Find Your Own Way: Weakly-Supervised Segmentation of Path Proposals for Urban Autonomy

Dan Barnes, Will Maddern and Ingmar Posner

***Abstract*— We present a weakly-supervised approach to segmenting proposed drivable paths in images with the goal of autonomous driving in complex urban environments. Using recorded routes from a data collection vehicle, our proposed method generates vast quantities of labelled images containing proposed paths and obstacles without requiring manual annotation, which we then use to train a deep semantic segmentation network. With the trained network we can segment proposed paths and obstacles at run-time using a vehicle equipped with only a monocular camera without relying on explicit modelling of road or lane markings. We evaluate our method on the large-scale KITTI and Oxford RobotCar datasets and demonstrate reliable path proposal and obstacle segmentation in a wide variety of environments under a range of lighting, weather and traffic conditions. We illustrate how the method can generalise to multiple path proposals at intersections and outline plans to incorporate the system into a framework for autonomous urban driving.**

I. Introduction

Road scene understanding is a critical component for decision making and safe operation of autonomous vehicles in urban environments. Given the structured nature of on-road driving, all autonomous vehicles must follow the 'rules of the road'; crucially, driving within designated lanes in the correct direction and negotiating intersections.

Current commercial systems that perform driver assistance and on-road autonomy typically depend on visual recognition of lane markings and explicit definitions of lanes and traffic rules, and therefore rely on simple road layouts with clear markings (e.g. well-maintained highways) [1], [2]. To extend these systems beyond multi-lane highways to complex urban environments and rural or undeveloped locations without clear or consistent lane markings, an alternative approach is required.

In this paper we present a weakly-supervised approach to segmenting *path proposals* for a road vehicle in urban environments given a single monocular input image. Our approach is capable of segmenting the proposed path for a vehicle in a diverse range of road scenes, without relying on explicit modelling of lanes or lane markings. We define the term *path proposal* as a route a driver would be expected to take through a particular road and traffic configuration. We present a novel method of automatically generating labelled images containing path proposals. Our method leverages both the behaviour of the data collection vehicle driver and additional sensors mounted to the vehicle, illustrated in Fig. 1. Using this approach we can generate vast quantities of labelled training data without any manual annotation, spanning a wide variety of road and traffic configurations under a number of different lighting and weather conditions limited only by the time spent driving the data collection vehicle. We use this data to train an off-the-shelf deep semantic segmentation network (e.g. SegNet [3]) to produce path proposal segmentations using *only* a monocular input image[1].

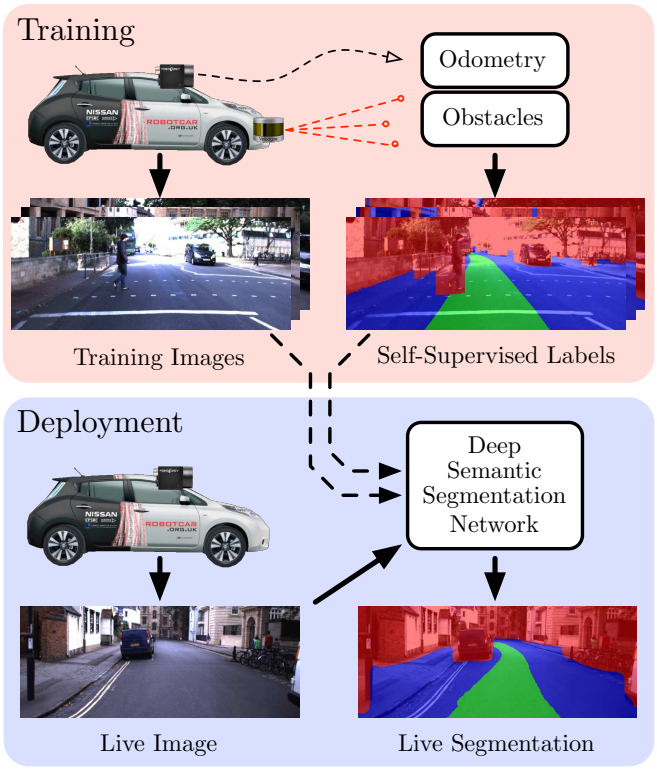

Fig. 1. Weakly-supervised path proposal segmentation using our approach. A data collection vehicle equipped with a camera as well as odometry and obstacle sensors is used to collect vast quantities of data during normal driving (top). The odometry and obstacle data is projected into the training images to generate weakly-supervised labels relevant for on-road autonomy, which are then used to train a deep semantic segmentation network. At run-time, a vehicle equipped with only a monocular camera can perform live segmentation of the drivable path and obstacles using the trained network (bottom), even in the absence of explicit lane markings.

We evaluate our approach using two large-scale autonomous driving datasets: the KITTI dataset [4], collected in Karlsruhe, Germany, and the large-scale Oxford RobotCar Dataset[2], consisting of over 1000km of recorded driving in Oxford, UK, over the period of a year. For each of

Authors are from the Oxford Robotics Institute, Dept. Engineering Science, University of Oxford, UK. {dbarnes,wm,ingmar}@robots.ox.ac.uk

[1] http://youtu.be/rbZ8ck_1nZk
[2] http://robotcar-dataset.robots.ox.ac.uk

these datasets we make use of the additional sensors on the vehicle and the trajectory taken by the driver as the weakly-supervised signal to train a pixelwise semantic classifier. We present segmentation results on the KITTI Road [5], Object and Tracking benchmarks and investigate the performance under different lighting and weather conditions using the Oxford dataset.

II. Related Work

Traditional methods of camera-based drivable path estimation for road vehicles involve preprocessing steps to remove shadow and exposure artefacts [6], [7], extraction of low-level road and lane features [8], [9], fitting road and lane models to feature detections [10], [11], and temporal fusion of road and lane hypotheses between successive frames [12], [13]. While effective in well-maintained road environments, these approaches suffer in the presence of occlusions, shadows and changing lighting conditions, unstructured roads, and areas with few or no markings [2]. Robustness can be significantly increased by combining images with radar [14] or LIDAR [15] but at an increased sensor cost.

More recently, advances in image processing using deep learning [16] have led to impressive results on the related problem of *semantic segmentation*, which aims to provide per-pixel labels of semantically meaningful objects for input images [3], [17], [18]. Deep networks make use of the full image context to perform semantic labelling of road and lane markings, and hence are significantly more robust than previous feature-based methods [3]. However, for automated driving these approaches depend on large-scale manually-annotated road scene datasets (notably CamVid [19] and Cityscapes [20], consisting of 700 and 5,000 labelled frames respectively), for which the labels are time-consuming and expensive to produce.

The challenges in building large-scale labelled datasets has led some researchers to consider virtual environments, for which ground-truth semantic labels can be rendered in parallel with synthetic camera images. Methods using customised video game engines have been used to produce hundreds of thousands of synthetic images with corresponding ground truth labels [21], [22]. While virtual environments allow large-scale generation of ground-truth semantic labels, they present two problems: firstly, rendering pipelines are typically optimised for speed and may not accurately reflect real-world images (both above approaches suggest rendered images are used only for augmenting real-world datasets and hence manual labelling is still necessary); secondly, the actions of the vehicle and all other agents in the virtual world must be pre-programmed and may not resemble real-world traffic scenarios. A recent method uses sparse 3D prior information to transfer labels to real-world 2D images [23] but requires sophisticated 3D reconstructions and manual 3D annotations.

Some approaches have proposed bypassing segmentation entirely and learning a direct mapping from input images to vehicle behaviour [24], [25]. These methods also use the driver of the data collection vehicle to generate the supervised labels for the network (e.g. steering angle) and have recently demonstrated impressive results in real-world driving tests [26], but it is not clear how this approach generalises to scenarios where there are multiple possible drivable paths to consider (e.g. intersections). Our proposed approach instead uses the data collection vehicle driver to implicitly label proposed paths in the image, but still allows a planning algorithm to choose the best path for the current route.

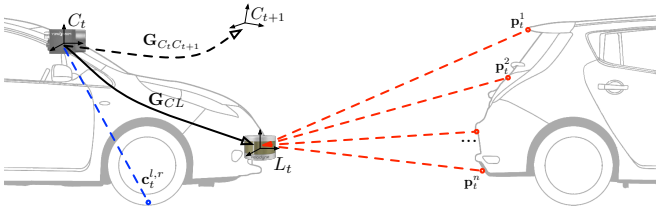

Fig. 2. Sensor extrinsics for weakly-supervised labelling. The survey vehicle (left) is equipped with a camera C and obstacle sensor L, e.g. a LIDAR scanner. The extrinsic transform $\mathbf{G}_{CL}$ between the camera and LIDAR is found using a calibration routine. The contact point $\mathbf{c}^{\{l,r\}}$ of the left and right wheels on the ground relative to the camera frame C is also measured at calibration time. At time t, the LIDAR scanner observes a number of points $\mathbf{p}_t^{1\cdots n}$ on obstacles, including other vehicles on the road (right). The relative pose $\mathbf{G}_{C_tC_{t+1}}$ of the camera between time t and $t+1$ is determined using vehicle odometry, e.g. using a stereo camera.

III. Weakly-Supervised Segmentation

In the following section we outline our approach for generating weakly-supervised training data for proposed path segmentation using video and sensor data recorded from a manually-driven vehicle.

A. Sensor Configuration

In addition to a monocular camera to collect input images, our approach depends on the following two capabilities for the data collection vehicle:

Vehicle odometry: a method of estimating the motion of the vehicle is required. For this we use stereo visual odometry [27], although other methods using inertial systems or wheel odometry would suffice.

Obstacle sensing: a method of detecting the 3D positions of impassible objects (both static and dynamic) in front of the vehicle is necessary to ensure that dynamic objects are not accidentally included in the drivable label area. For this we use a LIDAR scanner, though other methods that use dense stereo [28] or automotive radar would also be suitable.

Note that these additional sensing capabilities are only required for collecting training data; the resulting network only requires a monocular input image. Fig. 2 illustrates the sensor extrinsics for a vehicle equipped with a stereo camera and LIDAR sensor.

B. Weakly-Supervised Labelling

To generate class labels for pixels in the input image, we make use of large quantities of recorded data from the data collection vehicle driven by a human driver in a variety of traffic and weather conditions. We follow the general

approach of methods that learn to drive by demonstration [29], [30], and assume the *proposed path* corresponds to the one chosen by the driver of the data collection vehicle in each scenario. Labels are then generated by projecting the *future* path of the vehicle into each image, over which object labels as detected by the LIDAR scanner are superimposed as follows:

1) Proposed path projection: To project the future path of the vehicle into the current frame, it is necessary to know the size of the vehicle and the points of contact with the ground during the trajectory. We assume the position of the contact points $\mathbf{c}_{\{l,r\}}$ of the front left and right wheels on the ground relative to the camera C is determined as part of a calibration procedure. The position of the contact point $\mathbf{c}_{\{l,r\}}$ in the current camera frame C_t after k frames is then found as follows:

$$^{C_t}\mathbf{c}_{\{l,r\},k} = \mathbf{K}\mathbf{G}_{C_tC_{t+k}}\mathbf{c}_{\{l,r\}} \tag{1}$$

where $\mathbf{K}$ is the perspective projection matrix for the camera C and $\mathbf{G}_{C_tC_{t+k}}$ is the $\mathbb{SE}(3)$ chain of relative pose transforms formed by vehicle odometry from frame t to frame $t+k$ as follows:

$$\mathbf{G}_{C_tC_{t+k}} = \mathbf{G}_{C_tC_{t+1}} \times \mathbf{G}_{C_{t+1}C_{t+2}} \times \cdots \times \mathbf{G}_{C_{t+k-1}C_{t+k}} \tag{2}$$

Proposed path pixel labels are then formed by filling quadrilaterals in image coordinates corresponding to sequential future frames. The vertices of the quadrilateral are formed by the following points in camera frame C_t:

$$\left\{^{C_t}\mathbf{c}_{l,j}, ^{C_t}\mathbf{c}_{l,j-1}, ^{C_t}\mathbf{c}_{r,j-1}, ^{C_t}\mathbf{c}_{r,j}\right\} \tag{3}$$

where the index variable $j = \{1 \dots k\}$. An illustration of the proposed path projection and labelling process is shown in Fig. 3. The choice of frame count k depends on the look-ahead distance required for path labelling and the accuracy of the vehicle odometry system used to provide relative frame transforms. In practice we choose k such that the distance between first and last contact points $\left\|\mathbf{G}_{C_tC_{t+k}}\mathbf{c}_{\{l,r\}} - \mathbf{c}_{\{l,r\}}\right\|$ exceeds 60 metres. Different camera setups with higher viewpoints may require greater path distances, but accumulated odometry error will affect far-field projections.

2) Obstacle projection: For some applications it may be sufficient to use just the proposed path labels to train a semantic segmentation network. However, for on-road applications in the presence of other vehicles and dynamic objects, a naive projection of the path driven will intersect vehicles in the same lane and label them as drivable paths as illustrated in Fig. 4. This may lead to catastrophic results when the labelled images are used to plan paths for autonomous driving, since vehicles and traffic may be labelled as traversable by the network.

We make use of the obstacle sensor mounted on the vehicle, in our case a LIDAR scanner. Each 3D obstacle point $\mathbf{p}_t^i$ observed at time t is projected into the camera frame C_t as follows:

$$^{C_t}\mathbf{p}_t^i = \mathbf{K}\mathbf{G}_{CL}\mathbf{p}_t^i \tag{4}$$

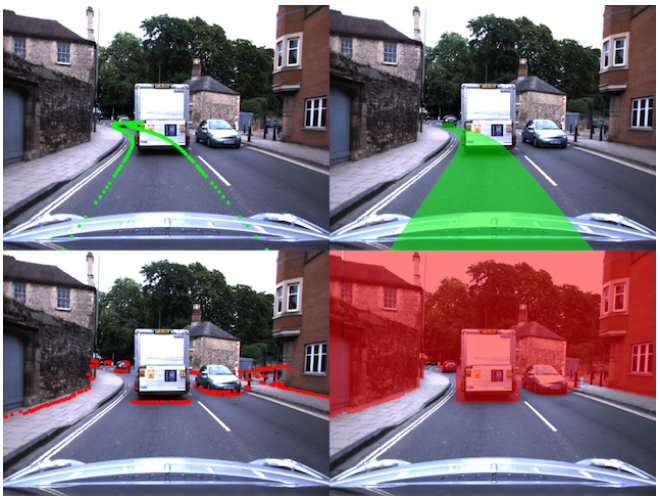

Fig. 3. Ground contact point (top) and obstacle point (bottom) projection into images. At time t, ground contact points $\mathbf{c}_{\{l,r\},j}$ (green) corresponding to the path of the vehicle up to k frames ahead are projected into the current image (top left). Pixel labels corresponding to drivable paths are filled in by drawing quadrilaterals between the left and right contact points between two successive frames (top right). At the same time, obstacle points $\mathbf{p}_t^i$ (red) from the current LIDAR scan are projected into the image (bottom left). Pixel labels corresponding to obstacles are formed by extending each of these points to the top of the image (bottom right). Note that the top and bottom sections of the image corresponding to the sky and vehicle bonnet are removed before training.

where $\mathbf{K}$ is the camera projection matrix and $\mathbf{G}_{CL}$ is the $\mathbb{SE}(3)$ extrinsic calibration between the camera and LIDAR sensor. For each camera-frame point $^{C_t}\mathbf{p}_t^i$, we take an approach inspired by "stixels" [28], [31] and label all pixels in the image on and above the point as an obstacle. This ensures all locations above and behind the detected obstacle are labelled as non-drivable, as illustrated in Fig. 3. Obstacle pixel labels take precedence over proposed path labels to ensure correct labelling of safe drivable paths as illustrated in Fig. 4.

In most images, there will be locations labelled as neither proposed path nor obstacle. These correspond to locations which the vehicle has not traversed, and no positive identification of obstacles have been made. Typically these areas correspond to the road area outside the current lane (including lanes for oncoming traffic), kerbs, empty pavements and ditches. We refer to these locations as "unknown area", as it is not clear whether the vehicle should enter these spaces during autonomous operation; this would be a decision for a higher-level planning framework as discussed in Section VII.

C. Semantic Segmentation

Once proposed path, obstacle and unknown area labels are automatically generated for a large number of recorded images, they can be used to train a semantic segmentation network to classify new images from a different vehicle equipped with only a monocular camera. We make use of SegNet [3], a deep convolutional encoder-decoder architecture for pixelwise semantic segmentation. Although higher-performing network architectures now exist (e.g. [18]), Seg-

Fig. 4. Proposed path labels for an input image (left) before (middle) and after (right) applying obstacle labels from the LIDAR scanner. Without the obstacle labels, the proposed path (middle, green) intersects vehicles in the same lane as the path driven by the data collection vehicle, which in this case will erroneously label sections of the white van as drivable route. Adding labels for obstacles (right, red) ensures that dynamic objects including the van, cyclist and pedestrian are marked as non-drivable. Note that static obstacles such as the road sign and the building are also labelled as obstacles, which correctly handles occlusions (e.g. as the path turns right after the traffic lights).

TABLE I

VEHICLE AND SETUP SUMMARY

Vehicle	**Oxford RobotCar Nissan LEAF**	**KIT AnnieWAY VW Passat**
Camera Sensor	Point Grey Bumblebee XB3	2 x Point Grey Flea2
Input Resolution	640 x 256	621 x 187
LIDAR	SICK LD-MRS 4-beam	Velodyne HDL-64E 64-beam
Vehicle Width	2.43 m	2.2 m

TABLE II

TRAINING IMAGE SUMMARY STATISTICS

Dataset	**Condition**	**Training Images**
KITTI	City	1264
	Residential	20734
	Road	2445
	Total	**24443**
Oxford	Overcast	17085
	Sun	16299
	Rain	9822
	Night	4170
	Snow	2604
	Total	**49980**

Net provides real-time evaluation on consumer GPUs, making it suitable for deployment in an autonomous vehicle.

The weakly-supervised labelling approach described in this section can generate vast quantities of training data, limited only by the length of time spent driving the data collection vehicle. However, the types of routes driven will also bias the input data, as most on-road driving is performed in a straight line; a random subsample of the training data will consist mostly of straight-line driving. In practice we subsample the data to 4Hz, before further subsampling based on turning angle. For each frame we compute the average yaw rate $\bar{\Delta\psi}$ per frame for the corresponding proposed path as follows:

$$\bar{\Delta\psi} = \frac{1}{k}\sum_{i}^{k} \psi\left(\mathbf{G}_{C_{t+i-1}C_{t+i}}\right) \tag{5}$$

where $\psi(\mathbf{G})$ is a function that extracts the Euler yaw angle ψ from the $\mathbb{SE}(3)$ transform matrix $\mathbf{G}$. We then build a histogram of average yaw rates and randomly sample from the histogram bins to ensure an unbiased selection of different turning angles.

IV. EXPERIMENTAL SETUP

We build two different models for evaluation: one using the KITTI Raw dataset [4] and one using the Oxford RobotCar dataset. These datasets were collected using different vehicles with different sensor setups, summarised in Table I.

A. Platform Specifications

Both vehicles are equipped with stereo camera systems, and we use the stereo visual odometry approach in [27] to compute the relative motion estimates required in Eq. 2. The images from the cameras are cropped and downscaled to the resolutions listed in Table I before training. The Oxford RobotCar is equipped with a SICK LD-MRS LIDAR scanner, which performs obstacle merging and tracking across 4 scanning planes in hardware. We use points identified as "object contours" to remove erroneous obstacles due to noise and ground-strike. The Velodyne HDL-64E mounted on the AnnieWAY vehicle does not perform any object filtering, and hence we use the following approach to detect obstacles: we fit a ground plane to the 3D LIDAR scan using MLESAC [32], and treat all points more than 0.25m above this plane as obstacles, as illustrated in Fig. 5. This approach effectively identifies obstacles the vehicle may collide with even in the presence of pitching and rolling motions. The camera-LIDAR calibration $\mathbf{G}_{CL}$ for the RobotCar vehicle was determined using the method in [33]; for the AnnieWAY vehicle the calibration provided with the KITTI Raw dataset was used.

B. Network Training

For the KITTI model, we made use of the available City, Residential and Road data from the KITTI Raw dataset. For the Oxford model, we selected a diverse range of weather conditions for each traversal of the route, including 9 overcast, 8 with direct sun, 4 with rain, 2 at night and 1 with snow; each traversal consisted of approximately 10km of driving. The number of labelled images used to train each model is shown in Table II and some examples are shown in Fig. 6. In total we used 24,443 images to train the KITTI model, and 49,980 images for the Oxford model.

For both datasets we built semantic classifier models using the standard SegNet convolutional encoder-decoder architecture. The same SegNet parameters were used for both datasets, with modifications only to account for the differences in input image resolution. We randomly split the input data into 75% training and 25% validation sets, performed training for 100 epochs then selected the best-performing model according to the results on the validation set. The training time totalled 10 days for KITTI using a GTX Titan GPU and 20 days for Oxford using a GTX Titan

Fig. 5. Obstacle labelling using Velodyne data for the KITTI dataset. Raw Velodyne scans (left) contain returns from the road surface as well as nearby obstacles. We fit a ground plane using MLESAC and retain only points 0.25m above the plane (middle). We then label pixels using the approach in Section III-B.2 (right) to ensure accurate labels on obstacles while retaining drivable surfaces on the ground.

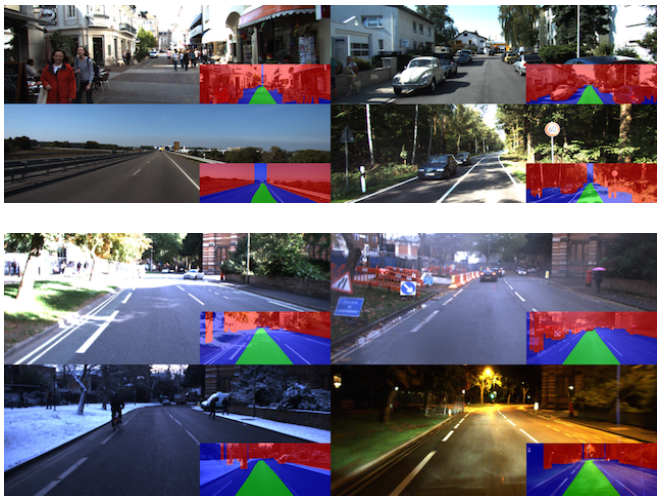

Fig. 6. Example training images with weakly-supervised labels from the KITTI (top) and Oxford (bottom) datasets. The weakly-supervised approach generates proposed path and obstacle labels for a diverse set of locations in the KITTI dataset, and a diverse set of conditions for the same location in the Oxford dataset. No manual annotation is required to generate the labels.

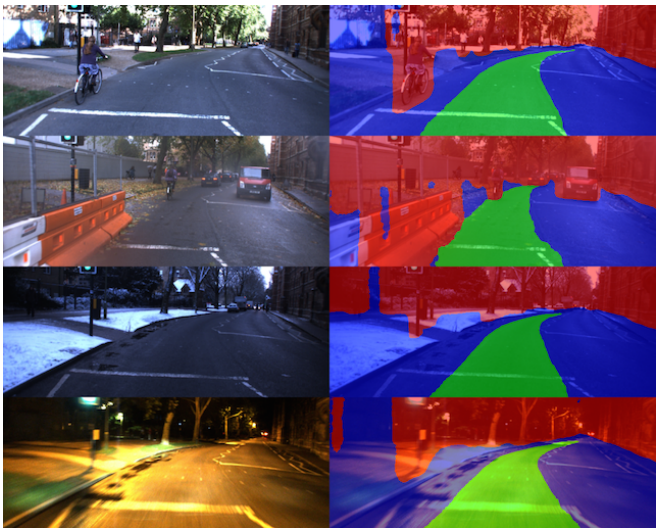

Fig. 7. Semantic segmentation on frames captured at the same location under different conditions. Despite significant changes in appearance between sunny, rainy, snowy and night-time conditions, the network correctly segments the proposed drivable path and labels obstacles including cyclists, other vehicles and road barriers.

X GPU; future training times can be reduced using a different architecture or making use of a GPU cluster.

For the comparison using the KITTI Road benchmark presented in Section V-B.1, we trained an additional SegNet model on only the training images provided for the Ego-Lane Estimation Evaluation. Note that these ground truth images were not provided to the model trained using the weakly-supervised approach described above. For the object detection evaluation using the KITTI Object and Tracking datasets, we have ensured that there is no overlap between images selected to train the weakly-supervised labels and the images with ground truth labels used in the evaluation.

V. Results

For reliable on-road driving, the semantic segmentation must function in multiple environments under the range of lighting, weather and traffic conditions encountered during normal operation. In this section we evaluate the performance of both the KITTI model and Oxford model under a range of different test conditions.

A. Oxford Dataset

We evaluate the Oxford model by generating ground truth labels for a further four datasets not used for training, consisting of 2,718 images in sunny conditions, 2,481 images in cloudy conditions, 2,340 images collected at night and 1,821 images collected in the rain, for a total of 9,360 test images. Table III presents the segmentation results for the three classes in each of the four different conditions in the test datasets listed above, where the "All" column shows the mean for each metric across all classes. The model provides good performance across the different conditions with mean intersection-over-union (IoU) scores exceeding 80% in all cases, with the highest performance in cloudy weather and lowest at night, due to the reduced image quality in low-light conditions. Fig. 7 illustrates the output of the network for four images of the same location under different conditions. Despite significant changes in lighting and weather, the network correctly determines the proposed path through the crossing and identifies obstacles (e.g. construction barriers). This result demonstrates that the weakly-supervised approach can be used to train a single network that segments proposed paths and obstacles across a wide range of conditions without explicitly modelling environmental changes due to lighting, weather and traffic. Fig. 8 presents a number of locations where the network proposed a valid path in the absence of explicit road or lane markings, instead using the context of the road scene to infer the correct route.

B. KITTI Benchmarks

To demonstrate how our weakly-supervised labelling approach can lead to useful performance for autonomous driving tasks, we evaluate it on two different benchmarks

TABLE III

SEGMENTATION RESULTS FOR OXFORD TEST DATA ACROSS VARYING CONDITIONS

Condition	Proposed Path		Obstacle		Unknown Area		All	
Night	PRE	86.50%	PRE	93.60%	PRE	88.88%	PRE	89.66%
	REC	87.75%	REC	93.71%	REC	88.31%	REC	89.92%
	IoU	77.18%	IoU	88.06%	IoU	79.53%	IoU	81.59%
Rain	PRE	89.55%	PRE	94.04%	PRE	91.41%	PRE	91.66%
	REC	86.97%	REC	96.88%	REC	88.73%	REC	90.86%
	IoU	78.95%	IoU	91.27%	IoU	81.90%	IoU	84.04%
Overcast	**PRE**	**91.13%**	PRE	94.76%	**PRE**	**93.41%**	**PRE**	**93.10%**
	REC	**92.63%**	REC	96.68%	**REC**	**90.53%**	**REC**	**93.28%**
	IoU	**84.97%**	IoU	91.77%	**IoU**	**85.09%**	**IoU**	**87.27%**
Sun	PRE	89.50%	**PRE**	**94.85%**	PRE	92.56%	PRE	92.30%
	REC	89.53%	**REC**	**97.01%**	REC	90.05%	REC	92.20%
	IoU	81.02%	**IoU**	**92.16%**	IoU	83.97%	IoU	85.72%

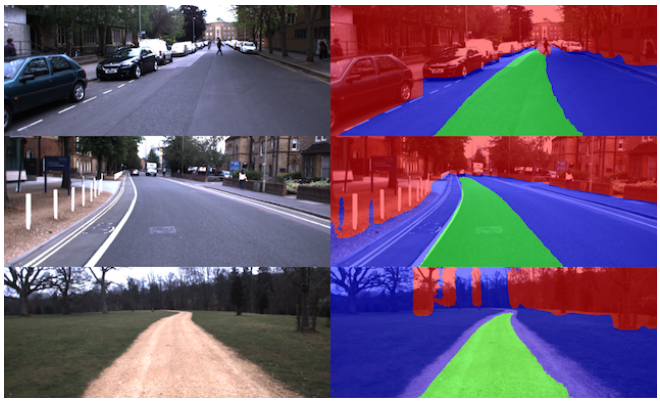

Fig. 8. Path proposals in locations without explicit lane dividers or road markings. Using the context of the road scene the network infers the correct proposed path (top, middle), even for gravel roads never seen in the training data (bottom).

from the KITTI Vision Benchmark Suite[3]: ego-lane segmentation and object detection. However, neither of these benchmarks are an exact match for the segmentation results provided by the network, as they were designed for different purposes. Accordingly, we present alternative metrics based on the provided ground truth to quantitatively evaluate our system. Note that the following two sections present different evaluation metrics for the same model trained on the same input data and should be interpreted in concert; the network produces both path and obstacle labels for each test image even when only one class is under evaluation.

1) Ego-lane Segmentation: The closest analogue to a *proposed path* in the KITTI benchmark suite is the ego-lane, consisting of the entire drivable surface within the lane the vehicle currently occupies [5]. The ego-lane dataset consists of 95 training and 96 test images, each with manually annotated ground truth labels. We trained an additional SegNet model on the provided ground truth training images to compare to our model trained on weakly-supervised labelled images, as detailed in Section IV-B. The results of both models on the KITTI website benchmark is shown in Table IV. Fig. 9 illustrates a sample network output for both models. The weakly-supervised model outperforms the model trained on the provided ground-truth images, with a 20% increase in max F-score and 15% increase in precision exceeding 90% in total, despite never making use of manually annotated ground truth images or explicit encoding of lane markings. Although the overall performance is not competitive with those generated by more sophisticated network architectures on the KITTI leaderboard (due to the different definition of ego-lane and proposed path), this result strongly indicates that the weakly-supervised approach generates segmentations useful for real-world path planning. The differences in the number of training images used for each model is illustrative of the fact that manually-annotated datasets will always be more be more time-consuming and expensive to produce than our weakly-supervised approach; even if manually annotated data is also available, for many tasks our approach could be used as pre-training to further improve results.

2) Object Detection: While the KITTI benchmark suite does not contain a semantic segmentation benchmark, it does contain object instance bounding boxes in both the Object and Tracking datasets. The definition of an *object* in the KITTI benchmark (an individual instance of a vehicle or person within a bounding box) differs significantly from our definition of an *obstacle* as part of the weakly-supervised approach (any part of the scene the vehicle might collide with). However, we can evaluate object detection performance by ensuring that every *object* instance provided by the KITTI Object and Tracking benchmarks was also classified as an *obstacle* by our segmentation approach; hence we aim for the highest *pixel-wise recall* score. For each object instance we evaluate the number of pixels within the bounding box classified as an obstacle using our weakly-supervised approach, as illustrated in Fig. 10. We present three different recall metrics: pixel recall, which includes all pixels under all bounding boxes for each object class, and two variants of instance recall, which requires a certain fraction of obstacle-labelled pixels within each bounding box instance before the object is considered as "detected" (thresholds of 50% and 75% are presented). We present recall results on the data provided as part of the Object and Tracking datasets (consisting of 15,047 images with 87,343 total object instances) in Table V, and an example detection is shown in Fig. 10. We have combined the object classes as follows: car, van, truck and tram labels are grouped as Vehicle; pedestrian, person sitting and cyclist labels are

[3] http://www.cvlibs.net/datasets/kitti/

TABLE IV

EGO-LANE SEGMENTATION RESULTS ON THE KITTI ROAD BENCHMARK

Training	Benchmark	MaxF	AP	PRE	REC	FPR	FNR
Provided	UM_LANE	52.42%	37.85%	77.88%	39.50%	1.98%	60.50%
Weakly-Supervised	UM_LANE	**72.88%**	**64.49%**	**92.78%**	**60.01%**	**0.82%**	**39.99%**

Fig. 9. Example ego-lane segmentation results using the KITTI Road dataset. For the given input image (left), a SegNet model trained on the small number of manually-annotated ground truth images (middle) performs poorly in comparison with the model trained on the much larger weakly-supervised dataset (right) generated without manual annotation.

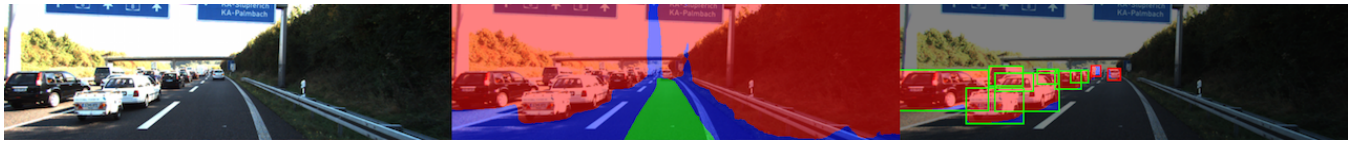

Fig. 10. Example object detection results using obstacle segmentation. For a given input image (left), the network labels areas corresponding to proposed path, obstacle and unknown area (middle). For each ground-truth bounding box provided in the KITTI Object and Tracking datasets, we compute the ratio of pixels labelled as obstacle by our method (right). For each object instance, we consider it detected (green outline) if more than 75% of the pixels within the bounding box are labelled as obstacles. Note that even for failed detections (red outline), a number of the pixels were still labelled as obstacle, and due to the tight obstacle outlines provided by our method we may miss portions of the bounding box (e.g. undercarriage of vehicles at bottom left).

TABLE V

OBSTACLE SEGMENTATION RESULTS ON THE KITTI OBJECT AND TRACKING DATASETS

Metric	Vehicle	Person	Misc	All
Pixel Recall	93.73%	92.47%	94.11%	93.53%
Instance Recall (>50%)	99.52%	99.65%	99.29%	99.55%
Instance Recall (>75%)	98.15%	97.38%	96.73%	97.93%

grouped as Person, and all others are grouped as Misc. The results show that the weakly-supervised segmentation approach is reliably labelling *objects* as *obstacles* regardless of object class (and performs especially well for an instance recall threshold of 50%); this is critical to avoid planning trajectories that intersect other vehicles or road users.

C. Limitations

Under some conditions the network fails to produce useful proposed path segmentations, as illustrated in Fig. 11. These failure cases are mostly due to limitations of the sensor suite (poor exposure or low field of view), and could be addressed using a larger number of higher-quality cameras.

D. Route Generalisation

As the weakly-supervised labels are generated from the recording of a data collection trajectory, it can only provide one proposed path per image at training time. However, at intersections and other locations with multiple possible routes, at test time the resulting network frequently labels multiple possible proposed paths in the image as shown in Fig. 12; this is an important step towards decision-making for topological navigation within a road network. Currently we have no ground truth to evaluate route generalisation; we present qualitative results here for illustration only and plan to further characterise this effect in a future publication.

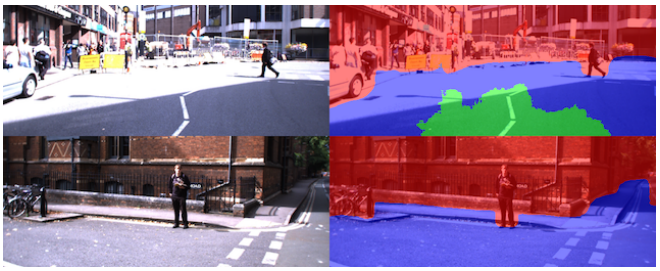

Fig. 11. Proposed path segmentation failures. (Top) Overexposed or underexposed images will lead to incorrect path segmentation; this could be addressed by using a high-dynamic-range camera. (Bottom) At some intersections during tight turns, there is no clear path to segment as it falls outside the field of view of the camera; using a wider field of view lens or multiple cameras in a surround configuration would address this limitation.

VI. CONCLUSIONS

In this paper we have outlined our approach for weakly-supervised labelling of images for proposed path segmentation during on-road driving using only a monocular camera. We have demonstrated that by leveraging multiple sensors and the behaviour of the data collection vehicle driver, we are able to generate vast quantities of semantically-labelled training data relevant for autonomous driving applications; crucially, we do not require any manual labelling of images in order to train our segmentation network. Our approach does not depend on specific road markings or explicit modelling of lanes to propose drivable paths. We evaluated the approach in the context of ego-lane segmentation and obstacle detection using the KITTI dataset, outperforming networks trained on manually-annotated training data and providing reliable

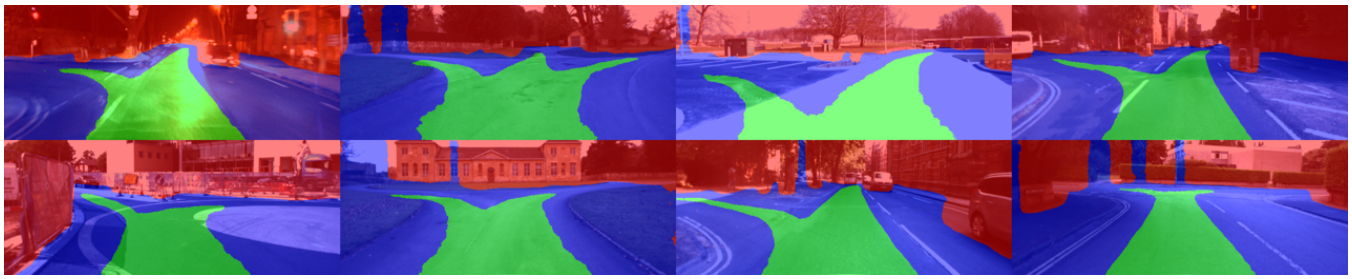

Fig. 12. Proposed path generalisation to multiple routes. At intersections and roundabouts the network will often label different possible paths, which can then be leveraged by a planning framework for decision making during autonomous navigation.

obstacle detections. We also demonstrated the robustness of the trained network to changes in lighting, weather and traffic conditions using the large-scale Oxford RobotCar dataset, with successful proposed path segmentation in sunny, cloudy, rainy, snowy and night-time conditions. We plan to integrate the network with a planning framework that includes our previous work on topometric localisation across experiences [34] as well as our semantic map-guided approach for traffic light detection [35] to enable fully autonomous driving in complex urban environments.

VII. Acknowledgements

The authors would like to gratefully acknowledge support of this work by the UK Engineering and Physical Sciences Research Council (EPSRC) Doctoral Training Partnership (DTP) and Programme Grant EP/M019918/1.

References

[1] S. Yenikaya, G. Yenikaya, and E. Düven, "Keeping the vehicle on the road: A survey on on-road lane detection systems," *ACM Computing Surveys (CSUR)*, vol. 46, no. 1, p. 2, 2013.

[2] A. B. Hillel, R. Lerner, D. Levi, and G. Raz, "Recent progress in road and lane detection: a survey," *Machine vision and applications*, vol. 25, no. 3, pp. 727–745, 2014.

[3] V. Badrinarayanan, A. Handa, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling," *arXiv preprint arXiv:1505.07293*, 2015.

[4] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *The International Journal of Robotics Research*, p. 0278364913491297, 2013.

[5] J. Fritsch, T. Kuehnl, and A. Geiger, "A new performance measure and evaluation benchmark for road detection algorithms," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, 2013, pp. 1693–1700.

[6] J. M. Álvarez, A. M. López, and R. Baldrich, "Shadow resistant road segmentation from a mobile monocular system," in *Iberian Conference on Pattern Recognition and Image Analysis*. Springer, 2007, pp. 9–16.

[7] I. Katramados, S. Crumpler, and T. P. Breckon, "Real-time traversable surface detection by colour space fusion and temporal analysis," in *International Conference on Computer Vision Systems*. Springer, 2009, pp. 265–274.

[8] J. C. McCall and M. M. Trivedi, "Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation," *IEEE transactions on intelligent transportation systems*, vol. 7, no. 1, pp. 20–37, 2006.

[9] K. Yamaguchi, A. Watanabe, T. Naito, and Y. Ninomiya, "Road region estimation using a sequence of monocular images," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*. IEEE, 2008, pp. 1–4.

[10] R. Labayrade, J. Douret, J. Laneurit, and R. Chapuis, "A reliable and robust lane detection system based on the parallel use of three algorithms for driving safety assistance," *IEICE transactions on information and systems*, vol. 89, no. 7, pp. 2092–2100, 2006.

[11] A. S. Huang and S. Teller, "Probabilistic lane estimation for autonomous driving using basis curves," *Autonomous Robots*, vol. 31, no. 2-3, pp. 269–283, 2011.

[12] R. Jiang, R. Klette, T. Vaudrey, and S. Wang, "New lane model and distance transform for lane detection and tracking," in *International Conference on Computer Analysis of Images and Patterns*. Springer, 2009, pp. 1044–1052.

[13] H. Sawano and M. Okada, "A road extraction method by an active contour model with inertia and differential features," *IEICE transactions on information and systems*, vol. 89, no. 7, pp. 2257–2267, 2006.

[14] B. Ma, S. Lakshmanan, and A. O. Hero, "Simultaneous detection of lane and pavement boundaries using model-based multisensor fusion," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, no. 3, pp. 135–147, 2000.

[15] A. S. Huang, D. Moore, M. Antone, E. Olson, and S. Teller, "Finding multiple lanes in urban road networks with vision and LIDAR," *Autonomous Robots*, vol. 26, no. 2-3, pp. 103–122, 2009.

[16] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[17] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[18] G. Papandreou, L.-C. Chen, K. Murphy, and A. L. Yuille, "Weakly-and semi-supervised learning of a DCNN for semantic image segmentation," *arXiv preprint arXiv:1502.02734*, 2015.

[19] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88–97, 2009.

[20] 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," *arXiv preprint arXiv:1604.01685*, 2016.

[21] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3234–3243.

[22] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," *arXiv preprint arXiv:1608.02192*, 2016.

[23] J. Xie, M. Kiefel, M.-T. Sun, and A. Geiger, "Semantic instance annotation of street scenes by 3D to 2D label transfer," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[24] D. A. Pomerleau, "ALVINN: An autonomous land vehicle in a neural network," DTIC Document, Tech. Rep., 1989.

[25] U. Muller, J. Ben, E. Cosatto, B. Flepp, and Y. L. Cun, "Off-road obstacle avoidance through end-to-end learning," in *Advances in neural information processing systems*, 2005, pp. 739–746.

[26] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, *et al.*, "End to end learning for self-driving cars," *arXiv preprint arXiv:1604.07316*, 2016.

[27] W. Churchill, "Experience based navigation: Theory, practice and implementation," Ph.D. dissertation, University of Oxford, Oxford, United Kingdom, 2012.

[28] D. Pfeiffer and U. Franke, "Efficient representation of traffic scenes by means of dynamic stixels," in *Intelligent Vehicles Symposium (IV), 2010 IEEE*. IEEE, 2010, pp. 217–224.

[29] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey

of robot learning from demonstration," *Robotics and autonomous systems*, vol. 57, no. 5, pp. 469–483, 2009.

[30] D. Silver, J. A. Bagnell, and A. Stentz, "Learning autonomous driving styles and maneuvers from expert demonstration," in *Experimental Robotics*. Springer, 2013, pp. 371–386.

[31] T. Scharwächter, M. Enzweiler, U. Franke, and S. Roth, "Stixmantics: A medium-level model for real-time semantic scene understanding," in *European Conference on Computer Vision*. Springer, 2014, pp. 533–548.

[32] P. H. Torr and A. Zisserman, "Mlesac: A new robust estimator with application to estimating image geometry," *Computer Vision and Image Understanding*, vol. 78, no. 1, pp. 138–156, 2000.

[33] G. Pascoe, W. Maddern, and P. Newman, "Direct visual localisation and calibration for road vehicles in changing city environments," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 9–16.

[34] C. Linegar, W. Churchill, and P. Newman, "Made to measure: Bespoke landmarks for 24-hour, all-weather localisation with a camera," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 787–794.

[35] D. Barnes, W. Maddern, and I. Posner, "Exploiting 3D semantic scene priors for online traffic light interpretation," in *2015 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2015, pp. 573–578.