Article

The International Journal of Robotics Research
1–15

DOI: 10.1177/0278364917730603
journals.sagepub.com/home/ijr

Learn from experience: probabilistic prediction of perception performance to avoid failure

Corina Gurău[1], Dushyant Rao[1], Chi Hay Tong[2], and Ingmar Posner[1]

Abstract

Despite significant advances in machine learning and perception over the past few decades, perception algorithms can still be unreliable when deployed in challenging time-varying environments. When these systems are used for autonomous decision-making, such as in self-driving vehicles, the impact of their mistakes can be catastrophic. As such, it is important to characterize the performance of the system and predict when and where it may fail in order to take appropriate action. While similar in spirit to the idea of introspection*, this work introduces a new paradigm for predicting the likely performance of a robot's perception system based on past experience in the same workspace. In particular, we propose two models that probabilistically predict perception performance from observations gathered over time. While both approaches are place-specific, the second approach additionally considers appearance similarity when incorporating past observations. We evaluate our method in a classical decision-making scenario in which the robot must choose* when *and* where *to drive autonomously in 60 km of driving data from an urban environment. Results demonstrate that both approaches lead to fewer false decisions (in terms of incorrectly offering or denying autonomy) for two different detector models, and show that leveraging visual appearance within a state-of-the-art navigation framework increases the accuracy of our performance predictions.*

Keywords

Robotics, object detection, introspection, performance estimation, autonomous driving

1. Introduction

As a result of recent advances in computer vision and machine learning, autonomous systems are now being deployed in complex operational scenarios in the real world. As many of these systems rely on learning from data, it is impossible to formally verify before deployment that they will behave as expected at test time. Furthermore, lower-level mistakes of a perception module can propagate to the higher-level decision-making procedures of an autonomous system. Without checks on the reliability of the information propagated, the safety of the robot and its surroundings is compromised. Our goal in this work is to equip an autonomous system with the introspective capability of predicting when it is about to make a mistake. Just as we have the ability to identify ambiguous or difficult situations, such as an approaching busy intersection or a narrow and crowded street, an autonomous system should be able to foresee its perceptual shortcomings and communicate them to a human operator. While significant effort is being devoted to building high-performance perception systems (Badrinarayanan et al., 2017; Cai et al., 2016; He et al., 2016; Ren et al., 2015), the problem of predicting their failure in action has not yet received the attention it deserves. As robots will share complex, continuously-evolving, dynamic workspaces with human beings, it is critical to analyze and predict how robustly their perception systems function at any given moment in time. One of the main reasons why perception systems do not behave as expected is the change in the conditions in which they operate—numerous external factors can lead to the appearance of the world during testing varying to extents that are unobserved during training. These factors can be weather conditions, time of day, illumination, structural changes, or anything that alters the visual appearance of a place. Our work is mainly motivated by our previous observations that perception performance for mobile robots is *environment-dependent*. Performance is excellent in some places of operation, while in others, failure occurs more often (Hawke et al., 2015). The *shift* in the data distribution that has

[1]Oxford Robotics Institute, Oxford University, UK
[2]Oxbotica, UK

Corresponding author:
Corina Gurău, Oxford Robotics Institute, Oxford University, 23 Banbury Road, Oxford OX2 6NN, UK.
Email: corina@robots.ox.ac.uk

not been accounted for at training time causes the learned model to generalize poorly to the data that it is about to encounter. Flagging up such situations is highly useful—it provides an automatic indicator of when the system should not be trusted.

This work treats a generic perception system as a black box and investigates the effect of changing operational conditions on its performance during repeated test runs of the same route. In particular, we are interested in discovering how many perception mistakes can be avoided if, occasionally, control is handed over from the autonomous system to the human operator in a principled fashion. Requiring a human driver to intervene in an autonomous operation falls under the *shared-autonomy* paradigm, in which the robot is able to seamlessly take control of the vehicle and give operation back to the driver as a function of the environment. To achieve this, the robot must also fulfill the *autonomy-on-offer* paradigm, in which the robot characterizes its own performance and determines whether or not it is confident in its abilities and hence capable of autonomous operation. Our proposed framework is kept separate from the perception system itself and is able to predict performance before the actual perception task has been executed. It can be seen as a favorable alternative to employing the uncertainty measure typically associated with a learning algorithm in order to assess how much trust can be placed on the detector's predictions. Intrinsic uncertainty measures are often erroneous in practice, particularly when the algorithm is presented without sample data or when the algorithm itself is not inherently probabilistic (Grimmett et al., 2016). We propose to show the usefulness of our reliability measure in the context of autonomy-on-offer, in which the robot has to decide when to ask for help. Some example decisions can be seen in Figure 1. While in some cases the reasons for failing at a certain visual task are apparent—such as overexposure or underexposure of a significant part of the input image—in other cases, more analysis is needed to understand the underlying causes of failure. Our work is unique in that we explicitly consider the location of the system when predicting perception performance. This means that we can predict likely failures in advance, even prior to commencing a route, and thereby highlight difficult sections of the proposed route by leveraging place-dependent cues. This is crucial, as research shows that it can take up to 15 s for an operator to resume control, and up to 40 s to stabilize control of the vehicle (Merat et al., 2014). As a result, we envision such introspection techniques to become an integral part of fleet scheduling systems for autonomous vehicles in the future. Our second approach also considers similarity of appearance between current and previous traversals (at the expense of this ability to predict this far in advance), which ensures that prior observations that differ greatly from the current frame, owing to, for example, vastly different lighting or weather conditions, are removed from consideration in the prediction model.

This paper augments and extends work by Gurau et al. (2016), with a more comprehensive exposition, additional experimental analysis with different detection models, and extended discussion. The key contributions of this work are:

- The concept of *performance records*: a framework which incorporates place-specific performance estimates gathered over time, in order to allow the robot at test time to estimate the likelihood of making a mistake;
- Two approaches to building performance records, one of which makes use of the visual appearance of a place; and
- A view of autonomy-on-offer as a decision-making problem that provides motivation and a use-case for performance records, and allows the robot to optimally choose to offer or deny autonomy on over 60 km of driving data in an urban environment.

The paper is structured as follows. We describe the related work in Section 2. In Section 3, we present our proposed approach for building performance records by incorporating observations of performance over time and describe a potential use-case for our estimates—offering or denying autonomy. Section 4 details the experimental validation, including details of the detector models and data used, as well as performance estimates across large areas of operation. Finally, we conclude in Section 6 with a short summary of our discoveries and a discussion on future work.

2. Related work

Perception systems are often deployed in challenging real-world environments, using noisy sensors in operating conditions that change with time. For autonomous vehicles, this shift is typically caused by changing weather, location, camera viewpoint, and other such factors. As a result, the performance of perception algorithms employed on these vehicles can be unreliable.

This lack of reliability has been observed by Peynot et al. (2009, 2010), who attribute it to sensor data integrity and analyze the effects of challenging operational conditions on the perceptual integrity of the robot. Hoiem et al. (2012) aim to understand and distinguish between the different failure modes of an object detector. They too acknowledge that the quality of the input plays a significant part in the outcome and show how different object characteristics correlate with the performance score. Similar problems are also reported for localization performance. Churchill et al. (2015) and Dequaire et al. (2016) propose to embed spatial models of expected localizer performance in localization maps in order to aid trajectory planners. Several works in the literature address the fluctuating performance levels of a machine learning system induced by a visual shift in the data. This is often due to training set bias, which can be dangerous, as the resulting models generalize poorly

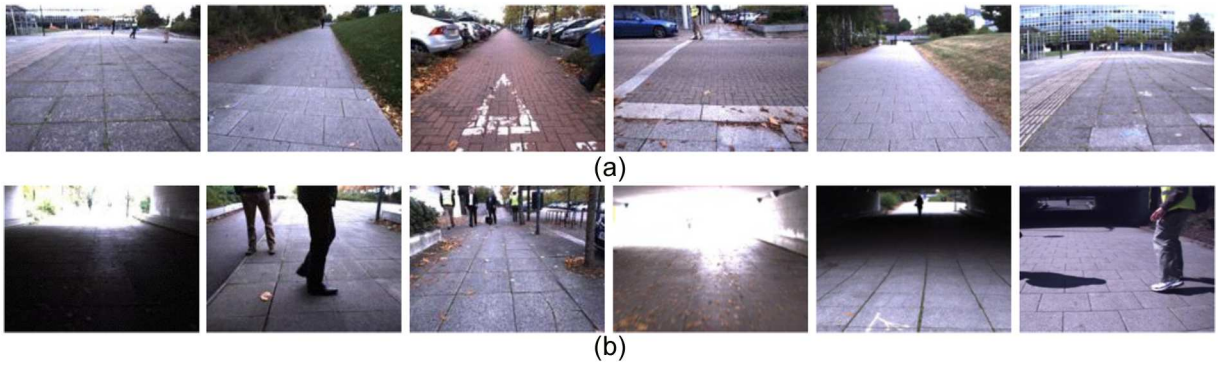

Fig. 1. Example data encountered by a robot as it traverses an urban environment in the proximity of pedestrians, cyclists, and other road users. On some sections of the road on which it believes its perception system is underperforming, the robot can ask to switch control back to a human operator (b). Alternatively it continues to operate autonomously (a).

across place and time, meaning that they might not respond well to new testing conditions. The works of Khosla et al. (2012) and Torralba and Efros (2011) identified early on that the mismatch between the training and test data distributions is a major cause of poor model performance, and, as a result, they advocate the use of cross-dataset evaluations. Gurau et al. (2014) describe the sensitivity of object detectors to such factors as weather and location, and train local experts by incorporating place-specific hard negative examples in the training procedure. When generic negative training data are replaced with the detector's mistakes, they are able to significantly improve detection results by specifically targeting areas of improvement for each location. This idea of adapting a model to the data it is most likely to observe during operation has been particularly successful in the context of object detection through the process of hard negative mining (Felzenszwalb et al., 2010). Similarly, in the case of training neural networks for object detection, the process of fine-tuning adjusts the model weights to be more representative of a specific set of data (Yosinski et al., 2014).

The problem of dataset bias has also been addressed by the computer vision community through *domain adaptation*; learning a mapping between two different domains, such that a model trained on one domain performs just as well on the other when used in combination with the learned mapping (Kulis et al., 2011). With the advent of deep learning techniques and, subsequently, deep domain adaptation, effort has been devoted to learning features that are invariant to this change of domain (Ganin and Lempitsky, 2015). In this scenario, the adaptation happens directly in the training process, where the model is penalized if it learns untransferable features that are biased toward one domain. This can be implemented under the recently proposed framework of adversarial training, whereby an additional adversarial network is trained to discriminate between the two domains, given the internal representations of the first network. The original network is then also trained to fool the adversary (i.e. by adding the negative of the discriminator loss), thereby encouraging it to learn representations that are domain-invariant (Tzeng et al., 2016).

While these methods seek to adapt a model to new environments, this work instead focuses on the critical ability to characterize and predict the performance of a perception system in new conditions or environments. This higher-level characterization of when and where an algorithm fails is similar in spirit to the concept of *introspection* introduced by Grimmett et al. (2016). In that work, the authors looked at the introspective capacity of different classification frameworks, which refers to a classifier's ability to assign an appropriate measure of confidence to any test data. Mistakes are not considered catastrophic when they are made with high uncertainty, as this gives the system the ability to ask for help and correct itself. McAllister et al. (2017) also looked at model uncertainty and proposed the use of Bayesian deep learning to increase vehicle safety autonomously. This assessment of trust or confidence has been widely employed by the active learning community in order to guide the data selection process (Holub et al., 2008; Kapoor et al., 2010). It has also been examined in the face verification domain through standardized image quality metrics and performance assessment systems. For instance, Dutta et al. (2015) present a generative model that maps image quality (including facial pose and illumination) to verification performance, without analyzing the distribution of the classifier's scores or its reported uncertainty. Gurari et al. (2016) propose similar ideas for the task of predicting the accuracy of a segmentation algorithm: they use a linear regressor to predict the Jaccard index (the fraction of pixels that are common to the segment and ground truth) based on simple features characterizing the shape and geometry of the segmented region.

A key benefit of our framework is that it is independent of the classification algorithm and its associated uncertainty measure. It bears some similarity with the work of Zhang et al. (2014), who introduces ALERT, a system used to predict the accuracy of a computer vision system for various tasks. The authors propose a regression model to predict

task performance directly from the input data and show that incorporating these predictions can help improve the performance of a downstream application. Our work is also similar to that of Daftry et al. (2016), in which the authors propose a convolutional neural network (CNN) architecture that uses the current image feed (and the resulting optical flow images) to predict the performance of a trajectory planner on a micro-aerial vehicle. They too motivate their work in the context of *introspection*, and also argue that performance prediction should be modeled directly as a function of the input data, so that it is independent of the system it is characterizing.

We share with these methods an aspiration to reliably flag a warning when we believe that the input data (in our case, camera images) do not have the characteristics required for high performance and the robot is more likely to fail at the vision task it has to perform. However, our work stands apart from that of Daftry et al. (2016) and Zhang et al. (2014), as our approach is tailored specifically to robot perception and explicitly exploits location and past experiences of the robot in that place of operation. Given that autonomous vehicles are likely to traverse the same routes repeatedly, these experiences provide useful contextual information, which could guide the robot's future decision-making process. To the best of our knowledge, this is the first approach to predict the performance of a perception system as a function of appearance, space, and time.

3. Approach

Given that robots often perform repeat traversals of the same workspaces (Furgale and Barfoot, 2010; Linegar et al., 2015; McManus et al., 2013; Paton et al., 2017), we propose to make use of the robot's past experience to estimate its capabilities. In the context of autonomous vehicles, this is not an unusual problem setting, as they rarely operate in completely unknown environments. If a robot has traversed a route in the past, then we would like to leverage its past experience to predict performance in subsequent visits of the same place. Our work is based on the assumption that the same physical location, under similar driving conditions, leads to a similar perception outcome. Figure 2 illustrates this setup: we repeatedly traverse the same route and gather performance estimates along it in order to create performance records. At test time, we utilize the records to predict how the system will perform on the current traversal. To evaluate our system, we further employ the predictions made with the purpose of deciding on whether the robot should offer or deny autonomy to the driver. The performance that we want to estimate is that of an image-based pedestrian detector (more details in Section 4) but the idea can be extended to any learning algorithm whose performance varies with operational conditions. We describe our approach for estimating detection performance at a particular location in Section 3.1 and formulate the offering or denial of autonomy as a decision-making problem in Section 3.2.

3.1. Estimating perception performance

For a traversal $\mathcal{T} = \{\ell_1, \ell_2, \ldots, \ell_N\}$ of a route, we denote as ℓ_i the ith location along it. We keep track of performance at every location ℓ_i by modeling detections at that location as events. While true positive detections indicate the success of the detection system, false positives and false negatives indicate failure, so we record observations $x_i^j \in \{0, 1\}$ such that

$$x_i^j = \begin{cases} 1 & \text{if the } j\text{th observation at } \ell_i \text{ is a true positive} \\ 0 & \text{if the } j\text{th observation at } \ell_i \text{ is a false positive or a false negative} \end{cases} \tag{1}$$

We let the observations x be modeled by a Bernoulli random variable: $x \sim Ber(\theta)$ with probability mass function

$$f(x;\theta) = \theta^x (1-\theta)^{1-x}, \qquad x \in \{0, 1\} \tag{2}$$

This θ can be thought of as the probability of success of the detection system. We make the assumption that the set of observations $\mathcal{X}_i = \{x_i^1, x_i^2, \ldots, x_i^{n_i}\}$ is conditionally independent given θ_i, and we explicitly condition each observation x_i on θ_i to express the likelihood of successful performance for a particular location ℓ_i as

$$p(\mathcal{X}_i|\theta_i) = \prod_{j=1}^{n_i} p(x_i^j|\theta_i) = \theta_i^{k_i}(1-\theta_i)^{n_i-k_i} \tag{3}$$

where k_i represents the number of observations indicating good performance ($x_i = 1$) out of a total of n_i observations at location ℓ_i along the route. Using Bayes' theorem, we calculate the probability of the detector being successful at location ℓ_i as

$$p(\theta_i|\mathcal{X}_i) = \frac{p(\mathcal{X}_i|\theta_i)\, p(\theta_i)}{\int_{\theta_i} p(\mathcal{X}_i|\theta_i)\, p(\theta_i)} \tag{4}$$

We represent the prior on the probability of success of the detector $p(\theta_i)$ as a beta density of the form

$$p(\theta_i;\alpha,\beta) = \frac{1}{B(\alpha,\beta)}\theta_i^{\alpha-1}(1-\theta_i)^{\beta-1}, \qquad 0 \le \theta_i \le 1 \tag{5}$$

where $\alpha > 0$, $\beta > 0$, and $B(\alpha, \beta)$ is the beta function. Our canonical prior at a new location that we see for the first time. where we have no knowledge of the success of the detector. is given by $\alpha = 1, \beta = 1$.

Since the beta distribution is a conjugate prior to the Bernoulli distribution, the posterior $p(\theta_i|\mathcal{X}_i)$ is also a beta distribution. The hyperparameters of the posterior are updated as

$$\widehat{\alpha}_i = \alpha + k_i, \qquad \widehat{\beta}_i = \beta + n_i - k_i \tag{6}$$

This gives us a simple procedure for incorporating observations over time. We refer to all $p(\theta_i; \widehat{\alpha}, \widehat{\beta})$ at locations ℓ_i as the performance record of the detection system on a chosen route after traversal $\mathcal{T}$ and use it to estimate the likely performance of the robot at test time.

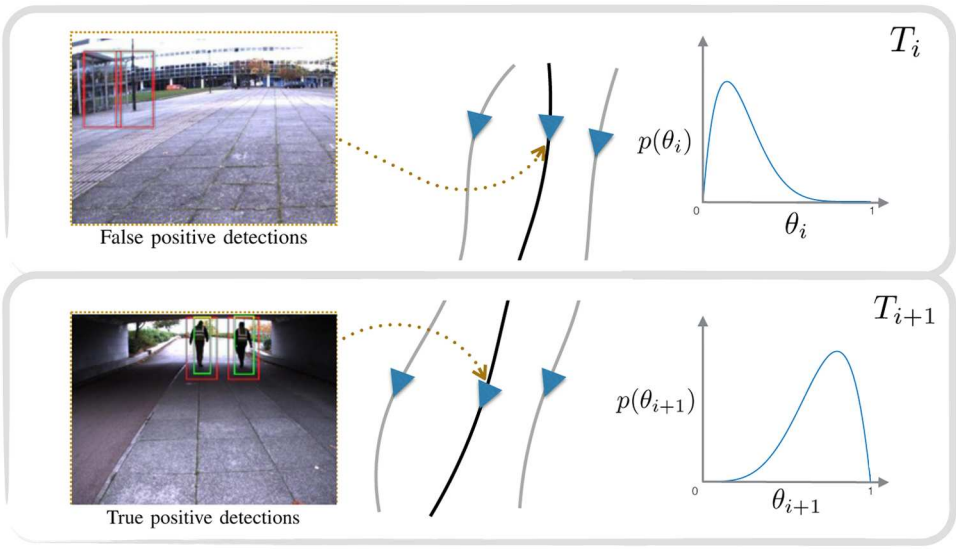

Fig. 2. Overview of proposed method. A new traversal (black line) of a route that has been traveled previously (gray lines) can make use of past estimates of detection performance. For instance, at Location A where we have repeatedly observed false positive detections, the performance record yields a low probability of success for the detector, while at Location B, where the detector has only produced true positive detections, the probability of success is very high.

3.2. Decision-making using a performance record

We are considering the case in which the robot has to decide between operating autonomously and asking a human operator to take over control and perform the task reliably on its behalf. In this simplified scenario, the robot has only two actions available: a^0, *denying autonomy*, and a^1, *offering autonomy*, at every location along a driving route. The robot should choose action a^0 when it believes that its perception system is failing and a human operator should take over control and it should choose action a^1 when it believes that its perception system is functioning well and it can reliably operate autonomously.

We assume that there are only two states that the perception system can be in: failing (and producing false detections), or performing well (and the robot presents no risk when operating autonomously). Following standard one-shot decision theory, we assign a loss to state-action pairs, which reflects how serious it is to take action a^i when the actual state is s^j, for $i,j \in \{0,1\}$

$$L(a,s) = \begin{pmatrix} 0 & L_{\text{offer}} \\ L_{\text{deny}} & 0 \end{pmatrix}$$

The optimal action is the one that minimizes the expected loss. This is computed as

$$\overline{L}_\tau(a) = \sum_i p(s^i)\, L(a,s^i) \tag{7}$$

To compute the probability of the perception system being in state $p(s^i)$, we introduce hyperparameter τ and denote by s^0 the event that the perception system is failing at location L_i. We compute its probability as

$$p(s^0|\theta,\tau) = p(\theta \leq \tau) = \int_0^\tau p(\theta;\widehat{\alpha},\widehat{\beta})\,\mathrm{d}\theta \tag{8}$$

where $p(\theta;\widehat{\alpha},\widehat{\beta})$ has been estimated using the performance records proposed. We denote by s^1 the event that the perception system is performing well and compute the probability of it happening as $p(s^1|\tau) = 1 - p(s^0|\tau)$. Thus, τ can be thought of as a *decision* threshold (not to be confused with the threshold of the detection system itself), which modulates the level of confidence that is required before autonomy is offered by the robot.

The ratio of the losses associated with each action usually depends on the application domain and it ultimately comes down to a choice that the system designer has to make. In this case, it is a trade-off between false positive and false negative interventions, so one might prefer to set the cost of inconveniencing a human operator as less than the cost of failing to detect an object. Figure 3 shows the effect of adjusting the losses associated with each type of error on the actions selected. Type I, or false positive errors, correspond to situations in which the robot denies autonomy (a^0) but its perception system is in reality performing well (s^1) and incurs a loss of L_{deny}. Type II, or false negative errors, occur when the robot fails to recognize that it is underperforming (s^0) and continues to operate autonomously (a^1). Figure 3 shows that by making type I errors more expensive (increasing L_{offer}), the robot is more conservative, and employs the safer action of denying autonomy more often. Ultimately, both types of error decrease the user's trust in the system. Section 4 presents results for equal costs of the actions as well as for an increased cost of offering autonomy.

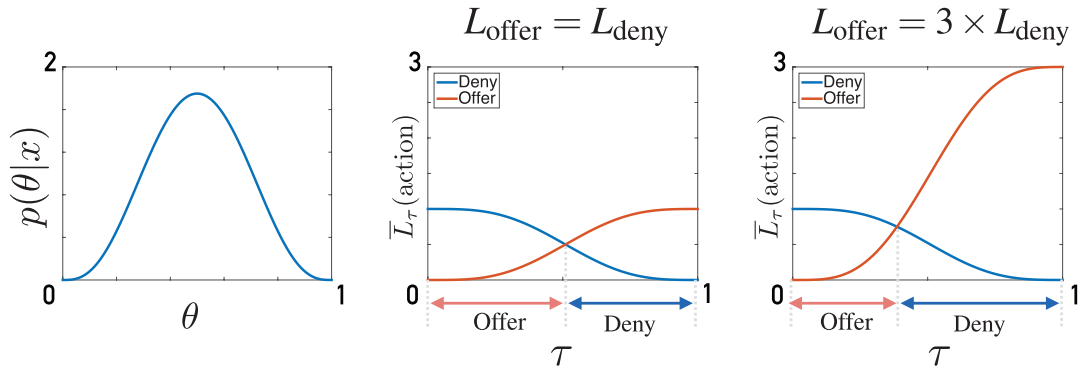

Fig. 3. Expected loss of choosing an action for an example posterior distribution $p(\theta|x)$ (left figure). Two different loss matrices are used, one in which the costs of the actions are equal (middle figure) and one in which L_{offer} is three times larger than L_{deny} (right figure). When $L_{\text{offer}} = L_{\text{deny}}$, for $\tau = 0.6$ (gray line), the action chosen by the robot is to offer autonomy because it has a lower expected loss $\bar{L}_{\tau=0.6}$. However, by setting $L_{\text{offer}} = 3 \times L_{\text{deny}}$, the optimal action becomes to deny autonomy. Increasing the cost of operating autonomously, L_{offer}, results in a more cautious system that generally prefers the safe action of denying autonomy.

3.3. Performance records and the experience paradigm

In the initial phase of performance map building, to assign observations of performance to locations in the world, we make use of the robot's navigation system, as well as a relational database framework (Nelson et al., 2016), to store and retrieve relevant information at runtime. The data and meta-data in which we are interested, denoted $\mathcal{D}$, consists of three-dimensional tuples at n discretized locations along a trajectory

$$\mathcal{D} = \left\{(\phi_i, X_i, L_i)\right\}_{i=1}^{n} \tag{9}$$

where ϕ_i is an image, X_i is a set of observations of performance, as described in Section 3.1, and L_i is a location along the trajectory that the robot has traversed.

At runtime, we make use of $\mathcal{D}$ to retrieve past performance at each test location L_{test} and estimate performance levels online. We propose two different methods of retrieving observations associated with a location. First, we look at geographical proximity given by GPS measurements and consider the observations in a local neighborhood $N(L_{\text{test}})$, where

$$N(L_{\text{test}}) = \left\{i : \|L_i - L_{\text{test}}\| \leq \epsilon\right\} \tag{10}$$

and ϵ is a prespecified radius. While this distance metric is useful for gathering all the observations close to a desired location, it does not take into account which of them are most relevant in terms of the appearance of the live frame. Imagine the following test case: while driving at night, past observations gathered at night-time should be more relevant than observations gathered in daytime. Similarly, detection in bright sunny conditions might have a different outcome than detection during rain. In these situations, having a distance metric that also incorporates visual similarity is crucial.

The second method we propose employs experience-based navigation (Churchill and Newman, 2013; Linegar et al., 2015) to find the *closest location*, in terms of both appearance and location to a live frame. Experience-based navigation is a vast-scale, camera-based localization system, which gathers many representations of the same place under various conditions (*experiences*) and stores them in a graph structure. In this graph, the nodes represent landmarks and the edges between them contain six degrees of freedom transformations between the nodes. Loop closures are detected using FAB-MAP (Cummins and Newman, 2011). Most importantly for our application, experience-based navigation searches the experience graph, in a local neighborhood of the robot's position estimate, to find the images that best match the live frame. As in Figure 4, on a traversal of the trajectory, each frame may be localized against one or more frames from past traversals. Our intuition is that if two images have enough corresponding features to localize the robot, then they would produce a similar detection outcome. We denote the method of estimating performance using all past observations, regardless of the visual appearance of the environment by LOC, since it only incorporates observations that are close in location. We denote the second method, which leverages experience-based navigation to explicitly distinguish between different appearances of the world and select observations from locations that are close both in physical distance and visual appearance by APP. This method first involves localizing a

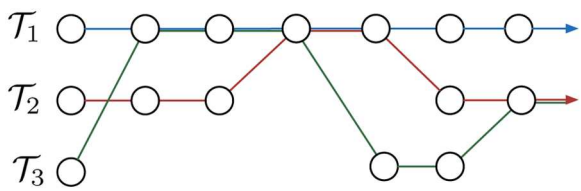

Fig. 4. Experience-based navigation framework. Each frame along a new traversal of a route can be matched to previous frames as a function of appearance. For example, individual frames in traversal 3 may be matched to frames in traversals 1 or 2.

live frame against an existing experience-based navigation experience graph. Examples of past observations selected by LOC and APP are shown in Figure 5. Whereas LOC selects images from past traversals that are closest according to GPS, APP collects images that are most visually similar, in some cases compromising proximity for appearance. We expect APP to give more accurate estimates of performance, as it has access to the outcome (the performance record) of images that are very similar to the live frame in terms of viewpoint, lighting, and even weather conditions. We further refer the reader to Linegar et al. (2015) for a comprehensive description of the experience-based navigation framework employed.

4. Experimental results

We evaluate the two methods proposed for estimating performance, LOC and APP, on 60 km of driving data gathered in an urban environment in Milton Keynes, England, over the course of 6 months. The same route has been traversed eight times under different environmental conditions using the data collection platform shown in Figure 6, yielding a total of 70k image frames. Some examples can be seen in Figures 1 and 5.

4.1. Pedestrian detector

We evaluate our two approaches using two distinct image-based pedestrian detectors. The first is a support vector machine on aggregate channel features (ACF) (Dollár et al., 2014) trained on the INRIA person dataset (Dalal and Triggs, 2005) with five rounds of hard negative mining on data collected in central Oxford, which we refer to as SVM+ACF. The second detector used is MS-CNN (Cai et al., 2016), which achieves state-of-the-art performance on the KITTI (Geiger et al., 2013) and Caltech (Dollár et al., 2009) datasets. We operate MS-CNN at a detection threshold of 0.8, as this presents the best combination of precision and recall. This block can easily be substituted for any other pedestrian detector or, more generally, performance records can be built for any image-based vision system for which some kind of supervisory signal exists.

4.2. Surrogate ground truth

To assess the performance of our object detection system, as well as the quality of our predictions, we require ground truth annotations. Naturally, it is impractical to annotate such large datasets manually. Further, because our approach explicitly exploits repeat traversals of a given region or trajectory, widely used benchmarking datasets, such as KITTI, are unsuitable for the task. To be able to achieve our goal, we make use of a surrogate metric of performance, which compares the image detections against laser detections (projected into the image) to obtain performance scores for the 2D pedestrian detector.

The laser detector used to provide the surrogate ground truth metric was trained on KITTI Velodyne data (Geiger et al., 2013) and achieves strong detection performance, as described in Wang and Posner (2015). To further improve the fidelity of this surrogate ground truth, the 3D laser detections are passed into a Kalman filter, ensuring that objects are consistently detected and tracked in consecutive frames. Note that although we require the laser sensor to build the performance record at training time, we do not require the sensor at test time. Further, by using a separate modality, such as Lidar, to generate the surrogate ground truth, the labels are insensitive to the types of appearance-based variation that plague the visual perception system and lead to degraded performance. We estimate performance and take optimal actions either using only the performance record and the location of the robot (required by LOC), or using the performance record and the incoming image feed (required by APP).

We have conducted an additional experiment to verify that treating laser detections as ground truth leads to a meaningful evaluation score. For this experiment, we use the F_1 metric to provide a single measurement of performance per image. The F_1 score is computed as

$$\mathrm{F}_1 = \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FP} + \mathrm{FN}} \tag{11}$$

where TP, FP, and FN are true positives, false positives, and false negatives, respectively.

By applying the SVM+ACF detector to one dataset with (manually annotated) ground truth labels, consisting of 3627 pedestrians, we compute two sets of F_1 scores. The first set is obtained by comparing the SVM+ACF detections with ground truth annotations, and the second set is obtained by comparing them with the surrogate ground truth: the detections provided by the 3D laser. In all experiments presented in this paper, a detection is deemed to be a true positive if it overlaps a ground truth box with intersection-over-union greater or equal to 0.5. Figure 7 shows a strong correlation between the two sets of scores; this leads us to believe that our surrogate metric has empirically the same effect as using ground truth annotations. The heatmap, computed using kernel density estimation, has a peak at 1, meaning that the detector has a perfect F_1 score against both ground truth and surrogate ground truth, to which most of our detections contribute. However, the laser detector is not perfect—there exists a set of detections that score highly against ground truth but not against the laser detector. Therefore, using a proper ground truth metric would further benefit the results. Nonetheless, by employing our surrogate metric of performance, the evaluation becomes entirely self-supervised. All performance scores can be retrieved effortlessly as soon as the robot has finished operating.

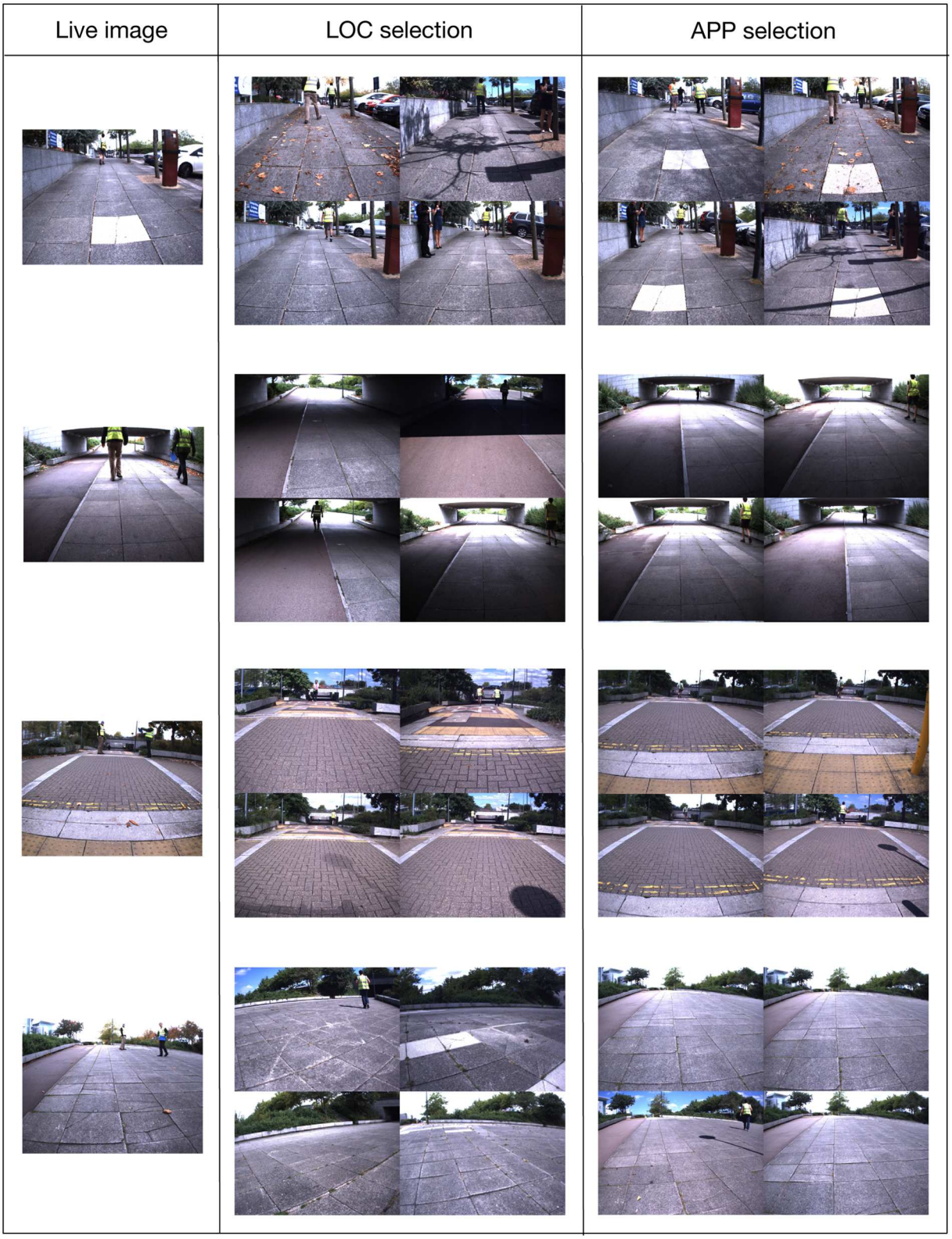

Fig. 5. Example images from a test route in Milton Keynes. The figure shows the live frame from a test traversal (left), a set of images that LOC selects for building the location-specific performance record (center) and a set of images that APP selects for the same purpose through experience-based navigation, the image-based localization system (right). Overall, the latter method results in a more accurate performance record.

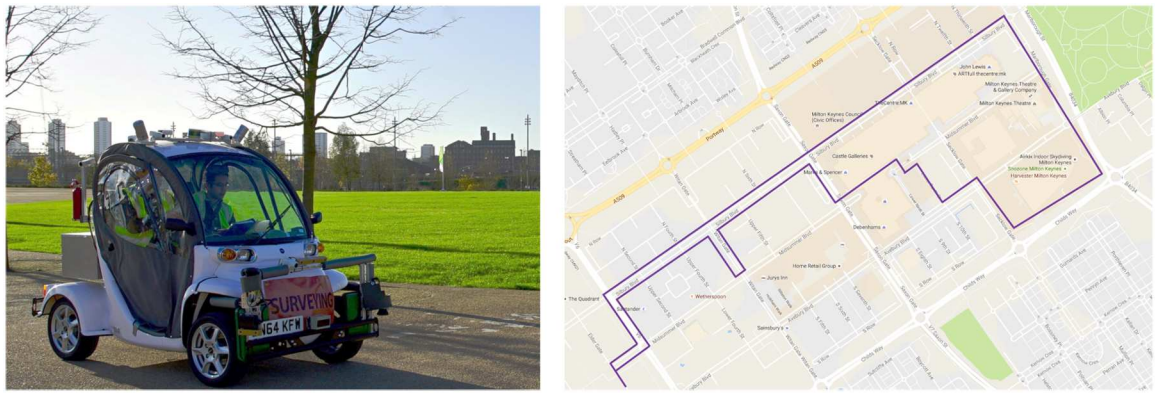

Fig. 6. Platform and route chosen for experiments. The vehicle is equipped with a Bumblebee3 stereo camera, Velodyne Lidar HDL32E, and an INS system for data collection. We produce both 2D and 3D pedestrian detections in image and laser data along the route in Milton Keynes, England, shown on the right.

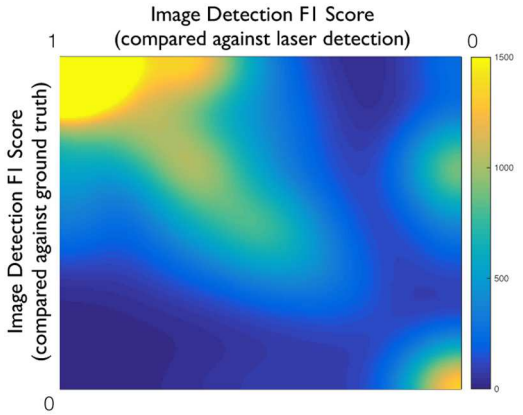

Fig. 7. Heatmap showing a strong correlation between performance scores for the SVM+ACF detector computed in two different ways. First, the 2D detection F_1 score is computed using ground truth annotations (y-axis). Second, the same score is computed using laser detections instead of ground truth (x-axis). The diagonal elements, as well as the peaks at 1 and 0, allow us to believe that very little noise will be introduced by using a surrogate metric of performance.

4.3. Performance estimation and decision-making

To evaluate the accuracy of the performance prediction methods, we propose a metric that shows the usefulness of LOC and APP in a practical scenario. We compare the total number of mistakes the robot makes while employing the two methods, as described in Section 3. We also show two baseline cases: 'always-yes', corresponding to always operating autonomously, and 'always-no', corresponding to always asking for help. What we refer to as *mistakes* are the outcomes of the following two cases:

- Choosing to deny autonomy when there are no false positive and no false negative detections in an image (i.e. perfect detector performance). These errors are of type I.
- Choosing to offer autonomy when there is at least one false detection in an image (i.e. imperfect detector performance). These errors are of type II, or critical errors as, in our situation, they can have severe consequences.

All decisions are taken on a per-frame basis before detection actually occurs. Additionally, for LOC, the estimates of performance do not rely on the live image, so they can be estimated even before the robot traverses the test route. We found that mistakes at a particular location in the world are indeed correlated (up to a difference in the appearance of the environment), and are a good indication of future performance. All figures show the results obtained in an evaluation of all traversals in a leave-one-out fashion and an equal cost ($L_{\text{offer}} = L_{\text{deny}}$) for each type of mistake.

Figure 8(a) shows the total percentage of mistakes made (in terms of incorrectly offering or denying autonomy) as a function of the sliding parameter τ, the *decision* threshold, which characterizes how good the past performance needs to be in order to decide to offer autonomy according to the LOC and APP methods. A value of $\tau = 0$ corresponds to the scenario of always offering autonomy (indicated as the constant *always-yes* line) and a value of $\tau = 1$ corresponds to always denying autonomy (*always-no*). The percentage of mistakes made for the *always-yes* case indicates the fraction of frames in which a false positive or negative detection was made (hence, offering autonomy was incorrect), and the *always-no* case indicates the complement: the fraction of frames that exhibited perfect detection (such that denying autonomy was incorrect). Interestingly, the percentage of mistakes for *always-yes* is higher for MS-CNN than for SVM+ACF, which indicates that the MS-CNN produces false detections on more *frames*. Based on visualization of the detections, we suspect that this may be due to MS-CNN finding correct detections where our surrogate ground truth has failed to do so. While further analysis on this would be

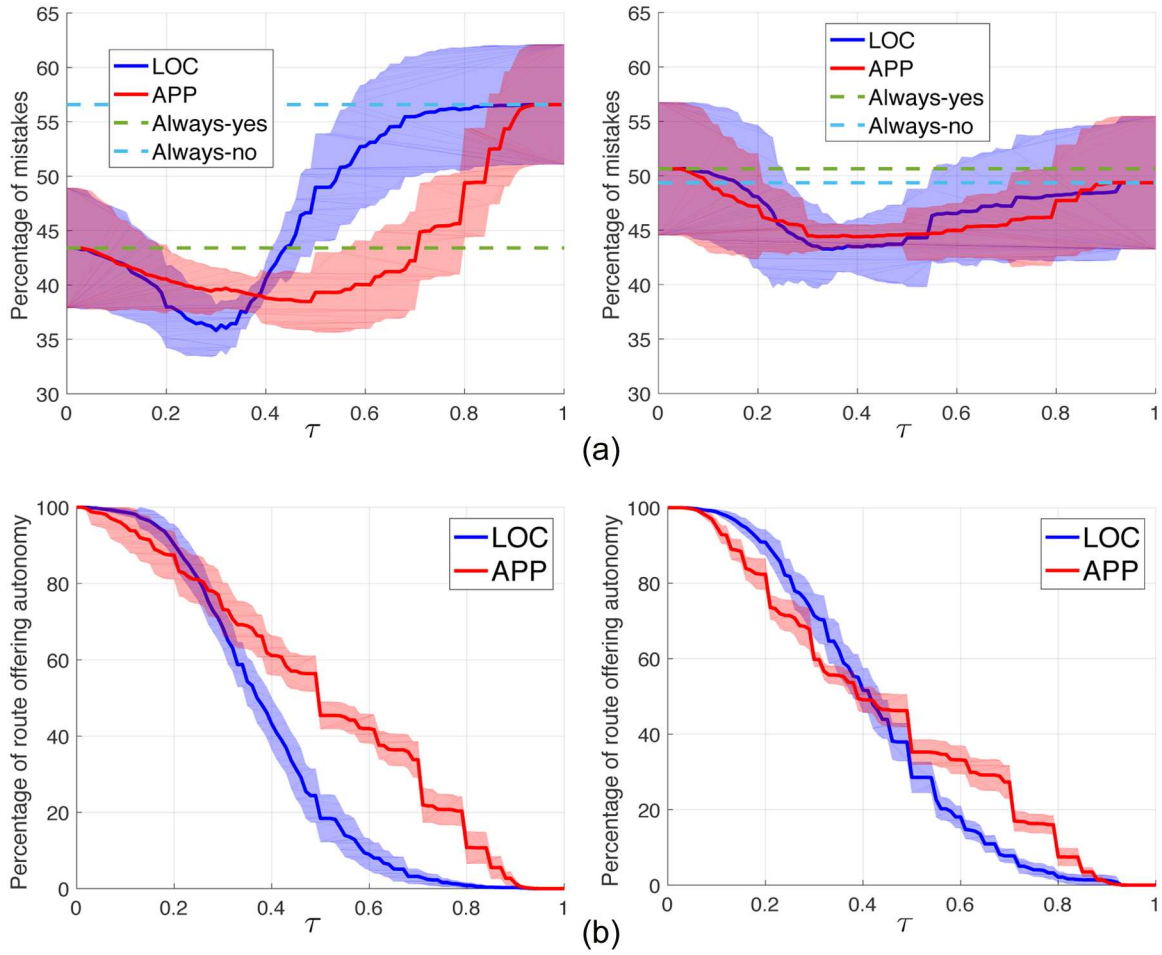

Fig. 8. Outcome of decision-making process. Although the robot does not offer autonomy for the entire trajectory (b), the number of perception mistakes is reduced (a). The evaluation is performed for all datasets in a leave-one-out fashion and for all values of the hyperparameter τ. Solid lines show the mean percentage of mistakes (a) and the mean percentage of autonomy offered (b), respectively. Shaded regions indicate one standard deviation from the mean.

beneficial, we leave it to future work, since it is only tangentially relevant to the purpose of this experiment: to accurately characterize the performance of different detectors (regardless of their actual accuracy).

Both of these baseline options are considerably higher than the methods proposed, which encourages us to believe that if we allow the robot to deny autonomy occasionally, the overall performance on a task is improved. The proposed methods help both the SVM+ACF and the MS-CNN detector, with APP being particularly important for SVM+ACF, and less so for MS-CNN. We believe this is because the latter is a higher capacity detector, which may already be appearance-invariant. Furthermore, for most values of τ, APP drives autonomously for a larger percentage of the route, as shown in Figure 8(b). We attribute the result that APP has fewer mistakes and offers more autonomy than LOC to the fact that it selects observations more carefully from past traversals.

The choice of τ not only has an influence on the total percentage of mistakes and the amount of autonomy offered, but also on the specific type of mistakes. For lower values of τ, both methods are more permissive of driving that leads to more false negative mistakes (type II). The robot fails to recognize that the perception system is operating poorly. For higher values of τ, both methods deny autonomy more often, which leads to more false positive mistakes (type I). These errors correspond to the less severe scenario of stopping the vehicle from driving despite good performance; therefore, higher values of τ might be preferable in practice.

In addition to τ, encouraging the robot to take either action can be achieved by adjusting the $L_{\text{offer}}/L_{\text{deny}}$ ratio, such that the action that incurs a lower cost will be selected more often (as demonstrated by Figure 3). For $L_{\text{offer}} = 3 \times L_{\text{deny}}$, for both prediction methods and detectors used in the experiments, we can reduce the number of type II errors even for the same value of τ. This is shown in Table 1, for $\tau = 0.6$. While it is straightforward to see that APP produces fewer type I errors than LOC, for type II errors, LOC seems to be preferable. This is because type II errors are computed strictly on the frames on which the decision taken

Table 1. Percentage of decision-making mistakes, separated into type I (denying autonomy despite performing well) and type II (offering autonomy despite performing poorly), as well as the percentage of the route driven autonomously (A). Results are shown for the two prediction methods proposed, LOC and APP, and for both pedestrian detectors employed, SVM+ACF and MS-CNN. The value of τ (the hyperparameter at which the action is taken) is set to 0.6. The table also shows how increasing the cost of action, L_{offer}, reduces the number of type II errors but also reduces the percentage of the route driven autonomously. The smallest percentage of errors of each type is shown in bold. While LOC appears to outperform APP in terms of type II errors, this is because type II errors are computed solely on the occasions in which autonomy was offered (thus the amount of data is smaller to begin with).

		$L_{\text{offer}} = L_{\text{deny}}$			$L_{\text{offer}} = 3 \times L_{\text{deny}}$		
		Type I (%)	Type II (%)	A(%)	Type I (%)	Type II (%)	A(%)
SVM+ACF	LOC	51.39	**1.36**	6.57	54.96	**0.51**	6.57
	APP	**27.66**	12.38	**41.31**	**38.69**	6.19	**24.09**
MS-CNN	LOC	39.65	**6.88**	16.59	44.69	**2.53**	7.19
	APP	**30.7**	14.1	**32.7**	**38.58**	7.37	**18.15**

was to offer autonomy, which is initially lower for LOC, as shown by the percentage of the route driven autonomously (A(%)). This holds for both SVM+ACF and MS-CNN.

Table 2 shows that, for a fixed percentage of the route driven autonomously (set to 30%, 50%, and 70%, respectively), APP also makes fewer type II errors (not seen in Table 1 owing to varying percentage of autonomy). This result is computed for the case of $L_{\text{offer}} = L_{\text{deny}}$ and the SVM+ACF detector.

Throughout these experiments, the robot has the opportunity to switch between two states of autonomy every few centimeters of the route it traverses, when, in practice, transitioning between the two would be highly inconvenient. For instance, for the MS-CNN detector with a detection threshold as well as a decision threshold of 0.7, the average length of a segment on which the robot does not switch its state is 1 m, with the longest correct prediction of performance amounting to 55 m and the longest incorrect prediction of performance to 30 m. Just as false image detections are sporadic, so are the decisions to switch the current state of autonomy. This creates a segmentation of the trajectory (into on and off sections of the route) that the two methods proposed do not correct for. While this work is mostly concerned with the decisions themselves, future work will address the stability of the predictions.

Figure 9 shows a qualitative assessment, made using the APP approach, to predict the performance of the MS-CNN detector. In particular, we look at examples in which the model correctly predicts success with high probability (i.e. when the past performance records are all correct detections), shown in Figure 9(a), and examples where the model correctly predicts failure with high probability (all performance records incorrect), shown in Figure 9(b). The success examples in Figure 9(a) all represent straightforward conditions (without adverse lighting or weather) in which a detector could be expected to perform well. The failure examples in Figure 9(b) particularly indicate the ability of the performance records to keep track of both false positive detections (such as a bike rack in the rightmost image that always produces spurious detections), and false negative detections (such as those caused by under- or overexposure in the left two images).

Equivalent results for the SVM+ACF detector are shown in Figure 10. In a similar vein, the frames correctly predicted to have good performance were obtained under standard operating conditions, while some of the predicted failures indicate adversity in the environment (such as the pole in the second image which is falsely detected as a pedestrian), or in the conditions (over- or underexposure in the last two images). Interestingly, some of the changes in lighting conditions are also tied to location: underexposure is usually a result of entering a tunnel (or indeed, any other location likely to produce extensive shadows), while overexposure occurs when exiting these covered or shadowed regions. Such factors could therefore also be largely captured by the LOC approach.

Thus, these results illustrate the most significant benefit of the performance records: in making the performance prediction specific to place or time and appearance, it becomes possible to consider, simultaneously, both fixtures in the environment that might cause problems (such as a bike rack) or appearance changes that could contribute to failure (such as exposure conditions).

5. Discussion

5.1. Reasoning with model uncertainty

A very simple alternative to the proposed methods is to utilize the inherent uncertainty in the outputs of the model itself (i.e. the probability score of each detection) as a predictor of confidence. One may use the entropies of the detection probabilities as an uncertainty measure, or indeed the probability values themselves (as this is directly mapped to the entropy)

Such methods (Grimmett et al., 2016) are significantly different, such that a like-for-like comparison is not straightforward. Indeed one may use the model uncertainty in the prediction of each detection to get an estimate of how likely that detection is to be correct. However, LOC and APP solve a slightly different problem: rather than looking

Table 2. Percentage of decision mistakes (type I and type II) for an equal amount of the route driven autonomously (30%, 50%, and 70%). The pedestrian detector used is SVM+ACF and equal costs are assumed for each action. APP outperforms LOC in terms of both type I and type II errors.

	30% autonomy		50% autonomy		70% autonomy	
	Type I (%)	Type II (%)	Type I (%)	Type II (%)	Type I (%)	Type II (%)
LOC	25.88	12.99	19.13	18.39	4.6	33.07
APP	**21.73**	**12.57**	**17.28**	**15.94**	**4.2**	**29.37**

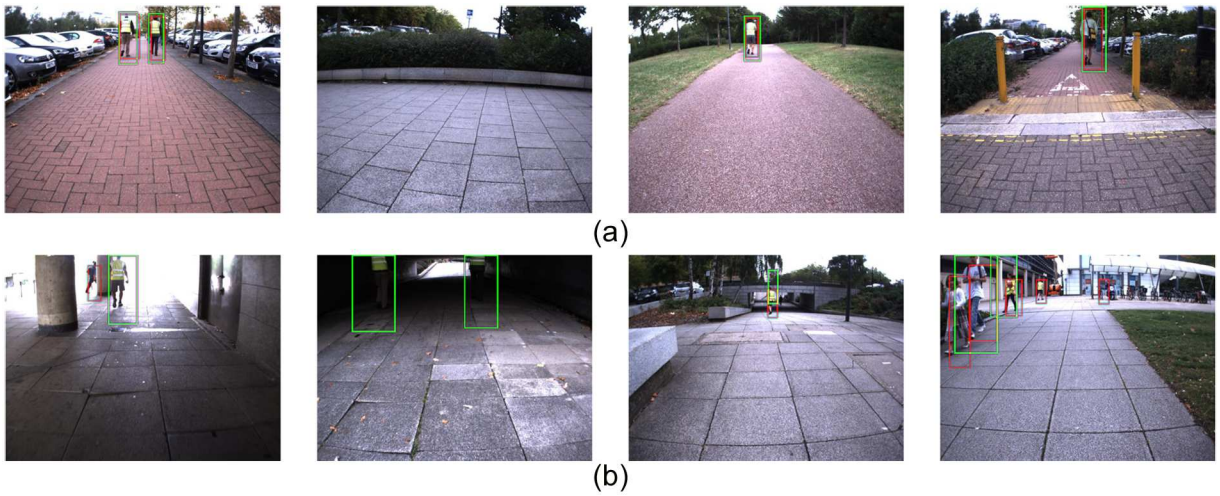

Fig. 9. Example images and their corresponding pedestrian detections using MS-CNN with a detector threshold of 0.8. The output of the pedestrian detector is shown in red and the ground truth is shown in green, with the goal of this work being to predict the detector accuracy correctly based on performance records. The figure shows examples in which we correctly predict success with high probability (a), or correctly predict failure with high probability (b), using the APP approach.

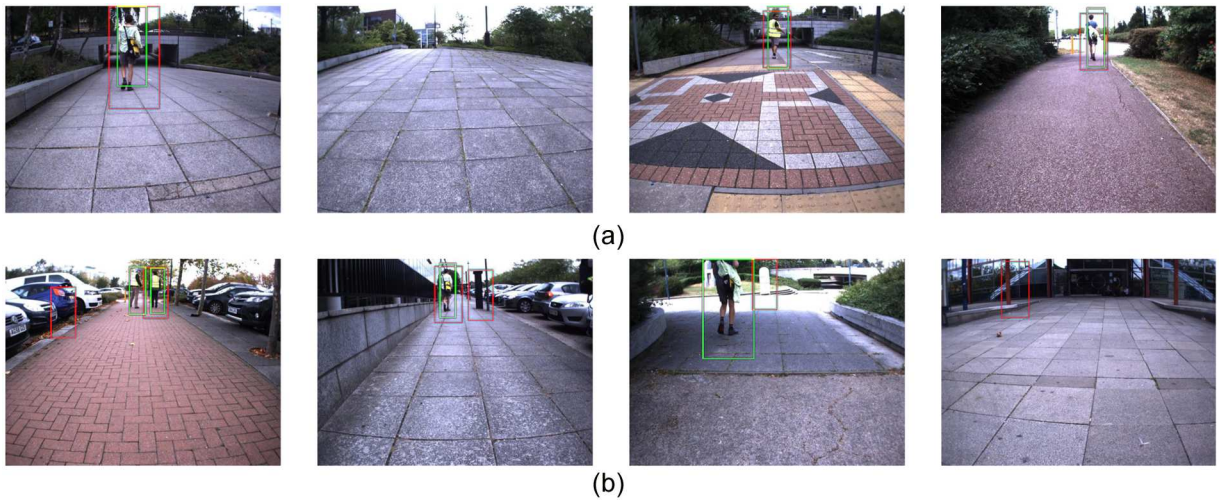

Fig. 10. Example images and their corresponding pedestrian detections using the SVM+ACF detector with a threshold of 0. The scores are not calibrated probabilistically and 0 was chosen because it provides the best trade-off between precision and recall. The output of the pedestrian detector is shown in red and the ground truth is shown in green, with the goal of this work being to predict the detector accuracy correctly, based on performance records. The figure shows examples in which we correctly predict success with high probability (a), or correctly predict failure with high probability (b), using the APP approach.

at the uncertainty of each detection sample (image crop), they classify an entire image captured at a certain location by looking back at past performance. We believe that making a fair comparison between retrospectively looking at past performance and introspectively looking at model uncertainty would require altering the initial frameworks in order to bring them to a comparable state.

Consider trying to characterize the performance of a pedestrian detector (with a threshold on the detection score for each box) using each of the two frameworks. On the one hand, LOC and APP do not use the incoming detections at all to take a decision (only to build the performance record). Therefore, this record incorporates only observations coming from detections above a certain threshold. On the other hand, an uncertainty-based framework cannot take decisions based on just a subset of the detection boxes, as they would merely correspond to detections with a high probability of success, which would unfairly bias the robot toward offering autonomy (since the low-probability detections have been removed). If, however, the low-probability detections are not removed, then the performance record will mostly be populated by observations of false detections, which would bias the robot toward not offering autonomy. This situation arises from the fact that introspection (as formulated by Grimmett et al. (2016)) addresses the problem of classifier uncertainty and not sliding-window detection uncertainty.

An additional concern in using the detector score or entropy as an indicator of confidence is that both thresholds (the *detection* threshold and the *decision* threshold) are coupled: a conservative decision threshold simply corresponds to a high detection threshold, and vice versa. Thus, we lose the ability to characterize decision-making performance as a function of τ for a single detection threshold. In other words, it is not possible to analyze decision-making performance and produce plots that are equivalent to Figure 8 for a single detector threshold.

5.2. Correcting for mistakes

Actively testing the value as well as the implications of the decision-making process is vital for long-range robot autonomy. We envision as future work not only predicting when a mistake is about to happen and switching the state of autonomy, but actively trying to correct for that mistake. For instance, if we separate the types of failure into false positives and false negatives and anticipate each type, we could dynamically change the operating threshold for the detection system, or intervene with place-specific experts to correct the detection outcome.

6. Conclusion

This paper introduces a framework to predict the performance of a perception system on an autonomous vehicle as a function of space, time, and appearance, and determine when to relinquish control to the human user. Two approaches are proposed: one that uses a *performance record* of the detector at nearby locations on previous traversals, and a second that additionally incorporates appearance-based cues. Experiments with different detector models indicate that the proposed methods produce fewer decision-making mistakes than keeping the robot autonomy always on or off. Moreover, selecting past observations from similar environmental conditions in an appearance-based fashion further reduces the number of mistakes. Our work calibrates the outcome of a vision system to what the world looks like at the time of operation rather than an a-priori validation set by keeping track of its past performance. We believe that the proposed records can improve with more experience in the same workspace and represent a step toward reliable vision systems operating in the real world. Future work will address how to predict in advance exactly when the driver has to intervene (in terms of a *time to failure*) in order to notify them accordingly, whilst utilizing appearance information from the current traversal.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the European Community's Seventh Framework Programme (grant number FP7-610603 (EUROPA2)) and the UK Engineering and Physical Sciences Research Council (grant number EP/J012017/1).

References

Badrinarayanan V, Kendall A, and Cipolla R (2017) SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Epub ahead of print 2 January 2017. DOI: 10.1109/TPAMI.2016.2644615.

Cai Z, Fan Q, Feris RS, et al. (2016) A unified multi-scale deep convolutional neural network for fast object detection. In: Leibe B, Matas J, Sebe N, et al. (eds.) *Computer Vision ECCV 2016* (*Lecture Notes in Computer Science*, vol. 9908). Cham: Springer, pp. 354–370.

Churchill W and Newman P (2013) Experience-based navigation for long-term localisation. *The International Journal of Robotics Research* 32(14): 1645–1661.

Churchill W, Tong CH, Gurau C, et al. (2015) Know your limits: Embedding localiser performance models in teach and repeat maps. In: *Proceedings of the IEEE international conference on robotics and automation (ICRA)*, Seattle, WA, 26–30 May 2015, pp. 4238–4244. Piscataway, NJ: IEEE.

Cummins M and Newman P (2011) Appearance-only SLAM at large scale with FAB-MAP 2.0. *The International Journal of Robotics Research* 30(9): 1100–1123.

Daftry S, Zeng S, Bagnell JA, et al. (2016) Introspective perception: Learning to predict failures in vision systems. In: *IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Daejeon, South Korea, 9–14 October 2016, pp. 1743–1750. Piscataway, NJ: IEEE.

Dalal N and Triggs B (2005) Histograms of oriented gradients for human detection. In: *Computer society conference on computer vision and pattern recognition (CVPR)*, San Diego, CA, 20–25 June 2005, vol. 1, pp. 886–893. Piscataway, NJ: IEEE.

Dequaire J, Tong CH, Churchill W, et al. (2016) Off the beaten track: Predicting localisation performance in visual teach and repeat. In: *Proceedings of the IEEE international conference on robotics and automation (ICRA)*, Stockholm, Sweden, 16–21 May 2016, pp. 795–800. Piscataway, NJ: IEEE.

Dollár P, Appel R, Belongie S, et al. (2014) Fast feature pyramids for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36(8): 1532–1545.

Dollár P, Wojek C, Schiele B, et al. (2009) Pedestrian detection: A benchmark. In: *IEEE conference on computer vision and pattern recognition*, Miami, FL, 20–25 June 2009, pp. 304–311. Piscataway, NJ: IEEE.

Dutta A, Veldhuis R, and Spreeuwers L (2015) Predicting face recognition performance using image quality. *arXiv* arXiv:1510.07119.

Felzenszwalb P, Girshick R, McAllester D, et al. (2010) Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32(9): 1627–1645.

Furgale P and Barfoot TD (2010) Visual teach and repeat for long-range rover autonomy. *Journal of Field Robotics* 27(5): 534–560.

Ganin Y and Lempitsky V (2015) Unsupervised domain adaptation by backpropagation. *Proceedings of Machine Learning Research* 37: 1180–1189.

Geiger A, Lenz P, Stiller C, et al. (2013) Vision meets robotics: The KITTI dataset. *The International Journal of Robotics Research* 32(11): 1231–1237.

Grimmett H, Triebel R, Paul R, et al. (2016) Introspective classification for robot perception. *The International Journal of Robotics Research* 35(7): 743–762.

Gurari D, Jain S, Betke M, et al. (2016) Pull the plug? Predicting if computers or humans should segment images. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, NV, 27–30 June 2016, pp. 382–391. Piscataway, NJ: IEEE.

Gurau C, Hawke J, Tong CH, et al. (2014) Learning on the job: Improving robot perception through experience. In: *Autonomously learning robots workshop at neural information processing systems (NIPS)*, Montreal, Quebec, Canada, 12 December 2014.

Gurau C, Tong CH, and Posner I (2016) Fit for purpose? Predicting perception performance based on past experience. In: Kuli D, Nakamura Y, Khatib O, et al. (eds.) *2016 International Symposium on Experimental Robotics. ISER 2016* (*Springer Proceedings in Advanced Robotics*, vol 1.) Cham: Springer, pp. 454–464.

Hawke J, Gurau C, Tong CH, et al. (2015) Wrong today, right tomorrow: Experience-based classification for robot perception. In: Wettergreen D and Barfoot T (eds.) *Field and Service Robotics* (*Springer Tracts in Advanced Robotics*, vol 113). Cham: Springer, pp. 173–186.

He K, Zhang X, Ren S, et al. (2016) Deep residual learning for image recognition. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, 27–30 June 2016, pp. 770–778. Piscataway, NJ: IEEE.

Hoiem D, Chodpathumwan Y, and Dai Q (2012) Diagnosing error in object detectors. In: *Proceedings of the 12th European conference on computer vision*, Florence, Italy, 7–13 October 2012, vol. III, pp. 340–353. Berlin: Springer-Verlag.

Holub A, Perona P, and Burl MC (2008) Entropy-based active learning for object recognition. In: *IEEE computer society conference on computer vision and pattern recognition workshops, 2008. CVPRW'08*. Anchorage, AK, 23–28 June 2008. Piscataway, NJ: IEEE.

Kapoor A, Grauman K, Urtasun R, et al. (2010) Gaussian processes for object categorization. *International Journal of Computer Vision* 88(2): 169–188.

Khosla A, Zhou T, Malisiewicz T, et al. (2012) Undoing the damage of dataset bias. In: Fitzgibbon A, Lazebnik S, Perona P, et al. (eds.) *Computer Vision ECCV 2012*. (*Lecture Notes in Computer Science*, vol. 7572). Berlin: Springer, pp. 158–171.

Kulis B, Saenko K, and Darrell T (2011) What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In: *IEEE conference on computer vision and pattern recognition (CVPR)*, Colorado Springs, CO, 20–25 June 2011, pp. 1785–1792. Piscataway, NJ: IEEE.

Linegar C, Churchill W, and Newman P (2015) Work smart, not hard: Recalling relevant experiences for vast-scale but time-constrained localisation. In: *IEEE international conference on robotics and automation (ICRA)*, Seattle, WA, 26–30 May 2015, pp. 90–97. Piscataway, NJ: IEEE.

McAllister R, Gal Y, Kendall A, et al. (2017) Concrete problems for autonomous vehicle safety: Advantages of Bayesian deep learning. *Proceedings of the twenty-sixth international joint conference on artificial intelligence (IJCAI)*, Melbourne, Australia, 19–25 August 2017, pp. 4745–4753. IJCAI.

McManus C, Furgale P, Stenning B, et al. (2013) Lighting-invariant visual teach and repeat using appearance-based lidar. *Journal of Field Robotics* 30(2): 254–287.

Merat N, Jamson AH, Lai FC, et al. (2014) Transition to manual: Driver behaviour when resuming control from a highly automated vehicle. *Transportation Research Part F: Traffic Psychology and Behaviour* 27: 274–282.

Nelson P, Linegar C, and Newman P (2016) Building, curating, and querying large-scale data repositories for field robotics applications. In: Wettergreen D and Barfoot T (eds.) *Field and Service Robotics* (*Springer Tracts in Advanced Robotics*, vol. 113). Cham: Springer, pp. 517–531.

Paton M, Pomerleau F, MacTavish K, et al. (2017) Expanding the limits of vision-based localization for long-term route-following autonomy. *Journal of Field Robotics* 34(1): 98–122.

Peynot T, Scheding S, and Terho S (2010) The Marulan data sets: Multi-sensor perception in a natural environment with challenging conditions. *The International Journal of Robotics Research* 29(13): 1602–1607.

Peynot T, Underwood J, and Scheding S (2009) Towards reliable perception for unmanned ground vehicles in challenging conditions. In: *IEEE/RSJ international conference on intelligent robots and systems (IROS)*, St. Louis, MO, 10–15 October 2009, pp. 1170–1176. Piscataway, NJ: IEEE.

Ren S, He K, Girshick R and Sun J (2015) Faster R-CNN: Towards real-time object detection with region proposal networks. In: *Advances in neural information processing systems (NIPS)* (eds. C Cortes, ND Lawrence, DD Lee, et al.), Montreal, Quebec, Canada, 7–12 December 2015, pp. 91–99. Red Hook, NY: Curran Associates, Inc.

Torralba A and Efros AA (2011) Unbiased look at dataset bias. In: *IEEE conference on computer vision and pattern*

recognition (CVPR), Colorado Springs, CO, 20–25 June 2011, pp. 1521–1528. Piscataway, NJ: IEEE.

Tzeng E, Hoffman J, Saenko K, et al. (2016) Adversarial discriminative domain adaptation. In: *Advances in neural information processing systems (NIPS) workshop on adversarial learning*, Barcelona, Spain, 9 December 2016.

Wang DZ and Posner I (2015) Voting for voting in online point cloud object detection. In: Kavraki LE, Hsu D and Buchli J (eds) *Proceedings of robotics: Science and systems*. Rome, Italy, 13–17 July 2015. Robotics Science and Systems Foundation.

Yosinski J, Clune J, Bengio Y, et al. (2014) How transferable are features in deep neural networks? In: *Proceedings of the 27th international conference on neural information processing systems, NIPS'14*, Montreal, Quebec, Canada, 8–13 December 2014, pp. 3320–3328: Cambridge, MA: MIT Press.

Zhang P, Wang J, Farhadi A, et al. (2014) Predicting failures of vision systems. In: *IEEE conference on computer vision and pattern recognition (CVPR)*, Columbus, OH, 23–28 June 2014, pp. 3566–3573. Piscataway, NJ: IEEE.