Bayesian Viewpoint-Dependent Robust Classification under Model and Localization Uncertainty

Yuri Feldman and Vadim Indelman

Abstract— **We propose an algorithm for robust visual classification of an object of interest observed from multiple views using a black-box Bayesian classifier which provides a measure of uncertainty, in the presence of significant ambiguity and classifier noise, and of localization error. The fusion of classifier outputs takes into account viewpoint dependency and spatial correlation among observations, as well as pose uncertainty when these observations are taken and a measure of confidence provided by the classifier itself. Our experiments confirm an improvement in robustness over state-of-the-art.**

I. Introduction

Object detection and classification is a component of situational awareness important to many autonomous systems and key to many tasks, especially, but not only, involving direct interfacing to humans. The mobility of robotic systems is widely exploited to overcome classical limitations of static, one-point-of-view approach to image classification such as occlusions, class aliasing (due to classifier imperfections or objects that appear similar from certain viewpoints), imaging problems, false detections. It is done by accumulating classification evidence across multiple observations and viewpoints, including a recent surge in active methods for autonomous classification, where next viewpoints are automatically selected. Variations in object appearance are often directly addressed using offline-built class models for inference rather than raw classifier measurements. Especially in the active methods, such models are often themselves spatial and view-dependent (Fig. 1). As was shown by Teacy et al. [18] and Velez et al. [19] view-dependent models can allow for better fusion of classifier measurements by modelling correlations among similar viewpoints instead of the common but usually false assumption of independence of measurements.

Reliance on spatial models however introduces new problems, as robot localization is usually not precisely resolved, leading to errors when matching measurements against the model. This is aggravated in the presence of classifier measurements actually not complying to the model, as may happen for example when a classifier is deployed in an environment different in appearance from the one it was trained on, for example - in another country where objects semantically identical to the ones in the training set look differently. In the latter case, classifier output would often be arbitrary, rather than reflect the actual uncertainty, known as *model uncertainty* [7]. In the domain of Bayesian deep learning, methods exist to approximate the above as network posterior [4], [7], [12], for example using test-time dropout [6], which allows to (approximately) obtain it for virtually any deep learning-based classifier without change in model.

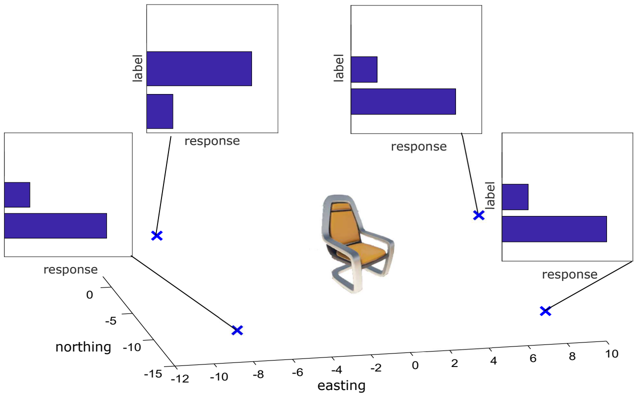

Fig. 1: GP model for class created from classifier output samples (entire classification vectors) at known relative locations around object. In our simulation scenarios, samples and their locations were specified manually for each of the classes.

Existing classification fusion methods do not address model uncertainty. Indeed, with few exceptions most current methods discard also the classification vector commonly output by the classifier, only using the most likely class (component with highest response) for belief update. Likewise, most methods ignore uncertainty in localization, assuming it perfectly known.

In this paper, we seek to (a) develop a method for fusing responses of a classifier which provides a model uncertainty measure, while (b) accounting for viewpoint-dependent variations in object appearance and correlations in classifier responses, and (c) accounting for localization uncertainty. We confirm in simulation that our method provides robustness with respect to the above sources of uncertainty compared to current methods.

II. Related Work

Visual classification fusion methods can be roughly split into methods directly using classifier scores [11], [14], and methods matching classifier measurements to a statistical model [1], [2], [13], [18], [19], or fusing them using a specially trained classifier [15]. The rationale for using a class model rather than individual classifier measurements lies in the variation in object appearance and background with viewpoint, which cannot always be correctly captured by the classifier, as well as situations where training data is not representative of the test data, e.g. where classifier was not or cannot be retrained specifically for the domain where it is deployed and therefore its responses cannot be directly relied upon. Among these, viewpoint-dependent object appearance (and hence, classifier response) are accounted for

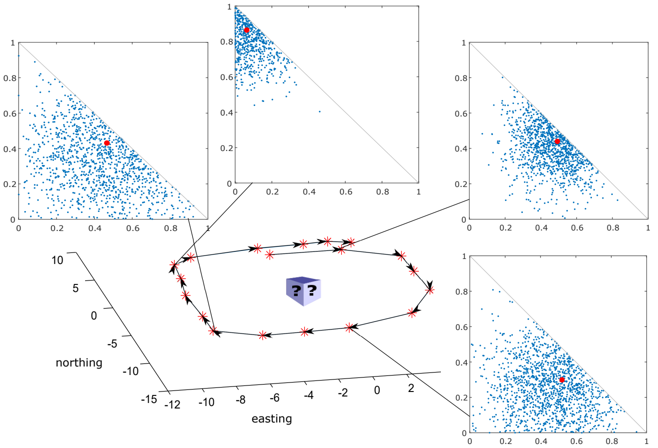

Fig. 2: Robot acquires observations along track in the vicinity of the object of interest. At each time step, classifier outputs a cloud of classification vectors reflecting the model uncertainty, unlike a single vector measurement (red dot) or a component thereof in other approaches.

by [13] as classifier noise, while other works [1]–[3], [18], [19] model spatial variation directly.

A common assumption to many classification methods is that of statistical independence of individual class measurements. This assumption is generally false, e.g. observations from the same or similar poses are likely to be extremely similar, resulting in similar classifier responses. Considering these observations as independent leads to an overly-confident posterior. Velez et al. [19] and Teacy et al. [18] deal with this by learning Gaussian Process regressors [16] to describe both per-class spatial variation in classifier responses and spatial statistical dependence.

Another often violated common assumption is that of known robot localization [1], [14], [18], [19]. This is specifically the weakness of methods modelling spatial variation of classifier responses, as localization error may introduce class aliasing when matching classifier responses against a spatial model. Becerra et al. [2], [3] address these aspects directly, however, while resorting to pose space discretization and considering the corresponding motion planning problem within a POMDP framework.

Recently, Gal et al. [5], [6] have shown that (Monte-Carlo, MC) dropout can be interpreted as an approximation to *model uncertainty* of a bayesian neural network classifier, expressed as posterior distribution of the output for Gaussian priors on weights. *Model uncertainty* quantifies the reliability of classifier responses for given input data [9], [10], complementing the classifier response vector (softmax output). While there are other ways of approximating network posterior [4], [12] and other reliability measures [17], MC dropout is both theoretically grounded and practical since it requires no change in architecture or special heavy computations which are not otherwise part of the model.

Building upon the approach of Teacy et al. [18], we develop a method for object classification from multiple views which is aware of classifier model uncertainty, robot localization uncertainty, and accounts for spatial correlation among views.

III. Notations and Problem Formulation

Consider a robot traversing an unknown environment, taking observations of different scenes. Robot motion between times t_k and t_{k+1} is initiated by a control input u_k, that may originate from a human user, or be determined by a motion planning algorithm. We denote the robot pose at time instant k by x_k, and by $X_{0:k} = \{x_0, \dots, x_k\}$ the sequence of poses up to that time. Let $\mathcal{H}_k = \{\mathcal{U}_{0:k-1}, \mathcal{Z}_{0:k}\}$ represent the history, comprising observations $\mathcal{Z}_{0:k} = \{z_0, \dots, z_k\}$ and controls $\mathcal{U}_{0:k-1} = \{u_0, \dots, u_{k-1}\}$ up until time k. We focus on the task of classification of a single object belonging to one of N_c known classes, denoted by indexes $\mathcal{C} = \{1, \dots, N_c\}$.

Our goal is to maintain the classification posterior, or *belief*, at time instant k:

$$b[c_k] \doteq \mathbb{P}(c \mid \mathcal{H}_k). \tag{1}$$

The classification posterior is the probability of the object in question to belong to class $c \in \mathcal{C}$, given all measurements and user controls up to time k. In calculating this posterior we want to take into account spatial correlation among measurements, model uncertainty, as well as uncertainty in the positions from which these measurements are taken (localization uncertainty).

A. *Classifier Model*

Commonly, the classifier output can be interpreted as a categorical distribution over classes (e.g. by applying softmax to its outputs). However, high responses may be unstable, specifically, when inputs are far from training data. We use the technique proposed by Gal and Gahrahmani [7] to obtain an approximation for the *model uncertainty* of the neural network classifier we are using. In short, for every classifier input we perform several forward passes applying random dropouts at each pass, to obtain a set of classification vectors, characterizing the uncertainty, yielding classifier output as shown in Fig. 2. We chose this method for its simplicity, although Myshkov and Julier [12] show that it may underestimate the model uncertainty in some cases. Formally, we assume that the robot has at its disposal an object classifier unit, which, given observation z_k (e.g., an image), calculates a set of outputs $\mathcal{S}_k \triangleq \{s_k\}$, where each output $s_k \in \mathbb{R}^{N_c \times 1}$ represents a categorical belief over the class of the observed object, i.e. $\sum_{i=1}^{N_c} s_k^{(i)} = 1$.

The set S_k can be interpreted as an approximation to the distribution

$$\mathbb{P}(s \mid z_k), \tag{2}$$

carrying information of the classifier's *model uncertainty* [9] for the given input z_k.

B. *Viewpoint-Dependent Class Model*

For the class likelihood we use a model similar to the one proposed by Teacy et al. [18]. For a single classifier measurement s (categorical vector) made from relative pose $x^{(rel)}$, the class likelihood is a probabilistic model

$$\mathbb{P}(s \mid c, x_k^{(rel)}), \tag{3}$$

where $c \in \mathcal{C}$ is the object class, and the k subscript denotes time index. Denoting object pose in global frame as o we can explicitly write

$$x_k^{(rel)} \doteq x_k \ominus o. \tag{4}$$

The dependence of the model in Eq. (3) on viewpoint naturally captures view-dependent variations in object appearance. Further, to incorporate the notion that similar views tend to yield similar classifier responses and in particular, are not independent, we consider the joint distribution

$$\mathbb{P}(\mathcal{S}_{0:k} \mid c, \mathcal{X}_{0:k}^{(rel)}), \tag{5}$$

characterizing the classification unit's outputs $\mathcal{S}_{0:k} \doteq \{S_0, \ldots, S_k\}$ when viewing an object of class c from a sequence of relative poses $\mathcal{X}_{0:k}^{(rel)} \doteq \{x_0^{(rel)}, \ldots, x_k^{(rel)}\}$. Similar to [18], [19], we represent this joint distribution with a Gaussian Process, learned using the classifier unit. Explicitly, we model training set classifier response when viewing object of class c from relative pose $x^{(rel)}$ as

$$s^{(i)} = f_{i|c}(x^{(rel)}) + \varepsilon, \tag{6}$$

where the i index denotes component i of classification vector s, $\varepsilon \sim N(0, \sigma_n^2)$ i.i.d. noise, and (dropping the (rel) superscript for clarity)

$$f_{i|c}(x) \sim \mathcal{GP}\left(\mu_{i|c}(x), k_{i|c}(x, x)\right), \tag{7}$$

where $\mu_{i|c}$ and $k_{i|c}$ are the mean and covariance functions defining the GP

$$\mu_{i|c}(x) = \mathbb{E}\{s^{(i)} \mid c, x\} \tag{8}$$

$$k_{i|c}(x, x') = \mathbb{E}\{(f_{i|c}(x) - \mu_{i|c}(x))(f_{i|c}(x') - \mu_{i|c}(x'))\} \tag{9}$$

We thus model the classification vector for each class c with independent, per-component GP's. Note also the Gaussian approximation of the distribution of the classification vector, which resides in the simplex (other representations exist, which however are not readily interpreted as a spatial model).

For the covariance we use the squared exponential function:

$$k_{i|c}(x, x') = \sigma_{i|c}^2 \exp(-\frac{1}{2}(x - x')^T L_{i|c}^{-1}(x - x')), \tag{10}$$

where $\sigma_{i|c}^2$ is the variance, and $L_{i|c}$ is the length scale matrix, determining the rate of the covariance decay with distance. These parameters can be learned from training data, however in our simulations they were set by hand.

Denote the training set for class c as $\{S_T^c, X_T^c\}$, with S_T^c classifier measurements, and X_T^c the corresponding poses, and denote (test-time) measurements as $S = \mathcal{S}_{0:k}$ and $X = \mathcal{X}_{0:k}^{(rel)}$. Further, the following equations Eqs. (11-14) all hold per vector-component (joined in Eq. (15)), i.e. for simplifying notation we drop the i index in $\mathcal{S}^{(i)}$, $\mathcal{S}_T^{(i)}$ and $k_{i|c}$.

We follow [16] and model the joint distribution of classifier measurements (per-component) for object of class c as

$$\mathbb{P}(S_T^c, S \mid c, X_T^c, X) = \\ N\left(0, \begin{bmatrix} K_c(X_T^c, X_T^c) + \sigma_n^2 I & K_c(X_T^c, X) \\ K_c(X, X_T^c) & K_c(X, X) \end{bmatrix}\right), \tag{11}$$

where K_c is the matrix produced by application of kernel k_c on all pairs of input vectors. We thus obtain the conditional distribution for classifier measurements of object of class c

$$\mathbb{P}(\mathcal{S}_{0:k} \mid c, X_T^c, S_T^c, \mathcal{X}_{0:k}^{(rel)}) = N(\mu, \Sigma), \tag{12}$$

with

$$\mu = K_c(X, X_T^c) \cdot H \cdot S \tag{13}$$

$$\Sigma = K_c(X, X) - K_c(X, X_T^c) \cdot H \cdot K_c(X_T^c, X), \tag{14}$$

and where $H \doteq \left(K_c(X_T^c, X_T^c) + \sigma_n^2 I\right)^{-1}$.

We finalize by combining the per-component models into a joint class likelihood as

$$\mathbb{P}(S \mid c, X_T^c, S_T^c, X) = \prod_i \mathbb{P}(S^{(i)} \mid c, X_T^c, S_T^{c,(i)}, X) \tag{15}$$

Note that this approach somewhat differs from [18], where inference from training data is done by offline learning of GP mean and covariance functions rather than using a joint distribution as in Eq. (11).

IV. Approach

To account for both localization and model uncertainty we rewrite Eq. (1) as marginalization over latent robot and object poses, and over classifier outputs. We start by marginalizing over robot pose history and object pose

$$b[c_k] = \mathbb{P}(c \mid \mathcal{H}_k) = \int_{\mathcal{X}_{0:k}, o} \mathbb{P}(c, \mathcal{X}_{0:k}, o \mid \mathcal{H}_k) \, d\mathcal{X}_{0:k} do, \tag{16}$$

which, using chain rule, can be written as

$$b[c_k] = \int_{\mathcal{X}_{0:k}, o} \underbrace{\mathbb{P}(c \mid \mathcal{X}_{0:k}, o, \mathcal{H}_k)}_{(a)} \underbrace{\mathbb{P}(\mathcal{X}_{0:k}, o \mid \mathcal{H}_k)}_{(b)} \, d\mathcal{X}_{0:k} do. \tag{17}$$

Term (a) above is the classification belief given relative poses $\mathcal{X}_{0:k}^{(rel)}$ which are calculated from $\mathcal{X}_{0:k}$ and o via Eq. (4). Term (b) represents the posterior over $\mathcal{X}_{0:k}$ and o given observations and controls thus far. As such, this term can be obtained from existing SLAM approaches. One can further rewrite the above equation as

$$b[c_k] = \underset{\mathcal{X}_{0:k}, o}{\mathbb{E}} \{\mathbb{P}(c \mid \mathcal{X}_{0:k}^{(rel)}, \mathcal{H}_k)\}, \tag{18}$$

where the expectation is taken with respect to the posterior $p\left(\mathcal{X}_{0:k}, o \mid \mathcal{H}_k\right)$ from term (b). In practice, in this work we assume that object orientation relative to the robot is known (leaving o with 3 degrees of freedom), and so this posterior can be computed using SLAM methods (see Section IV-B), which commonly model this posterior with a Gaussian distribution. We then use the obtained distribution to approximate the expectation in Eq. (18) using sampling.

In the following we detail the computation of the terms (a) and (b) of Eq. (17).

A. Classification Under Known Localization

In this section we develop the update of classification belief given known pose history, term (a) in Eq. (17), when receiving new measurements at time step k, while accounting for correlations with previous measurements and model uncertainty.

To simplify notation, we shall denote history of observations, controls and (known) relative poses as

$$H_k \doteq \mathcal{H}_k \cup \mathcal{X}_{0:k}^{(rel)} \equiv \{\mathcal{U}_{0:k-1}, \mathcal{Z}_{0:k}, \mathcal{X}_{0:k}^{(rel)}\}. \tag{19}$$

We start by marginalizing term (a) over model uncertainty in the classifier output at time k

$$\mathbb{P}(c \mid H_k) = \int_{s_k} \mathbb{P}(c \mid s_k, H_k) \cdot \mathbb{P}(s_k \mid H_k)\ ds_k. \tag{20}$$

Assuming s_k carries the class information from measurement z_k, and that $s_k \sim p(s_k \mid z_k)$ we can rewrite this as

$$\mathbb{P}(c \mid H_k) = \int_{s_k} \mathbb{P}(c \mid s_k, H_k \setminus \{z_k\}) \cdot \mathbb{P}(s_k \mid z_k)\, ds_k. \tag{21}$$

In our case, $\{s_k\}$ are samples from $p(s_k \mid z_k)$, so we can approximate the integral as

$$\mathbb{P}(c \mid H_k) \approx \frac{1}{n_k} \sum_{s_k \in \mathcal{S}_k} \mathbb{P}(c \mid s_k, H_k \setminus \{z_k\}). \tag{22}$$

To calculate the summand, we apply Bayes' law and then smoothing over class in the denominator

$$\mathbb{P}(c \mid H_k) = \sum_{s_k} \frac{\eta(s_k)}{n_k} \cdot \mathbb{P}(s_k \mid c, H_k \setminus \{z_k\}) \cdot \mathbb{P}(c \mid H_k \setminus \{z_k\}) \tag{23}$$

with

$$\eta(s_k) \doteq 1 / \sum_{c \in \mathcal{C}} \mathbb{P}(s_k \mid c, H_k \setminus \{z_k\}) \mathbb{P}(c \mid H_k \setminus \{z_k\}). \tag{24}$$

Note that the denominator in $\eta(s_k)$ is a sum of numerator (summand) terms in Eq. (23) for the different classes and can be computed efficiently (but cannot be discarded altogether due to the dependence on s_k). Further, note that

$$\mathbb{P}(c \mid H_k \setminus \{z_k\}) = \mathbb{P}(c \mid \mathcal{X}_{0:k}^{(rel)}, \mathcal{Z}_{0:k-1}) \tag{25}$$

$$= \mathbb{P}(c \mid \mathcal{X}_{0:k-1}^{(rel)}, \mathcal{Z}_{0:k-1}) = \mathbb{P}(c \mid H_{k-1}). \tag{26}$$

As $\mathbb{P}(c \mid H_{k-1})$ has been computed in the previous step, we are left to compute the class likelihood term $\mathbb{P}(s_k \mid c, H_k \setminus \{z_k\})$. This term involves past observations $\mathcal{Z}_{0:k-1}$ but not classifier outputs $\mathcal{S}_{0:k-1}$, which need to be introduced to account for spatial correlation with s_k using Eq. (5). Marginalizing over $\mathcal{S}_{0:k-1}$ (recall that in our notation $\mathcal{S}_{0:k-1} \cup \{s_k\} = \mathcal{S}_{0:k}$) yields

$$\mathbb{P}(s_k \mid c, H_k \setminus \{z_k\}) = \int_{\mathcal{S}_{0:k-1}} \mathbb{P}(\mathcal{S}_{0:k} \mid c, H_k \setminus \{z_k\})\ d\mathcal{S}_{0:k-1}$$

$$= \int_{\mathcal{S}_{0:k-1}} \mathbb{P}(s_k \mid c, \mathcal{S}_{0:k-1}, H_k \setminus \{z_k\}) \cdot \mathbb{P}(\mathcal{S}_{0:k-1} \mid c, H_k \setminus \{z_k\})\ d\mathcal{S}_{0:k-1}, \tag{27}$$

where we applied smoothing to separate between past classifier outputs $\mathcal{S}_{0:k-1}$ for which observations $\mathcal{Z}_{0:k-1}$ are given and the current output s_k. The first term in the product reduces to $\mathbb{P}(s_k \mid c, \mathcal{S}_{0:k-1}, \mathcal{X}_{0:k}^{(rel)})$, a conditioned form of the class model Eq. (12) (and thus Gaussian, which we treat explicitly later in Eq. (30) and on). This term represents the probability to obtain classification s_k when observing an object of class c from relative pose $x_k^{(rel)}$ given previous classification results and relative poses. The second term in Eq. (27) can be approximated using Eq. (2) for the individual observations z_i, i.e.

$$\mathbb{P}(\mathcal{S}_{0:k-1} \mid c, H_k \setminus \{z_k\}) = \mathbb{P}(\mathcal{S}_{0:k-1} \mid \mathcal{Z}_{0:k-1}) \approx \prod_{i=0}^{k-1} \mathbb{P}(s_i \mid z_i)$$

Note that class c and poses $\mathcal{X}_{0:k-1}^{(rel)}$, both members of H_k can be omitted since conditioning on observations determines classifier outputs up to uncertainty due to classifier intrinsics (model uncertainty). The approximation is in the last equality, since in general classifier outputs $s_0, \ldots, s_{k-1}$ are interdependent through classifier parameters. We can now rewrite $\mathbb{P}(s_k \mid c, H_k \setminus \{z_k\})$ from Eq. (27) as

$$\int_{\mathcal{S}_{0:k-1}} \mathbb{P}(s_k \mid c, \mathcal{S}_{0:k-1}, \mathcal{X}_{0:k}^{(rel)}) \cdot \prod_{i=0}^{k-1} \mathbb{P}(s_i \mid z_i)\ d\mathcal{S}_{0:k-1}. \tag{28}$$

Assuming classifier output Eq. (2) is Gaussian, we denote

$$\mathbb{P}(s_i \mid z_i) = N(\mu_{z_i}, \Sigma_{z_i}), \tag{29}$$

where μ_{z_i} and Σ_{z_i} are estimated from $\mathcal{S}_i$. Since class model is Gaussian, see Eq. (12), the first term in the integrand in Eq. (28) is a Gaussian that we denote as

$$\mathbb{P}(s_k \mid c, \mathcal{S}_{0:k-1}, \mathcal{X}_{0:k}^{(rel)}) = N(\mu_{k|0:k-1}, \Sigma_{k|0:k-1}) \tag{30}$$

where, utilizing standard Gaussian Process equations [16],

$$\mu_{k|0:k-1} = \mu_k + \Omega \cdot (\mathcal{S}_{0:k-1} - \mu_{0:k-1}) \tag{31}$$

$$\Sigma_{k|0:k-1} = K(x_k, x_k) - \Omega \cdot K(\mathcal{X}_{0:k-1}, x_k) \tag{32}$$

with $\Omega \doteq K(x_k, \mathcal{X}_{0:k-1}) K\left(\mathcal{X}_{0:k-1}, \mathcal{X}_{0:k-1}\right)^{-1}$.

Using these relations, the integrand from Eq. (28) is a Gaussian distribution over $\mathcal{S}_{0:k}$, that can be inferred as follows.

$$\mathbb{P}(s_k \mid c, \mathcal{S}_{0:k-1}, \mathcal{X}_{0:k}^{(rel)}) \cdot \prod_{i=0}^{k-1} \mathbb{P}(s_i \mid z_i) = \tag{33}$$

$$\eta \exp\left\{-\frac{1}{2}\left(\|s_k - \mu_{k|0:k-1}\|^2_{\Sigma_{k|0:k-1}} + \sum_{i=0}^{k-1} \|s_i - \mu_{z_i}\|^2_{\Sigma_{z_i}}\right)\right\},$$

where η only depends on $\mathcal{X}_{0:k}^{(rel)}$. Using Eq. (31) we can write

$$s_k - \mu_{k|0:k-1} = s_k - \mu_k - \Omega \cdot (\mathcal{S}_{0:k-1} - \mu_{0:k-1}) \tag{34}$$

$$= [-\Omega \quad I]\,(\mathcal{S}_{0:k} - \mu_{0:k}) \tag{35}$$

We have that the integrand Eq. (33) from Eq. (28) is proportional to a joint Gaussian distribution $N(\mu_J, \Sigma_J)$ with

$$\Sigma_J = \left(\Sigma_s^{-1} + \Sigma_z^{-1}\right)^{-1} \tag{36}$$

$$\mu_J = \Sigma_J^{-1} \cdot \left(\Sigma_s^{-1}\mu_s + \Sigma_z^{-1}\mu_z\right), \tag{37}$$

where

$$\mu_s = \begin{pmatrix} \mu_0 \\ \vdots \\ \mu_{k-1} \\ \mu_k \end{pmatrix} \quad \mu_z = \begin{pmatrix} \mu_{z_0} \\ \vdots \\ \mu_{z_{k-1}} \\ 0 \end{pmatrix}, \tag{38}$$

and

$$\Sigma_s^{-1} = [-\Omega \quad I]^T\, \Sigma_{k|0:k-1}^{-1}\, [-\Omega \quad I] \tag{39}$$

$$\Sigma_z^{-1} = \begin{pmatrix} \Sigma_{z_0}^{-1} & 0 & \cdots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & \Sigma_{z_{k-1}}^{-1} & 0 \\ 0 & \cdots & 0 & 0 \end{pmatrix} \tag{40}$$

Finally, the class likelihood from Eq. (27) is the marginal distribution of the above. Specifically, the integral is directly calculated by evaluation at s_k of a Gaussian pdf with the components corresponding to s_k from μ_J and Σ_J as mean and covariance.

So far, we have described how to update the class belief given known localization, term (a) of Eq. (17), upon arrival of new measurements. We now proceed to describe how the localization belief, term (b), is computed.

B. Localization Inference

In this work we assume that object *orientation* relative to the robot is known (perhaps, detected from observations $\mathcal{Z}$), and so o has three degrees of freedom (location). Hence, term (b) of Eq. (17) is essentially a SLAM problem with the robot pose history $\mathcal{X}_{0:k}$ and one landmark, the target object pose o, to be inferred. Specifically, we can express the target distribution as marginalization over all landmarks $\mathcal{L}$, except the object of interest

$$\mathbb{P}(\mathcal{X}_{0:k}, o \mid \mathcal{H}_k) = \int_{\mathcal{L}} \mathbb{P}(\mathcal{X}_{0:k}, o, \mathcal{L} \mid \mathcal{H}_k)\ d\mathcal{L}. \tag{41}$$

This can be computed using state of the art methods such as iSAM2 [8].

V. Results

We present experimental results for a MATLAB simulation, in which classifier measurements are generated using the GP model of the ground truth class, along a predetermined track. The class inference algorithm needs to fuse these measurements into a posterior over classes, essentially identifying which of the known GP models is the more likely origin of the measurements. We study robustness of our algorithm to model and localization uncertainty, and compare it to the state of the art.

A. Compared Approaches and Performance Metrics

We compare the results of three methods. One is our own, which we denote `Model with Uncertainty`, which takes into account spatial correlations, as well as uncertainty in pose and classifier model uncertainty. The second is `Model Based`, similar to the method described by Teacy et al. [18] but with GP defined as in Eq. (11) (and [16]), which takes into account spatial correlation, but not uncertainties. The third is `Simple Bayes`, which directly uses the classifier scores and assumes spatial independence between observations, as in e.g. Patten et al. [14].

We compare the methods above with relation to the following metrics: (i) probability of ground-truth class; (ii) mean squared detection error; and (iii) most likely-to-ground truth ratio.

The mean squared detection error (MSDE) is defined as

$$MSDE \doteq \frac{1}{N_c} \sum_{c' \in \mathcal{C}} \left(\delta_c(c') - \mathbb{P}(c' \mid \mathcal{H})\right)^2 \tag{42}$$

Here c is the ground truth class and $\delta_c(c')$ is 1 if $c = c'$ and 0 otherwise. This measure was also used in [18].

The most likely-to-ground truth ratio (MGR) is defined as

$$MGR \doteq \frac{\arg\max_{c'}\ \mathbb{P}(c' \mid \mathcal{H})}{\mathbb{P}(c \mid \mathcal{H})} \tag{43}$$

for ground truth class c. Roughly, this measure penalizes high confidence in the wrong class. In a way it "demands" ground truth class to be most (possibly, equally) likely.

We now proceed to detail the experiments and the results.

B. Simulation Experiments

Statistics (over realizations) for the three algorithms have been collected for several scenarios. In each scenario, GP models were created for three classes, by manually specifying classifier response for chosen relative locations around the origin (i.e. locations assumed to be in object-centered coordinates) in the 2D plane, see Fig. 1. Note that GP model for a class describes classifier responses for *all* classes, (see Eq. (15) and Section III-B). Another simplifying assumption is that object orientation is known, see also Section IV.

During simulation, the robot moves along a pre-specified trajectory and observes a single object from different viewpoints, see Fig. 2 for an example trajectory. At each time step the algorithm receives new classifier measurements and updated pose belief (simulating a SLAM solution). Classifier measurements are generated using the GP model of a "ground truth" class (the simulation of measurements is detailed in the next subsections), which needs to be inferred by the algorithm using the measurements.

We next present results on two scenarios highlighting our main contributions.

Algorithm 1 Procedure for simulating Classifier Outputs at step k

```
Input: $\mathcal{S}_{0:k-1}, \mathcal{X}^{(rel)}_{0:k}, \sigma^2_{max}, N_{samples}$
$s^{nominal} \sim \mathbb{P}(s \mid c, \mathcal{S}_{0:k-1}, \mathcal{X}^{(rel)}_{0:k})$    ▷ See Eq. (30)
$\sigma^2_u \sim Uni(0, \sigma^2_{max})$    ▷ Choose uncertainty level
$s^{noised} \sim N(s^{nominal}, \sigma^2_u I)$    ▷ Uncertain classification
$samples \leftarrow \emptyset$
for $N_{samples}$ times do    ▷ Simulating dropout
    $s \sim N(s^{noised}, \sigma^2_u I)$
    $samples \leftarrow samples \cup \{s\}$
end for
return $s^{nominal}, s^{noised}, samples$
```

1) Model Uncertainty Scenario: Model uncertainty expresses the reliability of the classifier output. High model uncertainty corresponds to situations where classifier input far from training data, often due to an unfamiliar scene, object or viewpoint pictured, causes output that may be arbitrary. We simulate this with two steps, performed at each time instant: first, nominal "true" measurement $s^{nominal}$ is generated from GP model of ground truth class. The level of model uncertainty σ^2_u is selected at each time step uniformly between 0 and σ^2_{max} (a parameter). It is then used as standard deviation of a Gaussian centered at the true measurement to generate a simulated noised measurement s^{noised}. The `Model Based` and `Simple Bayes` algorithms receive s^{noised} as classification measurement and are not aware of the uncertainty. Our method receives samples (simulating outputs of several forward passes applying dropouts) drawn from a Gaussian distribution centered at s^{noised} with standard deviation σ^2_u. Alg. 1 summarizes this process.

First scenario shows the effects of considerable model uncertainty, with no localization errors (perfect localization). Fig. 3 shows plots of GP model of ground truth class and simulated classifier measurements (s^{noised}) over robot track (left) and per-component as a function of time (right). Fig. 4 shows the statistics described above (probability assigned to ground truth class and Eqs. (42-43)) along with percentiles (over scenario realizations) as patches of varying saturation, with a 10% step: median is plotted darkest, the patch around it contains the runs between 40th and 60th percentile, the next one between 30th and 70th, etc. The area above and below the plots contains the top and bottom 10% of the runs respectively. Top row shows comparison of our method (blue) to `Model Based` (green), bottom - to `Simple Bayes` (in red).

An immediate observation in comparison to `Model Based` (first row) is that our percentiles are more concentrated, which means that method results are more stable. For example, in more than 20% of the runs (bottom lightest patch and below), probability of correct class (left column) for `Model Based` in time step 15 is less than 0.2 (compared to more than 0.33 for ours). Indeed, in more than 20% of the runs the MGR (middle column) for `Model Based` at iteration 15 is higher than 1, which means that a wrong (most likely) class was assigned probability more than twice higher than the correct one, i.e. wrong class was chosen with high confidence. The MSDE plot displays similar behavior. In the bottom row, drop of accuracy of `Simple Bayes` around time step 15 is the result of an "inverse" measurement in the model, meaning that from a certain angle, classifier response suggests a different class (see for example in Fig. 1). This illustrates well the difference from our method, which matches the entire sequence of measurements against a model, and thus can use also "inverse" measurements to classify correctly (on the downside, requiring a class model).

2) Localization Uncertainty Scenario: In methods making use of spatial class models, localization errors may cause classification aliasing when acquired measurements correspond to the model of a wrong class, because of the spatial shift in the query. To exemplify this, in this scenario, we introduced (a constant) bias in easting coordinate (the robot moves eastward in a straight line), causing aliasing between models (with no model uncertainty). Consider Fig. 5. The left plot as before shows GP mean of the correct class model (blue) and classifier output over robot track (red). It also shows the GP mean of the model of a wrong class (yellow). In the center plot, classifier outputs for label 2 (red) compared without localization bias against the corresponding GP component of the ground truth class model (blue) show a clear match. After introducing a bias of -8 units in easting (right plot) classifier responses (red) are matched against shifted spatial models, making the wrong class (yellow) a more likely match until around time step 16, after which the the blue line can be matched correctly in spite of the shift. The effects of this on performance are shown in Fig. 6. While our method, aware of the localization uncertainty (standard deviation) accumulates classification evidence gracefully, the `Model Based` method infers the wrong class with high confidence (as can be seen in the MGR plot, center) peaking at around time step 15, after which disambiguating measurements start to arrive. In the bottom row of the same figure, `Simple Bayes` method performs well (closely following the line from Fig. 5), since classifier measurements are stable and not ambiguous (the aliasing happens when trying to match against the different models).

VI. Conclusions

We described a method for classification from multiple views of an object of interest, by fusing classifier measurements which include a model uncertainty measure, and explicitly treating viewpoint-variability and spatial correlations of classifier outputs, as well as uncertainty in the localization. Our simulation experiments confirm increased robustness to the above sources of uncertainty as compared to current methods. In particular, our statistical analysis results suggest that in simulation cases where other compared methods inferred a wrong class with high confidence in a significant percentage of the runs due to noisy measurements of class or location, our method was aware of, and reported, high uncertainty, and was generally able to gracefully accumulate classification evidence.

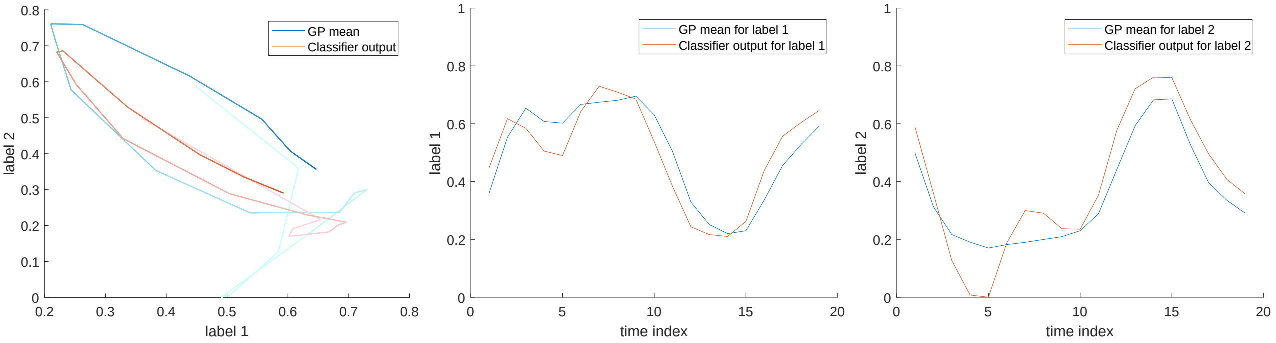

Fig. 3: Scenario with model uncertainty (no localization errors). Standard deviation for the noised observation chosen in $[0, 0.3]$. Left: mean of GP model for ground truth class and simulated (noised) classifier measurement over robot trajectory, plots of response for 1st label against response for 2nd label. More intense color corresponds to later time index. Center and right: first and second components over time indices, respectively.

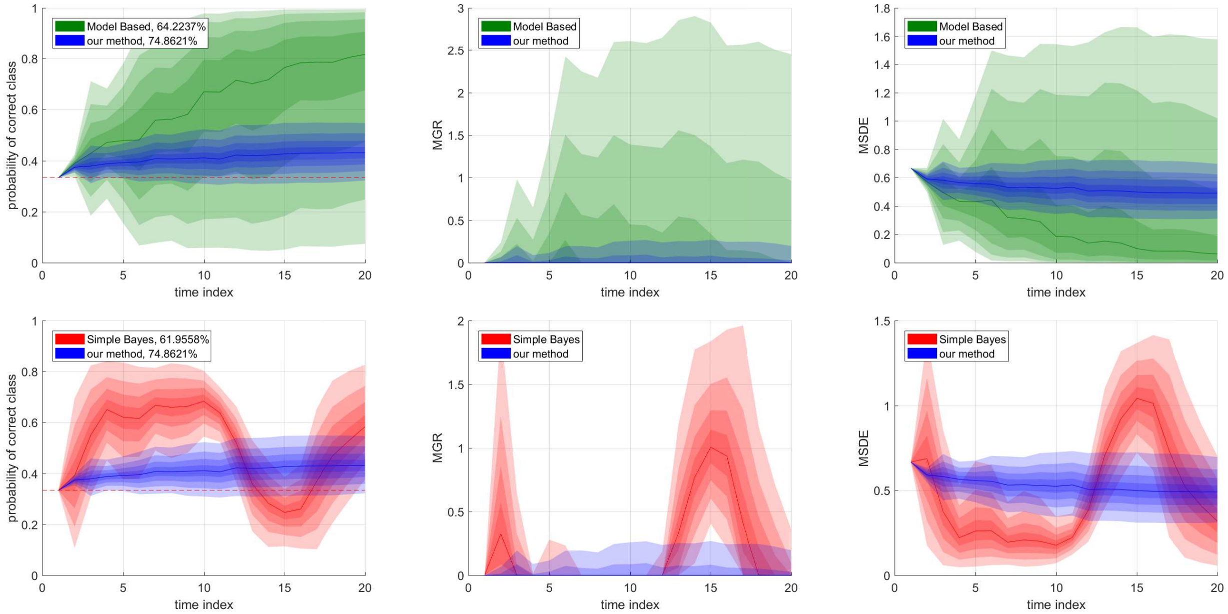

Fig. 4: Left column: probability of correct class, middle: MGR, right: MSDE. Legend in the leftmost column shows percentage of time steps where most likely class was the correct one. Color patches denote percentiles of the respective methods, one step in lightness denotes 10% percentile step and median is plotted darkest, so that the patch around the median comprises values between the 40th and the 60th percentile, the next darkest between the 30th and 70th and so on. Dashed line denotes the uninformative prior (probability of $1/3$ for label) Top row: comparison of our method to Model Based, bottom row: to Simple Bayes. While probability of correct class in our method rises slowly, it fluctuates significantly less over realizations, and the correct class is chosen more often. In both plots, Simple Bayes method performs poorly where an "inverse" measurement (see Section V-B.1) exists in the model, around time index 15.

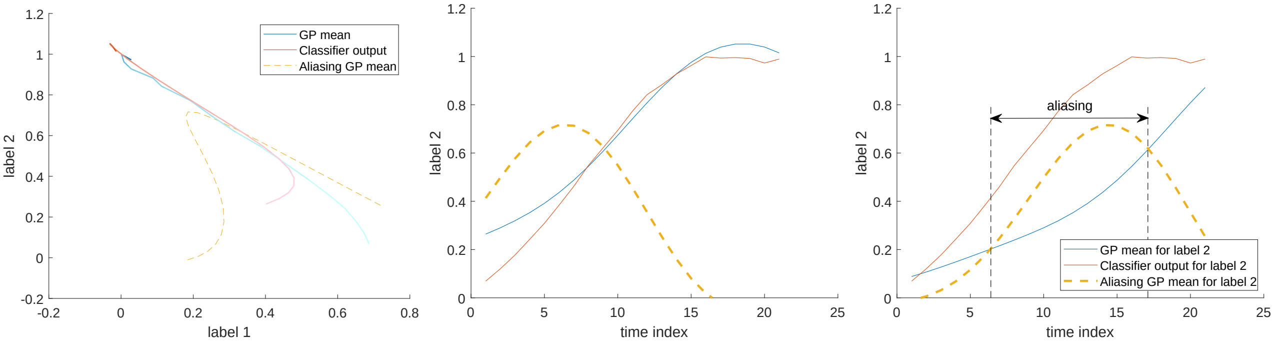

Fig. 5: Scenario with localization bias in the x axis. Time index corresponds to x coordinate (robot motion is a straight line, in the direction of the x axis). Left: as before, simulated classifier output generated from GP of ground truth class. Center: we concentrate on responses for class 2 (2nd component of classification vectors). Classifier output (red) matches GP of ground truth class (in blue) at true position. Right: bias in the x axis means that classifier output is effectively compared to a shifted model, better matching GP of a wrong class (yellow). This leads to classification errors unless accounting for localization uncertainty.

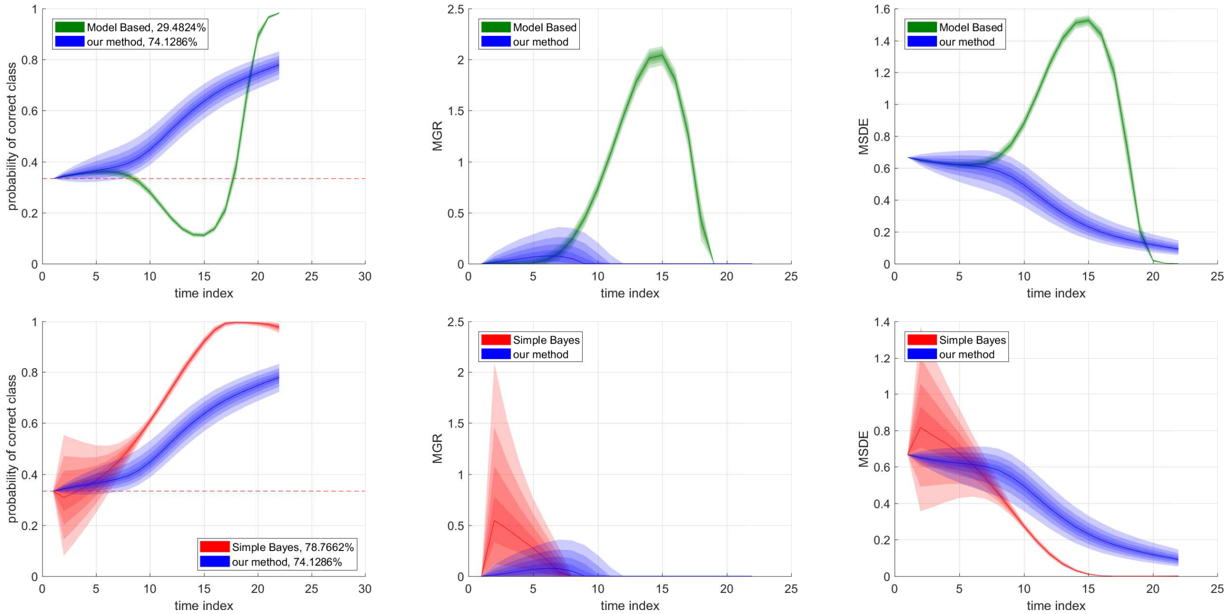

Fig. 6: Left column: probability of correct class, middle: MGR, right: MSDE. Localization bias of -8 units in the x axis causes severe aliasing in Model Based method resulting in a wrong class being inferred with high confidence. Our method is aware of localization uncertainty of standard deviation 16, and is able to recover. Simple Bayes method does not experience aliasing, as it uses the raw measurements directly, rather than matching them to a model.

One limitation of the proposed approach is the requirement that object orientation be known. While this may be the case for example in ground target search [18], where objects are geographic cells, in general this does not hold. Another limitation is that since it is an inference method, it depends on externally collected measurements, possibly insufficient or in a sub-optimal way. Possible future work may target these issues.

References

[1] N. Atanasov, B. Sankaran, J.L. Ny, G. J. Pappas, and K. Daniilidis. Nonmyopic view planning for active object classification and pose estimation. *IEEE Trans. Robotics*, 30:1078–1090, 2014.

[2] Israel Becerra, Luis M Valentín-Coronado, Rafael Murrieta-Cid, and Jean-Claude Latombe. Appearance-based motion strategies for object detection. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 6455–6461. IEEE, 2014.

[3] Israel Becerra, Luis M Valentín-Coronado, Rafael Murrieta-Cid, and Jean-Claude Latombe. Reliable confirmation of an object identity by a mobile robot: A mixed appearance/localization-driven motion approach. *Intl. J. of Robotics Research*, 35(10):1207–1233, 2016.

[4] C.M. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer-Verlag, Secaucus, NJ, USA, 2006.

[5] Yarin Gal. *Uncertainty in Deep Learning*. PhD thesis, PhD thesis, University of Cambridge, 2017.

[6] Yarin Gal and Zoubin Ghahramani. Bayesian convolutional neural networks with bernoulli approximate variational inference. *arXiv preprint arXiv:1506.02158*, 2016.

[7] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Intl. Conf. on Machine Learning (ICML)*, 2016.

[8] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert. iSAM2: Incremental smoothing and mapping using the Bayes tree. *Intl. J. of Robotics Research*, 31:217–236, Feb 2012.

[9] Alex Kendall and Roberto Cipolla. Modelling uncertainty in deep learning for camera relocalization. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 4762–4769. IEEE, 2016.

[10] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *arXiv preprint arXiv:1703.04977*, 2017.

[11] Beipeng Mu, Shih-Yuan Liu, Liam Paull, John Leonard, and Jonathan How. Slam with objects using a nonparametric pose graph. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2016.

[12] Pavel Myshkov and Simon Julier. Posterior distribution analysis for bayesian inference in neural networks. *Advances in Neural Information Processing Systems (NIPS)*, 2016.

[13] Shayegan Omidshafiei, Brett T Lopez, Jonathan P How, and John Vian. Hierarchical bayesian noise inference for robust real-time probabilistic object classification. *arXiv preprint arXiv:1605.01042*, 2016.

[14] T. Patten, M. Zillich, R. Fitch, M. Vincze, and S. Sukkarieh. Viewpoint evaluation for online 3-d active object classification. *IEEE Robotics and Automation Letters (RA-L)*, 1(1):73–81, January 2016.

[15] Sudeep Pillai and John Leonard. Monocular slam supported object recognition. In *Robotics: Science and Systems (RSS)*, 2015.

[16] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning*. The MIT press, Cambridge, MA, 2006.

[17] Amit Shaked and Lior Wolf. Improved stereo matching with constant highway networks and reflective confidence learning. *arXiv preprint arXiv:1701.00165*, 2016.

[18] WT Teacy, Simon J Julier, Renzo De Nardi, Alex Rogers, and Nicholas R Jennings. Observation modelling for vision-based target search by unmanned aerial vehicles. In *Intl. Conf. on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1607–1614, 2015.

[19] Javier Velez, Garrett Hemann, Albert S Huang, Ingmar Posner, and Nicholas Roy. Modelling observation correlations for active exploration and robust object detection. *J. of Artificial Intelligence Research*, 2012.