RECOGNIZING CARS IN AERIAL IMAGERY TO IMPROVE ORTHOPHOTOS

Franz Leberl
Stefan Kluckner
Georg Pacher
Helmut Grabner
Horst Bischof
Michael Gruber*

Institute for Computer Graphics and Vision
Graz University of Technology, Austria
{leberl,bischof,kluckner,pacher,hgrabner}@icg.tugraz.at

*Microsoft Photogrammetry, Austria
michgrub@microsoft.com

ABSTRACT

The creation of "clutter-free" orthophotos and 3D GIS data bases is desireable. Cars parked in the streets or driving on the road are of little interest in an orthophoto or in a GIS. We are demonstrating the ability of automatically detecting nearly all cars shown in an urban aerial photography at a ground sampling distance in the range of 8 cm to 15 cm. The image shapes from those cars can be inpainted to create a car-less orthophoto. In addition, the shapes can also be used to remove the effect of cars on the digital elevation model by filtering the elevations representing cars and thereby improving the "Bald Earth" DTM. We study the use of redundancy from image overlaps and the introduction of color to further improve the score beyond the current high success.

INTRODUCTION

The systematic creation of models of the real world to support the locational awareness on the Internet, or to grow the current 2D approach to car navigation into a full 3D experience, is an expensive proposition if based on manual methods. This motivates the work to replace such massive labor by automated procedures. Systems that are currently being set up to produce 3D urban models for Internet applications can rely on a rich literature, a great variety of methods and academic projects that demonstrate fully automatic approaches to 3D reconstruction by shape-from-stereo or laser scanning, e.g. (Werner 2002).

In recent years, our team has inspired the development of a fully automated work-flow to recover 3D city models from highly overlapping aerial images produced by the UltraCam from Microsoft Photogrammetry (formerly Vexcel Imaging). In its most recent incarnation, each UltraCam image resolves 14430 x 9420 pixels. Success in the robust automatic processing of the data depends on a high inter-image redundancy based on an 80% along-track overlap, thus within an individual flight line, and a 60% across-track, thus from one flight line to the next. In much of our work, we employ a ground sampling distance (GSD) for the digital aerial images at 8 cm and 15 cm.

Intermediate data sets get computed consisting of a Digital Surface Model (DSM) of the terrain, which then gets separated into the “Bald Earth” (a Digital Terrain Model (DTM)) representing the terrain off which the vertical objects get stripped, as well as those vertical objects themselves. The data exist in the form of “point clouds”. Associated with the points, or triangles formed from the points, are patches of photo texture. From those intermediate results, one builds both the 2-dimensional orthophotos as well as the 3-dimensional “city models”. The

orthophotos are created by projecting the photo texture either onto the “Bald Earth” DTM for so-called “Traditional Ortho Photos” or onto the DSM to produce a so-called “True” or “Reflective Surface Ortho Photo”.

The 2D orthophotos provide a simpler data structure, smaller data volume and a great ease of use and orientation.
By contrast, the 3D models consist of more data, need a more complex data structure and are more difficult to use and navigate. Therefore, all location-aware web sites offer imagery in the form of 2D orthophotos.

In this paper, we want to go a step further and develop semantic knowledge about the objects in the terrain, and initially use that knowledge to affect the orthophotos. In subsequent steps, we will use that knowledge also to improve the 3-dimensional city models.

The paper is structured as follows. We first review the 3D city modeling approach into which the semantic knowledge needs to be fed. Then we address object recognition in aerial images and we introduce an on-line Boosting variant. Furthermore we demonstrate an initial application of this semantic knowledge creation to detect cars. We demonstrate how the detected cars can be grouped together for extracting a street layer. The street layer forms an important part of the DTM. We will finally conclude that the method produces reliable and accurate results that outperform a manual approach of car detection and removal.

CURRENT 3D CITY MODEL GENERATION

Camera Station Information

The current approach of city modeling starts with a block of overlapping aerial photographs, typically obtained from a series of parallel flight lines, each representing a so-called “strip” of overlapping images. The coordinates and attitude of each camera station (thus the exterior orientation) are computed by an automatic process that searches for a great number of tie-points among overlapping images, perhaps 10,000 in a single image, and then computes the exterior orientation from the geometry of the image block. This relies on “redundancy” not only by image overlap, but also by the use of a very high number of tie points. The approach is described in (Zebedin, 2006).

Surface Modeling

The computed camera orientation parameters feed into an area based image matching algorithm to produce a dense range patch for each of the input images. Those patches then get converted into a seamless DSM. The particular implementation of the approach has been described by (Klaus, 2006). The range images are computed from three input images (a reference image and its two immediate neighbors) using a plane sweeping approach. The planesweep uses the normalized cross correlation as similarity measure and produces a so-called 3D depth space which contains the depth hypotheses and their associated correlation values. The final range image is computed using a semi-global optimization approach proposed by (Klaus, 2006). It employs a redundant and overlapping set of patches, each derived from an image triplet. Given an 80% forward-overlap, each ground point will be imaged onto 5 images. This overlap results in three image triplets per ground point, ready to get merged.

Creating Data Products in 2D and 3D

Combining the resulting terrain surface with the image textures produces a result as shown in Figure 1. The actual work-flow is fairly complex and includes a separation of the surface data into the “Bald Earth” and its vertical objects, and the point clouds do get triangulated and thinned out to be acceptable in an Internet application. At this time, the methods have advanced to the point where data can get produced fully automatically, and with minimal need for manual editing.

Figure 1, A generated 3D model from a part of the city center of Graz in Austria.

Adding Semantic Knowledge

The drawback of these automatic reconstruction methods is that only point-clouds (converted into triangles) and textured 3D models are provided which do not present any semantic information. Therefore, we cannot query the data sets according to content nor can we produce specific thematic maps of specific object classes. In order to have a proper interpretation of the scene and to build better 3D models higher level knowledge about the object is required. This has been recognized by recent research work, for example (Cornelis, 2006 and Foerstner, 2006). We believe that semantic information can be extracted fairly reliably, and that this increases the usefulness and value of the data tremendously.
Let us therefore proceed with a presentation of a promising method for object detection applicable to aerial images, and show how this affects the 2-dimensional orthophotos, using the example of cars.

OBJECT DETECTION

Background

In terrain images, we have permanent objects that we want to represent in a location-aware system. However, we also have objects that are irrelevant because they are not a permanent feature of the terrain. This includes cars and people. We initially focus on cars, although considerable work has been done already to detect pedestrians (Viola, 2003), faces (Rowley, 1998 and Viola, 2001), or cars (Agarwal, 2004), bikes (Opelt, 2004), and other visual objects. One sometimes denotes this as a “visual categorization” as opposed to “specific object recognition" (Fergus, 2003 and Opelt, 2004). See (Ponce, 2006) for a recent overview of research in the area of visual categorization.

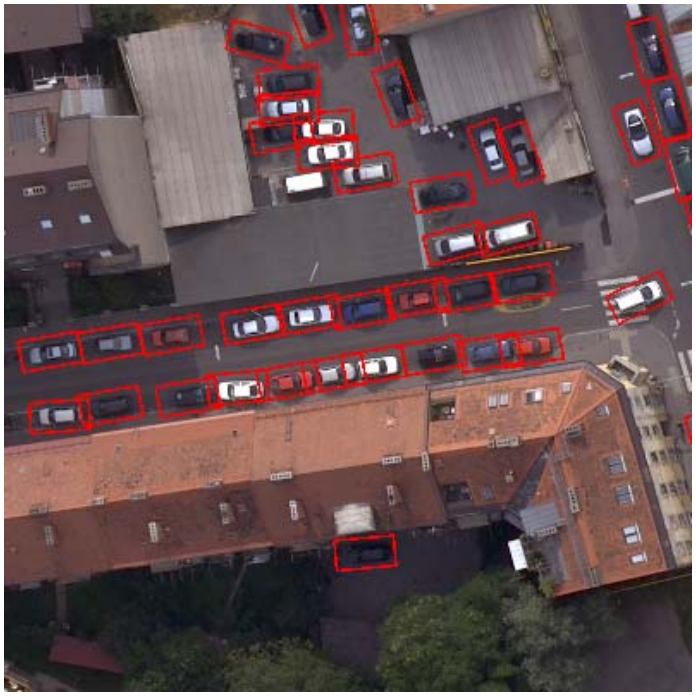

Figure 2, A typical scene in an urban environment with marked cars as ground truth.

The predominant paradigm for object detection is now based on classification approaches which scan the whole image by sliding a window over it, at different resolutions, to extract features such as edges, corners, texture, and classify this window as one containing the object of interest or not. Usually some post-processing by a local maxima search or similar approaches is necessary to avoid multiple detections. At the core of these object detection approaches is a classifier, e.g. AdaBoost (Freund, 1997), Winnow (Littlestone, 1987), neural Network (Rowley, 1998) or support vector machine (Vapnik, 1995). The proposed approaches have achieved considerable success in the above mentioned applications. We present in this paper an approach that follows this paradigm; as classifier we use an on-line variant of Boosting (Grabner, 2006).

In general, Boosting is a classifier combination method, which combines several weak classifiers to form a strong one. Many researchers have analyzed and applied Boosting for different tasks. Here, we can apply Boosting for feature selection, as introduced by Tieu and Viola (Tieu, 2000). The basic idea is that each feature corresponds to a weak classifier. The application of Boosting to these features gets us an informative subset of features. The on-line selection of significant weak classifiers is performed on so-called selectors. Each selector consists of a set of weak classifiers and can be seen as an auxiliary classifier that switches between the weak learners. Each training sample trains all weak classifiers and the selector with the lowest error on the samples is then selected. Each weak learner directly corresponds to the response of a single extracted feature.

Image Features

Using features instead of raw pixel values increases the robustness and integrates invariance in the classifier. In principle we can choose among many different features, but since we are working with large images, and since these large images need to be scanned by the classifier we look for features that can be computed quickly, for example by using efficient integral images. Our choices are Haar-like features (Viola, 2001) and orientation histograms (Levi, 2004), which can be computed via the integral structures.

Training

The car detection problem is treated as a binary classification problem: car versus background. Usually this requires a large amount of pre-labeled data, perhaps in the order of ten-thousand images. However since we have an on-line learning method (Grabner, 2006), which is sufficiently fast for interactive work, we attack training as an interactive learning problem. The key idea is that the user has to label only those examples that are not correctly classified by the current classifier.

We evaluate the current classifier on an image. The human supervisor labels informative samples, e.g. marks a wrongly labeled example which can either be a false detection or a missed car. The new updated classifier gets applied again on the same image or on a new image, and the process continues iteratively until a satisfactory detection performance is achieved. This is a fully supervised interactive learning process. This process permits us to update the parameters of the classifier in a greedy manner with respect to minimizing the detection error. This results in very fast training with a minimum number of samples. It avoids labeling redundant samples that do not contribute to the current decision boundary.

Detection

After training, the overall detection results from the exhaustive application of the trained classifier on the images. Since we know the resolution of the image we do not need to search cars at various scales. However, we need to cope with the car's orientation and therefore search cars at different image rotations. Instead of training the classifier with different orientations we train it at one canonical orientation, and evaluate it by rotating the image with 15 degree increments. A car is considered to be detected if the output confidence value of the classifier is above a threshold, i.e. zero. The lower the threshold, the more likely an object is detected as a car, but on the other hand the more likely a false positive will occur. For post processing we could use non-maximum suppression. A slight improvement of the localization can get achieved using the mean-shift algorithm (Comaniciu, 1999). The confidence values of the classifier are an input to the mean shift procedure. The mean shift algorithm is non-parametric and iteratively locates density extrema or modes of a given distribution. We rely on a Gaussian kernel, its width is related to the size of the car. Starting from an initial location the local mean shift vector maximizes the underlying probability density function in a gradient descent manner.

Street Layer Extraction

The street layer defines regions where cars can definitely appear. In this work, we propose to use the best car detections to extract this street layer. It consists of all flat areas on the ground, which are not covered by vegetation or water. Since most parts of the street layer are connected, a connectivity measure can be used as verification for the initial points. We use the car detection results as these initial points. It is obvious that the car detections can include some false positive detections. Therefore a graph-based grouping technique performs an outlier removal by including context information, such as height data and a color similarity measurement as proposed in (Pacher, 2008).
We apply the idea of graph-based grouping to link the best car detections. Similar to (Pacher, 2008), we define two requirements for a valid linkage between two detections. First, a linkage may not include discontinuities concerning their involved height values and additionally, the color values along the connecting path must correspond to a specific street related color distribution. These two requirements constrain the construction of a weighted bi-directional graph G = (V,E). The set of vertices V is 4-neighborhood connected according to a regular grid covering the overall image. In order to reduce computation time, the grid spacing is fixed to 1.5 meters for all evaluated data sets.

The color distribution along streets is estimated using probes around the detected cars. In order to have an isotropic color feature space, the image is converted to the three dimensional CIE Luv color space. Assuming normal distributed color values, the distribution of extracted color values can be easily modeled by using the estimated mean and covariance matrix. The Bhattacharyya distance is applied as a similarity measure between

Normal distributions similar to (Donoser, 2007). The discretized Bhattacharyya distance denotes a measure for statistical separability of probability distributions. Therefore, it gives a quantitative statement whether the color distributions are likely to be the same or not. By applying this distance measure patch-wise on each pixel of a test image the image can be roughly segmented. To reduce the enormous computation time, integral image structures are used.

Then, the Dijkstra algorithm (Dijkstra, 1959) finds single shortest pathes with lowest costs depending on the assigned weights. A linkage between the car detections is determined with respect to the maximal costs for each computed path. By tracing all low-cost linkages between the car detections, we establish a connectivity measurement for all detected cars.

Image Redundancy

To obtain an enlarged coverage of the street layer extraction we exploit the available image redundancy. Independently, the car detections and the estimated connecting paths are extracted from three neighboring images. The collected paths provide an increased number of control points and facilitate the loop closing of streets. Since the camera orientations of each image are known, the control points are transformed into the common world coordinate system using the height data. As the next step, the interpolation based street layer extraction step is performed using this world coordinate system.

The extraction and grouping of the car detections in overlapping images improves the description of streets characteristics using additional representative control points. To find areas on similar ground level a DTM can be estimated and compared to the available 3D height information. A thin plate spline (TPS) interpolation (Bookstein, 1989) offers a closed-form solution for surface interpolation from a set of control points.

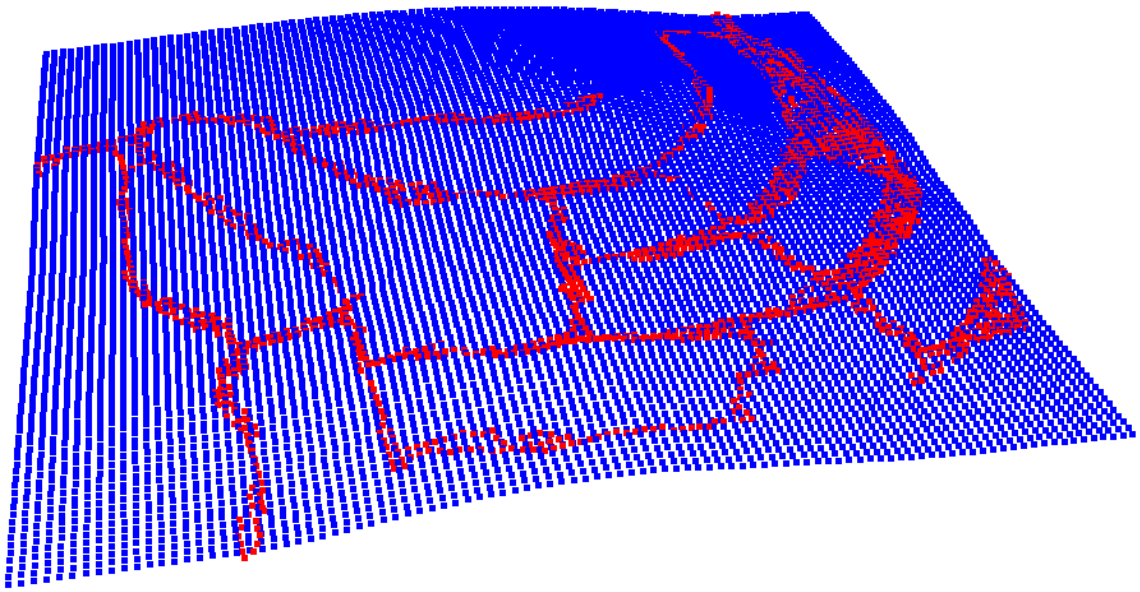

Figure 3, A TPS interpolated DTM for the San Francisco data set: The blue points represents the interpolated grid, the red points indicate the extracted control grid.

Using the interpolated surface, the street layer can be easily extracted by considering areas in the range data, which are at approximately the same height level as the estimated DTM. These difference computations result in a rough ground layer segmentation. Non-street related regions, such as grassland or water bodies located at the same height level are still included. Therefore, the estimated mean color models are used to refine the segmentation result by evaluation of the Bhattacharyya distances for all pixels in the aerial image. A morphological post processing is

performed to add isolated areas to the street layer.

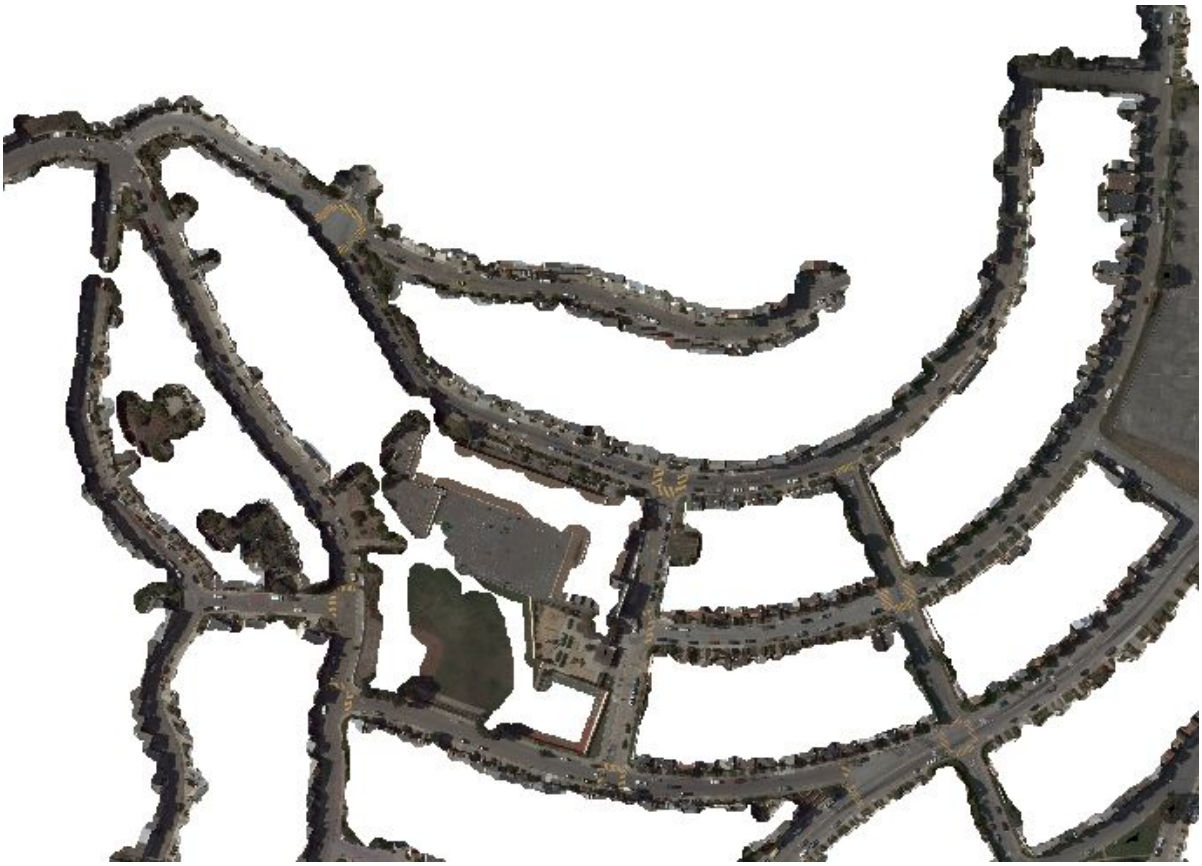

Figure 4, Extracted street layer using the interpolated DTM and color models of a San Francisco scene.

Experimental Results

For a quantitative evaluation of the car detection, we use the recall-precision curves (RPC) (Agarwal, 2004). We manually establish a reference set of cars, representing the ground truth with #nP cars. Of the total detected cars, #TP are the true positives and #FP the false positives. The precision rate (PR) shows the accuracy of the prediction of the positive class. The recall rate (RR) shows how many of the known total number of positive samples we are able to identify. The F-Measure (Fm) is the harmonic mean between RR and PR, a type of average between recall and precision rate. For a visual evaluation of the detector, we plot RR against 1-PR.

$$PR = \frac{\#TP}{\#TP + \#FP} \qquad RR = \frac{\#TP}{\#nP} \qquad Fm = \frac{2 \cdot RR \cdot PR}{RR + PR}$$

We accept a detection as TP if the center of the detection corresponds to the annotated ground truth car with a maximum city block distance of approximately 1.8 meters. In addition we require that the detected orientation to be within 16 degrees of the ground truth. These values remain constant for all experiments.

Data Set

Our initial experience includes the aerial imagery of the city of Graz in Austria and a scene of San Francisco.

The images were acquired by the UltraCam camera. The camera produces 4 color channels in red-green-blue-near infrared, and the images used initially have a ground resolution of 8 cm and 15 cm, respectively.

The fixed GSD of the aerial images supports a fixed-size rectangle to reflect the size of a typical passenger car. That rectangle covers the car in its center and some surrounding texture. This is to include some context information of a car so that the car is analyzed with its surrounding background. Usually the boundary of a car is a rectangle with a length twice its width. In our case, we have chosen the patch size to be 2.8 meters x 5.6 meters. The initial training and testing sets are selected on two non-overlapping parts of the aerial images.

Car Detection

The classifier is improved on-line by the user. During training we label 1250 samples, of which 500 are positive, each sample containing a car, and 750 are negative, each showing diverse background patches. The more informative the samples are, the faster the system learns. Moreover, the training samples can be diversified and adjusted during training to capture the variability of the real images. After training we apply the trained classifiers to our test scenes.

Results

Figure 6 presents the detection results using an extracted street layer for the Graz and the San Francisco data set. We processed images with 6500 x 4500 pixels and 4500 x 3000 pixels, respectively. The scene of Graz includes 618 cars and the San Francisco data set contains 302 cars. Note, for the street layer extraction process we draw the best car detections. Figure 6 represents a visual result of recognized cars using an on-line classifier.

Figure 5, Visual detection results on a Graz data set scene. The blue rectangles indicate a true positive car detection.

For the detection in the Graz data set, we propose a common car size of 35 x 70 pixels. In contrast to the Graz scene, we train the San Francisco classifier with a size of 25 x 45 pixels. The reduced amount of detail, by using a 25 x 45 pixels patch, results in a slightly decreased recall rate as shown in Figure 5. The street layer extraction

method is compared to the refined classification results as proposed in (Zebedin, 2006). For a quantitative evaluation of the street layer extraction, again the RR and the PR for pixels are used. The PR measures the areas which are falsely classified as street layer, while the RR measures the right classified areas. Experiments show that more than 95% of all pixels are correctly classified as street layer. Here, the PR is of minor importance for the improving the car detection process. In Figure 7, some visual car detection results with and without using a street layer mask are shown.

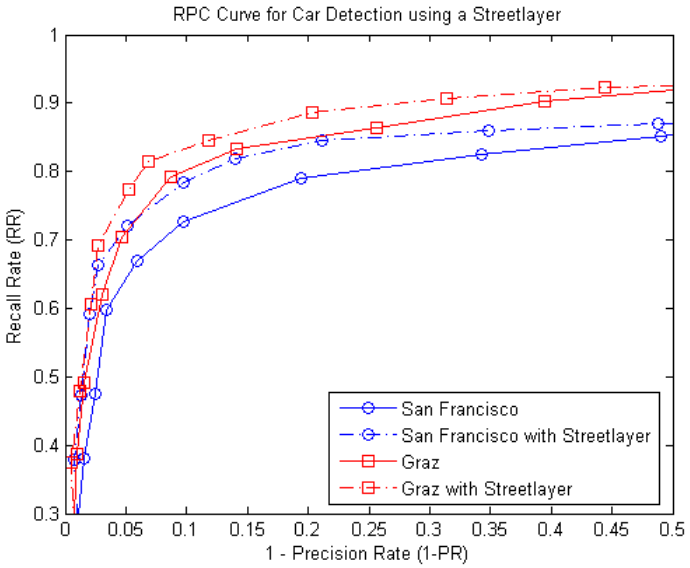

Figure 6, RPC curve showing the car detections results using the extracted street layer.

CONCLUSION

We are making the case for the development of rich semantic knowledge about the objects presented in aerial digital photography. We believe that this is the "next big step" in automatic 3D modeling of the human habitat. The semantic information consists of labels attached to objects in the scene. This is a classical task of object recognition/classification. The paper briefly reviews how this task has been tremendously advanced by the vision research community in recent years. We believe that automatic and robust procedures can be developed for recognizing most common objects in aerial photography, such as people, cars, trees, buildings, streets, chimneys, skylights etc. We refer to these as "human scale objects". Some of these objects must be maintained in a 3D model of an urban environment, others are of such a transient nature that they are to be removed. And some can be replaced by generic models rather than a specific 3D representation of the specific object, if that object is for example a tree or shrub. Semantic information will have a significant effect on many tasks. We think of the way we search in images, how we can compress, publish and broadcast them on the Internet, how urban scenes get visualized.

Our efficient car detector exploits an on-line Boosting algorithm for training the detector. We use on-line learning with human interaction to build a suitable detector from only a few labeled samples, and we can successively improve the detector to achieve satisfactory results. We believe that our procedure can be applied to a range of different objects, not just cars. Furthermore, this paper proposed a strategy to reduce the number of false positive detections of the car detection process by introducing an extracted street layer as context information. We have shown how the detection of cars and available height information can be used to extract a layer, where the detected cars can definitely appear. A color model of the street facilitates the correct modeling of the ground. The application of the street layer to the car detection process results in a decrease by a factor of 2 of the FPR. Additionally, we used the high image redundancy to improve the generation of the DTM.

Our hope for further improvements is nourished by the availability of overlapping images, typically 5 images per ground patch. We envision an approach that automatically improves the detector in an unsupervised fashion. The image overlaps will help to obtain increased car detection rates and improve the street layer extraction method.

We speculate that a combination of approaches will achieve a better performance and also a greater level of generalization towards use for various objects and types of aerial imagery.

ACKNOWLEDGMENTS

This work has been supported by the APAFA Project No. 813397, financed by the Austrian Research Promotion Agency (http://www.ffg.at) and the Virtual Earth Academic Research Collaboration funded by Microsoft.

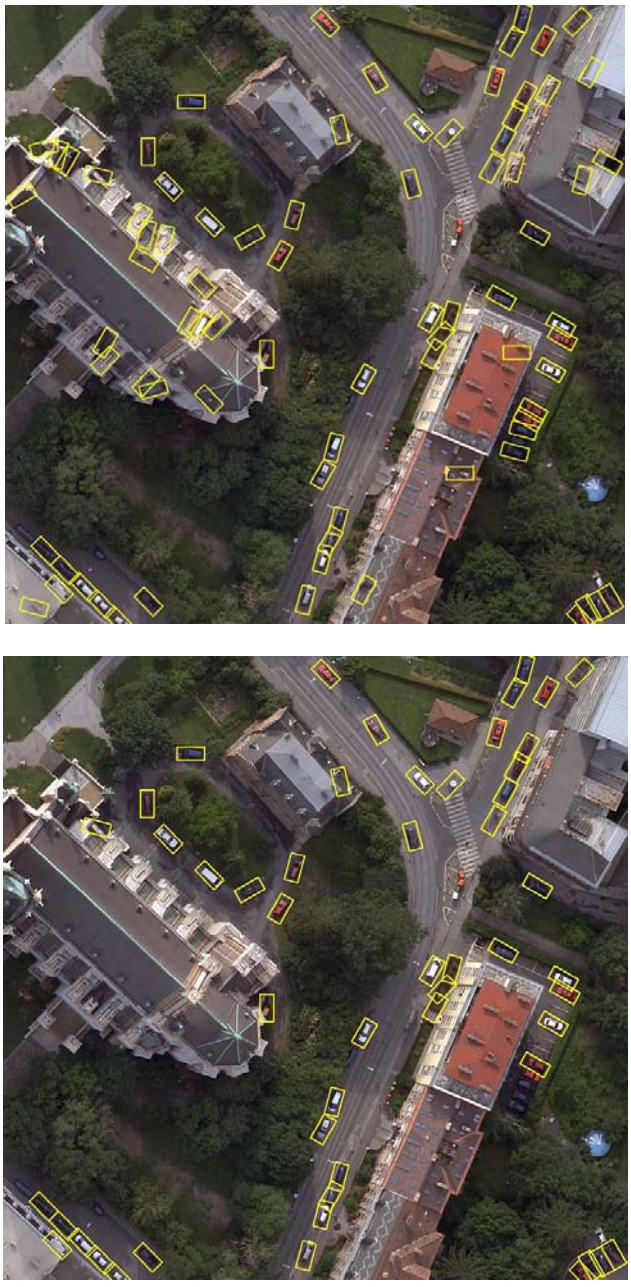

Figure 7, Visual results: the top scene demonstrates the detected cars without using a street layer,

the lower scene shows the improved car detection with reduced false positive detections.

REFERENCES

S. Agarwal, A. Awan, and D. Roth (2004). Learning to detect objects in images via a sparse, part-based representation. PAMI, 26(11):1475–1490.

F. L. Bookstein (1989). Principal warps: thin-plate splines and the decomposition of deformations. PAMI, 11(6):567–585

D. Comaniciu and P. Meer (1999). Mean shift analysis and applications. In Proceedings ICCV, pages 1197–1203.

N. Cornelis, B. Leibe, K. Cornelis, and L. van Gool (2006). 3d city modeling using cognitive loops. In Third International Symposium on 3D Data Processing, Visualization and Transmission, 3DPVT

Edsger W. Dijkstra (1959). A note on two problems in connexion with graphs. Numerische Mathematik, 1:269–271.

M. Donoser, and H. Bischof (2007). Roi-seg: Unsupervised color segmentation by combining differently focused sub results. In Proceedings CVPR.

R. Fergus, P. Perona, and A. Zisserman (2003). Object class recognition by unsupervised scale-invariant learning. In In Proceedings CVPR, pages 264–271.

W. Foerstner (2006). Etrims. http://www.ipb.uni-bonn.de/projects/etrims.

Y. Freund and R. Schapire (1997). A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences, 55(1):119–139.

H. Grabner and H. Bischof (2006). On-line boosting and vision. In Proceedings CVPR, pages 260–268.

A. Klaus, M. Sormann, and K. Karner (2006). Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In Proceedings ICPR, pages 15–18.

K. Levi and Y. Weiss (2004). Learning object detection from a small number of examples: The importance of good features. In Proceedings CVPR, pages 53–60.

N. Littlestone (1987). Learning quickly when irrelevant attributes abound. Machine Learning, 2:285–318.

T. Ojala, M. Pietikäinen, and T. Mäenpää (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. PAMI, 24(7):971–987.

A. Opelt, M. Fussenegger, A. Pinz, and P. Auer (2004). Weak hypotheses and boosting for generic object detection and recognition. In Proceedings ECCV, pages 71–84.

N. Oza and S. Russell (2001). Online bagging and boosting. In Proceedings Artificial Intelligence and Statistics, pages 105–112.

J.-H. Park and Y.-K. Choi (1996). On-line learning for active pattern recognition. In IEEE Signal Processing Letters, pages 301–303.

G. Pacher, S. Kluckner, and H. Bischof (2008). An Improved Car Detection using Street Layer Extraction, In Proceedings Computer Vision Winter Workshop., pages 1-8.

J. Ponce, M. Herbert, C. Schmid, and A. Zisserman (2006). Toward Category-Level Object Recognition. Springer.

H. Rowley, S. Baluja, and T. Kanade (1998). Neural network-based face detection. PAMI, 20(1):23–38.

K. Tieu and P. Viola (2000). Boosting image retrieval. In Proceedings CVPR, pages 228–235.

V. Vapnik (1995). The Nature of Statistical Learning Theory. Springer.

P. Viola and M. Jones (2001). Rapid object detection using a boosted cascade of simple features. In Proceedings CVPR, pages 511–518.

P. Viola, M.J. Jones (2003). Detecting pedestrians using patterns of motion and appearance. In Proceedings ICCV, pages 734–741.

T. Werner and A. Zissermann (2002). New techniques for automated architecture reconstruction from photographs. In Proceedings of the 7th European Conference on Computer Vision, pages 541–555. Springer.

B. Wu and R. Nevatia (2007). Simultaneous object detection and segmentation by boosting local shape feature based classifier. In Proceedings CVPR, pages 1180–1187.

L. Zebedin, A. Klaus, B. Gruber-Geymayer, and K. Karner (2006). Towards 3d map generation from digital aerial images. Int. Journal of Photogrammetry and Remote Sensing, pages 413–427.