TorontoCity: Seeing the World with a Million Eyes

Shenlong Wang[1,2], Min Bai[*,1,2], Gellert Mattyus[*,1,2], Hang Chu[*1] , Wenjie Luo[1,2], Bin Yang[1,2], Justin Liang[1,2], Joel Cheverie[1], Sanja Fidler[1], Raquel Urtasun[1,2]

[1]Department of Computer Science, University of Toronto, [2]Uber Advanced Technologies Group

{slwang, mbai, mattyusg, chuhang1122, wenjie, byang, justinliang, joel, fidler, urtasun}@cs.toronto.edu

Abstract

In this paper we introduce the TorontoCity benchmark, which covers the full greater Toronto area (GTA) with $712.5km^2$ *of land,* $8439km$ *of road and around* $400,000$ *buildings. Our benchmark provides different perspectives of the world captured from airplanes, drones and cars driving around the city. Manually labeling such a large scale dataset is infeasible. Instead, we propose to utilize different sources of high-precision maps to create our ground truth. Towards this goal, we develop algorithms that allow us to align all data sources with the maps while requiring minimal human supervision. We have designed a wide variety of tasks including building height estimation (reconstruction), road centerline and curb extraction, building instance segmentation, building contour extraction (reorganization), semantic labeling and scene type classification (recognition). Our pilot study shows that most of these tasks are still difficult for modern convolutional neural networks.*

1. Introduction

> ”It is a narrow mind which cannot look at a subject from various points of view.”
>
> *George Eliot, Middlemarch*

In recent times, a great deal of effort has been devoted to creating large scale benchmarks. These have been instrumental to the development of the field, and have enabled many significant break-throughs. ImageNet [10] made it possible to train large convolutional neural networks, initiating the deep learning revolution in computer vision in 2012 with SuperVision (commonly refer as AlexNet [17]). Efforts such as PASCAL [12] and Microsoft COCO [19] have pushed the performance of segmentation and object detection approaches to previously inconceivable levels. Similarly, benchmarks such as KITTI [13] and Cityscapes [9] have shown that visual perception is going to be an important component of advanced driver assistance systems (ADAS) and self-driving cars in the imminent future.

However, current large scale datasets suffer from two shortcomings. First, they have been captured by a small set of sensors with similar perspectives of the world, e.g., internet photos for ImageNet or cameras/LIDAR mounted on top of a car in the case of KITTI. Second, they do not contain rich semantics and 3D information at a large-scale. We refer the reader to Fig. 2 for an analysis of existing datasets.

In this paper, we argue that the field is in need of large scale benchmarks that allow joint reasoning about geometry, grouping and semantics. This has been commonly referred to as the three R's of computer vision. Towards this goal, we have created the TorontoCity benchmark, covering the full greater Toronto area (GTA) with $712.5km^2$ of land, $8439km$ of road and around $400,000$ buildings. According to the census, $6.8million$ people live in the GTA, which is around 20% of the population of Canada. We have gathered a wide range of views of the city: from the overhead perspective, we have aerial images captured during four different years as well as LIDAR from airborne. From the ground, we have HD panoramas as well as imagery and LIDAR data captured from a moving vehicle driving around in the city. We are also augmenting the dataset with imagery captured from drones.

Manually labeling such a large scale dataset is not feasible. Instead, we propose to utilize different sources of high-precision maps to create our ground truth. Compared to online map services such as OpenStreetMap [1] and Google Maps, our maps are much more accurate and contain richer meta-data, which we exploit to create a wide variety of diverse benchmarks. This includes tasks such as building height estimation (reconstruction), road centerline and curb extraction, building instance segmentation, building contour extraction (reorganization), semantic labeling and scene type classfication (recognition). Participants can exploit any subset of the data (e.g., aerial and ground images) to solve these tasks.

One of the main challenges in creating TorontoCity was aligning the maps to all data sources such that the maps can produce accurate ground truth. To alleviate this problem, we have created a set of tools which allow us to reduce the labeling task to a simple verification process, speeding up

[*]indicates equal contribution

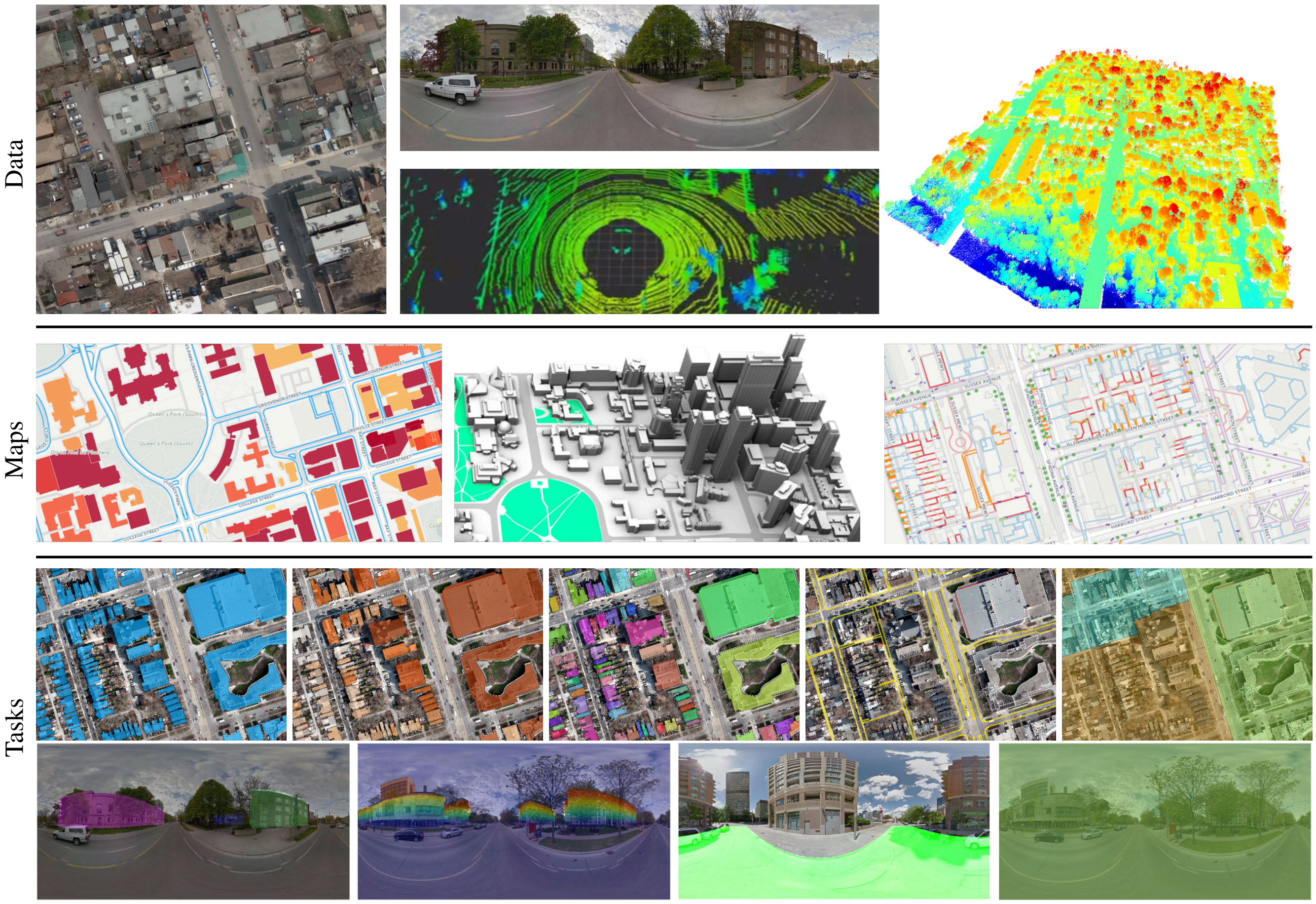


Figure 1: Summary of the TorontoCity benchmark. Data source: aerial RGB image, streetview panorama, street-view LIDAR, airborne LIDAR; Maps: buildings and roads, 3D buildings, property meta-data; Tasks: semantic segmentation, building height estimation, instance segmentation, road topology, zoning segmentation and classification.

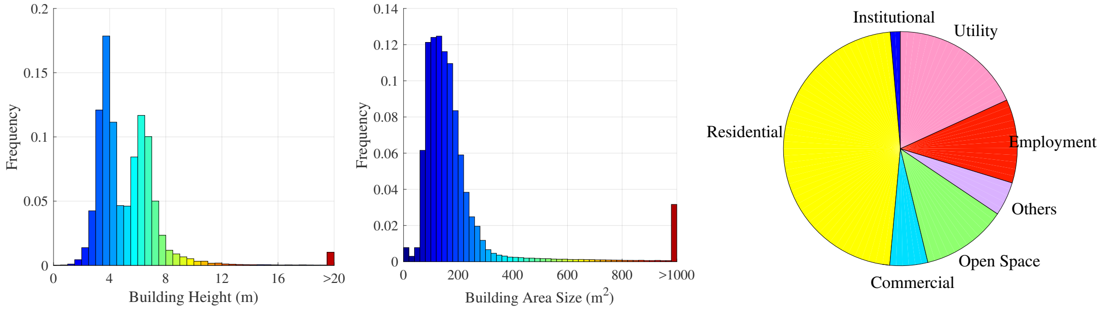
labeling, thus making TorontoCity possible.

We perform a pilot study using the aerial images captured in 2011 as well as the ground panoramas. Our experiments show that state-of-the-art methods work well on tasks such as semantic segmentation and scene classification, however, tasks such as instance segmentation, contour extraction and height estimation remain an open problem. We believe our benchmark provides a great platform for developing and evaluating new ideas, particularly techniques that can leverage different viewpoints of the world. We plan to extend the benchmark in the near future with tasks such as building reconstruction, facade parsing as well as tree, traffic light and traffic sign detection, for which our maps provide accurate ground truth. We have only scratched the surface of TorontoCity's full potential.

2. Related Work

Automatic mapping, reconstruction and semantic labeling from urban scenes have been an important topic for many decades. Several benchmarks have been proposed to tackle subsets of these tasks. KITTI [13] is composed of stereo images and LIDAR data collected from a moving vehicle, and evaluates SLAM, optical flow, stereo and road segmentation tasks. Cityscapes [9] focuses on semantic and instance annotations of images captured from a car. Aerial-KITTI [22] augments the KITTI dataset with aerial imagery of a subset of Karlsruhe to encourage reasoning of semantics from both ground and bird's eye view.

The photometry community has developed several benchmarks towards urban scene understanding [16, 25, 31, 26, 21]. TUM-DLR [16] and ISPRS Multi-Platform [25] benchmarks contain imagery captured through multiple perspectives from UAV, satellite images and handheld cameras. Oxford RobotCar contains lidar point cloud and stereo images captured from a vehicle [21]. However, these benchmarks do not offer any semantic ground-truth for benchmarking purposes. Perhaps the most closely related dataset to ours is the ISPRS Urban classification and building reconstruction benchmark [31], where the task is to extract urban object, such as building, road and trees from both aerial images and airborne laserscanner point clouds. However, this dataset has a relatively small coverage and does not provide ground-view imagery. In contrast, TorontoCity is more



Dataset	ISPRS	TUM-DLR	Aerial KITTI	KITTI	RobotCar	Ours
Location	Vaihingen/Toronto	Munich	Karlsruhe	Karlsruhe	Oxford	Toronto
Aerial Coverage (km^2)	3.49+1.45	8.32	3.23	-	-	712
Ground Coverage (km)	-	<1	<20	39.2	10	>1000(pano)[1]
Aerial RGB	yes	yes	yes	-	-	yes
Aerial LIDAR	yes	yes	-	-	-	yes
Ground Panorama	-	-	-	-	yes	yes
Ground LIDAR	-	yes	-	yes	yes	yes
Aerial Resolution (pixel/cm^2)	8	50	9	-	-	10
Repeats	-	-	-	partial	x10	x4 (aerial)
Top Semantic GT (# of classes)	100% (8)	-	100% (4)	-	-	100% (2 + 8)
Top Geometric GT (source)	dense (lidar)	dense (lidar)	-	-	-	dense
Ground Semantic GT (# of classes)	-	-	dense (4)	object (3)	-	dense (2) / image (6)
Ground Geometric GT (source)	-	-	-	sparse (lidar)	sparse (lidar)	dense (map+lidar)

Figure 2: Statistics of our data and comparison of current state-of-the-art urban benchmarks and datasets.

than two orders of magnitude bigger. Furthermore, we offer many different perspectives through various sensors, along with diverse semantic and geometric benchmarks with accurate ground-truth. The readers may refer to Fig. 2 for a detailed comparison against previous datasets.

A popular alternative is to use synthetic data to generate large scale benchmarks [4, 6, 27, 30, 32, 14, 7, 29]. Through 3D synthetic scenes and photo-realistic renderers large-scale datasets can be easily created. To date, however, these datasets have been focused on a single view of the world. This contrasts TorontoCity. Unlike other benchmarks, our input is real-world imagery, and the large-scale 3D models are a high-fidelity modeling of the real world rather than a synthetic scene.

Maps have been proven useful for many computer vision and robotics applications [35, 24, 23, 36, 22], including vehicle detection and pose estimation [24], semantic labeling and monocular depth estimation [35] as well as HD-map extraction [22]. However, there has been a lack of literature that exploit maps as ground-truth to build benchmarks. This is mainly due to both the lack of high-fidelity maps to provide pixel-level annotation and the lack of accurately geo-referenced imagery that aligned well with the maps. One exception is [36], where the streettree catalog is used to generate ground-truth for tree detection. [37] utilizes 3D building models to generate correspondences from multiple streetview images. In this paper, we use maps to create multiple benchmarks for reconstruction, recognition and reorganization from many different views of the world.

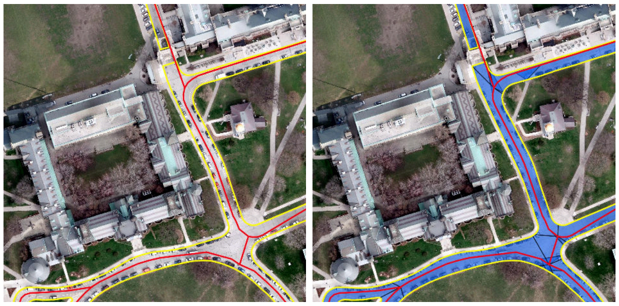


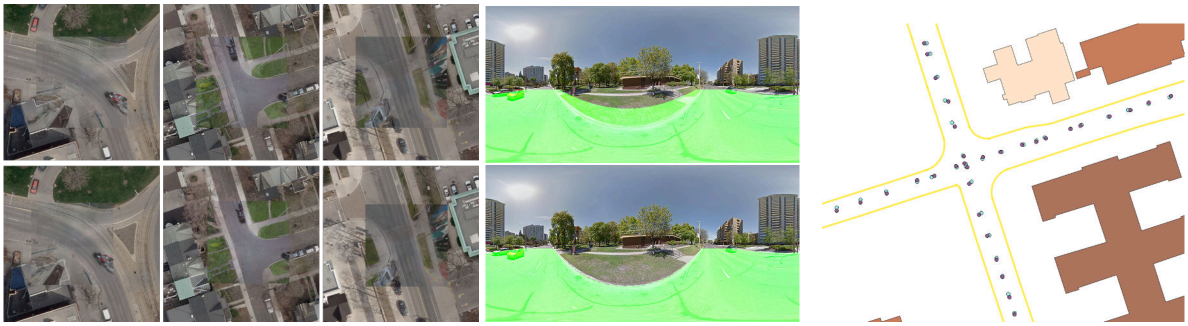
Figure 3: Road surface generation: (left) input data with curbs (yellow) and center lines (red). Extracted road surface is the union of polygons shown in blue and black. Note that a formulation ensuring connectivity is needed, otherwise the road surface would contain holes at intersections.

3. TorontoCity at a Glimpse

TorontoCity is an extremely large dataset enabling work on many exciting new tasks. We first describe the data in detail. In the next section we describe our efforts to simplify the labeling task, as otherwise it is infeasible to create such a large-scale dataset. We then show the challenges and metrics that will compose the benchmark. Finally, we perform a pilot study of how current algorithms perform on most tasks, and analyze the remaining challenges.

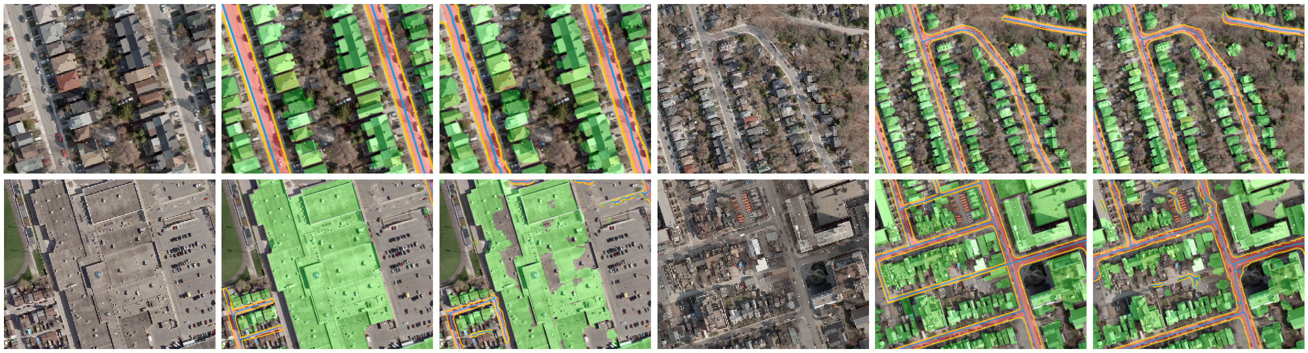
3.1. Dataset

Toronto is the largest city in Canada, and the fourth largest in North America. The TorontoCity dataset covers the greater Toronto area (GTA), which contains $712.5km^2$ of land, $8439km$ of road and around $400,000$ buildings. According to the census $6.8million$ people live in the GTA,



(a) NCC: before vs. after (b) Overlay: before vs. after (c) Location: before vs. after

Figure 4: Ground-aerial alignment



(a) Input (b) GT (c) ResNet56 (d) Input (e) GT (f) ResNet56

Figure 5: Examples of aerial semantic segmentation, road curb extraction, and road centerline estimation.

which is around 20% of the population of Canada.

We have gathered a wide range of views of the city: from the overhead perspective, we have aerial images captured during four different years (containing several seasons) as well as airborne LIDAR. From the ground, we have HD panoramas as well as imagery and LIDAR data captured from a moving vehicle driving around the city. In addition, we are augmenting the dataset with imagery captured from drones. Fig. 1 depicts some of the data sources that compose our dataset. We now describe the data in more details and refer the reader to Fig. 2 for a comparison against existing datasets.

Panoramas: We downloaded Google Streetview panoramas [2] that densely populate the GTA. On average, we crawled around 520 full 360° spherical panoramas for each km^2. In addition, we crawled the associated metadata, including the geolocation, address and the parameters of the spherical projection, including pitch, yaw and tilt angles. We resized all panoramas to 3200×1600 pixels.

Aerial Imagery: We use aerial images with full coverage of the GTA taken in 2009, 2011, 2012 and 2013. They are orthorectified to $10cm$/pixel resolution for 2009 and 2011, and 5 and $8cm$/pixel for 2012 and 2013 respectively. This contrasts satellite images, which are at best $50cm$/pixel. Our aerial images have four channels (i.e., RGB and Near infrared), and are 16 bit resolution for 2011 and 8 bit for the rest. As is common practice in remote sensing [5], we projected each image to the Universal Transverse Mercator (UTM) 17 zone in the WGS84 geodetic datum and tiled the area to $500 \times 500m^2$ images without overlap. Note that the images are not true orthophotos and thus facades are visible.

Airborne LIDAR: We also exploit airborne LIDAR data captured in 2008 with a Leica ALS sensor with a resolution of 6.8 points per m^2. The total coverage is 22 km^2. All of the points are also geo-referenced and projected to the UTM17 Zone in WGS84 geodetic datum.

Car setting: Our recording platform includes a set of cameras and a LIDAR mounted on top of a self-driving vehicle. All the sensors are calibrated and synchronized with a positioning system to record real-time geo-location and orientation information. While we have driven this platform for a relatively small area, we are actively collecting and aligning new data from ground-view vehicles.

3.2. Maps as Annotations

Manually labelling such a large scale dataset as TorontoCity is simply not possible. Instead, in this paper we exploit different sources of high-precision maps covering the whole GTA to create our ground truth.

Buildings: The TorontoCity dataset contains $400,000$ 3D buildings covering the full GTA. As shown in Fig. 2, the buildings are very diverse, with the tallest being the CN

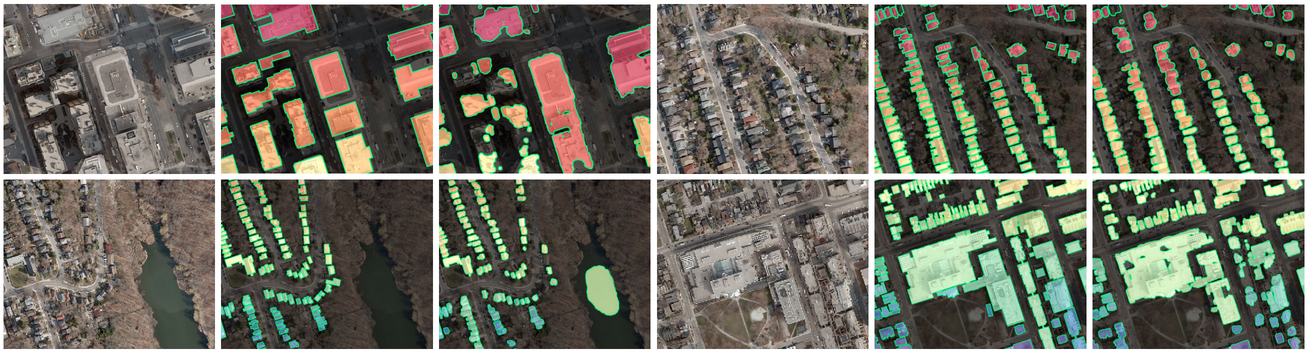


Figure 6: Examples of building instance segmentation.

Tower with 443m of elevation. Toronto contains many individual family houses, which makes tasks such as instance level segmentation particularly difficult. The mean height of each building is 4.7m, and the mean building area is $148m^2$. In contrast, the largest building has an area of $120,000m^2$. The level of detail of the 3D models varies per building (see Fig. 1). Many are centimeter accurate and contain other semantic info such as roof type, windows and balconies.

Roads: Our maps contain very accurate polylines representing streets, sidewalks, rivers and railways within the GTA. Each line segment is described with a series of attributes such as name, road category and address number range. Road intersections are explicitly encoded as intersecting points between polylines. Road curbs are also available, and describe the shape of roads (see Fig. 1).

Urban Zoning: Our maps contain government zoning information on the division of land into categories. This zoning includes categories such as residential, commercial, industrial and institutional. Multiple categories are allowed for one zone, e.g., commercial+residential. Note that understanding urban zoning is important in applications such as urban planning, real estate and law-enforcement.

Additional data: We have cartographic information with full coverage of the GTA including the location of all poles, traffic lights, street lights and trees. Additional metadata includes the height of the poles/traffic lights, model type of each street light, trunk radius and species of each tree. We plan to exploit this in the near future.

Comparison to Online Maps: Most of our maps were created by the City of Toronto. Compared to online map services such as OpenStreetMap [1] and Google Maps, our maps are much more accurate. We have centimeter accurate building contours and height estimates. Every building is geo-referenced with an address code used by the Canadian postal service, and most buildings have detailed 3D geometric shapes. Our road-network also contains centimeter accurate road curbs, which none of the online mapping services provide. Moreover, we have access to the location of primitives such as poles and trees as well as other metadata.

4. Maps for Creating Large Scale Benchmarks

In this section we describe our algorithms to automatically align maps with our sources of imagery. We then describe the alignment of the different road maps.

4.1. Aligning Maps with All Data Sources

Aerial images: Orthorectification and geo-referencing utilizes high-precision digital elevation maps captured by airborne LIDAR and an on-board high-precision navigation system. This makes the alignment between aerial images and our high-definition maps very accurate. However, small errors in elevation can still result in a small misalignment (see supplementary material). Furthermore, a small number of buildings in the aerial images have different shapes than the ones on the map. We utilize an efficient semi-automatic process to verify the building accuracy and correct misalignments. We develop a brush painting tool for in-house annotators to manually select buildings that are misaligned or have the wrong shape. This process takes 97s per $0.25km^2$. On average, 3.1% of the buildings are labelled as wrong shape and 12% are labelled as misaligned. We then design an algorithm that exploits semantic and geometric information to estimate the correct alignment. We trained a ResNet segmentation network over the training set to estimate building segmentation and we extracted boundaries using structured edges [11]. For each building instance we generate its binary mask and contour map, and apply 2D correlation filtering on the building segmentation and edge maps over a range of 50 pixels, which corresponds to shifts of up to ±5 meters. Following this, another manual pass which takes 70s per $0.25km^2$ is taken to identify rare mistakes in the automatic process. Finally wrong misalignments and shapes are identified as 'don't care regions'.

Panoramas: Panoramas are not perfectly aligned. As noted in [8], the geo-localization error can be up to $5m$ with an average of $1.5m$, while rotation is very accurate. As a

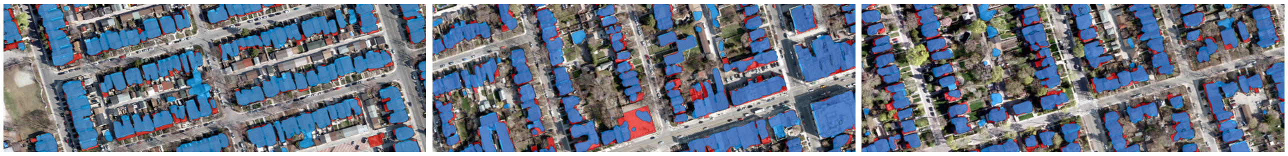


Figure 7: Qualitative results on building structured contour prediction: ResNet vs GT

Method	Road	Building	Mean
FCN [20]	74.94%	73.88%	74.41%
ResNet [15]	**82.72%**	**78.80%**	**80.76%**

Table 1: Aerial image semantic segmentation IoU.

Method	WeightedCov	AP	Re-50%	Pr-50%
FCN [20]	38.76%	15.79%	20.15%	33.67%
Resnet [15]	38.07%	22.49%	18.13%	43.42%
DWT [3]	**55.06%**	**25.62%**	**68.85%**	**65.14%**

Table 2: Building instance segmentation IoU.

consequence, projecting our maps will not generate good ground truth (see Fig. 4). To handle this issue, we design an alignment algorithm that exploits both aerial images and maps. Their information is complementary, as aerial images give us appearance, while maps give us sparse structures (e.g., road curves). For this, we first rectify the panoramas by projecting them onto the ground-plane and extract a 400 × 400 m ground plane region with 10cm/pixel resolution. We parameterize the alignment with three degrees of freedom representing the camera's offset and scale and perform a two step alignment process. We obtain a coarse alignment by maximizing a scoring function that compromises between appearance matching and a regularizer. In particular, we use normalized cross correlation (NCC) as our appearance matching and a Gaussian prior with mean $(0, 0, 2.5)m$ and diagonal covariance $(2, 2, 0.2)m$. We rescale both aerial and ground images to $[0, 1]$ before NCC. The solution space is a discrete search window in the range $[-10m, 10m] \times [-10m, 10m] \times [2.2m, 2.6m]$ with a step of $0.1m$. We use exhaustive search to perform this search, and exploit the fact that NCC can be computed efficiently using FFT and the Gaussian prior score is a fixed look-up-table. As shown in Fig. 4 this procedure produces very good coarse alignments. The alignment is coarse as we reason at the aerial images' resolution, which is relatively low. Our fine alignment then utilizes the road curves and aligns them to the boundary edges [11] in the panorama. We use a search area of $[-1m, 1m] \times [-1m, 1m]$ with a step of 5cm. This is followed by a human verification process that selects the images where this alignment succeeds. Mistakes in the alignment are due to occlusions (e.g., cars in the panoramas) as well as significantly non-flat terrain. Our success rate is 34.35%, and it takes less than 2s to verify an image. In contrast annotating the alignment takes 20s. We discard the misaligned panoramas with imperfect ground-truth.

4.2. Semantic Segmentation from Polyline Data

Our maps provide two types of road structures: curbs defining the road boundaries as well as center lines defining the connectivity (adjacency) in the street network. Unfortunately, these two sources are not aligned, and occasionally center lines are outside the road area. In this section we discuss how we exploit a Markov random field (MRF) to align road centerlines and curves. This allow us to generate the polygons describing the road surfaces. Fig. 3 shows an example of the road surface generation.

Let $y_i \in \{0, 1, \cdots, k\}$ be the assignment of the i-th curb segment to one of the k nearest centerline segments, where state 0 denotes no match. We define an MRF composed of unary and pairwise terms, which connects only adjacent curbs segments, and thus naturally form a set of chains. For the unary terms $\phi_{un}(y_i)$, we use the weighted sum of the distance of the curve to each centerline segment (condition on the state) and the angular distance between curves and centerlines. For the pairwise terms $\phi_{con}(y_i, y_{i+1})$, we employ a Potts potential that encourages smoothness along the road. This is important as otherwise there may be holes in places such as intersections, since the center of the intersection is further away from other points. Due to the chain structure of the graphical model, inference can be done exactly and efficiently in parallel for each chain using dynamic programming. Our formulation allows for multiple curbs to be matched to one road, which is needed as there are curbs on both sides of the centerline. We manually inspect the results and mark errors as "don't care" regions. We convert each continuous curb-road center line assignment to polygons which gives us the final road surface. We refer the reader to Fig. 3 for an example.

5. Benchmark Tasks and Metrics

We designed a diverse set of benchmarks to push computer vision approaches to reason about geometry, semantics and grouping. To our knowledge, no previous dataset is able to do this at this scale. In the evaluation server, participants can submit results using any subset of the imagery types provided in the benchmark (e.g., aerial images, panoramas, ground view LIDAR and camera data). In this section, we briefly describe the tasks and metrics, and refer the reader to the supplementary material for further details. Fig. 1 shows an illustration of some of our tasks.

Building Footprint and Road Segmentation: Our first task is semantic segmentation of building footprints and roads. Following common practice in semantic segmentation, we utilize mean Intersection-Over-Union (mIOU) as our metric. This is evaluated from a top-down view.

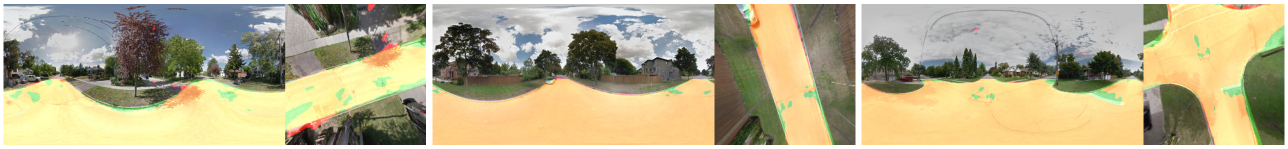


Figure 8: Examples of road segmentation. Left: panoramic view; right: top-down view. (TP: yellow, FP: red, FN: green)

	Road centerline						Road curb					
Method	F1$_{0.5}$	Pr$_{0.5}$	Re$_{0.5}$	F1$_{2}$	Pr$_{2}$	Re$_{2}$	F1$_{0.5}$	Pr$_{0.5}$	Re$_{0.5}$	F1$_{2}$	Pr$_{2}$	Re$_{2}$
FCN [20]	0.169	0.156	0.186	0.626	0.576	**0.687**	0.444	0.413	0.482	0.778	0.726	**0.837**
FCN + Close [20]	**0.173**	0.164	0.183	0.639	0.604	0.678	0.444	0.427	0.462	0.781	0.752	0.812
ResNet[15]	0.162	0.143	**0.186**	0.613	0.567	0.667	**0.575**	0.585	**0.566**	0.796	0.830	0.765
ResNet + Close [15]	0.162	**0.169**	0.155	**0.644**	**0.671**	0.619	0.568	**0.614**	0.529	**0.799**	**0.862**	0.745

Table 3: Road centerline and curb results. Metric: F1, Precision, Recall with minimal distance threshold 0.5m and 2m.

Building Footprint Instance Segmentation: Our second task is building instance segmentation. We adopt multiple metrics for this task, since there is no consensus in the community of what is the best metric. We thus evaluate weighted coverage (Cov), average precision (AP) as well as instance level precision and recall at 50%.

Building Structured Contours: Most semantic and instance segmentation algorithms produce ”blob”-like results, which do not follow the geometry of the roads and/or buildings. We thus want to push the community to produce instance segmentations that follow the structure of the primitives. Towards this goal, we define a metric that merges (in a multiplicative fashion) segmentation scoring with geometric similarity. In particular, segmentation is measured in terms of IOU, and we exploit the similarity between the turning functions of the estimated and ground truth polygons as a geometric metric. We refer the reader to the supplementary material for more details.

Road Topology: One of the remaining fundamental challenges in mapping is estimating road topology. In this task, participants are asked to extract polylines that represent road curbs and road centerlines in bird's eye perspective. We discretize both estimated and ground truth polylines in intervals of size 10cm. We define precision and recall as our metrics, where an estimated segment is correct if its distance to the closest segment on the target polyline set is smaller than a threshold (i.e., 0.5m and 2.0m).

Ground Road Segmentation: We use IOU as our metric.

Ground Urban Zoning Classification: This benchmark is motivated by the human's ability to recognize the urban function of a local region by its appearance. We use Top-1 accuracy as our metric and evaluate on the ground view.

Urban Zoning Segmentation: Our goal is to produce a segmentation in bird's eye view of the different urban zones including residential, commercial, open space, employment, *etc*. We utilize IOU as our metric.

Building Height Estimation: This tasks consists on estimating building height. Useful cues include size of the buildings, pattern of shading and shadows as well as the imperfect rectification in aerial views. We adopt root mean square error in the log domain (log-RMSE) as our metric.

Additional Tasks: We plan to add many tasks in the future. This includes detecting trees and recognizing their species. Moreover, the accurate 3D building models allow us to build a benchmark for normal estimation as well as facade parsing. We also plan to have benchmarks for detection and segmentation of traffic lights, traffic signs and poles. We also plan to release the raw map information for training and validation. We are just scratching the surface of the plethora of possibilities with this dataset.

Benchmarks: Our dataset as well as the reproducible code for all the baselines can be found at http://www.cs.toronto.edu/~torontocity/.

6. Experimental Evaluation

We perform a pilot study in a subset of TorontoCity, containing 125 km^2 region (50 km^2 for training, 50 km^2 for testing and 25km^2 for validation). The train/val/test regions do not overlap and are not adjacent. We utilize 56K streetview images around these regions (22K for training, 18K for validation and 16K for testing). Hyper-parameters are chosen based on validation performance, and all numbers reported are on the testing set. To perform the different segmentation related tasks, we train two types of convolutional networks: a variant of FCN-8 architecture [20] as well as a ResNet [15] with 56 convolutional layers. More details are in the supplementary material.

Semantic Segmentation: As shown in Tab. 1, both networks perform well. Fig. 5 illustrates qualitative results of ResNet56 output. It is worth noting that large networks such as ResNet56 can be trained from scratch given our large-scale dataset. Visually ResNet's output tends to be more sharp, while FCN's output is more smooth.

Method	AlexNet [17]	VGG-16 [33]	GoogleNet [34]	ResNet-152 [15]	AlexNet [17]	ResNet-32 [15]	GoogleNet [34]	NiN [18]
From-scratch	no	no	no	no	yes	yes	yes	yes
Top-1 accuracy	75.49%	79.12%	77.95%	**79.33%**	66.48%	75.65%	75.08%	**79.07%**

Table 4: Ground-Level Urban Zoning Classification

Method	WeightedCov	PolySim
FCN [20]	0.456	**0.323**
ResNet [15]	0.401	0.292
DWT [3]	**0.520**	0.240

Table 5: Building contour results.

Method	Residential	Open Space	Others
FCN [20]	**60.20%**	32.20%	**5.57%**
ResNet [15]	51.71%	**33.63%**	1.49%

Table 6: Qualitative results for urban zoning segmentation.

Instance Segmentation: We estimate instance segmentation by taking the output of the semantic labeling and performing connected-component labeling. Since convolutional nets tend to generate blob like structures, a single component might contain multiple instances connected with a small number of pixels. To alleviate this problem, we apply morphological opening operators over the semantic labeling masks (i.e., erosion followed by same size dilation). We also evaluated the deep watershed transform [3]. As shown in Tab. 2 and Fig. 6, this task is far from being solved. With more than $400,000$ buildings, the TorontoCity dataset provides an ideal platform for new developments.

Road Centerlines and Curbs: We compute the medial axis of the semantic segmentation to extract the skeleton of the mask as our estimate of road centerline. In order to smooth the skeletonization, we first conduct a morphological closing operator (dilation followed by erosion) over the road masks. To estimate road curbs, we simply extract the contours of the road segmentation and exploit closing operator. As shown in Table. 1, ResNet achieves the highest score in both tasks, and morphological filtering helps for both networks. Qualitative results are shown in Fig. 5. Note that there is still much room for improvement.

Building Contours: We compute building contours from our estimated building instances, and apply the Ramer-Douglas-Peucker algorithm [28] to simplify each polygon with a threshold of 0.5m. This results in polygons with 13 vertices on average. As shown in Tab. 5 and Fig. 7, this procedure offers reasonable yet not satisfactory results.

Ground Urban Zoning Classification: We train multiple state-of-the-art convolutional networks for this task including AlexNet [17], VGG-16 [33], GoogleNet [34] and ResNet-152 [15] that are fine-tuned from ImageNet [10]. We also train AlexNet [17], ResNet-32 [15], Network-In-Network [18] and ResNet-152 [15] from scratch over our ground-view panoramic image tiles. As shown in Table 1 ResNet-152 with pre-trained initialization achieves the best results. Net-in-net achieves the best performance among all models that are trained from scratch. For more details, please refer to the supplementary material.

Urban Zoning Segmentation: This is an extremely hard task from aerial views alone. To simplify it, we merged the zone-types into residential, others (including commercial, utility and employment) as well as open spaces (including natural, park, recreational *etc.*). Please refer to the supplementary material for detailed label merging. As shown in Tab. 6 more research is needed to solve this task.

Ground-view Road Segmentation: We utilize a subset of the labeled panoramas, which includes 1000 training, 200 validation and 800 testing images. The average IOU is 97.21%. The average pixel accuracy is 98.64% and average top-down IOU is 87.53%. This shows that a state-of-the-art neural network can nearly solve this task, suggesting that it is promising to automatically generate high-resolution maps by capturing geo-referenced street-view panoramas.

Building Height: No network was able to estimate building height from aerial images alone. This task is either too hard, or more sophisticated methods are needed. For example, utilizing ground imagery is a logical next step.

7. Conclusions

In this paper, we have argued that the field is in need of large scale benchmarks that allow joint reasoning about geometry, grouping and semantics. Towards this goal, we have created the TorontoCity benchmark, covering the full Greater Toronto area (GTA) with $712.5km^2$ of land, $8439km$ of road and around $400,000$ buildings. Unlike existing datasets, our benchmark provides a wide variety of views of the world captured from airplanes, drones, as well as cars driving around the city. As using human annotators is not feasible for such a large-scale dataset, we have exploited different sources of high-precision maps to create our ground truth. We have designed a wide variety of tasks and show that most of them remain challenging for convolutional networks. In the future we plan to add tasks such as building reconstruction, facade parsing as well as tree, traffic light and traffic sign detection.

8. Acknowledgements

This work was performed at the University of Toronto and funded by the support from NSERC, CFI, ORF, ERA, CRC. We thank NVIDIA for donating GPUs.

References

[1] Openstreetmap. https://www.openstreetmap.org/. 1, 5

[2] D. Anguelov, C. Dulong, D. Filip, C. Frueh, S. Lafon, R. Lyon, A. Ogale, L. Vincent, and J. Weaver. Google street view: Capturing the world at street level. *IEEE Computer*, 2010. 4

[3] M. Bai and R. Urtasun. Deep watershed transform for instance segmentation. In *CVPR*, 2017. 6, 8

[4] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. *IJCV*, 2011. 3

[5] W. Brooks. The universal transverse mercator grid. In *Proceedings of the Indiana Academy of Science*, 1973. 4

[6] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, 2012. 3

[7] C. Chen, A. Seff, A. Kornhauser, and J. Xiao. Deepdriving: Learning affordance for direct perception in autonomous driving. In *ICCV*, 2015. 3

[8] H. Chu, S. Wang, R. Urtasun, and S. Fidler. Housecraft: Building houses from rental ads and street views. In *ECCV*, 2016. 5

[9] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 1, 2

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 1, 8

[11] P. Dollár and C. L. Zitnick. Structured forests for fast edge detection. In *ICCV*, 2013. 5, 6

[12] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. 1

[13] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, 2012. 1, 2

[14] A. Handa, V. Patraucean, V. Badrinarayanan, S. Stent, and R. Cipolla. Scenenet: Understanding real world indoor scenes with synthetic data. *arXiv*, 2015. 3

[15] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *CVPR*, 2016. 6, 7, 8

[16] T. Koch, P. d'Angelo, F. Kurz, F. Fraundorfer, P. Reinartz, and M. Korner. The tum-dlr multimodal earth observation evaluation benchmark. In *CVPRW*, 2016. 2

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 1, 8

[18] M. Lin, Q. Chen, and S. Yan. Network in network. *ICLR*, 2014. 8

[19] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1

[20] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 6, 7, 8

[21] W. Maddern, G. Pascoe, C. Linegar, and P. Newman. 1 year, 1000km: The oxford robotcar dataset. *IJRR*, 2016. 2

[22] G. Máttyus, S. Wang, S. Fidler, and R. Urtasun. Enhancing road maps by parsing aerial images around the world. In *ICCV*, 2015. 2, 3

[23] G. Máttyus, S. Wang, S. Fidler, and R. Urtasun. Hd maps: Fine-grained road segmentation by parsing ground and aerial images. In *CVPR*, 2016. 3

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

[25] F. Nex, M. Gerke, F. Remondino, H. Przybilla, M. Bäumker, and A. Zurhorst. Isprs benchmark for multi-platform photogrammetry. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2015. 2

[26] J. Niemeyer, F. Rottensteiner, and U. Soergel. Contextual classification of lidar data and building object detection in urban areas. *ISPRS journal of photogrammetry and remote sensing*, 2014. 2

[27] N.Mayer, E.Ilg, P.Häusser, P.Fischer, D.Cremers, A.Dosovitskiy, and T.Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *CVPR*, 2016. 3

[28] U. Ramer. An iterative procedure for the polygonal approximation of plane curves. *Computer graphics and image processing*, 1972. 8

[29] S. R. Richter, V. Vineet, S. Roth, and V. Koltun. Playing for data: Ground truth from computer games. In *ECCV*, 2016. 3

[30] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *CVPR*, 2016. 3

[31] F. Rottensteiner, G. Sohn, M. Gerke, and J. D. Wegner. Isprs test project on urban classification and 3d building reconstruction. 2013. 2

[32] A. Shafaei, J. J. Little, and M. Schmidt. Play and learn: using video games to train computer vision models. *arXiv*, 2016. 3

[33] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv*, 2014. 8

[34] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 8

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

[36] J. D. Wegner, S. Branson, D. Hall, K. Schindler, and P. Perona. Cataloging public objects using aerial and street-level images-urban trees. In *CVPR*, 2016. 3

[37] A. R. Zamir, T. Wekel, P. Agrawal, C. Wei, J. Malik, and S. Savarese. Generic 3d representation via pose estimation and matching. In *ECCV*, 2016. 3