Enhancing Road Maps by Parsing Aerial Images Around the World

Gellért Máttyus
Remote Sensing Technology Institute
German Aerospace Center
gellert.mattyus@dlr.de

Shenlong Wang, Sanja Fidler and Raquel Urtasun
Department of Computer Science
University of Toronto
{slwang, fidler, urtasun}@cs.toronto.edu

Abstract

In recent years, contextual models that exploit maps have been shown to be very effective for many recognition and localization tasks. In this paper we propose to exploit aerial images in order to enhance freely available world maps. Towards this goal, we make use of OpenStreetMap and formulate the problem as the one of inference in a Markov random field parameterized in terms of the location of the road-segment centerlines as well as their width. This parameterization enables very efficient inference and returns only topologically correct roads. In particular, we can segment all OSM roads in the whole world in a single day using a small cluster of 10 computers. Importantly, our approach generalizes very well; it can be trained using only $1.5\ km^2$ *aerial imagery and produce very accurate results in any location across the globe. We demonstrate the effectiveness of our approach outperforming the state-of-the-art in two new benchmarks that we collect. We then show how our enhanced maps are beneficial for semantic segmentation of ground images.*

1. Introduction

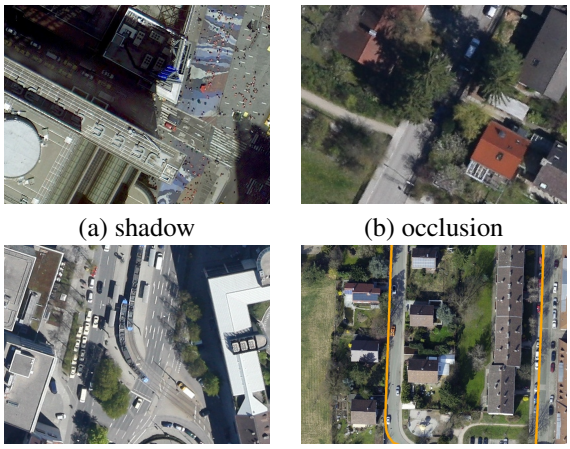
Over the past decades many contextual models have been developed to improve object recognition [41, 20, 24, 18, 19, 14, 15, 6, 10, 31, 39, 11, 16, 3, 13]. Particularly successful are approaches that use maps to improve localization [22], layout estimation [22] and holistic scene understanding [36]. Most self-driving cars (*e.g.*, Google car, participants of the DARPA urban challenge) rely on detailed maps of the environment to facilitate navigation and perception. These maps are typically obtained via costly manual intervention, limiting the applicability of current approaches.

An alternative are online resources such as the OpenStreetMap (OSM) project [1], which contains a cartographic map of the road topology with good coverage over almost the full world, with around 33,968,739 km of road data. This is advantageous as it is freely available on the web and the quality and quantity of the annotations are growing over time, as more users contribute to the project. However, the map information is noisy and partially missing as for example most roads do not contain information about their width.

In this paper we proposed to exploit aerial images in order to enhance open-source maps (*e.g.*, with road geometry). This is not an easy task as despite decades of research, large-scale automatic road segmentation from aerial images remains an open problem. Most approaches either do not deliver a topologically correct road network and/or rely on classifiers that have to be re-trained for each location in order to properly capture appearance variations. As a consequence they require tedious manual annotation for each region of the globe to be segmented. This annotation task takes around 8 hours per km^2, therefore, current approaches focus on a small set of locations.

In contrast, instead of framing the problem as semantic segmentation, we propose to use OpenStreetMap (OSM) to formulate the problem as inference in a Markov random field (MRF) which is directly parameterized in terms of the centerline of each OSM road segment as well as its width. This parameterization enables very efficient inference and returns the same topology as OSM. In particular, we can segment the OSM roads of the whole world in only 1 day when using a small cluster of 10 computers. Furthermore, our approach can be trained using only $1.5\ km^2$ of aerial imagery over Germany and is able to generalize to the entire world and produce state-of-the-art results without any further manual interaction. As we reason about the location of the centerline, we can handle and correct OSM mistakes as well as geo-localization/projection errors. This is not an easy task as illustrated in Fig. 1 due to shadows, occlusions and misalignments. Our energy encodes the appearance of roads, edge information, car detection, contextual features, relations between nearby roads as well as smoothness between the line segments. All our energy terms can be computed very efficiently via local, non-axis aligned integral images. Learning can also be done very efficiently using structured SVMs [33] taking 1 minute on a desktop

[1] www.openstreetmap.org



(a) shadow (b) occlusion

(c) vehicles (d) misaligned centerline

Figure 1. Road segmentation is challenging due to shadows, occluding trees and vehicles which make the appearance heterogeneous as well as OSM/projection misalignment errors.

computer.

The coverage of OSM is very high in most areas, and thus by employing our parameterization we did not miss roads in our datasets. We had to exclude lower road categories as they include forest tracks and pedestrian areas, which are not sufficiently visible in the aerial images. In other regions of the globe the coverage is not as dense and our approach might miss some roads. We refer the reader to the OSM project[2] for a more detailed explanation of the coverage and its growth, and [3] for a comparison with other maps. Detecting new roads that are missing in OSM is our plan for future work.

We demonstrate the effectiveness of our approach by extracting road information from aerial images from different camera sensors taken around the whole world (*e.g.*, Toronto, Sydney, New York, Manila, Nairobi). Importantly, we only employ $1.5\ km^2$ imagery over Germany captured by one camera sensor for training, illustrating the ability of our approach to generalize (domain adaptation). The aerial image datasets we are aware of are not labeled with the geometric information we want to extract. They either consider the road as a single centerline or label the other surfaces instead, *e.g.*, the ISPRS [4] contains the "impervious surfaces" class but no roads. Therefore we collect two new datasets namely Bavaria and aerial KITTI, which we manually annotate and show that our approach significantly outperforms all competitors. We then demonstrate the usefulness of our road priors for the task of semantic segmentation on KITTI ground images, and show that we can provide better cartographic priors than [36]. We will release code and datasets to reproduce all results on the paper.

[2] http://wiki.openstreetmap.org/wiki/Stats

[3] http://tools.geofabrik.de/mc

[4] http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html

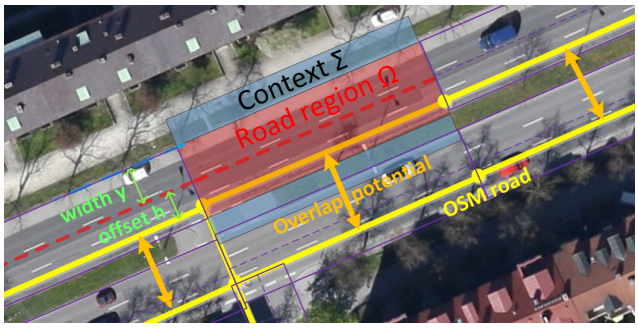


Figure 2. Illustration of the road centerline with the width parameterized by the center offset h and symmetrical width y. The direction and length of the rectangle Ω_i is defined by the p_i, p_{i-1} points given by the street database. The context is depicted as Σ.

2. Related Work

Road segmentation in aerial images has draw a lot attention for decades in the computer vision and remote sensing communities. However, it still remains an open-problem due to the difficulties in handling appearance variations and producing topologically correct segmentations. Early approaches search for objects that fulfill a pre-defined criteria. [2] defines a geometric-stochastic model and estimates the roads by tiling the input image. [32] use a Point Process to simulate and detect a network of connected line segments. We refer the reader to [25] for a detailed literature review and comparison. These approaches, however, share a common drawback: they require manual parameter tuning. Learning based methods have been proposed to be more robust to appearance variations. Mnih and Hinton [26] proposed a two stage approach, where first a neural net is used to label patches independently. Road topology is then corrected using a post-processing step. This was extended in [27] to deal with noisy training labels by employing a robust loss function. However, this method suffers from block effects due to the patch-based prediction. [37] model the road classification as a CRF, where the high-order cliques are sampled over straight segments or junctions to maintain a road-like network structure. In [28] height-field contextual information captured from dense stereo matching is used to improve segmentation. This approach is computationally very expensive and results were shown in a single location. [4] sample graph junction-points using image consistency and shape priors, resulting in long computation times (4 min/image). [34] formulate the delineation of linear loopy structures as an Integer program. However, only simple suburban scenes were tackled.

Map information has been used in both computer vision and robotics communities. Aerial image and land cover attribute maps are exploited in [21] for single image geolocalization. Kalogerakis *et al.* [17] built a human travel prior from maps to geolocalize time-stamped photographs. Brubaker *et al.* [3] use road networks for self-localization.

Figure 3. Output of the car detector. This task is challenging due to the small resolution of the target objects. The left image was captured from Google Earth while the right is part of the Bavaria dataset which was used for training the detector.

In [24] various maps of New York city were used to detect and localize cars from ground images. [22] use floor plans to localize and reconstruct in 3D single images in apartments. [36] use OSM to generate a geographic prior for outdoor holistic scene understanding improving performance in 3D object detection, pose estimation, semantic segmentation and depth reconstruction.

In [30] the road segmentation in aerial images is formulated as a weakly supervised classification problem, in which superpixels that overlap with road vector data are adopted as positive samples. However, inaccuracies of the road vector data are not taken into account and the solution does not preserve topology. [40] consider road segmentation as a width estimation problem. By analyzing the spatial distribution of superpixel boundaries along the direction of the road, the road width is retrieved for each line segment independently. However, their approach is not robust to shadows and occlusion.

3. Enhancing Road Maps from Aerial Images

In this section we show how to enhance world maps by parsing aerial images. In particular, we frame the problem as the one of inference in a Markov random field where the noisy cartographic map is employed to directly parameterize the problem. This parameterization is very robust and enables efficient inference.

3.1. Energy Formulation

In OSM, each road centerline is defined as a polyline chain (*i.e.*, piece-wise linear curve) but no information about the road width is typically available. Unfortunately OSM roads are not very accurate as they are either edited by volunteers without explicit quality control, or computed automatically from GPS trajectories. Furthermore, geo-localization and projection errors make the vertices of the polyline poorly aligned with the center of the road in aerial images. We refer the reader to Fig. 1 for an illustration of the difficulties of the problem. Thus we re-reason about their true location. Given a geo-localized aerial image, we model each road with a set of random variables representing for each vertex of the polyline an offset in the normal direction as well as the width of the road segment. We refer the reader to Fig. 2 for an illustration of our parameterization.

More formally, let $\mathbf{h}^j = \{h_1^j, \cdots, h_{l_j}^j\}$ be a set of random variables encoding the offsets of each vertex of the polyline that defines the j-th road, where l_j is the number of vertices for that road and $h_i^j \in [-30, 30]$ pixels. Our images have a resolution of 13 cm/pixel. Denote $\mathbf{y}^j = \{y_1^j, \cdots, y_{l_j}^j\}$ the width of each segment that compose the j-th road, with $\mathbf{y}^j \in [24, 50]$ pixels. Note that the hypothesis spaces for h and y are defined based on our empirical estimate of maximal road width and OSM projection error. Further, let $\mathbf{h} = \{\mathbf{h}^1, \cdots, \mathbf{h}^L\}$ and $\mathbf{y} = \{\mathbf{y}^1, \cdots, \mathbf{y}^L\}$ be the set of offsets and widths for all roads respectively. Denote $\mathbf{x}$ the input aerial image. We define the energy of our road segmentation as a sum of potentials encoding the image evidence, the presence of car detections, smoothness between widths and offsets of consecutive road segments and overlap constraints between nearby parallel roads

$$
\begin{aligned}
E(\mathbf{h}, \mathbf{y}) = & \sum_{j=1}^{L} \sum_{i=1}^{l_j} \mathbf{w}_{road}^T \phi_{road}(h_i^j, y_i^j, \mathbf{x}) \\
& + \sum_{j=1}^{L} \sum_{i=1}^{l_j} \mathbf{w}_{ap}^T \phi_{ap}(h_i^j, y_i^j, \mathbf{x}) + \sum_{j=1}^{L} \sum_{i=1}^{l_j} \mathbf{w}_{car}^T \phi_{car}(h_i^j, y_i^j, \mathbf{x}) \\
& + \sum_{j=1}^{L} \sum_{i=1}^{l_j - 1} \mathbf{w}_{sm}^T \phi_{sm}(h_i^j, y_i^j, h_{i+1}^j, y_{i+1}^j) \\
& + \sum_{i,j,k,m \in P} \phi_{ol}(h_i^j, y_i^j, h_k^m, y_k^m) \qquad (1)
\end{aligned}
$$

Note that the overlap energy does not have a weight as it is a hard constraint. We use three types of appearance features: distance to edges, homogeneity of the region as well as its context, *i.e.*, $\phi_{app} = [\phi_{edge}, \phi_{hom}, \phi_{context}]$. We now describe our potentials in more details.

Road classifier: We employ a road classifier to compute for each pixel the likelihood of being road/non-road. The potential for each segment $\phi_{road}(h_i^j, y_i^j)$ is simply the sum of the likelihoods of all pixels in the non-axis aligned rectangle Ω_i^j defined by h_i^j, y_i^j (see Fig. 2 for an example).

$$
\phi_{road}(h_i^j, y_i^j) = \sum_{p \in \Omega_i^j(h_i^j, y_i^j)} \varphi(p) \qquad (2)
$$

with $\varphi(p)$ the classifier score at pixel p. Note that this can be very efficiently computed using non-axis aligned integral images. Since we know the orientation of each segment, only a single integral image is necessary per segment. The integral image is also local to the segment, as the hypothesis space covers regions near the original OSM vertices.

Method	*Bavaria*						*Aerial KITTI*					
	IoU		F1		$\overline{\Delta h}$[m]	$\overline{\Delta y}$[m]	IoU		F1		$\overline{\Delta h}$[m]	$\overline{\Delta y}$[m]
	GT	Oracle	GT	Oracle			GT	Oracle	GT	Oracle		
Road Unary [38]	49.7	48.3	66.4	65.1	–	–	32.8	31.2	49.4	47.6	–	–
OSMxSeg	61.6	60.6	76.2	75.5	–	–	50.3	48.8	67.0	65.6	–	–
FSeg [40]	63.0	65.3	77.3	79.0	2.11	1.15	55.4	58.6	71.3	73.9	2.32	1.25
OSMFixed	64.7	66.9	78.6	80.2	1.75	1.45	51.0	53.8	67.6	70.0	2.38	1.21
Ours	**73.5**	**77.2**	**84.8**	**87.2**	**1.30**	**0.97**	**71.8**	**77.5**	**83.6**	**87.4**	**0.91**	**0.79**
Oracle	86.5	100	92.7	100	0	0	84.2	100	91.4	100	0	0

Table 1. Performance of our method vs baselines. The IoU and F1 values are in %, while $\overline{\Delta h}$, $\overline{\Delta y}$ are the mean absolute error of the offset and width measured in meters.

Method	*Bavaria*						*Aerial KITTI*					
	IoU		F1		$\overline{\Delta h}$[m]	$\overline{\Delta y}$[m]	IoU		F1		$\overline{\Delta h}$[m]	$\overline{\Delta y}$[m]
	GT	Oracle	GT	Oracle			GT	Oracle	GT	Oracle		
Road+Edge+Car	72.2	75.7	83.8	86.2	1.57	1.10	70.7	76.3	82.8	86.5	1.05	0.84
Road+Edge+Car+‖	72.8	76.4	84.2	86.7	1.39	1.03	**71.8**	**77.6**	**83.6**	**87.4**	**0.91**	**0.79**
Edge+Hom+Context+Car	64.8	68.4	78.6	81.2	1.58	1.26	63.6	67.2	77.8	80.3	1.61	1.36
Edge+Hom+Context+Car+‖	69.7	72.6	82.1	84.2	1.52	1.09	63.5	67.4	77.7	80.5	1.49	1.26
All	73.0	76.2	84.4	86.5	1.51	1.08	71.2	76.8	83.2	86.9	1.05	0.84
All+‖	**73.5**	**77.2**	**84.8**	**87.2**	**1.30**	**0.97**	**71.8**	77.5	**83.6**	**87.4**	**0.91**	**0.79**
	Domain shift (train on one dataset, test on the other)											
Road+Edge+Car	70.0	74.3	82.4	85.2	1.45	1.06	66.0	71.0	79.5	83.0	1.33	0.89
Road+Edge+Car+‖	70.7	75.2	82.8	85.8	1.30	0.99	66.8	72.0	80.1	83.7	1.18	0.83
Edge+Hom+Context+Car	69.1	71.5	81.7	83.4	1.73	1.11	59.3	63.5	74.4	77.6	1.63	1.17
Edge+Hom+Context+Car+‖	70.4	73.4	82.7	84.6	1.43	0.98	62.0	65.7	76.5	79.3	1.57	1.36
All	70.8	75.1	82.8	85.8	1.37	1.02	**67.7**	72.8	**80.7**	84.3	1.20	0.84
All+‖	**71.7**	**76.1**	**83.5**	**86.4**	**1.27**	**0.93**	**67.7**	**73.2**	**80.7**	**84.6**	**1.08**	**0.79**

Table 2. Performance on Bavaria and Aerial KITTI with various features configurations. The IoU and F1 values are in %, while $\overline{\Delta h}$, $\overline{\Delta y}$ are the mean absolute error of the offset and width measured in meters. The ‖ symbol denotes the overlap potential between parallel roads.

Edge: We expect the boundaries of road segments to match image appearance boundaries. Towards this goal, we compute edges using the line detector of [7], and define the potential as the distance d from each rectangle boundary pixel to the closest image edge

$$\phi_{edge}(h_i^j, y_i^j) = \sum_{p \in \partial\Omega_i^j(h_i^j, y_i^j)} \min_{e \in \mathcal{E}} d(p, e) \qquad (3)$$

with $\partial\Omega$ the boundary of the rectangle Ω, p a pixel and $\mathcal{E}$ the set of all lines returned by the line detector. We adopt the distance transform of [8] to accelerate the computation.

Object detector: We train a car detector using the detector of [23]. Note that this task is extremely challenging as on average a car has only 30×12 pixels (see Fig. 3). We form a 2D feature for each car by computing $[s \cdot \sin(\Delta\alpha), s \cdot \cos(\Delta\alpha)]$, with $\Delta\alpha$ the angle between the segment and the car and s the confidence of the detector. The car potential ϕ_{car} is simply the sum of the features of all the detected cars that are inside the rectangle. Given the car features, the potentials can be computed efficiently using accumulators in a local region around each segment.

Homogeneity: An important property of roads is that they are typically free of obstacles (otherwise we could not drive on them) and therefore we expect their appearance to be homogeneous. This is violated if there are vehicles, shadows or if our aerial view of the road is obstructed by trees, bridges or tunnels. In those cases we expect the other potentials to correct the mistakes. We capture homogeneity by first transforming the image into Luv space and computing for each channel the standard deviation of the appearance inside the rectangle. This potential can be efficiently computed using two non-axis aligned integral images per channel: one computing the sum of intensities and the other the sum of square intensities. Note that this calculation was used in [35] to normalize the Haar-like features in a subwindow.

Context features: This feature encodes the fact that the road looks different than the area around it. Similar to [35], we compute the difference between the means of pixel intensities in the context and road rectangles, Σ_i^j and Ω_i^j respectively (see Fig. 2). The potential is computed by aggregating the difference across all Luv channels. Again, we use integral images for efficiency.

(Toronto: Pearson Airport) (NYC: Times square) (Nairobi, Kenya)*

(Manila, Philippines) (Mexico City) (Kyoto: Kinkakuji)

(Sydney: At Harbour bridge) (St. Moritz, Switzerland) (Manaus, Brazil)*

Figure 4. Segmentation results on several cities over the world using the edge, homogeneity, context, car and overlap features. Note that the MRF was trained only on 1.5 km^2 imagery from the Aerial KITTI dataset. * Indicates satellite image.

Smoothness: The widths and offsets along the same road tend to be similar in nearby segments. Our smoothness potentials for both h and y are defined between consecutive segments along the same road as a weighted sum of ℓ_0 and ℓ_2 norms.

Overlap: This is a hard constraint encoding the fact that two parallel roads can not overlap. We enforce this for all roads that have similar orientation (within 20 degrees) and are close enough that they could overlap.

3.2. Inference

Inference in our model can be done by computing the minimum energy configuration

$$\{\mathbf{h}^*, \mathbf{y}^*\} = \text{argmin}_{\mathbf{h},\mathbf{y}} E(\mathbf{h}, \mathbf{y}) \tag{4}$$

with $E(\mathbf{h}, \mathbf{y})$ the total energy defined in Eq. (1). Note that due to the overlap constraint, the graphical models might contain loops. As a consequence exact inference is not possible. When there is no overlap, the graphical model is composed of set of chains and dynamic programing yields the exact solution. Inspired by the stereo work of [5], we employ block coordinate descent (BCD) to perform approximated inference. Towards this goal, we define each block in BCD to form a chain since we can then solve each step to optimality. We then alternate between going over all horizontal and vertical chains to propagate the information. Note that since we solve each sub-step to optimality this procedure is guaranteed to converge. We refer the reader to Fig. 5 for an illustration, where to simplify the figure we have collapsed the width and offset variables in a single variable $g_i^j = (h_i^j, y_i^j)$. It is important to note that each of the BCD steps (*i.e.*, optimization over a subset of variables) involve conditioning, and thus the pairwise potentials between a variable in the chain and a connected variable not in the chain are folded as unaries. Prior to BCD, we initialize all variables by performing inference along each road chain and ignoring the connections between nearby parallel roads. We refer the reader to Algorithm 1 for more details about the block coordinate descent.

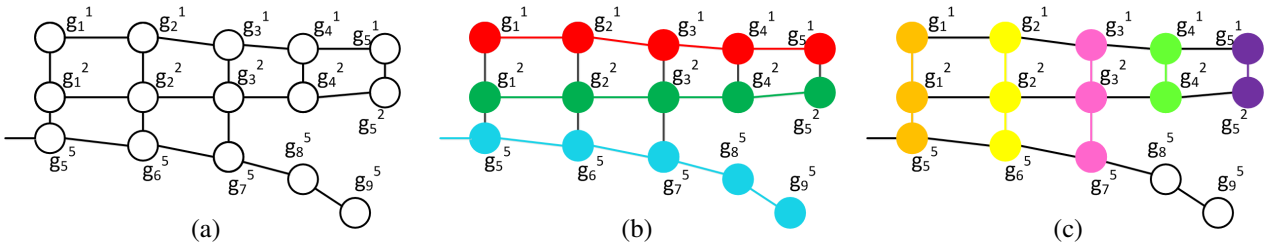


Figure 5. Illustration of our BCD inference. Note that $g_i^j = (h_i^j, y_i^j)$. (a) Graphical model consisting of 3 roads that have overlapping constraints (*i.e.*, vertical edges). We alternate between performing inference (b) over each road one at a time (red, green, blue), and (c) along chains on the vertical direction encoding the horizontal constraints, also one at a time (orange, yellow, pink, green, purple). Note that these operations involve conditioning, and thus the pairwise potentials between a variable in the chain and a connected variable not in the chain are folded as unaries.

Algorithm 1 Block coordinate descent inference (BCD)

1: Initialize $(\mathbf{h}, \mathbf{y})$ by minimizing Eq. (1) ignoring the overlap potentials
2: **repeat**
3: **for** all roads R_j **do**
4: Minimize Eq. (1) w.r.t $\mathbf{h}^j, \mathbf{y}^j$ holding the rest fixed.
5: **end for**
6: **for** all overlap chains O_i **do**
7: Minimize Eq. (1) over the variables in the overlap chain
8: **end for**
9: **until** no energy reduction or max number iterations

3.3. Learning

We learn the parameters of the MRF using a structural SVM (S-SVM)[33] by minimizing

$$\min_{\mathbf{w}\in\mathbb{R}^D} \frac{1}{2}||\mathbf{w}||^2 + \frac{C}{N}\sum_{n=1}^{N}\xi_n$$
$$\text{s.t. } \delta(\mathbf{h},\mathbf{y}) \geq \Delta(\mathbf{h},\mathbf{y}) - \xi_n, \forall(\mathbf{h},\mathbf{y}) \in \mathcal{H}\times\mathcal{Y}\setminus(y_n,h_n), \forall n \quad (5)$$

with $\delta(\mathbf{h},\mathbf{y}) = E(\mathbf{h},\mathbf{y}) - E(\mathbf{h}_n,\mathbf{y}_n)$ and $\mathcal{H}\times\mathcal{Y}$ the space of all possible labelings for $(\mathbf{h},\mathbf{y})$. Note that our definition is opposite from the one in [33], as we have defined the features in terms of an energy minimization and not a score maximization. We employ the parallel cutting plane implementation of [29] to learn the parameters. We use the intersection-over-union between the configuration and the ground-truth labels as our task loss. This can be computed as a pairwise term, and thus loss augmented inference can be done efficiently.

4. Experimental Evaluation

We perform our experiments on three different datasets: Bavaria, Aerial KITTI and World which were captured with different sensors. Note that we have access to RGB images without any elevation information. We conduct road pixel-wise annotations in all Bavaria and Aerial KITTI images.

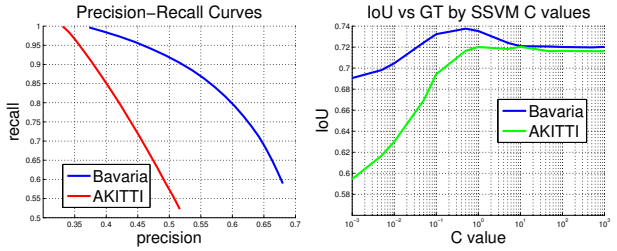


Figure 6. (Left) Precision-recall curve of the road classifier; (Right) Segmentation performance as a function of the structural SVM C parameter.

Note that the parameters not learned by S-SVM were set via four-fold cross-validation.

Bavaria: This dataset is a collection of ortho-rectified aerial images captured by a DSLR camera mounted on a plane flying around the Bavaria region in Germany [5]. It covers urban, suburban and rural areas with motorways. The resolution is 13 cm/pixel on the ground. The total area is 4.95 km^2 containing 103 km of road.

Aerial KITTI: This dataset consists of aerial images downloaded from Google Earth Pro over the city of Karlsruhe, Germany, covering the same area as the KITTI tracking benchmark [12]. The total area is 5.96 km^2 with 84 km of road. We resampled the images to be 13 cm/pixel resolution to be consistent with the Bavaria dataset.

World: This dataset consists of aerial images downloaded from Google Earth Pro of landmarks all over the world, including metropolitan areas in Toronto, New York, Sydney, Mexico City, Manaus, *etc.*, as well as rural areas in St. Moritz and Kyoto. For this dataset there is no annotation.

We use four metrics to measure performance: intersection over union, F_1 score, and mean of the absolute error of h and y. We consider two different ground truth labels when evaluating the performance: our human labeled road annotations as well as the maximum achievable score with respect to our model hypothesis, refer to as *Oracle*. The

[5] We will release these images and the ground truth upon publication.

Features	Time (s) per km	
	Accumulator	Inference
Road+Edge+Car	0.07	0.031
Road+Edge+Car+‖	0.069	0.092
All	0.126	0.032
All+‖	0.122	0.095

Table 3. Running time for feature accumulator calculation and inference under various configurations. In sec per km of road.

later can be computed by performing our MRF inference, by replacing our unary potentials with ground truth segmentations. For all quantitative experiments we perform four-fold cross-validation.

To compute our road classifier, we first convert the image to opponent Gaussian color space and extract a dense filter response map, with a filterbank composed of 17 edge-like filters [38]. We oversegment the image using SLIC [1] and calculate the mean and std of the filter responses in each superpixel. We then train a random forest classifier [7] with this $34D$ input feature. Note that this road classifier was used in [37] as unary potential. Fig. 6 shows the Precision-Recall curve of the road classifier on Bavaria and Aerial KITTI.

Comparison to baselines: We compare our approach to four baselines: The first one is the road classifier unary potential of [37], denoted *Road Unary*. The second baseline, denoted as *OSMxSeg*, is computed by segmenting the image into superpixels using [9] and labeling each super pixel as road if it is crossed by a road segment in OSM. We also reproduce the state-of-the-art method of [40], denoted as *FSeg*, which also uses the OSM road data. To illustrate the effectiveness of our cartographic prior, the last baseline, denoted *OSMFixed*, projects OSM into the image and utilizes an empirical estimate of the road width. As shown in Table 1 our approach significantly outperforms all baselines in both Bavaria and aerial KITTI datasets. (see qualitative results in Fig. 8). Fig. 10 shows a comparison to [40].

Importance of the features: Table 2 depicts inference results for different combinations of features. Note that every feature contributes, and good performance can be achieved without using a road classifier. As a consequence, we do not need new training data for each different location in the world as the other features are very robust to appearance changes.

Segmenting the world: Fig. 4 shows qualitative results from the *World* dataset with our model trained only on *AKITTI*. Our model works very well under many complex scenarios even with significant appearance changes, illustrating the generalization capabilities of our approach. Note that no re-training is necessary as we do not use the road classifier in our potentials.

(a) (b) (c)

Figure 7. Failure modes: (a) Missing turn lane intersection. (b) The extracted road is too narrow. (c) Road covered by trees.

	Sky	Build	Road	Sidewalk	Vege	Car
[36]	32.41	59.25	63.01	36.41	7.36	35.65
Ours	32.41	59.10	**78.71**	**41.96**	7.36	35.65

Table 4. Our method improves the geographic priors of [36]. All values are IoU in %.

Domain Adaptation: We next show our method's domain adaptation ability. Towards this goal, we trained one model on Aerial KITTI and evaluate its performance on Bavaria, and vise versa. As shown in Table 2 our algorithm outperforms all baselines despite the fact that it is trained with different imagery. Furthermore, performance drops less than 5% IoU when compared when we train on the same dataset we test on.

Processing time: We implemented our method in C++ without multi-threading and test it on a laptop with an Intel Core i7-4600M processor. As shown in Table 3, our approach takes less than 0.13 s for computing all feature accumulators per km of road and less than 0.1 s per km for inference. The feature computation (road classifier, edge, car detector) relies on external code which takes around 0.1s per km of road. According to this performance, we estimate our algorithm could approximately segment all the OSM roads in the world in 1 day using a small cluster of 10 machines. We use the parallel cutting-plane structured SVM of [29] to learn the parameters of the model. This takes only 1 minute on a desktop computer.

Ground-level Scene Understanding: In this experiment we show that our enhanced maps can be used to improve semantic segmentation of ground images from KITTI. Towards this goal, we replace the road prior used in [36] by the estimations of our method. This improves the geographic unary prior for the road class by 15%, see the Table 4. Qualitative results are shown in Fig. 9.

Failure modes and limitations: Fig. 7 depicts failure modes. (a) At intersections the OSM might not include the turn lanes and our model can not recover from this. (b) In some cases our features/weights are not good for the scene. This is more likely to happen in the strong generalization case. (c) The road can be (partly) covered, and we only extract the visible part of the road. Additional challenges are

(Bavaria: Motorway) (Bavaria: Urban 1) (Bavaria: Urban 2)

(AKITTI 1) (AKITTI 2) (AKITTI 3)

Figure 8. Results on Bavaria and Aerial KITTI.

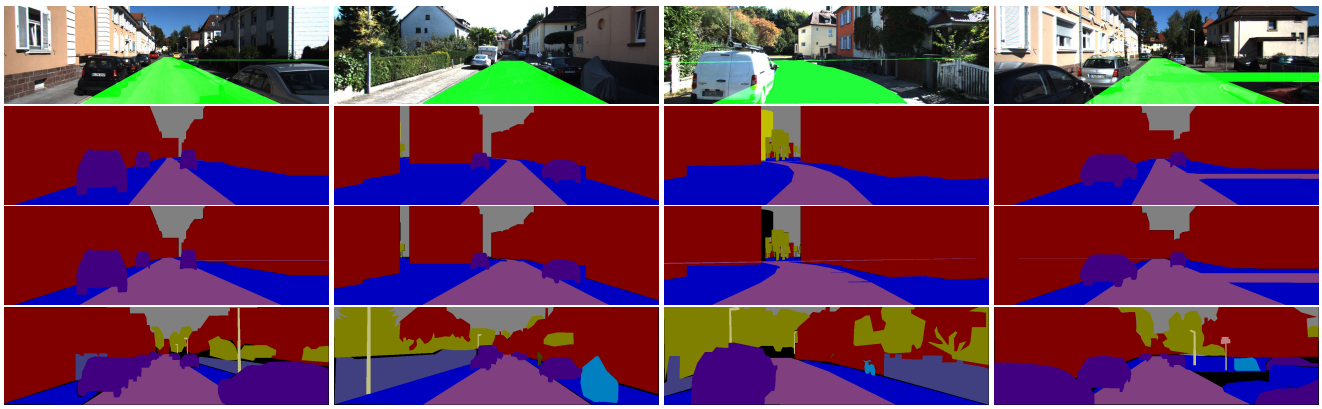


Figure 9. (Top) Our road extracted from aerial images (green) projected into Kitti ground images. (2nd row): Geographical unary of [36]. (3rd row): Geographical unary with our road estimate. (Bottom) Ground truth. Road (pink), sidewalk (blue), building (red), car (purple).

Image [40] Ours Ground truth

Figure 10. Comparison to [40]: Our approach works significantly better than the baseline.

posed by historical city centers were the roads might not be visible as well as developing countries, where only satellite images with much lower resolution than aerial images might be available.

5. Conclusion

We have presented an approach to enhance world maps by parsing aerial images. By parameterizing the problem in terms of OSM road segment centerlines and widths, we were able to extract road properties very efficiently. In particular, we can process the whole world in a single day using a small cluster. Importantly, our approach can be trained with as little as $1.5\ km^2$ aerial imagery from a single area and it is able to generalize to the full world. We have demonstrated the effectiveness of our approach in three different datasets captured by different sensors in different regions of the world.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *PAMI*, 2012. 7

[2] M. Barzohar and D. Cooper. Automatic finding of main roads in aerial images by using geometric-stochastic models and estimation. *PAMI*, 1996. 2

[3] M. A. Brubaker, A. Geiger, and R. Urtasun. Lost! leveraging the crowd for probabilistic visual self-localization. In *CVPR*, 2013. 1, 2

[4] D. Chai, W. Forstner, and F. Lafarge. Recovering line-networks in images by junction-point processes. In *CVPR*, 2013. 2

[5] Q. Chen and V. Koltun. Fast mrf optimization with application to depth reconstruction. In *CVPR*, 2014. 5

[6] S. K. Divvala, D. Hoiem, J. H. Hays, A. A. Efros, and M. Hebert. An empirical study of context in object detection. In *CVPR*, 2009. 1

[7] P. Dollár and C. L. Zitnick. Structured forests for fast edge detection. In *ICCV*, 2013. 4, 7

[8] P. Felzenszwalb and D. Huttenlocher. Distance transforms of sampled functions. Technical report, Cornell University, 2004. 4

[9] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 2004. 7

[10] S. Fidler, S. Dickinson, and R. Urtasun. 3d object detection and viewpoint estimation with a deformable 3d cuboid model. In *NIPS*, 2012. 1

[11] A. Geiger, M. Lauer, C. Wojek, C. Stiller, and R. Urtasun. 3d traffic scene understanding from movable platforms. *PAMI*, 2014. 1

[12] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The kitti dataset. *IJRR*, 2013. 6

[13] J. Hays and A. A. Efros. Im2gps: estimating geographic information from a single image. In *CVPR*, 2008. 1

[14] V. Hedau, D. Hoiem, and D. Forsyth. Recovering the spatial layout of cluttered rooms. In *ICCV*, 2009. 1

[15] D. Hoiem, A. A. Efros, and M. Hebert. Geometric context from a single image. In *ICCV*, 2005. 1

[16] D. Hoiem, A. A. Efros, and M. Hebert. Putting objects in perspective. *IJCV*, 2008. 1

[17] E. Kalogerakis, O. Vesselova, J. Hays, A. A. Efros, and A. Hertzmann. Image sequence geolocation with human travel priors. In *ICCV*, 2009. 2

[18] L. Ladickỳ, C. Russell, P. Kohli, and P. H. Torr. Graph cut based inference with co-occurrence statistics. In *ECCV*. 2010. 1

[19] L. Ladickỳ, J. Shi, and M. Pollefeys. Pulling things out of perspective. In *CVPR*, 2014. 1

[20] L. Ladickỳ, P. Sturgess, K. Alahari, C. Russell, and P. H. Torr. What, where and how many? combining object detectors and crfs. In *ECCV*. 2010. 1

[21] T.-Y. Lin, S. Belongie, and J. Hays. Cross-view image geolocalization. In *CVPR*, 2013. 2

[22] C. Liu, A. Schwing, R. Urtasun, and S. Filder. Rent3d: Floor-plan priors for monocular layout estimation. *CVPR*, 2015. 1, 3

[23] K. Liu and G. Mattyus. Fast multiclass vehicle detection on aerial images. *GRSL*, 2015. 4

[24] K. Matzen and N. Snavely. Nyc3dcars: A dataset of 3d vehicles in geographic context. In *ICCV*, 2013. 1, 3

[25] H. Mayer, S. Hinz, U. Bacher, and E. Baltsavias. A test of automatic road extraction approaches. In *ISPRS*, 2006. 2

[26] V. Mnih and G. E. Hinton. Learning to detect roads in high-resolution aerial images. In *ECCV*, 2010. 2

[27] V. Mnih and G. E. Hinton. Learning to label aerial images from noisy data. In *ICML*, 2012. 2

[28] J. A. Montoya-Zegarra, J. D. Wegner, L. Ladicky, and K. Schindler. Mind the gap: Modeling local and global context in (road) networks. In *GCPR*, 2014. 2

[29] A. G. Schwing, S. Fidler, M. Pollefeys, and R. Urtasun. Box In the Box: Joint 3D Layout and Object Reasoning from Single Images. In *ICCV*, 2013. 6, 7

[30] Y.-W. Seo, C. Urmson, and D. Wettergreen. Exploiting publicly available cartographic resources for aerial image analysis. In *SIGSPATIAL*, 2012. 3

[31] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*. 2012. 1

[32] R. Stoica, X. Descombes, and J. Zerubia. A gibbs point process for road extraction from remotely sensed images. *IJCV*, 2004. 2

[33] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *JMLR*, 2005. 1, 6

[34] E. Turetken, F. Benmansour, B. Andres, H. Pfister, and P. Fua. Reconstructing loopy curvilinear structures using integer programming. In *CVPR*, 2013. 2

[35] P. Viola and M. J. Jones. Robust real-time face detection. *IJCV*, 2004. 4

[36] S. Wang, S. Fidler, and R. Urtasun. Holistic 3d scene understanding from a single geo-tagged image. In *CVPR*, 2015. 1, 2, 3, 7, 8

[37] J. D. Wegner, J. A. Montoya-Zegarra, and K. Schindler. A higher-order crf model for road network extraction. In *CVPR*, 2013. 2, 7

[38] J. Winn, A. Criminisi, and T. Minka. Object categorization by learned universal visual dictionary. In *ICCV*, 2005. 4, 7

[39] J. Yao, S. Fidler, and R. Urtasun. Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation. In *CVPR*, 2012. 1

[40] J. Yuan and A. Cheriyadat. Road segmentation in aerial images by exploiting road vector data. In *COM.geo*, 2013. 3, 4, 7, 8

[41] M. Z. Zia, M. Stark, K. Schindler, and R. Vision. Are cars just 3d boxes?–jointly estimating the 3d shape of multiple objects. In *CVPR*, 2014. 1