Driven Learning for Driving: How Introspection Improves Semantic Mapping

Rudolph Triebel, Hugo Grimmett, Rohan Paul, and Ingmar Posner

Abstract This paper explores the suitability of commonly employed classification methods to action-selection tasks in robotics, and argues that a classifier's *introspective* capacity is a vital but as yet largely under-appreciated attribute. As illustration we propose an active learning framework for semantic mapping in mobile robotics and demonstrate it in the context of autonomous driving. In this framework, data are selected for label disambiguation by a human supervisor using uncertainty sampling. Intuitively, an introspective classification framework – i.e. one which moderates its predictions by an estimate of how well it is placed to make a call in a particular situation – is particularly well suited to this task. To achieve an efficient implementation we extend the notion of introspection to a particular sparse Gaussian Process Classifier, the Informative Vector Machine (IVM). Furthermore, we leverage the information-theoretic nature of the IVM to formulate a principled mechanism for forgetting stale data, thereby bounding memory use and resulting in a truly life-long learning system. Our evaluation on a publicly available dataset shows that an introspective active learner asks more informative questions compared to a more traditional non-introspective approach like a Support Vector Machine (SVM) and in so doing, outperforms the SVM in terms of learning rate while retaining efficiency for practical use.

1 Introduction

In answering the question 'where am I?' roboticists have gone to great lengths to model, manage and, indeed, exploit uncertainty. This, however, is not as yet the case when it comes to asking 'what is this?'. As we aspire to robust, long-term

Rudolph Triebel
Mobile Robotics Group, Oxford University, UK, e-mail: rudolph.triebel@in.tum.de,
who has since moved to Computer Vision Group, Technical University of Munich, Germany

Hugo Grimmett, Rohan Paul, and Ingmar Posner
Mobile Robotics Group, Oxford University, UK, e-mail: {hugo, rohanp, hip}@robots.ox.ac.uk

autonomous operation our systems have to contend with vast amounts of continually evolving, non-i.i.d. data from which information needs to be assimilated. This presents a challenge and an opportunity particularly to the robotics community as here the real cost of failure can be significant. We believe that *realistic* estimates of uncertainty are pivotal to achieving robust and efficient decision making in robotics. In particular, classification as a precursor to *action-selection* seems to be largely disregarded by the community.

We frame our argument in the context of offline semantic mapping. Significant progress in autonomous driving in recent years has inspired a view that successful autonomous operation in complex, dynamic environments critically depends on *a-priori* available semantic maps representing ostensibly permanent aspects of the environment such as lane markings, traffic light positions and road sign information (see, for example, [3, 22]). Owing to their safety-critical nature, these maps are typically created manually for particular routes [5]. This is, of course, an expensive process which scales badly with the number of routes for which autonomous operation is to be provided. Much, therefore, can be gained by reducing human involvement in this process and thus providing a robust and scalable solution.

A prominent approach to tackling such a challenge is that of *active learning*, where classification results are iteratively refined by asking a human supervisor for ground-truth labels in ambiguous cases and incorporating the added information into classifier training. To the best of our knowledge this paper is the first in robotics to present an efficient and scalable active learning framework for the task of offline semantic mapping. Crucially, however, our work is also set apart from the vast majority of the related works in active learning by the unusual stance we take with regards to uncertainty estimates in the system. Commonly, active learning relies on selecting data for human labelling using a variant of *uncertainty sampling*, by which data are selected according to how confident a classifier is in individual predictions (see, for example, [17]).

However, Grimmett *et al.* [7] show that several of the classification frameworks commonly used in robotics are unrealistically overconfident in their assessment of class membership. To characterise this attribute, the authors introduce the notion of the *introspective capacity* of a classification framework: the ability to estimate a classification confidence which realistically reflects how qualified the classifier is to make a particular class decision in each individual test instance. In this paper we show that *introspective classification* harbours significant benefits for active learning as compared to more traditional, non-introspective approaches. In particular, our contributions are

- the application of an active learning framework to semantic mapping in robotics,
- the application of the notion of introspection to the Informative Vector Machine (IVM) [10] as an efficient extension to [7],
- the application of the IVM specifically to achieve *introspective* active learning, which is demonstrated to lead to more effective information extraction over more traditional approaches, and
- the introduction of a principled mechanism for the IVM to *forget* less important data to provide for scalable, life-long active learning on a mobile robot.

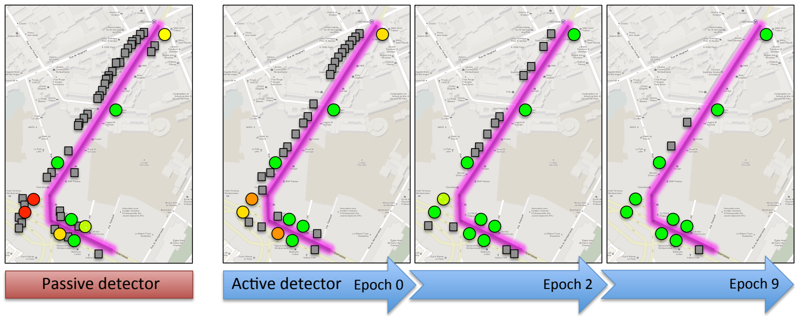

Fig. 1: Active learning in a semantic mapping context. This figure shows semantic maps indicating the positions of traffic lights along a street in Paris. Circles denote the locations of ground-truth traffic lights. The shading encodes the correctness of the classification output as provided by a probabilistic classifier: red denotes a recall of 0 (no detections), and green denotes a recall of 1 for that particular traffic light (all views of that object correctly detected). False positives are shown as grey squares. From left to right, we first see a typical passive detector, followed by our active-learning framework at epochs 0, 2, and 9 respectively. Note that in the active learning setting the shading of the circles progresses from red to green as a greater proportion of traffic lights are correctly detected with increasing confidence. Similarly the number of false positives reduces dramatically. By epoch 2 the active learning framework already outperforms the passive detector. In this paper we show that our formulation of an *introspective* active learning approach provides for more efficient information extraction – and thus a higher learning rate – over conventional active learning approaches. (This figure is best viewed in colour.)

The work presented here first appeared as a workshop paper by the same authors [21]. However, here we offer a more detailed treatment as well as the following significant extensions:

- the introspective capacity of the IVM is established, including the effects of varying the sparsity factor,
- qualitative results are included of when the IVM is confident (correctly and incorrectly) in its classifications, and
- timing information is provided regarding the training of an IVM.

We apply our framework to the detection of traffic lights in a real, third-party vision dataset and demonstrate iteratively improved semantic mapping, which makes efficient use of available label information. A typical qualitative example of our system output is shown in Fig. 1.

2 Related Works

Active learning is an established and vibrant field of research spanning a significant number of application domains. Consequently, a variety of methods have been proposed for selecting informative measurements for labelling and/or for incrementally training a learning algorithm. For example, Freund *et al.* [6] propose disagreement among a committee of classifiers as a criterion for active data selection. McCullum and Nigam [12] apply this to text classification using high label inconsistency as a query criterion coupled with expectation maximisation (EM) for online learning. More recently, Joshi *et al.* [8] address multi-class image classification using SVMs and propose criteria based on entropy and best-versus-second-best (BvSB) measures based on the hyperplane-margin for determining uncertain points. Tong and Koller [19] pick unlabelled data for query based on minimising the version space within a margin-based SVM formulation. Kapoor *et al.* [9] propose an active learning system for object categorization using a GP classifier where data points possessing large uncertainty (using posterior mean and variance) are queried for labels and used to improve classification.

Within the robotics community, active learning and directed information acquisition has received attention in recognition, planning and mapping tasks. For example, Dima *et al.* [4] present unlabelled data filtering for outdoor terrain classification tasks with the aim of reducing the amount of training data to be human-labelled. The approach relies on kernel density estimation over unlabelled data and estimating a "surprise" score for image patches, hence only querying the least likely samples given the density estimate for human labelling. In [14] the authors present a learning approach for continually improving place recognition perfomance by actively learning an appearance model of a robot's operating environment. The method uses probabilistic topic models and a measure of perplexity to identify least explained images which further drives retrieval of thematically linked samples leading to an improved workspace representation. Recent work by Tellex *et al.* [18] explores active information gathering for human-robot dialog. The authors formulate an information-theoretic strategy for asking clarifying questions to disambiguate the robot's belief over the mapping between phrases and aspects of the workspace.

While, to the best of our knowledge, this is the first work in robotics applying active learning to a semantic mapping task, our work is also set apart significantly from prior art in active learning in that we introduce and demonstrate the benefits of efficient *introspective* active learning. In this respect, the work most closely related to ours is that of [9] above, in which an inherently introspective classifier is used but its use is not motivated by its introspective qualities.

3 Introspective Classification

The introspective capacity of a classifier characterises its ability to *realistically* estimate the uncertainty in its predictions. Grimmett *et al.* [7] define the introspective capacity as a classifier's ability to moderate its output by an appropriate measure as to how 'qualified' it is to make a call given its own prior experience, usually in the form of training data. The intuition is that test data, which are in some form 'similar' to that seen in training, are classified with higher certainty than data which are more dissimilar. This points towards non-parametric approaches potentially being more introspective than parametric ones, as all the training data are available for inference in the former, whereas inference in the latter is based on parametric models learned from the data. Grimmett *et al.* [7] investigated several commonly used classification frameworks providing probabilistic output and found that a Gaussian Process classifier (GPC) [16] indeed is significantly more introspective than, for example, the more commonly used Support Vector Machine (see, for example, [1]) with a probabilistic calibration (such as, for example, provided by Platt *et al.* [15]).

In [7], this quality is attributed to a GPC's Bayesian treatment of predictive variance. Consider a set of training data $\{X, \mathbf{y}\}$, where $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_{|X|}\}$ denotes the set of feature vectors and $\mathbf{y}$ denotes the set of corresponding class labels. Probabilistic predictions for a test point, $\mathbf{x}_*$, are obtained in two steps. First, the distribution over the latent variable corresponding to the test input is obtained by

$$p(f_* \mid X, \mathbf{y}, \mathbf{x}_*) = \int p(f_* \mid X, \mathbf{x}_*, f) p(f \mid X, \mathbf{y}) df, \tag{1}$$

where $p(f \mid X, \mathbf{y})$ is the posterior distribution over latent variables. This is followed by applying a sigmoid function $\sigma(\cdot)$, which in our implementation is the cumulative Gaussian, and *marginalising* over the latent f_* to yield the class likelihood $p(y_* \mid X, \mathbf{y}, \mathbf{x}_*)$ as

$$p(y_* \mid X, \mathbf{y}, \mathbf{x}_*) = \int \sigma(f_*) p(f_* \mid X, \mathbf{y}, \mathbf{x}_*) df_*. \tag{2}$$

It is this *marginalisation* over all models induced by the training set, as opposed to relying on a single *minimisation*-based estimate, which accounts for a more accurate estimate of the inherent uncertainty in class distribution, and therefore endows GP classification with a high introspective capacity.

3.1 Efficiency by Sparsification

A key drawback of a GPC is its significant computational demand in terms of memory and run time. This is due to the fact that the GPC maintains a mean $\boldsymbol{\mu}$, as well as a covariance matrix Σ, which is computed from a kernel function and has size $|\mathbf{y}|^2$. A number of sparsification methods have been proposed in order to mitigate this computational burden. For efficiency, in this work we adopt one such sparsification method: the Informative Vector Machine (IVM) [10]. The main idea of this algorithm is to only use a subset of the training points denoted the *active set*, $\mathcal{I}$, from which an approximation $q(f \mid X, \mathbf{y}) = \mathcal{N}(f \mid \boldsymbol{\mu}, \Sigma)$ of the posterior distribution $p(f \mid X, \mathbf{y})$ is computed. The IVM algorithm computes $\boldsymbol{\mu}$ and Σ incrementally, and at every iteration j selects the training point $(\mathbf{x}_k, y_k)$ which maximizes the entropy difference ΔH_{jk} between q_{j-1} and q_j for inclusion into the active set. Because q is Gaussian, ΔH_{jk} can be computed by

$$\Delta H_{jk} = -\frac{1}{2} \log|\Sigma_{jk}| + \frac{1}{2} \log|\Sigma_{j-1}|. \tag{3}$$

The details of the implementation can be found in Lawrence *et al.* [11]. The algorithm stops when the active set has reached a desired size. In our implementation, we choose this size to be a fixed fraction γ of the training set q.

To find the kernel hyper-parameters $\boldsymbol{\theta}$ of an IVM, two steps are iterated a given number of times: the estimation of $\mathcal{I}$ given $\boldsymbol{\theta}$, and minimising the marginal likelihood $q(\mathbf{y} \mid X)$ given $\mathcal{I}$. Although there are no convergence guarantees, in practice already a small number of iterations are sufficient to find good kernel hyper-parameters.

Importantly for our work, since inference with the IVM is similar to that with a GPC, the IVM retains the model averaging described in Eq. (2). We argue therefore, that the IVM provides a significant and well-established improvement in processing speed over a GPC while maintaining its introspective properties (see Sec. 5 and 5.4 for details).

4 Scalable Active Learning: Drive, Ask, Improve

The power of an active learning framework lies in its ability to select a suitable training set in an application-oriented way. It thus inherently allows the system to adapt naturally to the non-stationarity of the data often encountered in long-term robotics applications. The active learning framework considered here is a supervised learning process by which a human operator provides class labels for machine-selected test data, which are then fed back into classifier training to improve the classification result of the next round. We examine performance over successive *epochs*, which each consist of (re-)training, classification, and user-feedback. The implementation of a scalable active learning framework requires two problems to be addressed: firstly, a subset of test data has to be selected for re-training such that classification performance increases in the next epoch. Secondly, measures have to be taken that guarantee that the training set is bounded in size, since otherwise the algorithm will sooner or later exhaust the resources of a finite-memory, real robotic system. We compare this active learning approach with a more conventional "passive" alternative, that is, training a classifier once without any subsequent human-feedback improvement.

We now outline the specific active learning algorithm employed in this work, before providing details of both our data selection strategy and our approach to forgetting (bounding the training set size).

4.1 The Active Learning Algorithm

Algorithm 1 describes our active learning framework which, for reasons given in Sec. 3, uses an IVM as the underlying classifier. It requires five different input parameters: the initial hyper-parameters $\boldsymbol{\theta}_0$ used for training the IVM, the fraction γ of training points that are used for sparsification, the batch size b, the normalised entropy (NE) threshold ϑ that a test point needs to exceed to be considered for re-training, and the maximum number of questions r that the algorithm may ask. The last is intended to minimise nuisance to a human operator due to being asked too many questions. The sub-routines in the algorithm are explained as follows.

Algorithm 1: Active Learning with an IVM

Data: training data $\mathcal{D} = (X, \mathbf{y})$, stream of test data X^*
Input: initial kernel parameters $\boldsymbol{\theta}_0$, batch size b, active set size fraction γ, minimal retraining score ϑ, maximum number of questions r
Output: stream of output labels $\mathbf{y}^*$

```
i ← 0
while X* ≠ ∅ do
    (θ_{i+1}, I_{i+1}) ← TrainIVM(X, y, γ, θ_0)
    move next b test points from X* into X*_i
    P ← ∅
    forall the x* ∈ X*_i do
        z ← IVMPrediction(I_{i+1}, θ_{i+1}, x*)
        s ← ComputeRetrainingScore(z)
        if s > ϑ then P ← P ∪ {(x*, s)}
    sort P by decreasing values of s
    D+ ← ∅
    for j ← 1 to MIN(r, |P|) do
        (x+_j, s_j) ← element j of P
        y+_j ← AskLabelFromUser (x+_j)
        D+ ← D+ ∪ (x+_j, y+_j)
    D ← D ∪ D+,   i ← i + 1
```

`TrainIVM` uses the current training set, the active set fraction γ, and the initial kernel parameters to find optimal kernel parameters $\boldsymbol{\theta}_{i+1}$ and an active set $\mathcal{I}_{i+1}$ as described in Sec. 3.1. Throughout this work we employ a squared exponential kernel (which is the same as the Radial Basis Function kernel) with additive white noise:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \sigma_f^2 e^{-\frac{(\mathbf{x}_i - \mathbf{x}_j)^2}{2l^2}} + \sigma_n^2 \delta_{ij}, \tag{4}$$

where δ_{ij} is the Kronecker delta, and $\boldsymbol{\theta} = \{\sigma_f^2, l, \sigma_n^2\}$ are the signal variance, the length scale, and the noise variance.

`IVMPrediction` returns an estimate of the probability z that the next test datum $\mathbf{x}^*$ has a particular class label, as given in Eq. (2). Based on this probability, the normalised entropy measure is then computed. The top ranked r test data exceeding the retraining threshold ϑ are labelled by the user and added to the training set for the next epoch.

4.2 Data Selection Strategy: What Questions to Ask?

The key element of an active learning algorithm is the strategy by which a new test point $\mathbf{x}^*$ is considered for re-training. In Algorithm 1, this is done in the subroutine `ComputeRetrainingScore`. An intuitive and well-explored indicator of which data might be suitable for inclusion is the classification *uncertainty* associated with $\mathbf{x}^*$. To characterise the uncertainty of the classification from the given class prediction $z = p(y_* \mid X, \mathbf{y}, \mathbf{x}_*)$, we adopt the measure of *normalised entropy* $H(z)$, such that for the binary case,

$$H(z) = -z \cdot \log_2(z) - (1-z) \cdot \log_2(1-z), \tag{5}$$

where $H(z) \in [0,1]$, with high values representing high uncertainty.

This, indeed is central to our work. While, in principle, any classification framework which provides a distribution over class labels as output can be used in our active learning framework, intuitively we expect those with more realistic estimates of these probabilities to be more effective for active learning. Thus, we expect more introspective classifiers to perform better in the sense that they will ask more informative questions, leading to a higher learning rate. In Sec. 5, we will show that this is indeed the case when comparing the proposed framework based on an IVM with one based on a more commonly used, probabilistically calibrated SVM.

4.3 Forgetting Uninformative Data to Bound Memory Use

The main problem with the active learning framework as we presented it so far is that in theory the training set can grow indefinitely, because there are no guarantees that the algorithm will stop asking new questions. This makes the algorithm less flexible, especially if the input data can not be guaranteed to be within certain locality bounds, for example in a life-long learning application. Therefore, and for run time efficiency, we bound the size of the training set by removing points from it when it exceeds a given target size n_t. To decide which points to remove, we leverage the information-theoretic instruments that the IVM already provides. After each training round, we keep the entropy differences given in Eq. (3) for all training points and sort them in increasing order. Those training data which correspond to the first $n_i - n_t$ values, where n_i is the current training set size, are then removed before training in the next epoch. Intuitively, this method discards the data that were least informative during the last training round. One caveat with this method is that it assumes independence between the training data, which is not generally given. For example, two data may both have small individual ΔH values, but when removing both of them the entropy could change significantly. In this work we acknowledge but do not explore this phenomenon. Instead, we note that in our experiments we did not observe a deterioration in classification performance when we applied our method for forgetting.

5 Experimental Results

In this section we investigate the performance of our introspective active learning approach in terms of learning rate, data selection strategy, classification performance and tractability. We compare and contrast our approach with one based on the much more commonly used SVM classifier (calibrated to provide probabilistic output). The task we set both learners is to detect traffic lights in a third-party image dataset. Specifically, we use the publicly available Traffic Lights Recognition (TLR) data set [13], which comprises 11,179 colour images taken at 25 Hz from a car driven through central Paris at speeds under 31 mph. It has ground-truth labels for traffic light positions and subtype labels 'green', 'orange', 'red', 'ambiguous' (though here we are only concerned with the detection of traffic lights, irrespective of their state). As recommended by the authors of the dataset, we disregard labels of type 'ambiguous' and exclude sections where the vehicle was stationary for long periods of time. We use data from the first 5,800 frames for training and the remainder for testing. We compute a template-based feature set inspired by Torralba *et al.* [20] which has a successful track record in the detection of traffic lights [7]. Each training or test window is represented by a feature vector of length 200.

When training the IVM we used an active set fraction γ of 0.2, which means that informative points will be added to the active set until its size is 20% of the training set size. We use a Squared Exponential (SE) with white noise kernel. Training such a classifier takes approximately 1.5 seconds on a single 3.4GHz core.

The SVMs used here are trained using *libsvm* [2], and use the isotropic Radial Basis Function (RBF) kernel, which is equivalent to the SE kernel used by the IVM. They are trained using 10-fold cross-validation on top of a grid-search over the parameters C (the penalty parameter for the error term) and γ (the inverse of the length scale for the isotropic RBF kernel), both in the space 2^k where $k = \{-7, -6, \dots, +4\}$. Training takes approximately 10 minutes.

5.1 Does Introspection Improve Active Learning?

One of the central claims of this paper is that the use of an introspective classifier will lead to more informative questions being asked of the human expert. In order to test this claim we perform a cross-over experiment (see Fig. 2) which starts with both an IVM and an SVM are initially trained on the same data, 200 traffic lights (positive) and 200 background patches (negative). Then, 1,000 new data (with a class fraction of 1:1, the same as during training) are shown to both classifiers for testing. Each chooses up to 50 data points (providing their normalised entropies are over a threshold empirically set to be $\vartheta = 0.97$) to add to their own training set for the next round, resulting in *two* new and different training sets: the 'IVM set' and the 'SVM set'. A new IVM and SVM are now trained on *each* of the two new sets and evaluated on a further 1,000 new data points. This process thus gives rise to four classifiers: two IVMs trained on data selected by an IVM and a SVM respectively, and two equivalent SVMs. We compute precision and recall for all four classifiers.

The results after 100 repetitions of this experiment are shown in Fig. 3. As expected, both the IVM and the SVM perform better when trained on the dataset chosen by the initial IVM, suggesting that the questions asked by the IVM tend to be more informative. An unpaired t-test shows this result to be significant to a level of over 95%.

The overall effect of introspection in an active learning setting seems to be an increased learning rate, a claim which we support with the following active learning experiment, performed over 11 epochs. As described in Sec. 4, our active learning algorithm is retrained after having seen a batch of test points, as opposed to running the training algorithm after every new datum encountered. Every epoch consists of a training phase, a classification phase, and a feedback phase. At the very start of epoch 0, the classifiers are trained on 50 positive (traffic light) windows and 500 negative (background) windows extracted at random from the training frames. We choose this class fraction disparity to reflect the fact that in real data sets, negative examples are much more prevalent than positive examples. During each classification phase, the classifiers are then tested on a batch of 1,000 windows extracted from the test frames. The class fraction for these test windows is 1:10, the same as for training. Next, the 50 points with the highest normalised entropy (providing they are over $\vartheta = 0.97$) are added to the training set, ready for retraining at the start of the next epoch. Note that each classifier (IVM and SVM) makes its own choices regarding which points to add for the next epoch.

The results are shown in Fig. 4, where the IVM learner starts off with a worse f_1 measure at epoch 0 but has already exceeded the SVM by epoch 2, and is better (with non-overlapping 95% confidence bounds) in the steady state from then onwards. The gradient of the plot in Fig. 4 is shown in Fig. 5, and shows that the rate of increase of f_1 measure (the learning rate) for the IVM is better than that of the SVM over the first few epochs, and then always at least as good subsequently.

Fig. 4 further serves to justify empirically our choice of normalised entropy as a valid criterion for data selection, by comparing it to randomly selecting new training data. Intuitively, both methods should improve classification by virtue of the fact that they increase the training set size. However, the results indicate that for both the IVM and the SVM, using normalised entropy leads to more rapidly improving classification performance.

5.2 Does Forgetting Affect the Performance?

Our work aims to contribute an introspective active learning algorithm that is efficient in terms of computational effort and scalable with respect to its memory requirements. In this section we investigate the efficacy of the mechanism we have put in place to provide this tractability: forgetting. In experiments thus far, new training data were added in each epoch. The IVM active set size is a fixed proportion of the training set size, which has the benefit of increasing classification performance, but is detrimental to processing time. In the context of a life-long-learner, this is not a scalable solution.

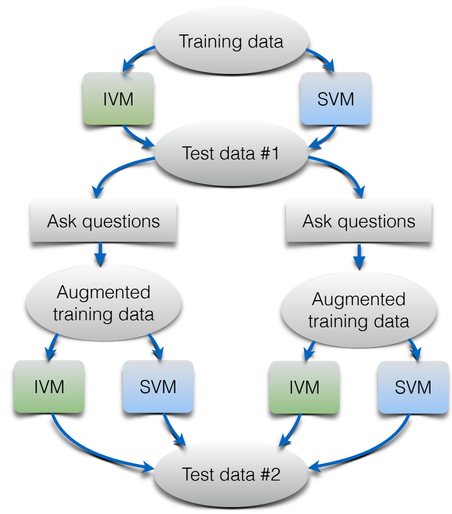

Fig. 2: Here we show the procedure for the cross-over experiment, designed to test whether one classifier chooses points which do not only benefit itself in the next round, but are consistently more useful for the other type of classifier as well. We compare an IVM and an SVM, and choose the test points with highest normalised entropy to be labelled to augment the original training set.

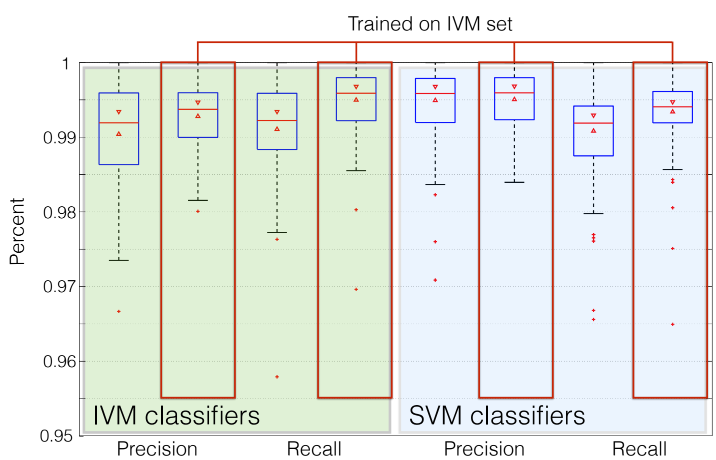

Fig. 3: Data selected by the IVM lead to an improved learning rate in terms of precision and recall for both an IVM and SVM over those selected by the SVM. Results are shown for 100 experimental runs, and increases are significant to the 95% level. See text and Fig. 2 for more details.

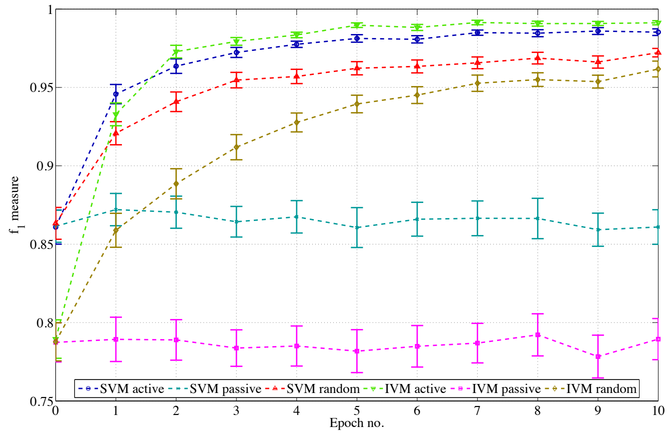

Fig. 4: Classification performance for both IVM and SVM variants as indicated by the f_1-measure after each epoch. Measurements are averaged over 100 runs. Error bars indicate the 95% confidence region of the mean. The IVM using a normalised entropy-based data selection strategy (IVM-active) consistently outperforms all other active learning variants in terms of learning rate and final classification performance.

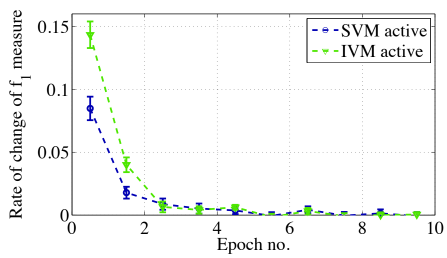

Fig. 5: The gradient of the f_1 measure of the active learners from Fig. 4

We therefore elect to cap the size of the training set at $n_t = 550$ data, which makes the computational effort constant. This *'IVM with forgetting'* learner can add new data, but only by simultaneously discarding enough data to reduce the training set size to the target size n_t. Fig. 6 (left) shows the training set size for the normal IVM with unbounded training set,and an IVM with forgetting, capped at 550 data (the initial training amount). Fig. 6 (right) shows the corresponding classification performance as characterised by the f_1 measure. It indicates that in this scenario, the IVM with forgetting mechanism has the same performance as the unbounded IVM. We note that this is likely to be dataset dependent.

5.3 *What Does the Active Learner Ask?*

In Fig. 7 we show the 27 most certain and 27 least certain test cases for an IVM at epochs 0, 3, and 10, and whether they were correctly classified or not. Firstly, it

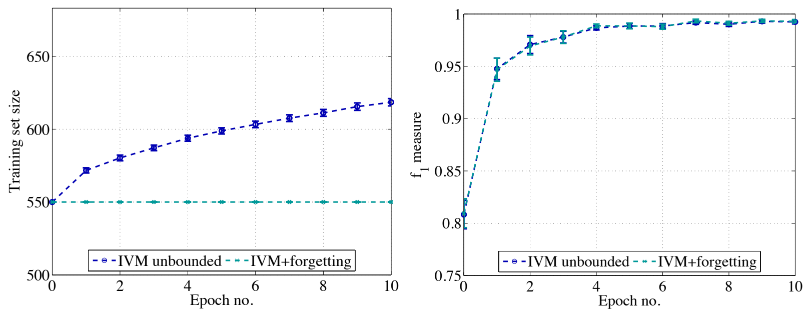

Fig. 6: *Forgetting* results in commensurate classification performance while successfully bounding the active set size of the classifier. Each datum represents the mean (and associated 95% confidence interval) over 100 experimental runs. **Left:** The evolution of the training set size. The IVM+forgetting learner has a target training set size n_t = 550, the initial training set size. **Right:** Classification performance with and without *forgetting*. For corresponding SVM results, see Fig. 5.

is reassuring to confirm that the certain classifications are always correct. At epoch 0 we see that the confident classifications are all of the background class, almost entirely of fairly uniformly textured surfaces like tarmac, and that the unconfident classifications are all regarding traffic lights. As the learners gather more data, the traffic lights which at epoch 0 were uncertain, are now very confident at epoch 3. At epoch 10, the uncertain group are more balanced in terms of traffic lights and background, and we see that although there is a little more variation in terms of the confident patches, they are very similar to the confidence classifications at epoch 3. This is consistent with the learning algorithm having reached an equilibrium after epoch 3 in Fig. 4.

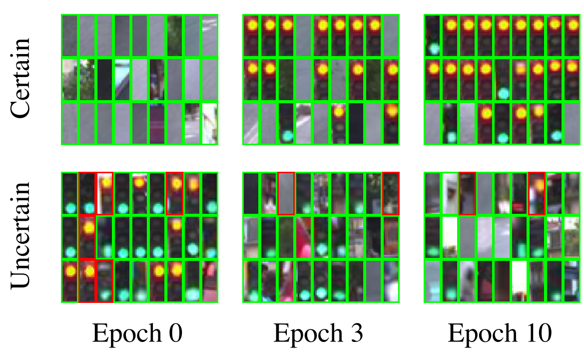

Fig. 7: The 27 most certain and 27 least uncertain test classifications of an IVM at epochs 0, 3, and 10 during the active learning experiment. A green border indicates a correct classification, and a red border indicates an incorrect classification.

5.4 The Effects of Sparsity

In [7] we showed that the GPC is more introspective than other more commonly used classification frameworks. In this paper we have argued the necessity of using a sparse formulation for the sake of computational complexity, however, it is necessary to ensure that the IVM is introspective in its own right. The useful characteristic of an introspective classifier is that it tends to be confident when it is making true predictions, and uncertain when it may be making false predictions. In addition, we would like to see whether the introspective quality changes with the active set size; intuitively, a truly introspective classifier will be more confident if it is exposed to more data, and vice versa.

Similarly to the approach in [7] we have plotted the cumulative true and false classifications against uncertainty in Fig. 8 for a single round of training and testing. In the legend, "IVM $\gamma = 0.4$" indicates an IVM with an active set fraction of 0.4, such that the active set contains 40% of the training set. These particular IVMs have been trained on 550 data and tested on 11000, with the ratio 1:10 positive:negative. There are several things to notice from the graph. Firstly, we can see that by looking at the curves for the IVMs with $\gamma = \{0.2, 0.4, 0.6, 0.8, 1.0\}$, indeed as we would hope, having a larger active set results in a more confident classifier; however it is interesting to see that there are diminishing returns: very little confidence is gained between an active set fraction of 0.6 and 1.0. Secondly and most importantly, *the IVM is introspective*: the incorrect classifications occur with *high* uncertainty, whereas the majority of the correct classifications occur with *low* uncertainty. Thirdly, we would expect that as the level of sparsity decreases, we approach the behaviour of the GPC, which is indeed what happens; the full GPC is commensurate with the IVMs with $\gamma = \{0.6, 0.8, 1.0\}$.

6 Conclusion

The contributions of this paper are three-fold: firstly, the notion of introspective classification introduced earlier shows promise in the context of active learning, where a reliable estimate of the classification uncertainty is required. We do this by showing an improvement in both classification performance and learning rate over a non-introspective classifier (Sec. 5.1). Secondly, an efficient version of the Gaussian Process Classifier, namely the Informative Vector Machine is used, which makes the approach particularly useful for robotics applications with large amounts of data. We show visual examples of where it is confused and where it is confident (Sec. 5.3), and use it to create the first offline semantic mapping algorithm via active learning. Finally, we present an information-theoretic solution to the problem of increasing memory requirements by forgetting the least informative data, which maintains a high classification performance in our experiments, but more extensive experimentation is required to confirm the success of this approach for the wider scope of mobile robotics applications.

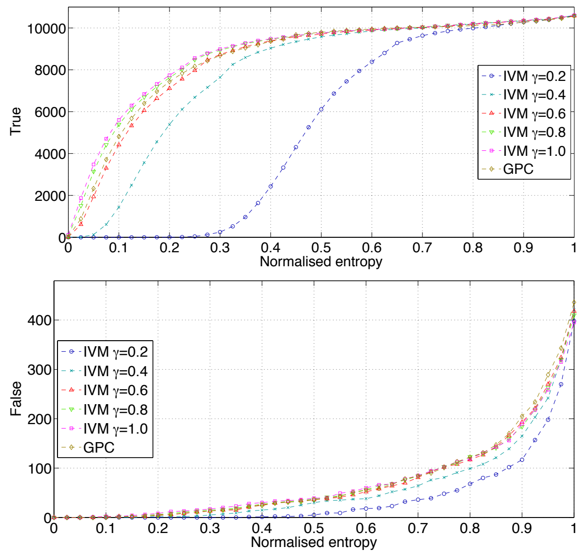

Fig. 8: The introspective capacity of the IVM. We show the number of true (**top**) and false (**bottom**) classifications (positive and negative classes together) which are made with a normalised entropy lower than a chosen value. For instance, if we were to threshold at NE = 0.5, we would have 6000 correct classifications with the IVM $\gamma = 0.2$ and < 10 incorrect classifications.

7 Acknowledgements

This work is funded under the European Community's Seventh Framework Programme (FP7/2007-2013) under Grant Agreement Number 269916 (V-CHARGE) and by the UK EPSRC Grant Number EP/J012017/1.

References

[1] C. J. C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

[2] C.C. Chang and C.J. Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[3] DARPA. Urban Challenge Route Network Definition File (RNDF) and Mission Data File (MDF) Formats, March 2007.

[4] C. Dima, M. Hebert, and A. Stentz. Enabling learning from large datasets: Applying active learning to mobile robotics. In *IEEE Intern. Conf. on Robotics and Automation (ICRA)*, volume 1, pages 108–114. IEEE, 2004.

[5] N. Fairfield and C. Urmson. Traffic light mapping and detection. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.

[6] Y. Freund, H.S. Seung, E. Shamir, and N. Tishby. Selective sampling using the query by committee algorithm. *Machine learning*, 28(2):133–168, 1997.
[7] H. Grimmett, R. Paul, R. Triebel, and I. Posner. Knowing when we don't know: Introspective classification for mission-critical decision making. In *Proc. IEEE Intern. Conf. on Robotics and Automation (ICRA)*, 2013.
[8] A.J. Joshi, F. Porikli, and N. Papanikolopoulos. Multi-class active learning for image classification. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2372–2379, 2009.
[9] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell. Gaussian processes for object categorization. *International Journal of Computer Vision*, 88(2):169–188, 2010.
[10] N. D. Lawrence, M. Seeger, and R. Herbrich. Fast sparse gaussian process methods: The informative vector machine. *Advances in neural information processing systems*, 15:609–616, 2002.
[11] N. D. Lawrence, J. C. Platt, and M. I. Jordan. Extensions of the informative vector machine. In *Proceedings of the First international conference on Deterministic and Statistical Methods in Machine Learning*, pages 56–87. Springer-Verlag, 2005.
[12] A. McCallum and K. Nigam. Employing em in pool-based active learning for text classification. In *Proceedings of ICML-98, 15th International Conference on Machine Learning*, pages 350–358, 1998.
[13] Robotics Centre of Mines ParisTech. Traffic lights recognition (TLR) data set.
[14] R. Paul and P. Newman. Self help: Seeking out perplexing images for ever improving navigation. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 445–451. IEEE, 2011.
[15] J. Platt. Probabilistic outputs for support vector machines and comparison to regularize likelihood methods. In A.J. Smola, P. Bartlett, B. Schoelkopf, and D. Schuurmans, editors, *Advances in Large Margin Classifiers*, pages 61–74, 2000.
[16] C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. MIT Press, 2006.
[17] B. Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2010.
[18] S. Tellex, P. Thaker, R. Deits, T. Kollar, and N. Roy. Toward information theoretic human-robot dialog. In *Robotics: Science and Systems*, 2012.
[19] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. *The Journal of Machine Learning Research*, 2: 45–66, 2002.
[20] A. Torralba, K. P. Murphy, and W. T. Freeman. Sharing visual features for multiclass and multiview object detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(5):854–869, 2007. ISSN 0162-8828.
[21] R. Triebel, H. Grimmett, R. Paul, and I. Posner. Introspective active learning for scalable semantic mapping. In *Workshop. Robotics Science and Systems (RSS)*, June 2013.
[22] C. Urmson et al. Autonomous driving in urban environments: Boss and the Urban Challenge. *J. of Field Robotics*, 25(8):425–466, 2008. ISSN 1556-4967.