conventional large-scale aerial imagery. We cover in this paper a four-step procedure of obtaining georeferenced 3D urban models from video sequences. The four steps of the procedure – orientation, dense reconstruction, urban terrain modeling and geo-referencing – are robust, straight-forward, and nearly fully-automatic. The two last steps – namely, urban terrain modeling from almost-nadir videos and co-registration of models – represent the main contribution of this work and will therefore be covered with more detail. The essential substeps of the third step include digital terrain model (DTM) extraction, segregation of buildings from vegetation, as well as instantiation of building and tree models. The last step is subdivided into quasi-intrasensorial registration of Euclidean reconstructions and intersensorial registration with a geo-referenced orthophoto. Finally, we present reconstruction results from a real data-set and outline ideas for future work.

1. INTRODUCTION AND PREVIOUS WORK

1.1 Motivation and problem statement

The technical equipment of the miniaturized unmanned aerial vehicles (MUAVs) has experienced a tremendous progress in recent years: historically, UAVs were simple remotely piloted drones, but autonomous control and capability to carry out pre-programmed flight plans is increasingly being employed in UAVs. In the same measure, the quality and diversity of the optical sensors such UAVs may carry onboard is becoming higher and more competitive to the conventional kinds of passive sensors, such as large-scale aerial imagery. The main advantages of small, cheap, easily usable aerial vehicles lie in cases of either time-critical or local, but rather detailed exploration of scenes. In the first case, UAVs are more suitable than the aerial imagery because of a significantly lower time needed to launch a mission, while in the second case, the mission can be performed on a lower cost.

Several possible applications of UAVs reach from civil engineering and urban planning over surveillance and rescue tasks to reconnaissance missions for defense research. In our previous work, (Solbrig et al. 2008), we discussed only 2D applications on local area exploration from UAV-based videos. These applications included mosaicing, detection of moving objects as well as annotation of objects of interest in a video. The core of the procedure was a real-time oriented geo-referencing of a video stream on an orthophoto. A 2D transformation that links a pixel of a video frame and the orthophoto is called homography.

However, in the case of a relatively low sensor altitude and a moderate focal length – needed in order to achieve a satisfactory resolution of the acquired images and videos – the presence of buildings and vegetation cannot be interpreted as a disturbing factor any longer. As a consequence, it is important for urban terrain modeling, to extend the existing concepts by 3D or, at least, 2.5D component. The counter-part of homography in scenes with spatial depth is called depth map, and it will be a very important intermediate result for our applications.

It is known (Bulatov 2011, Nistér 2001) that a detailed generic 3D modeling of urban terrain is a very challenging task because of a large topological variety of objects, that is, not only buildings with vertical walls, but also trees, bridges, underpasses, etc. Therefore we will consider in this paper a 2.5D urban terrain representation $z(x, y)$, in which we will search for two kinds of elevated objects, namely buildings and trees. In other words, it can be assumed that typical 3D aspects like balconies or objects below the tree crowns can be neglected because of sufficient altitude of the sensor platform. Finally, image-based georeferenciation can be used to fix the shortcomings of small and miniaturized UAVs regarding onboard navigation systems by registration of images or videos to referenced images such as orthophotos. Here we propose a real-time oriented approach, adjusted for agile UAVs equipped with video cameras.

1.2 Organization of the paper

The four steps of the procedure for geo-referenced urban terrain modeling after image acquisition are: extraction of camera orientation parameters, computation of depth maps, reconstruction of buildings and vegetation and, finally, geo-referencing (see Figure 1). Since the two last steps represent the main contributions of this work, they will be explained in more detail in Section 2. The results from a real data set are presented in Section 3, while conclusions and ideas for future research are summarized in Section 4.

* Corresponding author.

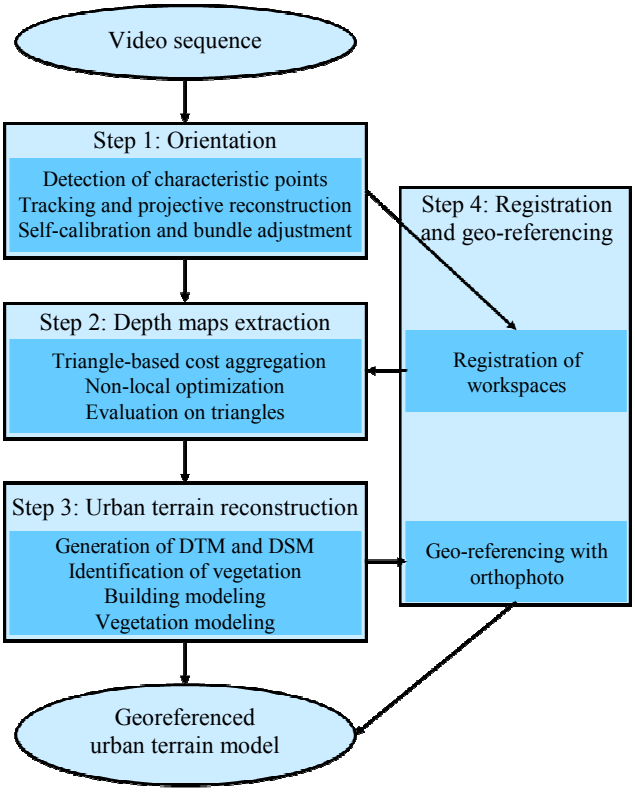

Figure 1: Overview of the algorithm.

1.3 Previous work

Numerous references exist about the first step of the procedure, namely computation of orientations. The *structure-from-motion approaches* (Bulatov 2008, Hartley and Zisserman 2002, Nistér 2001) do not use any additional information except the video stream itself and the fact of constant camera intrinsic parameters; the sensor trajectory is retrieved by means of algorithms of computer vision, (Hartley and Zisserman 2002).

The second step concerns depth map extraction. Here, an excellent survey (Scharstein and Szeliski 2002) can be recommended. We, however, will give a brief description of our approach for obtaining multi-view depth maps supported by triangular networks from already available points.

The third step contains building and vegetation modeling from depth maps. Most work has been elaborated for LIDAR point clouds (Gross et al. 2005, Rottensteiner 2010, Vosselman and Dijkman 2001), and so it is important to generalize different features of these algorithms for dense point clouds obtained by means of passive sensors. For example, the work of (Rottensteiner 2010) presupposes a color segmentation of pair of images and uses LIDAR points (sparse, but homogeneously distributed in the images) to determine initial orientation of planes. It is, however, not straight forward to generalize this approach to video sequences with hundreds of frames. The algorithm of (Vosselman and Dijkman 2001) is a generalization of the Hough-Transform for 3D dense point cloud. The image-based approach (Fischer et al. 1998) is a combination of *bottom-up* (data-driven) and *top-down* (model-driven) processes that can be significantly simplified if precise and accurate 3D information for a dense set of pixels is available. In this present work, we are inspired by the simple and reliable method (Gross et al. 2005) whose modifications for our applications are covered in the next sections.

Finally, with respect to geo-referencing, we mention a group of methods for a real-time-oriented matching of video streams and orthophotos (Lin et al. 2007 and Solbrig et al. 2008). Both authors use a 2D homography as a transformation function and differentiate between *intrasensorial* registration of different frames of the video sequence and *intersensorial* registration, i.e. matching of video frames and the orthophoto. For example, in (Solbrig et al., 2008), after the first frame of the video sequence is registered to the orthophoto, interest points in videos frames are tracked by the relatively fast KLT-tracking algorithm (Lukas and Kanade 1981) so that the homography between a frame and the orthophoto can be computed incrementally. This process is called intrasensorial registration. At the same time, by monitoring back-projection errors, a failed intrasensorial estimation of the homography can be detected and replaced by an intersensorial registration, i.e. matching SIFT points (Lowe 2004) of the frame and orthophoto. Since a homography is clearly insufficient for scenes with a non-negligible spatial depth, we decided to extend this approach for 2.5D situations, see Section 2.4.

2. PROPOSED ALGORITHM

2.1 Computation of camera orientation parameters

Reconstruction of the camera trajectory and a sparse point cloud from a moving sensor can be performed by a structure-from-motion algorithm. Characteristic points in the images of the sequence are tracked (Lukas and Kanade 1981) from frame to frame. From these correspondences, a projective reconstruction is carried out by methods of (Hartley and Zisserman 2002). Euclidean reconstruction by means of a self-calibration algorithm (Bulatov 2008) followed by bundle adjustment complete this step.

2.2 Depth maps computation

The output of the previous step includes the camera trajectory and a sparse point cloud. What we need is the depth information for every pixel of the so-called *reference image*. We take a short subsequence of 3 to 7 images; the reference image is typically the one in the middle of the subsequence. The core of the algorithm, which is described in more detail in (Bulatov 2011, Bulatov et al. 2011), consists of minimization, by means of the well-known semi-global method (Hirschmüller 2008), an energy function that is computed for every pixel and every label of the discretized depth scale.

The energy function consists of three terms. The first term is the aggregated value of the data terms (for examples, gray values differences), whereby the *aggregation function* should be robust against occlusions. The second term is a smoothness term that penalizes depth discontinuities of neighboring pixels. Contrary to most of the previous work, a triangulation-based smoothness term is introduced that biases the depth of the pixel in the direction of the value given by the triangular interpolation of depth values from already available points. Also, evaluation of triangles consistent or inconsistent with the surface is performed. This makes sense because many parts of the urban scenes consist of piecewise planar structures that can be modeled by triangles. Other advantages of this method include a robust treatment of textureless areas and disentanglement of discretization artifacts in triangles consistent with the surface.

2.3 Urban terrain reconstruction

Generation of DTM and DSM: There are plenty of algorithms for depth map fusion (Pock et al. 2011). For sake of simplicity,

the 3D points orthogonally projected in one cell of a rastered fragment of the *xy*-plane and the median *z*-value of these points is assigned to be the altitude of the cell in our DSM (digital surface model). To compute the digital terrain model, we incur the approach of (Gross et al. 2005) into our algorithm. First cells corresponding to the ground – those with minimum altitude within a circular filter – are fixed; whereby the circle radius corresponds to the smaller dimension of the largest building. Then, a modification of the Neumann differential equation

$\Delta b = 0$ for all non-fixed pixels b
$\partial b/\partial \mathbf{n} = 0$ for non-fixed pixels at the image boundary

is solved to obtain the DTM values for the remaining, unfixed pixels b. Here $\mathbf{n}$ is the outer normal vector at the boundary. Now the methods described in the following section are applicable to the difference image between DSM and DTM, denoted by B. The synthetic color/intensity image obtained by projecting the color/intensity values from frames of true video sequence into the rasterized *xy*-plane is denoted by J for later processing.

Building Modeling: The thresholded segmentation of B is used to extract elevated labeled regions which, by our assumption, correspond either to buildings or to vegetation (trees). We describe here the process of building reconstruction from these regions and leave the question of building identification until the next paragraph. The building outlines are extracted by fitting rectangular polygons in B. If there are small convexities or indentations in the building contour, short edges are removed by modifying the object contour through generalization. The area is changed as little as possible by adding to or removing from the object rectangular subparts. The generalization repeats until all short edges are removed. As a result of this first substep, building outlines are created.
The second substep consists of modeling roof planes with a slightly modified algorithm (Geibel and Stilla 2000). The normal vector of every internal building pixel x is determined by computing a local adaptive operator in a small window around x. Extraction of roof planes is performed by clustering these normal vectors and grouping connected pixels into regions. We use morphological operations to eliminate small holes caused by outliers and describe the roof surfaces by polygons. Finally, the walls of the buildings are constructed through the outer polygon edges of the roof surfaces (upper edge) and through the terrain height (lower edge) available from the DTM.
The last substep is texturing of building roofs and surrounding terrain by means of J. The question of texturing building walls with possibly available side views is left for future work.

Identifying and Modeling Vegetation: In order to classify elevated regions of B into building and trees in the absence of a short-wave infrared channel, we make use of two assumptions. First, it can be expected that at the moment of video rendering all trees have several characteristic colors; for example, during summer, the dominant color is green while during autumn, leaves of the trees often also take on red, yellow and orange color, see Figure 2. Second, regions corresponding to buildings usually contain many straight line structures. In our approach, straight line segments are determined in the reference frames of the video sequence with the method of (Burns et al. 1986) and projected into J and B. The lineness measure λ of an elevated region R of B is the sum of the length of all line segments entirely lying in $\underline{R}$ divided by the perimeter of R. Here $\underline{R}$ denotes a morphological dilatation of R. Of course, it is not enough to associate the regions with a high value of λ with buildings, because the most challenging kind of regions – namely, buildings parts of which are occluded by trees – falls into this category and identification of contours of such a region with rectangular polygons will then suffer from severe artifacts. Instead we propose to associate the regions with a quite low value of λ with isolated (groups of) trees, calculate the mean value and standard deviation of color values within this regions for each band (red, green and blue) and declare all cells with color values of a smaller deviation from the mean value than the standard deviation for each band as tree-like cells. The corresponding cells of the height map B are not included in the building reconstruction.
In the areas where several treelike pixels form a large enough region, 3D tree models are included in the visualization.
While the height of the tree is obtained from the height map, its appearance can be visualized at different resolutions and levels of detail. Real time image generation algorithm make extensive use of texture mapping functions of computer graphics boards. For such applications, a 3D tree model is built only by a few polygons whose transparency is modulated by images of trunks, branches and tree crowns. Different seasonal appearances can be easily extracted from image sets like those shown in Figure 2 (Godet 1986) and applied to the tree model in accordance with simulation time (time of the year). Applying random generators for tree placements and tree scales even large forest areas can be rendered in real time.


Figure 2: Seasonal variations of a pear tree.

More advanced modelling of natural objects like trees, bushes etc. is based on Lindenmayer systems, also termed L-Systems, (Prusinkiewicz 1980). These are production systems whose grammar describe e.g. how a plant grows (mainly the branching topology) by string-rewriting methods. Such a tree needs many polygons for an accurate 3D model representation (e.g. up to 100.000 polygons). In other words, the complexity of a whole urban model of a city, in terms of polygons, is comparable to the complexity of a single tree model. Currently, this high complexity prohibits them for use in real time scene generation, despite their advantageous 3D geometry. The number of polygons has to be reduced for real time visualization e.g. by dynamic polygonal representations (LOD) (Rossignac 1993) and by application of dynamic billboards.

2.4 Registration and Geo-referencing

Registration of workspaces: Three previous steps of the algorithm can be performed, independently from each other, by different UAVs with different kinds of sensors onboard and also in different areas of the region to be explored. In this case, a Euclidean reconstruction of different workspaces (by workspace, we denote the meta-data including camera trajectory, point cloud, and all other available information) is carried out in different coordinate systems in the course of the algorithm described in Sec. 2.1.
An obvious method of registration, namely, to use the navigation equipment onboard of the UAV, becomes less reliable for inexpensive and miniaturized UAVs. The reason is the low accuracy of data delivered by such a lightweight, inexpensive navigation unit. Therefore, an alternative, image-based approach

was developed to fuse pairs of workspaces into a common coordinate system. In other words, our task is to determine a spatial homography H connecting these workspaces. Such a homography is given by a regular 4×4 matrix in homogeneous coordinates. For two workspaces with sets of camera matrices $\boldsymbol{P}_1$ (= $P_{1,1}$, ..., $P_{1,K}$) and $\boldsymbol{P}_2$ (= $P_{2,1}$, ..., $P_{2,L}$) , we assume, without loss of generalization, that two images corresponding to camera matrices $P_{1,K}$ and $P_{2,1}$ cover an overlapping area of the data-set and that point correspondences c_1 and c_2 can be detected in these images by means of a matching operator (e.g. Lowe 2004). We now compute, by tracking points c_2 in other images of the second workspace as well as camera matrices $P_{2,1}$, $P_{2,2}$, ... several 3D points Y in the second coordinate system. By backward tracking points c_1 in other images of the first workspace and Y, we obtain the set of camera matrices $Q_{1,K}$, $Q_{1,K-1}$,... via camera resection algorithm (Hartley and Zisserman 2002). For more than one correspondent camera $Q_{1,K-n}$ to $P_{1,K-n}$, the initial value of the spatial homography H as a solution of the over-determined up-to-scale system of equations

$$\begin{bmatrix} P_{1,K}H \\ P_{1,K-1}H \\ \dots \end{bmatrix} \cong \begin{bmatrix} Q_{1,K} \\ Q_{1,K-1} \\ \dots \end{bmatrix}$$

is obtained via Direct Linear Transformation method and refined by means of an geometric error minimization algorithm.

Geo-referencing of workspaces: Here we consider again the case where the internal navigation of an UAV is not available and strive for an image-based matching of the video stream and a (geo-referenced) orthophoto I. Because of the 3D character of the scene, we cannot provide, contrary to (Solbrig et al. 2008), an accurate registration by means of a planar transformation (i.e. a 2D homography); also, because of large differences in scale of a video frame and the orthophoto, we cannot rely on a SIFT operator as a matching cost function any longer.
To overcome both problems, we decided to use the synthetic image J of the previous section instead of video frames. Considered from the nadir perspective and downsampled to the resolution of the orthophoto, J can be matched to I by means of a homography. Optionally, elevated regions can be identified by means of B and excluded from the matching procedure. We found that application of the Local Self Similarity-algorithm due to (Shechtman and Irani 2007) produced stabler correspondences (and, as a consequence, significantly better results) than comparable algorithms, such as SIFT (Lowe 2004) or SURF (Bay et al. 2006), if the radiometric differences between I and J are large. Similarly to (Solbrig et al. 2008), using of robust methods for outlier rejection, such as RANSAC (Fischler and Bolles 1981) accelerated by a $T_{1,1}$-test (Matas and Chum 2002), is indispensable for a reliable registration.

3. RESULTS

The input data set of this section is an UAV-based video recorded over the village Bonnland, in Germany. We used a FX 35 Lumix digital camera onboard of a quadro-copter md4-200, built by *Microdrones RC UAV*. The video frames contain 720×1280 pixels and are of resolution 25-40 pixel/m. After a structure-from-motion algorithm (Bulatov 2008) the depth maps supported by triangular meshes were obtained from 18 reference frames. We depict some of reference frames and the corresponding depth map in Figure 3, top. The complete camera trajectory and a point cloud from a union of three workspaces (as described at the beginning of Sec. 2.4) is presented in Figure 3, bottom. Five images were used for depth computation.

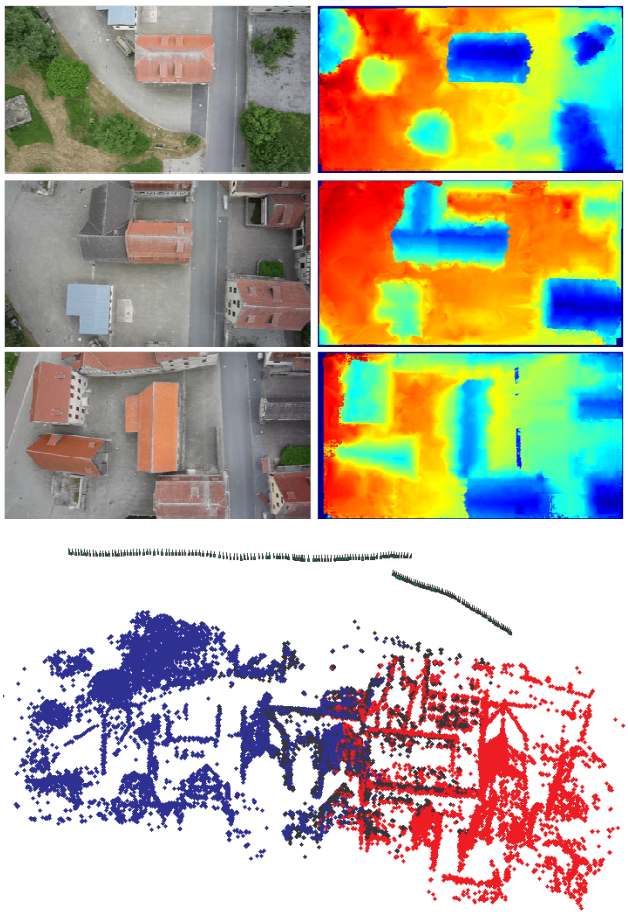

Figure 3: Top: three reference frames and corresponding depth maps obtained as described in Sec. 2.2 and the (intrasensorial) registration procedure of Sec. 2.4. Bottom: the complete camera trajectory and a sparse point cloud obtained as a union of three workspaces (coloured in blue, black and red). From the kink in the camera trajectory, it becomes evident that we are dealing with two different UAV-flights.

Since after the Euclidean reconstruction (not supported by internal navigation), the physical vertical direction does not coincide with the direction of the z-axis, the plane π through the camera positions is robustly calculated, and, since we know that the sensor altitude remained relatively constant, we rotated the point cloud to make the xy-plane parallel to π. By rasterization of the point cloud, as discussed at the beginning of Sec. 2.3, a synthetic image J with 903×652 cells is created. We show J and the difference image B of DSM and DTM in Figure 4, top. The labeled elevated regions are illustrated together with Burns-lines and lineness measures in Figure 4, bottom. The results of building reconstruction are illustrated by screenshots from different positions in Figure 5. For the building walls, a synthetic texture which is typical for the region of interest is taken.
The synthetic image J of Figure 4 has been geo-referenced as described in Sec. 2.4. The orthophoto, a fragment of which is shown in Figure 6, bottom right, is a product of *TopoSys*. Due to the low operating altitude of the UAV in our experiments (see Figure 6, top), the difference in scale between the video-frames and the orthophoto exceeds the tolerance of SIFT. On the other hand, the synthetic image J has a desired (lower) resolution and is ortho-rectified like the orthophoto itself. To cope with the scattered RGB-values, application of the very robust Local Self Similarity-algorithm can be recommended for successful registration, as illustrated in Figure 6, bottom.

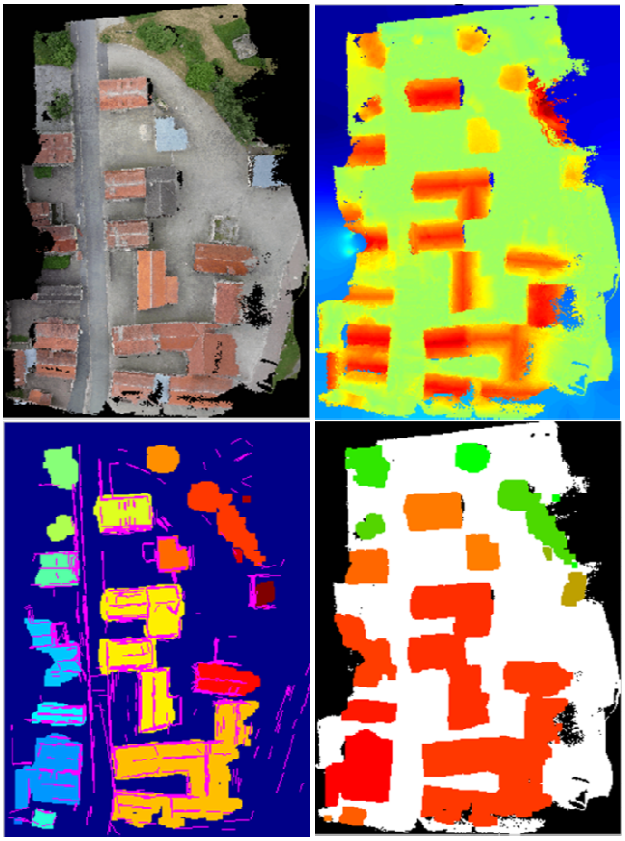

Figure 4: Top left: the synthetic image J illustrating the rasterization of the RGB-values from video frames into the xy-domain; top right the corresponding difference of DSM (fused rasterized depth maps) and DTM. Bottom left: regions found in B together with line segments depicted in magenta; bottom right: Lineness measure of the regions; those corresponding to areas of vegetation are marked in green and those to buildings in red.

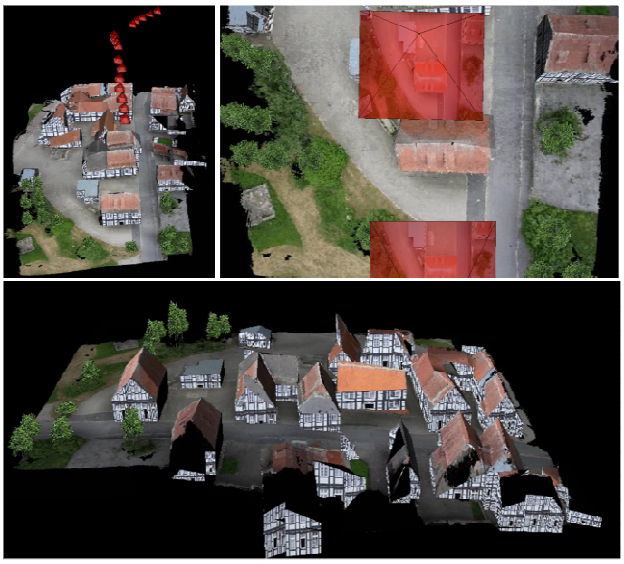

Figure 5: Three views of the textured model. Top left: oblique view. Top right: a nadir view from one of the reference camera perspective. The reference cameras are illustrated by red pyramids with corresponding reference images. Bottom: another view of the scene.

Figure 6: Top: the differences in resolution and appearance of the orthophoto (left) and a video frame (right) are enormous if the altitude of the sensor platform is low. Bottom: the workspace has been geo-referenced by registration of the synthetic image to an orthophoto.

4. CONCLUSIONS AND OUTLOOK

We presented a straight-forward algorithm from creating close range urban terrain models from (M)UAV-videos showing urban terrain. Two first substantial steps of the algorithm – image orientation and depth maps extraction – are fully automatic and do not require any additional knowledge. The extraction of depth maps from a short subsequence with an arbitrary number of not necessarily rectified images is widely supported by triangular networks from already available points. These meshes are very convenient to extract depth values in those regions, where the surface given by the triangular network nearly coincides with the real surface, especially in regions of homogeneous texture, while the computing time is extremely low. In order to complete dense reconstruction in regions inconsistent with the surface, an optimal trade-off between good results and computing time was made with the semi-global approach (Hirschmüller 2008). The most important work for the future is a better superposition of depth maps in order to improve – by means of visibility information and radiometric confidence values – the rasterized DSM.

The third step concerns building modeling and here it becomes clear that the modification of the three-step procedure of (Gross et al. 2005) can also automatically process dense point clouds obtained by passive sensors from the nadir perspective. The output is, in the majority of cases, the correct segregation of urban structures – building and vegetation – and building outlines. A separation of buildings and vegetation is performed by computing a lineness measure λ of every elevated region and specifying treelike pixels by means of their colors. This approach can be improved in two ways: first, we strive in the future for automatic selection of a threshold for λ and second, especially for spring and winter, it will be extremely important to give more weight to the lineness measure than to distribution of colors within treelike pixels.

While the height of trees is given by depth maps, the important parameter of diameter is not considered yet, in other words, all trees appear equally broad in our model. With respect to visualization, it is possible to adapt the appearance of trees for different seasons and times of day, and since, in addition, the geo-referencing procedure described in Section 2.4 allows re-

placing the orthophoto by some other geo-referenced view, the representation of the whole terrain can be easily instantiated for a broad spectrum of situations, which makes it a valuable tool for applications in augmented and virtual reality.
Image coordinates can be converted to geo-coordinates by registration to a (georeferenced) orthophoto. Thus, the precision of geolocation does not depend on the quality of onboard navigation systems. But then, the use of iconic information is a critical restriction in certain situations like homogeneous terrain. Structural image-matching methods, such as (Michaelsen and Jäger 2009) are put into the focus for future development.

References:

Bay, H., Tuytelaars, T., Van Gool, L., 2006, SURF: Speeded up robust features, In: Proc. *9th European Conference on Computer Vision in Graz*, Austria.

Bulatov, D., 2008. Towards Euclidean reconstruction from video sequences, In: *Int. Conf. Computer Vision Theory and Applications,* (2), pp. 476-483.

Bulatov, D., 2011. Textured 3D reconstruction of urban terrain from UAV-borne video sequences, Ph.D. thesis at the Leibniz University of Hannover, Germany, DGK-C Nr. 661, to appear.

Bulatov, D., Wernerus, P., Heipke, C., 2011. *Multi-view dense matching supported by triangular meshes*, ISPRS Journal of Photogrammetry and Remote Sensing, accepted for publication.

Burns, J.B., Hanson, A.R., Riseman, E.M. 1986. Extracting straight lines. *Transactions on Pattern Analysis and Machine Intelligence*, 8(4), pp. 425-455.

Fischer, A., Kolbe, T.H., Lang, F., Cremers, A.B., Förstner, W., Plümer, L., Steinhage, V., 1998. Extracting buildings from aerial images using hierarchical aggregation in 2D and 3D. *Computer Vision and Image Understanding* 72(2), pp. 185-203.

Fischler, M.A., Bolles, R.C., 1981. Random Sample Consensus: A paradigm for model fitting with applications to image analysis and automated cartograph, In: Communications of the ACM, 24(6), pp. 381-395.

Geibel, R., Stilla, U., 2000. Segmentation of Laser-altimeter data for building reconstruction: Comparison of different procedures, In: *Int. Arch. of Photogrammetry and Remote Sensing,* 33, part B3, pp. 326-334.

Godet, J.-D. 1991, *Gehölzführer: Bäume und Sträucher*, Mosaik Verlag GmbH, München, ISBN 3-576-10064-4.

Gross, H., Thönnessen, U., v. Hansen, W., 2005. 3D-Modeling of urban structures, In: *Joint Workshop of ISPRS/DAGM Object Extraction for 3D City Models, Road Databases, and Traffic Monitoring CMRT05*, In: *Int. Arch. of Photogrammetry and Remote Sensing*, 36, Part 3/W24, pp. 137-142.

Hartley, R., Zisserman, A. 2008. *Multiple View Geometry in Computer Vision*. Cambridge University Press.

Hirschmüller, H., 2008. Stereo processing by semi-global matching and mutual information, In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2), pp. 328-341.

Lowe, D. G., 2004. Distinctive image features from scale-invariant keypoints, *International Journal of Computer Vision*, 60(2), pp. 91-110.

Lin, Y., Yu Q., Medioni, G., 2007. Map-Enhanced UAV Image Sequence Registration, In: *IEEE Workshop on Applications of Computer Vision*.

Lucas, B., Kanade, T., 1981. An iterative image registration technique with an application to stereo vision, In: *Proc. 7th International Joint Conference on Artificial Intelligence* (IJCAI), pp. 674-679.

Matas, J., Chum, O., 2002. Randomized RANSAC with $T_{d,d}$ – test, In: *British Machine Vision Conf.*, vol. 2, pp. 448-457.

Michaelsen, E., Jäger., K., 2009 A GOOGLE-Earth based test bed for structural image-based UAV Navigation, *12th International Conf. on Information Fusion,* Seattle, USA. Proc. On CD, IEEE-ISIF, ISBN: 978-0-9824438-0-4, pp. 340-346.

Nistér, D., 2001. Automatic Dense Reconstruction from Uncalibrated Video Sequences. Ph.D thesis, Royal Institute of Technology KTH, Stockholm, Sweden.

Pock, T., Zebedin, L., Bischof, H., 2011. TGV-fusion. In: *Rainbow of Computer Science*. Springer-Verlag, to appear.

Prusinkiewics, P., Hanan, J. 1989, *Lindenmayer Systems, Fractals, and Plants*, Springer Verlag, New York, USA, Inc., ISBN 0-387-97092-4.

Rossignac, J., and Borrel, P. 1993, *Multi-resolution 3D approximations for rendering complex scenes*, In: Geometric Modeling in Computer Graphics, pp. 455-465, Springer-Verlag.

Rottensteiner, F., 2010. Roof plane segmentation by combining multiple images and point clouds, In: *Photogrammetric Computer Vision and Image Analysis Conference*, Int. Arch. of Photogrammetry and Remote Sensing, 38, part 3A, pp. 245-250.

Scharstein, D., Szeliski, R., 2002. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1), pp.7-42. Images, ground truth and evaluation results available at: http://vision.middlebury.edu/~schar/stereo/web/results.php (accessed 28 Feb. 2010).

Shechtman, E., Irani, M., 2007. Matching local self-similarities across images and videos, In: *IEEE Conf. on Computer Vision and Pattern Recognition*, Minneapolis, USA.

Solbrig, P., Bulatov, D., Meidow, J., Wernerus, P., Thönnessen, U., 2008. Online annotation of airborne surveillance and reconnaissance videos, In: *11th International Conf. on Information Fusion,* Köln, Germany, pp. 1131-1138.

Vosselman, G., Dijkman, S., 2001. 3D building model reconstruction from point clouds and ground plans, In: *Int. Arch. of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 34, part 3/W4, pp. 37-44.