Wrong Today, Right Tomorrow: Experience-Based Classification for Robot Perception

Jeffrey Hawke†, Corina Gurău†, Chi Hay Tong and Ingmar Posner

Abstract This paper is about building robots that get better through use in their *particular* environment, improving their perceptual abilities. We approach this from a life long learning perspective: we want the robot's ability to detect objects in its specific operating environment to evolve and improve over time. Our idea, which we call Experience-Based Classification (EBC), builds on the well established practice of performing hard negative mining to train object detectors. Rather than cease mining for data once a detector is trained, EBC continuously seeks to learn from mistakes made while processing data observed during the robot's operation. This process is entirely *self-supervised*, facilitated by spatial heuristics and the fact that we have additional scene data at our disposal in mobile robotics. In the context of autonomous driving we demonstrate considerable object detector improvement over time using 40km of data gathered from different driving routes at different times of year.

1 Introduction

Object detection forms one of the cornerstones of autonomous operation in complex, dynamic environments. Whether it concerns the detection of assets for the purpose of infrastructure survey, the detection of wares and co-workers for applications in logistics, or the detection of other traffic participants in an autonomous driving context, object detectors need to provide fast, reliable performance *across* a number of workspaces. This is explicitly encouraged in the machine vision community as witnessed by, for example, the ImageNet Large Scale Visual Recognition Challenge [6].

Jeffrey Hawke, Corina Gurău, Chi Hay Tong, and Ingmar Posner
University of Oxford, Oxford OX1 3PJ, United Kingdom, e-mail: [jhawke, corina, chi, ingmar]@robots.ox.ac.uk
† J. Hawke and C. Gurău contributed equally to this work.

However, while much progress is being made, error rates of state-of-the-art approaches are still prohibitive, particularly for safety critical applications (e.g. [1] for the case of pedestrian detection). This is often due to a significant amount of variation in the negative class which, in reality, is not captured in the training data. While it is relatively easy to obtain negative samples, computational limits imply that we should only include ones that have a large effect on the decision boundary. The standard method for obtaining relevant negative samples is known as hard negative mining (HNM) [25, 13], which is commonly used to bootstrap the underlying classifier used in an object detector. HNM is widely considered a mandatory part of detector training, where the classifier is first trained on the original training data and then used to perform object detection on a *labelled* dataset. False positives are identified using the ground truth labels provided and included for classifier retraining. This provides considerable improvement over the original detector, but the data used for negative mining strongly influences the resulting performance due to dataset bias [28, 21]. In robotics, where we have a limited range of operation and are not as concerned with general performance, biasing the detector's performance to our workspace is a powerful tool.

In robotics, in order to improve performance for a particular application, scene context – obtained through online sensing or contained in (semantic) map priors – is commonly leveraged as a filter (e.g. [15, 24]). Typically, this takes the form of discarding detections as spurious if certain validation criteria are not met (e.g. a car needs to be found on or near a road [22]).

In this work we also exploit scene context to validate the detections obtained. However, we advocate a radically different detector deployment model from the status quo, which leads to *self-supervised* and *environment-dependent* performance improvement over the lifetime of the detector. This reflects our desire for lifelong learning systems which excel in a robot's specific application domain instead of providing mediocre performance everywhere.

Our approach, is from one perspective, a simple and straightforward one and yet it brings remarkable and profound benefits to our problem domain: some applications of embedded perception can afford to trade generality for specificity. Robotic agents should adjust to a vanishingly small subset of all possible workspaces: the ones they operate in, or 'experience', on a daily basis.

Inspired by hard negative mining, we continue to train our detectors in a self-supervised learning by exploiting scene context from the robot's operating environment. This is achieved by continuously feeding back into the training process of the detector any false positives identified by a validation step throughout the lifetime of the system. We call this process Experience-Based Classification (EBC).

In effect, EBC automatically trains detectors for specific operating environments. While this may lead to overfitting to the background encountered, we argue that this is exactly what is required in mobile robotics where autonomous agents often traverse the same workspace over and over again. In fact, EBC relies on this behaviour and, inspired by recent work in the vision community such as [7], exploits similarities in geo-spatially related lo-

Fig. 1: Images from the same route at two different times of year (January and May) on which we performed pedestrian detection. While the pedestrians look similar in all images, the background class is quite different, with visible seasonal effects. This suggests the need for environment-dependent classifiers. False positives are shown in purple, while true positives have a yellow bounding box. A few iterations of EBC over the course of a few days show great improvement.

cations. Furthermore, the self-supervised, operational nature of this approach means that it can incorporate considerably more data in training than conventionally performing HNM on small canonical datasets. This opens up the possibility for life-long learning on robot perception, building up a collection of environment dependent object detectors over the robot's lifetime.

EBC is agnostic to the application domain, detection framework and object class considered. However, in this paper we frame the discussion in the context of pedestrian detection for autonomous driving (see Figure 1). We utilise the fact that object detection is often performed alongside navigation and that current navigation solutions localise against a previously-acquired map [14, 3, 23]. This provides the scene context for EBC.

2 Related Work

One common approach to pedestrian detection from monocular imagery utilises a linear SVM classifier on Histogram of Oriented Gradients (HOG) features [5]. The use of a linear model permits efficient sliding window computations [9] when a sliding window detector is implemented using this classifier. More recent work in pedestrian detection has extended this to use alternative feature types such as Aggregate Channel Features (ACF) [8], or alternative classifiers such as Latent SVMs with deformable parts models [13], and decision trees with Adaboost [1]. For our sliding window detector, we elected to use the same feature type as the current state of the art pedestrian detector (Aggregate Channel Features), but with a simpler linear classifier model and a reduced number of scales.

3D scene information has been primarily used in object detection to generate Regions of Interest (ROIs). For example, a ground plane computed from stereo imagery can provide a search space for detections (e.g. [15, 24]), or enforce scale [16]. Enzweiler et al. [10] extend this idea by maintaining a height-based representation of the local environment to generate ROIs, and Ess et al. [11] jointly infer the depth, ground plane and object detections.

Instead of generating ROIs to present to our classifier, we invert the order and apply scene information after we compute detections. While both approaches provide us with a set of valid positive classifications, this ordering also allows us to obtain a set of informative negative data samples that can be used for detector improvement.

As mentioned in the introduction, the conventional approach for obtaining these hard negatives when initially training a detector is Hard Negative Mining (HNM), performed on a labelled training dataset. Initially introduced by Sung and Poggio [25] as a bootstrap method for expanding the training set, Felzenszwalb et al. [13] tailor it for structural SVMs by defining 'hard' negatives as examples that are incorrectly classified or within the margin of the SVM classifier. HNM has also been used for multiclass object detection [20], where positive samples of other classes can serve as hard negatives.

Instead of HNM, Hariharan et al. [18] suggested training Linear Discriminant Analysis (LDA) classifiers with an extremely large negative class. This was made possible for SVM classifiers by Henriques et al. [19], who used block-circulant decomposition to train with an approximation to the set of all negative samples from a series of images. In effect, training with a vast set of negatives reduces the need to specifically mine for hard negatives. While efficient, the training remains limited by computational resources, and does not escape the core requirement of labelled data.

This prior work on HNM is complementary to our work on EBC. HNM still forms a critical step when initially training a detector, and EBC builds on this to continue bootstrapping the detector to the robot's operating environment. In addition, many of these techniques to extend HNM could equally be applied to EBC. This paper is an extension of a previous workshop paper

[17], which showed that EBC is comparable to HNM on the same labelled data. Here we consider the effect of place and season on life-long learning.

In all these approaches, labelled data are used to identify negative samples. While the labelling effort may be tolerated for individual datasets, real-world operation is subject to variation from seasonal, lighting and environmental changes. This has a significant impact on detection performance, but manually labelling data for all of these scenarios is impractical for life-long learning in robotics. EBC is able to meet these requirements by identifying relevant samples in a self-supervised manner.

We share some similarities with the concept of group induction [27], where self-supervised training is performed by alternating between classifying unlabelled tracks and incorporating the most confident positive classifications in retraining. Our approach differs by the fact that we use an external signal in the form of an environmental prior to provide labels for the whole scene. This allows us to focus only on hard samples and provides a means to automatically train our detectors for specific environments.

3 Framework Description

EBC augments a standard perception pipeline by introducing a scene filtering step after object detection, a memory bank of negative samples and classifier retraining. Our implementation of this system is depicted in Figure 2. The following sections describe the function of each component in further detail and provide specific information about our implementation.

3.1 Object Detector

In general terms, an object detector processes a data stream and produces detections. EBC serves as a wrapper for this, providing additional training samples for lifelong improvement. In this work we employ a linear SVM classifier to classify whether an image patch is part of the positive class or the negative class. Given an input image, we first compute features for the entire image, and then employ a sliding window approach to obtain classification scores. Multiscale detection is performed by resizing the image and repeating the process. Finally, non-maximal suppression is used to filter out overlapping detections. The output is a set of bounding boxes which correspond to subwindows that score above a threshold, which are deemed to be positive detections. Further detail on the object detector specifics can be found in Section 4.

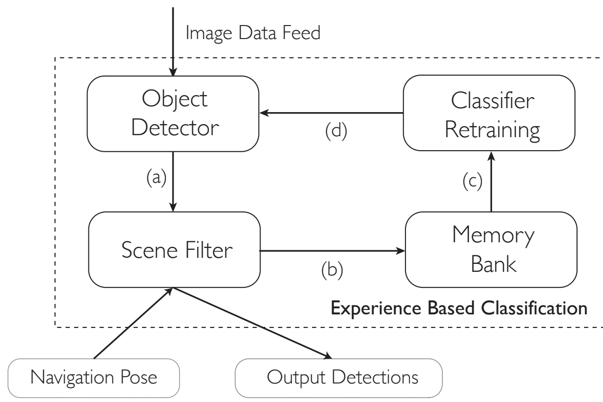


Fig. 2: The EBC architecture implemented in this paper. (a) An object detector provides detections based on the image feed. (b) A scene prior is used to filter out detections that do not touch the ground plane or have an unexpected scale. (c) Rejected samples are stored. (d) The detector is retrained at the end of an experience using additional rejected samples. The EBC detector improves through successive outings as it automatically adjusts to what it experiences.

3.2 Scene Filter

The scene filter is a core component of the EBC framework. Given a set of detections, the scene filter employs local context to filter out false positives according to strong heuristics. Accepted detections are passed on to the remainder of the perception pipeline, while rejected detections are stored in the memory bank. Since the rejected samples are detections that scored highly in the previous step these are by definition hard negatives.

Given localisation information and a 3D scene prior, we first look up the local ground plane for our current location, then project the local ground plane into the image. This is used by a first filter, which rejects detections that lie off the ground plane for the current navigation frame. Our second filter then projects each remaining bounding box into the 3D scene to ensure detections are of a viable scale. The application of these heuristics is illustrated in Figure 3. The scene filtering step should be conservative to avoid rejecting a valid detection (true positive), which may lead to semantic drift [4]. The goal of this filtering component is to reduce the number of false positives while not introducing false negatives.

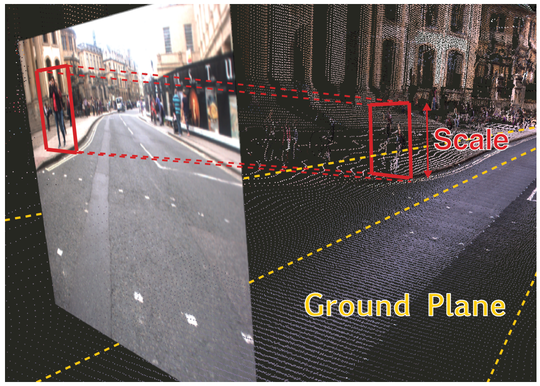


Fig. 3: An illustration of the scene filter employed in this work, using localisation information and heuristics such as scale and ground correction.

3.3 Memory Bank and Retraining

The final step of the EBC cycle augments the original training set with the rejected samples and retrains the classifier model. Since these additional negatives are obtained during operation, each subsequent training cycle further adapts the classifier to the specific environment. It should be noted that data streams gathered from mobile robotic platforms tend to be spatially and temporally correlated. This can cause problems in retraining as most classifiers assume independent, identically distributed data. Subsampling may be required to avoid these issues.

4 Experimental Evaluation

4.1 Methodology

We seek to evaluate the implications of an object detector learning from the environment it experiences. We do this by taking a common baseline detector model, then use this to train separate detectors for different classes of data, comparing their performance on test datasets from these same classes. To do this, we put a baseline pedestrian detector through successive training cycles on urban driving datasets gathered from two different routes in Oxford at different times of year. We anticipate that the detector which has learned from operating data which most closely matches the test data (place and season) will perform the best, as the detector becomes fitted to the operating environment.

A single experiment consisted of the baseline detector being presented with driving data from successive days, with a detector retraining step between each dataset. This process follows the EBC system architecture diagram in Figure 2. For a given detector model, the image data from the single specified urban driving dataset was processed to compute detections. The detections were then processed by the scene filter to validate or reject the samples according to spatial heuristics. The resulting negative samples were then sampled (taking the top 10 false positives from a random frame in every second of time), aggregated with prior rejected negative data samples, then used to retrain the detector along with the original training data. The negative data was weighted to ensure the class balance from the original training data was maintained.

Each experimental run was evaluated against a manually labelled test dataset which shared the same location and environmental conditions as one of the training categories.

4.2 Baseline Detector

Our baseline pedestrian detector used a classifier trained using LIBLINEAR [12] on the INRIA Pedestrian Dataset [5] with Aggregate Channel Features and a similar training methodology to [8]. We performed ten-fold cross validation on a training set consisting of 1237 cropped positive pedestrian samples, and 12180 sampled negatives (10 windows per negative image). The final step in the detector training process is a bootstrapping step consisting of ten consecutive cycles of HNM using the INRIA negative images. In each HNM cycle the classifier was presented with 10000 random negative cropped samples extracted from the negative images. Misclassified 'hard' negatives were saved and used as additional training data to retrain the classifier.

4.3 Datasets

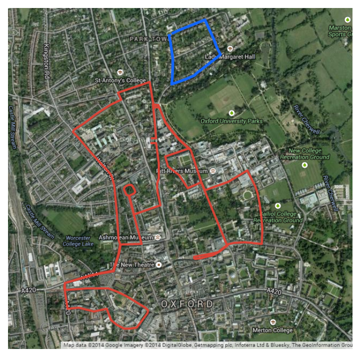
To show the impact of environmental variation and to evaluate our self-supervised learning approach, we used twelve different urban driving training datasets gathered with a Bumblebee2 stereo camera mounted on our Wildcat vehicle (Figure 4a) driving around Oxford. These unlabelled datasets were gathered from two different routes from successive outings at different times of year. We allocated these datasets to three categories based on the route and time of year (season), with 4 training datasets per category. These categories are referred to in this section as *North Oxford January*, *North Oxford May*, and *Central Oxford August*. A map of the routes is provided in Figure 4b, with no overlap between the North Oxford and Central Oxford routes. For all datasets, we used only the left stereo image with a capture rate of 20Hz.

For evaluation, we used an additional manually labelled test dataset from the *North Oxford January* category. This provided a total of 40km of unlabelled training data and 2km of labelled test data. The datasets are summarised in Table 1.

Route		Train 1	Train 2	Train 3	Train 4	Train Total	Test
North Oxford January	Distance (km)	2.60	2.01	1.92	1.92	8.45	1.99
	Image frames	12782	9436	8172	8215	38605	9155
	Time (min)	10.7	7.86	6.81	6.84	32.2	7.63
North Oxford May	Distance (km)	1.43	1.95	1.01	1.01	5.40	-
	Image frames	6066	8676	4001	3977	22720	-
	Time (min)	5.06	7.23	3.33	3.31	18.9	-
Central Oxford	Distance (km)	6.91	6.79	6.68	6.56	26.94	-
	Image frames	36472	27720	27607	23463	115262	-
	Time (min)	30.4	23.1	23.0	19.6	96.1	-

Table 1: A summary of the datasets used for training and evaluation.

a



b

Fig. 4: The Wildcat vehicle (left) used to gather the image data, and a map (right) depicting the routes for our datasets, where we gathered images in January, May, and August. The difference in time of year provided seasonal variation, which affects the visual appearance of the scene. The two North Oxford routes, illustrated in blue, provided variation in season, and the Central Oxford route, depicted in red, provided a difference in location.

4.4 Results

We trained three different detectors from the same starting base detector, one per category. These detectors are referred to by their category name: *North Oxford January* and *North Oxford May*, and *Central Oxford August*. Each detector was evaluated against a separate test dataset also derived from the North Oxford route during January.

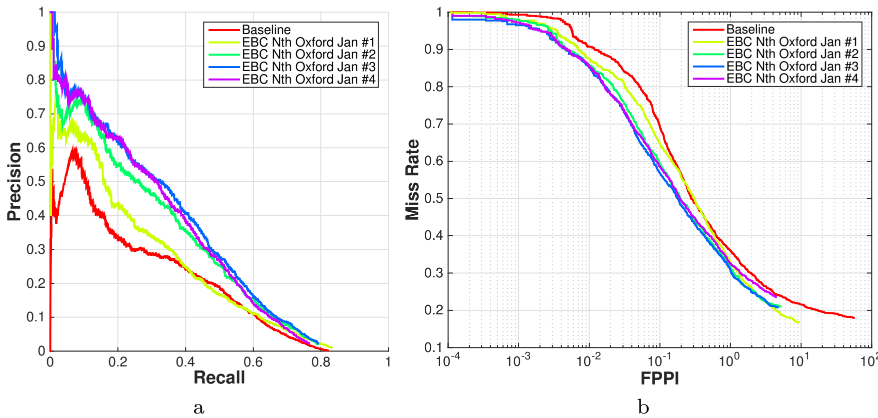


Fig. 5: The precision-recall (left) and miss rate-false positives per image (right) performance of the detector during learning, tested on the North Oxford January data. The PR curve performance increases with the first three datasets observed, moving to the top right corner of the graph. This then settles with a very slight performance drop on the fourth dataset. The same trend is visible in the MR-FPPI graph with the curves moving to the lower left corner.

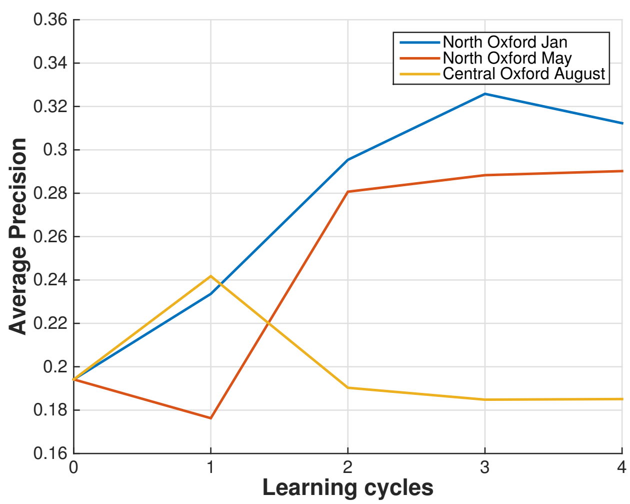


Fig. 6: The average precision (obtained by computing the area under a PR curve) for three detectors trained using EBC, evaluated on the North Oxford January test dataset. Learning from operating on the same route (North Oxford) improves performance over the baseline detector, with the detector shown data from the same season as the evaluation set performing the best (January). The detector which learned from operation on a different route and time of year (Central Oxford August) does not improve performance.

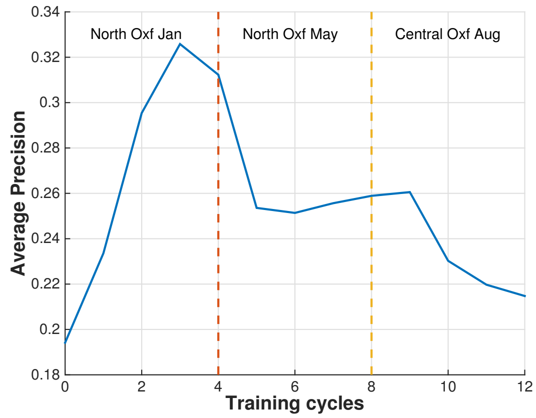


Fig. 7: The average precision (obtained by computing the area under a PR curve) for a detector trained using EBC on all datasets (both routes), evaluated on the North Oxford January test dataset. The detector shows a performance improvement with January data, but drops as it incorporates data from dissimilar environments, changing seasons to May, then changing route and season to August.

Firstly, the results in Figure 5 show that we are able to improve the perceptual performance of a detector by training it on data gathered from the environment it operates in. However place is clearly important in Figure 6. We see that both detectors trained in the same place (North Oxford) improve notably in performance, whereas the detector trained in a different place (Central Oxford) does not.

Secondly, the same figure shows that there is a seasonal effect in addition to the spatial similarities. There are clear perceptual differences between the two North Oxford seasons, and a detector trained on a driving route in January has improved performance when operating in January compared to a detector which learned from the same driving route in May.

These results support our argument for experience-specific classifiers. However, while it is clear that a detector trained for its operating environment is better than the general baseline detector, this raises questions around the necessary spatial and temporal resolution for these experiences in robot perception. To confirm the value of experience-specific classifiers, we also investigated the effects of simply amalgamating all the training data into one classifier, with all twelve datasets processed in temporal order (January through August). The results in Figure 7 show that this detector is not comparable to the detector trained only on data from the same place and season as the evaluation data, with performance degrading substantially as dissimilar data is observed and learned. This result adds further weight to the argument for training place dependent classifiers, and emphasises the need for research into what defines a 'place'. Our trials considered a small set of possible places and conditions, and it is likely that experiences in robot perception will be influenced by more than simply season and route. These factors could include weather, lighting, and additional environmental changes such as traffic.

Finally, we note that we have only showed the raw detector performance in our experimental trials. Since the scene filter is already incorporated into the EBC framework, we can also validate our detections while running online if a 3D scene prior and localisation information is available. The performance increase from the scene filter on the detector's output decreases over successive training cycles, with a large initial improvement tapering off to a very small difference by the end of our trials in Figure 8. The small difference at the end may be attributed to the fact that the ACF model is sufficiently expressive to cover what the current scene filter is able to invalidate. Further investigation is needed into the unintended slight drop in precision at higher recall when using the scene filter. Additional checks may be needed to achieve better results, potentially including computationally expensive offline checks.

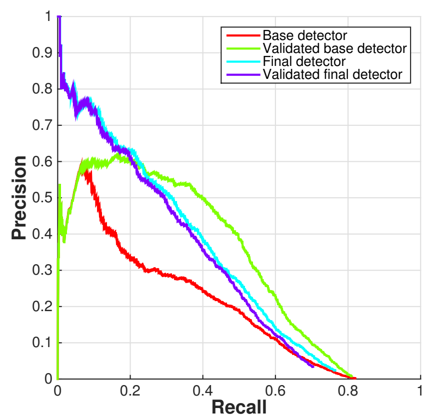


Fig. 8: Performance increase provided by the scene filter (referred to as 'validating' a classifier) when applied to both the base classifier and the final EBC detector on the North Oxford January test dataset. The invalidated data from the scene filter facilitates learning from the environment.

5 Conclusions

Though general object detection remains a noble goal, applications in robotics tend to be constrained to particular operating environments. We can exploit this fact to obtain practical systems which excel in a specific application domain. This is a major step towards reliable performance for real-world safety-critical systems. In particular, we make use of scene context to validate detections, and feed the rejected samples back to retrain the detector. This augmentation to the standard perception pipeline provides self-supervised environment-dependent improvement over the lifetime of the system. We call this process Experience-Based Classification.

Using urban driving data, we demonstrate that EBC provides a means to improve a general baseline object detector beyond what conventional negative data mining on a training dataset achieves. This suggests great utility in training experience-specific classifiers, potentially leading to life-long learning in robot perception without the need for human assistance. Perceptual systems benefit from being trained to suit the local environment and their performance varies as the robot experiences different environments.

Our experimental results show that environment-specific tuning provides benefits in performance at the cost of generality, but the results raise a number of research questions, primarily around what defines a robot's perceptual experience. While we manually divided the datasets here, we require an automated method to determine when to train new classifiers based on some metric of difference between perceptual experiences. This could be achieved through localisation, with a new detector model for every small map segment. However this approach would not accommodate normal variation in weather, lighting, and seasons. We believe that there is some benefit in pursuing a data driven approach, transferring classifiers to different locations with similar observed environmental conditions. Probabilistic topic modelling [2] offers a possible mechanism for this. Finally, as we desire lifelong learning, we must address the issues of positive mining [26], further scene filter checks (including expensive offline checks), semantic drift [4], and when to 'forget' data.

Acknowledgements

The authors gratefully acknowledge the support of this work by the EU project FP7-610603 (EUROPA2), EPSRC grant EP/J012017/1 and the UK Space Agency grant ST/L002981/1.

References

1. Benenson, R., Mathias, M., Timofte, R., Van Gool, L.: Pedestrian detection at 100 frames per second. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pp. 2903–2910 (2012)
2. Blei, D.M.: Probabilistic topic models. Communications of the ACM **55**(4), 77–84 (2012)
3. Churchill, W., Newman, P.: Experience-based navigation for long-term localisation. International Journal of Robotics Research (IJRR) **32**(14), 1645–1661 (2013)
4. Curran, J.R., Murphy, T., Scholz, B.: Minimising semantic drift with mutual exclusion bootstrapping. In: Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics, pp. 172–180 (2007)
5. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 886–893 (2005)

6. Deng, J., Dong, W., Socher, R., Li, L., Li, K., Fei-fei, L.: ImageNet: A large-scale hierarchical image database. In: Proceedings of the IEEE Conference on Computer Vision and Patter Recognition (CVPR), pp. 248–255. Miami, Florida, USA (2009)
7. Doersch, C., Singh, S., Gupta, A., Sivic, J., Efros, A.A.: What makes paris look like paris? ACM Trans. Graph. **31**(4), 101 (2012)
8. Dollár, P., Appel, R., Belongie, S., Perona, P.: Fast feature pyramids for object detection. PAMI (2014)
9. Dubout, C., Fleuret, F.: Exact acceleration of linear object detectors. In: Proceedings of the European Conference on Computer Vision (ECCV), 7574, pp. 301–311. Florence, Italy (2012)
10. Enzweiler, M., Hummel, M., Pfeiffer, D., Franke, U.: Efficient stixel-based object recognition. In: 2012 IEEE Intelligent Vehicles Symposium (IV), pp. 1066–1071 (2012)
11. Ess, A., Leibe, B., Gool, L.V.: Depth and appearance for mobile scene analysis. In: Proceedings of the International Conference on Computer Vision (ICCV) (2007)
12. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: A library for large linear classification. Journal of Machine Learning Research (JMLR) **9**, 1871–1874 (2008)
13. Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE Transactions on Pattern Analysis and Machine Intelligence **32**(9), 1627–1645 (2010)
14. Furgale, P., Barfoot, T.D.: Visual teach and repeat for long-range rover autonomy. Journal of Field Robotics **27**(5), 534–560 (2010)
15. Gavrila, D.M., Munder, S.: Multi-cue pedestrian detection and tracking from a moving vehicle. Int J Comput Vision **73**(1), 41–59 (2007)
16. Gerónimo, D., Sappa, A.D., Ponsa, D., López, A.M.: 2d–3d-based on-board pedestrian detection system. Computer Vision and Image Understanding **114**(5), 583–595 (2010)
17. Gurau, C., Hawke, J., Tong, C.H., Posner, I.: Learning on the job: Improving robot perception through experience. In: Neural Information Processing Systems (NIPS) Workshop on "Autonomously Learning Robots". Montreal, Quebec, Canada (2014)
18. Hariharan, B., Malik, J., Ramanan, D.: Discriminative decorrelation for clustering and classification. European Conference on Computer Vision (2012)
19. Henriques, J., Carreira, J., Caseiro, R., Batista, J.: Beyond hard negative mining: Efficient detector learning via block-circulant decomposition. In: 2013 IEEE International Conference on Computer Vision (ICCV), pp. 2760–2767 (2013)
20. Kanezaki, A., Inaba, S., Ushiku, Y., Yamashita, Y., Muraoka, H., Kuniyoshi, Y., Harada, T.: Hard negative classes for multiple object detection. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) (2014)
21. Khosla, A., Zhou, T., Malisiewicz, T., Efros, A.A., Torralba, A.: Undoing the damage of dataset bias. In: Computer Vision–ECCV 2012, pp. 158–171. Springer (2012)
22. Petrovskaya, A., Thrun, S.: Model based vehicle detection and tracking for autonomous urban driving. Autonomous Robots **26**(2-3), 123–139 (2009)
23. Stewart, A., Newman, P.: LAPS - localisation using appearance of prior structure: 6-DOF monocular camera localisation using prior pointclouds. In: Proc. IEEE International Conference on Robotics and Automation (ICRA). Minnesota, USA (2012)
24. Sudowe, P., Leibe, B.: Efficient use of geometric constraints for sliding-window object detection in video. In: Proceedings of the International Conference on Computer Vision Systems (ICVS) (2011)
25. Sung, K.K., Poggio, T.: Example-based learning for view-based human face detection. IEEE Transactions on Pattern Analysis and Machine Intelligence **20**(1), 39–51 (1998)
26. Teichman, A., Thrun, S.: Tracking-based semi-supervised learning. International Journal of Robotics Research (IJRR) **31**(7), 804–818 (2012)
27. Teichman, A., Thrun, S.: Group induction. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2757–2763. Tokyo, Japan (2013)
28. Torralba, A., Efros, A.A.: Unbiased look at dataset bias. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pp. 1521–1528. IEEE (2011)