Image and Vision Computing 39 (2015) 1–9

Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

Feature selection for position estimation using an omnidirectional camera ☆

Huan N. Do [a,1], Mahdi Jadaliha [a,1], Jongeun Choi [a,b,*], Chae Young Lim [c]

[a] Department of Mechanical Engineering, Michigan State University, East Lansing, MI 48824, USA
[b] Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI 48824, USA
[c] Department of Statistics and Probability, Michigan State University, East Lansing, MI 48824, USA

ARTICLE INFO

Article history:
Received 26 September 2013
Received in revised form 19 September 2014
Accepted 1 April 2015
Available online 8 May 2015

Keywords:
Vision-based localization
Appearance-based localization
Feature selection
Gaussian process regression
Hyperparameter estimation
Empirical Bayes methods

ABSTRACT

This paper considers visual feature selection to implement position estimation using an omnidirectional camera. The localization is based on a maximum likelihood estimation (MLE) with a map from optimally selected visual features using Gaussian process (GP) regression. In particular, the collection of selected features over a surveillance region is modeled by a multivariate GP with unknown hyperparameters. The hyperparameters are identified through the learning process by an MLE, which are used for prediction in an empirical Bayes fashion. To select features, we apply a backward sequential elimination technique in order to improve the quality of the position estimation with compressed features for efficient localization. The excellent results of the proposed algorithm are illustrated by the experimental studies with different visual features under both indoor and outdoor real-world scenarios.

1. Introduction

Minimizing levels of location uncertainties in sensor networks or robotic sensors is important for regression problems, e.g., prediction of environmental fields [1,2]. Localization of a robot relative to its environment using vision information (i.e., appearance-based localization) has received extensive attention over the past few decades from the robotic and computer vision communities [3–5]. Vision-based robot positioning may involve two steps. The first step involves learning some properties of vision data (features) with respect to the spatial position where observation is made, so-called mapping. The second step is to find the best match for the new spatial position corresponding to the newly observed features, so-called matching. The mapping from these visual features to the domain of the associated spatial position is highly nonlinear and sensitive to the type of selected features. In most cases, it is very difficult to derive the map analytically. The features shall vary as much as possible over the spatial domain while varying as small as possible for a given position over the disturbance. For example, they should be insensitive to changes in illumination and partial obstruction.

Motivated by the aforementioned situations, we consider the problem of selecting features from the original feature set in order to improve the localization performance of a robot. The central assumption when using a feature selection technique is that the original feature set contains many redundant or irrelevant features.

To facilitate further discussion, let us consider a configuration where the input vector X is the robot position and the output feature vector Y is the collection of extracted features from the vision data. We first build a feature map F at a robot location X such that $F(X) = Y$.

In order to reduce position estimation error, the ideal subset is defined as follows:

$$Y_{opt} = \arg\min_{\hat{Y}} \left\| X - F^{-1}\left(\hat{Y}\right) \right\|^2,$$

where $\hat{Y}$ is a vector that consists of the selected entries of the original vector Y. However with a high cardinality of the original feature set, the optimal solution relies on the combinatorial optimization which is not feasible.

On the other hand, using the mutual information criterion, F and F^{-1} could be chosen as follows:

$$F(X) = \arg\max_{Y} I(X,Y), \quad F^{-1}(Y) = \arg\max_{X} I(X,Y),$$

☆ This paper has been recommended for acceptance by Enrique Dunn.

* Corresponding author at: 428 S. Shaw Lane, Room 2459, East Lansing, MI 48824, USA. Tel.: +1 517 432 3164; fax: +1 517 353 1750.
E-mail addresses: dohuan@msu.edu (H.N. Do), jadaliha@msu.edu (M. Jadaliha), jchoi@egr.msu.edu (J. Choi), lim@stt.msu.edu (C.Y. Lim).

[1] The first two authors have the equal contributions.

http://dx.doi.org/10.1016/j.imavis.2015.04.002
0262-8856/

where $I(X,Y) = \iint \mathbb{P}(X,Y) log\left(\frac{\mathbb{P}(X,Y)}{\mathbb{P}(X)\mathbb{P}(Y)}\right)$ is the mutual information of X and Y. Note that, in the case where $\mathbb{P}(X)$ and $\mathbb{P}(Y)$ are constant then $F(X)$ is obtained by maximizing the log-likelihood function. Peng et al. [6] show that by using mutual information, one can achieve a recognition rate higher than 90% while just using 0.61% of feature space for a classification problem. However, the approach based on mutual information could suffer from its computational complexity [7].

In order to make a fast and precise estimation, most of the existing localization algorithms extract a small set of important features from the robotic sensor measurements. The features used in different approaches for robotic localization range from *C.1*: artificial markers such as color tags [8] and barcodes (that need to be installed) [9], *C.2*: geometric features such as straight wall segments and corners [10], and to *C.3*: natural features such as light and color histograms [11]. Most of the landmark-based localization algorithms are classified in *C.1* and *C.2*. It is shown in [12] that autonomous navigation is possible for outdoor environments with the use of a single camera and natural landmarks. In a similar attempt, [13] addressed the challenging problem of indoor place recognition from wearable video recording devices.

The localization methods which rely on artificial markers (or static landmarks) have disadvantages such as lack of flexibility and lack of autonomy. A method is described in [14] that enables robots to learn landmarks for localization. Artificial neural networks are used for extracting landmarks. However, the localization methods which rely on dynamic landmarks [14] have disadvantages such as lack of stability. Furthermore, there are reasons to avoid the geometric model as well, even when a geometric model does exist. Such cases may include: 1) the difficulty of reliably extracting sparse, stable features using geometrical models, 2) the ability to use all sensory data directly rather than a relatively small amount of abstracted discrete information obtained from feature extraction algorithms, and 3) the high computational and storage costs of dealing with dense geometric features.

In contrast to the localization problem with artificial markers or popular geometrical models, there are a growing number of practical scenarios in which global statistical information is used instead. Some works illustrate localization using various spatially distributed (continuous) signals such as distributed wireless Ethernet signal strength [15], or multi-dimensional magnetic fields [16]. In [17], a neural network is used to learn the implicit relationship between the pose displacements of a 6-DOF robot and the observed variations in global descriptors of the image such as geometric moments and Fourier descriptors. In similar studies, gradient orientation histograms [18] and low dimensional representation of the vision data [19] are used to localize mobile robots. In [5], an algorithm is developed for navigating a mobile robot using a visual potential. The visual potential is computed from the image appearance sequence captured by a camera mounted on the robot. A method for recognizing scene categories by comparing the histograms of local features is presented in [20]. Without explicit object models, by using global cues as indirect evidence about the presence of an object, they consistently achieve an improvement over an orderless image representation [20].

The recent research efforts that are closely related to our problem are summarized as follows. The location for a set of image features from new observations is inferred by comparing new features with the calculated map [21–23]. In [24], a neural network is used to learn the mapping between image features and robot movements.

Similarly, there exists effort on automatically finding the transformation that maximizes the mutual information between two random variables [25].

Using Gaussian process (GP) regression, the authors of [21,26] present effective approaches to build a map from a sparse set of noisy observations taken from known locations using an omnidirectional camera. While the selection of visual features for such applications determines the ultimate performance of the algorithms, such a topic has not been investigated to date. Therefore, building on Brook's approach [21] our work expands it more on the feature extraction and selection in order to improve the quality of localization. A Bayesian point of view is taken to make the map using a GP framework.

The contributions of the paper are as follows. This paper provides a position estimation method using an omnidirectional camera. We present an approach to build a map from optimally selected visual features using GP regression. First, we describe how we extract some robust properties from vision data captured by an omnidirectional camera (Section 2). In particular, we describe how different transformations are applied to the panoramic image to calculate a set of image properties. We then transform the high dimensional vision data to a set of uncorrelated feature candidates. A multivariate GP regression with unknown hyperparameters is formulated to connect the set of selected features to their corresponding sampling positions (Section 3). An empirical Bayes method using a point estimate is used to predict the feature map. Next, a feature reduction approach is developed using the backward sequential elimination method such that an optimal subset of the features is selected to minimize the root mean square error (RMSE) and compress the feature size (Section 4). The effectiveness of the proposed algorithms is illustrated by experimental results under indoor and outdoor conditions. Additionally, we compare our results with another appearance-based localization method utilizing the bag of words (BOW) algorithm [27] (Section 5).

2. Image features

Conventional video cameras with projective lens have restricted fields of view. With different mirrors, 360° panoramic views can be achieved in a single image [28]. In this paper, to make localization insensitive to the heading angle, an omni-directional camera is used to capture a 360° view from the environment of a robot.

Before an omnidirectional image is processed, it is first unwrapped. When it comes from the camera, the image is a nonlinear mapping of a 360° panoramic view onto a donut shape. Recovering the panoramic view from the wrapped view requires the reverse mapping of pixels from the wrapped view onto a panoramic view [22,29]. Fig. 1-(a) and (b) shows the wrapped omnidirectional image and the unwrapped panoramic image, respectively.

We will use the notation $y^{[i]}$ generally for all types of image properties that will be extracted from image i. In particular, we will use the FFT coefficients, the histogram, and the Steerable Pyramid (SP) decomposition [30] as image properties [20]. These feature types and their properties (indicated by $y^{[i]}$) are briefly explained as follows:

FFT (128) The fast Fourier transform (FFT) is applied to the panoramic image to calculate a set of image properties y. For a square image of size $N \times N$, the two-dimensional FFT is given by

$$F^{[i]}(\rho, l) = \sum_{a=0}^{N-1} \sum_{b=0}^{N-1} f^{[i]}(a,b) e^{-j2\pi\left(\rho\frac{a}{N} + l\frac{b}{N}\right)},$$

where $f^{[i]}$ is the i-th two-dimensional realized image, and j is the imaginary unit. To use FFT, we convert panoramic color images to gray scale 128×128 pixel images, i.e., $[f(a, b)]$. Figs. 1(c) and (d) show the reduced size gray scale image and its two-dimensional FFT magnitude plot, respectively. Often in image processing, only the magnitude of the Fourier transformed image is utilized, as it contains most of the information of the geometric structure of the spatial domain image [31]. Additionally, the magnitude of the Fourier transformed panoramic image is not affected by the rotation in yaw angle.

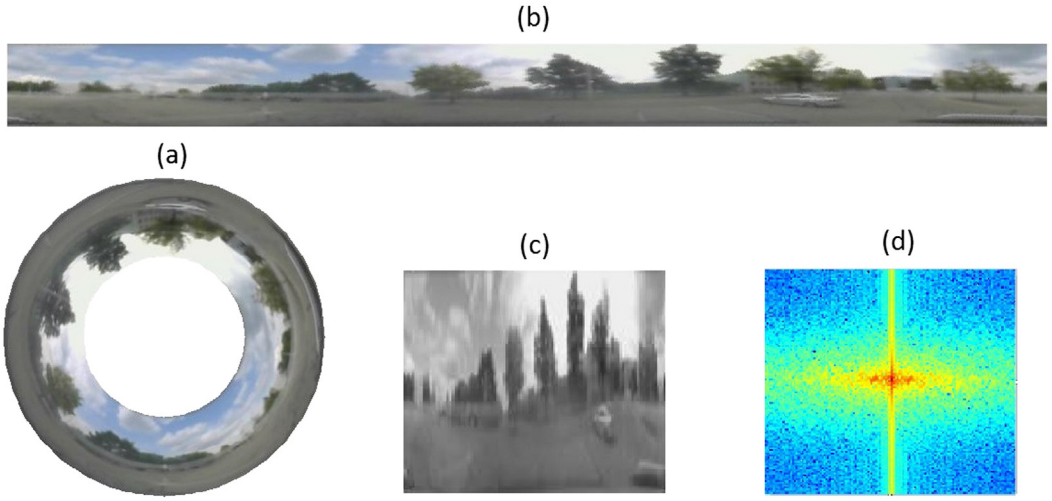

Fig. 1. Labels (a) and (b) show the wrapped omnidirectional image and the unwrapped panoramic image, respectively. Labels (c) and (d) show the reduced size gray scale image and the two-dimensional FFT magnitude plot, respectively.

In [22], it was shown that the first 15 components of FFT carry enough information to correctly match a pair of images. We specify the first 64 FFT components of each axis, e.g., $y^{[i]} = \{F^{[i]}(1, 0), \cdots, F^{[i]}(64, 0), F^{[i]}(0, 1), \cdots, F^{[i]}(0, 64)\}$ to be our 128-dimensional image properties of the FFT features.

Histogram (156) The image histogram [32] is a type of a histogram that acts as a graphical representation of the tonal distribution in a digital image. The number of pixels in each tonal bin of the histogram for the image is used as an image property from the histogram. Thus, the number of different tonal bins (which is 156) corresponds to the number of image properties from the histogram of the image.

SP (72) The Steerable Pyramid (SP) [30] is a multi-scale wavelet decomposition in which the image is linearly decomposed into scale and orientation subbands, and then the band-pass filters are applied to each subband individually. Using the method from [21], an image is decomposed by 4 scale and 6 orientations, which yields 24 subbands. Each subband is represented by three values, viz., the average filter responses from the top, middle, and bottom of the image such that we have 72 image properties for the SP decomposition. The multi-scale wavelet decomposition is also used widely by appearance-based place recognition methods [19,21].

SURF (64) The Speeded-Up Robust Features (SURF) [33] is a powerful scale- and rotation-invariant that utilizes Haar wavelet responses to produce a 64 dimensional descriptor vector for points of interest in an image. Furthermore, the SURF of each point of interest is calculated locally based on the neighborhood region around it.

In general, specific image processing to generate original features will affect the overall performance of the localization. These features are robust to changes in the yaw angle of the vehicle, which results in horizontal shifts of the pixels of the panoramic images. Additionally, images are converted into gray-scale for all types of features since the gray-scale images are less likely to be affected by illumination [34]. The presence of moving objects and occlusions is treated by modeling image features as Gaussian processes via vertical variability and measurement noise, respectively.

3. Gaussian process (GP) model

We propose a multivariate GP as a model for the collection of image features. A GP defines a distribution over a space of functions and it is completely specified by its mean function and covariance function. We denote that $y_\rho^{[i]} := y_\rho(s^{[i]}) \in \mathbb{R}$ is the i-th realization of the ρ-th image property and $s^{[i]} \in \mathcal{S}$ is the associated position where the realization occurs. Here $\mathcal{S}$ denotes the surveillance region, which is a compact set. Then, the accumulative image properties y is a random vector defined by $y = (y_1^T, \cdots, y_m^T)^T \in \mathbb{R}^{nm}$, and $y_\rho = (y_\rho^{[1]}, \cdots, y_\rho^{[n]}) \in \mathbb{R}^n$ contains n realizations of the ρ-th image property.

We assume that the accumulative image properties can be modeled by a multivariate GP, i.e. $y \sim \mathcal{GP}(\Gamma, \Lambda)$, where $\Gamma : \mathcal{S}^n \to \mathbb{R}^{mn}$ and $\Lambda : \mathcal{S}^n \to \mathbb{R}^{mn \times mn}$ are the mean function and the covariance function, respectively. However, the size and multivariate nature of the data lead to computational challenges in implementing the framework.

For models with multivariate output, a common practice is to specify a separable covariance structure for the GP for efficient computation. For example, Higdon [35] calibrated a GP simulator with the high dimensional multivariate output, using principal components to reduce the dimensionality. Following such model reduction techniques, we transform the vector y to a vector z such that its elements $\{z_\rho | \rho \in \Omega_m\}$, where $\Omega_m = \{1, \cdots, m\}$ are i.i.d.

The statistics of y can be computed from the learning data set.

$$\mu_y = \frac{1}{n}\sum_{i=1}^{n} y^{[i]}, \quad \sum_y = \frac{1}{n-1}\sum_{i=1}^{n} \left\| y^{[i]} - \mu_y \right\|^2.$$

The singular value decomposition (SVD) of Σ_y is a factorization of the form $\sum_y = USU^T$, where U is a real unitary matrix and S is a rectangular diagonal matrix with nonnegative real numbers on the diagonal. In summary, the transformation will be performed by the following formula.

$$z^{[i]} = S^{-1/2} U^T \left(y^{[i]} - \mu_y \right). \tag{1}$$

From now on, we assume that we applied the transformation given by Eq. (1) to the visual data. Hence, we have the zero-mean multivariate

GP: $z(s)\sim\mathcal{GP}(0,\mathcal{K}(s,s'))$, which consists of multiple scalar GPs that are independent of each other.

3.1. The ρ-th random field

In this subsection, we only consider the ρ-th random field (visual feature). Other scalar random fields can be treated in the same way. A random vector x, which has a multivariate normal distribution of mean vector μ and covariance matrix Σ, is denoted by $x\sim\mathcal{N}(\mu,\Sigma)$. The collection of n realized values of the ρ-th random field is denoted by $z_\rho := (z_\rho^{[1]},\cdots, z_\rho^{[n]})^T\in\mathbb{R}^n$, where $z_\rho^{[i]} := z_\rho(s^{[i]})$ is the i-th realization of the ρ-th random field and $s^{[i]} = \left(s_1^{[i]}, s_2^{[i]}\right)\in\mathcal{S}\subset\mathbb{R}^2$ is the associated position where the realization occurs. We then have $z_\rho(s)\sim\mathcal{N}(0,\Sigma_\rho)$, where $\Sigma_\rho\in\mathbb{R}^{n\times n}$ is the covariance matrix. The i,j-th element of Σ_ρ is defined as $\sum_\rho^{[ij]} = \mathbb{C}\mathrm{ov}(z_\rho^{[i]}, z_\rho^{[j]})$. In this paper, we consider the squared exponential covariance function [36] defined as

$$\sum\nolimits_\rho^{[ij]} = \sigma_{f,\rho}^2 exp\left(-\frac{1}{2}\sum_{\ell=1}^{2}\frac{\left(s_\ell^{[i]}-s_\ell^{[j]}\right)^2}{\sigma_{\ell,\rho}^2}\right). \tag{2}$$

In general, the mean and the covariance functions of a GP can be estimated a priori by maximizing the likelihood function [37].

The prior distribution of z_ρ is given by $\mathcal{N}(0,\Sigma_\rho)$.

A noise corrupted measurement $\tilde{z}_\rho^{[i]}$ at its corresponding location $s^{[i]}$ is defined as follows:

$$\tilde{z}_\rho^{[i]} = z_\rho^{[i]} + \epsilon_\rho^{[i]}, \tag{3}$$

where the measurement errors $\{\epsilon_\rho^{[i]}\}$ are assumed to be an independent and identically distributed (i.i.d.) Gaussian white noise, i.e., $\epsilon_\rho^{[i]}\overset{i.i.d.}{\sim}\mathcal{N}\left(0,\sigma_{\epsilon,\rho}^2\right)$. Thus, we have that

$$\tilde{z}_\rho\sim\mathcal{N}(0,R_\rho),$$

where $R_\rho = (\sum_\rho + \sigma_{\epsilon,\rho}^2 I)$. The log-likelihood function is defined by

$$L_{\theta,\rho} = -\frac{1}{2}\tilde{z}_\rho^T R_\rho^{-1}\tilde{z}_\rho - \frac{1}{2}log|R_\rho| - \frac{n}{2}log2\pi, \tag{4}$$

where n is the size of $\tilde{z}_\rho$.

The hyperparameter vector of the ρ-th random field is defined as $\theta_\rho = (\sigma_{f,\rho}, \sigma_{\epsilon,\rho}, \sigma_{1,\rho}, \sigma_{2,\rho})\in\mathbb{R}_{>0}^4$. Using the likelihood function in Eq. (4) the hyperparameter vector can be computed by the ML estimator

$$\overline{\theta}_\rho = \arg\max_\theta L_{\theta,\rho}, \tag{5}$$

which will be plugged in prediction as in an empirical Bayes way.

All parameters are learned simultaneously. If no prior information is given, then the maximum a posteriori probability (MAP) estimator is equal to the ML estimator [37].

In a GP, every finite collection of random variables has a multivariate normal distribution. Consider a realized value of the ρ-th random field $z_\rho^\star$ being taken from the associated location $s^\star$. The probability distribution $\mathbb{P}\left(z_\rho^\star\middle|s^\star,s,\tilde{z}_\rho\right)$ is a normal distribution with the following mean and variance.

$$\mu_\rho(s^\star) = C_\rho^T R_\rho^{-1}\tilde{z}_\rho,\quad \sigma_\rho^2(s^\star) = \sigma_{f,\rho}^2 - C_\rho^T R_\rho^{-1} C_\rho, \tag{6}$$

where the covariance $C_\rho := \mathbb{C}\mathrm{ov}(z_\rho^\star, z_\rho)\in\mathbb{R}^{1\times n}$ is defined similar to Eq. (2).

In order to estimate location $s^\star$, using the MAP estimator, we need to compute $\mathbb{P}\left(s^\star|\tilde{z}_\rho^\star,s,\tilde{z}_\rho\right)$, where the noisy observation $\tilde{z}_\rho^\star$ is the summation of the realized values of the random field $z_\rho^\star$ and a noise process.

$$\mathbb{P}\left(s^\star|\tilde{z}_\rho^\star,s,\tilde{z}_\rho\right) = \frac{\mathbb{P}\left(\tilde{z}_\rho^\star\middle|s^\star,s,\tilde{z}_\rho\right)\mathbb{P}(s^\star|s,\tilde{z}_\rho)}{\mathbb{P}\left(\tilde{z}_\rho^\star\middle|s,\tilde{z}_\rho\right)} \tag{7}$$

A MAP estimator given the collection of observations $\tilde{z}_\rho$ is a mode of the posterior distribution.

$$\overline{s}_\rho^\star = \arg\max_{s^\star\in\mathcal{S}}\mathbb{P}\left(s^\star|\tilde{z}_\rho^\star,s,\tilde{z}_\rho\right) \tag{8}$$

If $\mathbb{P}(s^\star|s,\tilde{z}_\rho)$ and $\mathbb{P}\left(\tilde{z}_\rho^\star\middle|s,\tilde{z}_\rho\right)$ are the uniform probabilities, then the MAP estimator is equal to the ML estimator, given by

$$\overline{s}_\rho^\star = \arg\max_{s^\star\in\mathcal{S}} L_\rho(s^\star), \tag{9}$$

where the ρ-th log-likelihood function, i.e., $L_\rho(s^\star)$, is defined as follows:

$$L_\rho(s^\star) = \frac{-1}{2}\left(\frac{\left|\tilde{z}_\rho^\star-\mu_\rho(s^\star)\right|^2}{\sigma_{\epsilon,\rho}^2+\sigma_\rho^2(s^\star)} + log\left(\sigma_{\epsilon,\rho}^2+\sigma_\rho^2(s^\star)\right) + log\,2\pi\right). \tag{10}$$

4. Localization and feature selection

Let Ω be the collection of indices that are associated to the multiple scalar random fields (of the multivariate GP). Provided that all scalar random fields (of the multivariate GP) are independent of each other, we then obtain a computationally efficient ML estimate of the location given the observations of all scalar random fields $\{\tilde{z}_\rho|\rho\in\Omega\}$ as follows:

$$\overline{s}_\Omega^\star = \arg\max_{s^\star\in\mathcal{S}}\sum_{\rho\in\Omega} L_\rho(s^\star), \tag{11}$$

where $L_\rho(s^\star)$ is the ρ-th log-likelihood function as given in Eq. (10).

In this paper, a backward sequential elimination technique [38] is used for the model selection. It is mainly used in settings where the goal is prediction, and one wants to estimate how accurately a predictive model will perform in practice. To this end, we divide the data set into two segments: one used to learn or train the GP model and the other used to validate the model.

The RMSE is used to measure the performance of GP models. It is defined by the following equation:

$$\mathrm{RMSE}(\Omega) = \sqrt{\frac{1}{n_c}\sum_{i=1}^{n_c}\left\|s_c^{[i]}-\overline{s}_\Omega^\star\right\|^2}, \tag{12}$$

where $\|\cdot\|$ is the Euclidean norm of a vector. In the case that $\Omega = \emptyset$, we define the following:

$$\mathrm{RMSE}(\emptyset) = \sqrt{\frac{1}{n_c}\sum_{i=1}^{n_c}\left\|s_c^{[i]}-\mathrm{median}(s_c)\right\|^2},$$

where median $(\cdot)$ is the median of a random vector. Assume that $\Omega_m = \{1,\cdots,m\}$ is the set of all features. Dupuis et al. [39] reported that the backward sequential elimination outperforms the forward sequential

selection. Thus, we use a backward sequential elimination algorithm as follows:

$$\Omega_{\ell-1} = \Omega_{\ell} - \arg\min_{\rho \in \Omega_{\ell}} \text{RMSE}(\Omega_{\ell} - \rho), \forall \ell \in \Omega_m, \tag{13}$$

where $\Omega_{\ell} - \rho = \{p | p \in \Omega_{\ell}, p \neq \rho\}$.

Finally a subset of features is selected as follows:

$$\Omega_{opt} = \arg\min_{\Omega = \Omega_1, \cdots, \Omega_m} \text{RMSE}(\Omega). \tag{14}$$

The optimum subset Ω_{opt} has the minimum RMSE among $\{\Omega_1, \cdots, \Omega_m\}$. The mapping and matching steps of the proposed approaches in this paper are summarized in Algorithms 1 and 2, respectively.

Algorithm 1. Learning maps from a sparse set of panoramic images observed in known locations.

Input:	#1. Training data set includes a set of panoramic images captured from known spatial sites,
Output:	#1. A linear transformation from image properties to uncorrelated visual features, #2. The estimated hyperparameter, the estimated mean and the estimated variance function of each independent visual feature,

1: extract image properties $y^{[i]}$ in the available learning data set.
2: use SVD to make a set of uncorrelated visual features $z^{[i]}$ using Eq. (1)
3: for each independent visual feature estimate hyperparameters using Eq. (5)
4: compute the mean function and variance function for each of independent features using Eq. (6)
5: choose optimal subset of visual features using Eq. (14) to eliminate some of the visual features that are worthless for the localization goal.

Algorithm 2. Localization using learned map of visual features.

Input:	#1. A linear transformation from image properties to uncorrelated visual features, #2. The estimated hyperparameter and the estimated mean and variance function of selected visual features,
Output:	#1. Position of newly captured images.

1: capture new images and obtain image properties $y^{\star}$.
2: compute the selected visual features $z^{\star}$ using Eq. (1)
3: compute the likelihood function of selected features $\rho \in \Omega_{opt}$ over the possible sampling positions using Eq. (10)
4: determine the estimated position $\bar{s}^{\star}_{\Omega_{opt}}$ using Eq. (11).

5. Indoor and outdoor experiments

In this section, we demonstrate the effectiveness of the proposed localization algorithms with experiments using different image features we discussed. We report results on two different data sets collected indoors (Case 1) and outdoors (Case 2).

5.1. Experimental setups

In Case 1, the Kogeto panorama lens was used to capture 360-degree images on an indoor corridor as illustrated in Fig. 2. In total, 207 pairs of the exact sampling positions on a regular lattice (7×2.7 m^2) were recorded manually and the corresponding panoramic images were collected.

In Case 2, we use a vision and GPS data acquisition circuit which consists of an Arduino microcontroller (Arduino MEGA board, Open Source Hardware platform, Italy), a Xsens GPS unit (MTi-G-700, Xsens Technologies B.V., Netherlands), a Raspberry Pi microcontroller (Raspberry Pi model B+, Raspberry Pi Foundation, United Kingdom) and a webcam (Logitech HD webcam C310, Logitech, Newark, CA, U.S.A.) glued to a 360 degree lens (Kogeto Panoramic Dot Optic Lens, Kogeto, U.S.A.). The data acquisition circuit was secured inside the vehicle while the omni-directional camera was fixed on the roof of the vehicle. The vehicle was driven through the surveillance area (Fig. 3). The surrounding scenes were recorded by the Raspberry Pi unit while the truth locations measured by the Xsens GPS unit were stored on the Arduino microcontroller. We collected 378 data points, on a 61×86 meter area on the campus of Michigan State University, East Lansing, MI, U.S.A. (see Fig. 3). Figs. 2 and 4 show the setups for Case 1 and Case 2, respectively.

The data sets are divided into 50% *learning*, 25% *backward sequential elimination* (*or validation*) and 25% *testing* data subsets. The learning data set is used to estimate the mean functions and the hyperparameters for the covariance functions to build GP models. The validation data set is used to select the best features in order to minimize the localization estimation RMSE and compress the feature. After the training and feature selection, we evaluate the performance of the selected model using the testing data set, which was not used for training or feature selection.

To analyze our results in a statistically meaningful way, we calculate the Bayesian Information Criteria (BIC) index for the model with all features and the one with the only selected features in addition to the RMSE. The BIC is a criterion for model selection based on the log likelihood with a penalty on the number of parameters to penalize over-fitting. The model with a smaller BIC index is less likely to be over-fitted [40].

5.2. Learning of GP models in an empirical Bayes approach

As illustrative examples for the case of utilizing FFT features of length 128, we apply the proposed algorithm to both data sets.

The variance of the random field σ_f^2, the spatial bandwidth $\sigma^2_{\ell,\rho}$, and the noise variance σ_ϵ^2 are estimated for each feature independently.

Fig. 2. Dot iPhone Panorama lens (left) and the indoor environment (right) for Case 1.

Fig. 3. Outdoor trajectory collected from a GPS unit.

Thus, for the FFT case, $128 \times 4 = 512$ hyperparameters need to be estimated in total for each experimental setup. The hyperparameters for Case 2 are estimated in the same manner. The 3D plots of the means and variances of the first three GP models for the case of 128 FFT features are shown in Fig. 5.

To study the effect of the turning angle of the vehicle (or the yaw angle) on the features, we run the algorithm with another data set in which the collected panoramic images are pre-processed so that the heading of the panoramic image is kept constant using the yaw angles from the GPS unit, denoted as (fixed angle) in Table 1.

All inferential algorithms are implemented using Matlab R2013a (The MathWorks Inc., Natick, MA, U.S.A.) on a PC (3.2 GHz Intel i7 Processor).

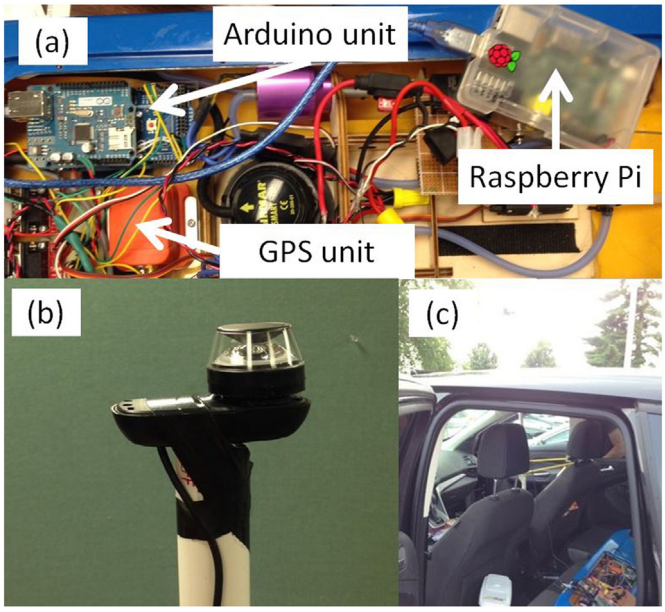

Fig. 4. (a) Data acquisition circuit, (b) panoramic camera and (c) vehicle used in Case 2.

5.3. Localization utilizing the bag of words (BOW)

We also compare the GP-based approach with a localization scheme based on the BOW. To have a fair comparison, we feed an identical data set to both of the methods. We utilize the SURF as the image descriptor for our BOW. We define the notation $y^{[i]}$ as a set of SURF points extracted from image i. Notice that the number of SURF points varies for different images. The region around each SURF point is represented by a descriptor vector of 64 lengths. The SURF points from the whole data set are accumulated and put into the k-means clustering [41]. Each centroid is defined as a codeword and the collection of centroids is defined as the codebook. Each SURF point is mapped into the index of the nearest centroid in the codebook. Therefore, we obtain a histogram of codewords for each image that indicates the appearance frequency of all codewords in the image. Lastly, the test set is classified by applying the k-nearest-neighbor classifier [42] based on the histogram of codewords.

We subsample 25% of the data to be the test set (the same test subset used in Table 1), which is associated with a newly defined label set, i.e., $\mathscr{T}_T := \{1, \cdots, n_T\}$. The label of each test data point $s^\star$ is assigned to the non-test data points within a 5 meter radius with respect to $s^\star$ (see Fig. 6). Such relabeled non-test data points are used for training the BOW. Since the BOW is mostly used for classification such as identical scene recognition [27], we define the localization error to compute the RMSE as follows.

Let $s_t(i) \in \mathcal{S}$ be the location of the test point i for all $i \in \mathscr{T}_T$.

Let $h^\star(i)$ be the predicted label for the test data point i. Then we define the error at test point i as follows:

$$\text{error}_i = \begin{cases} \|s_t(i) - s_t(h^\star(i))\| & \text{if } i \neq h^\star(i), \\ 0 & \text{if } i = h^\star(i), \end{cases} \tag{15}$$

for all $i \in \mathscr{T}_T$.

5.4. Experimental results

5.4.1. Our method over different features

The indoor and outdoor performances under different image features are summarized in Tables 1 and 2, respectively. We consider three different types of appearance-based features such as the FFT [17], the histogram [32], and the SP decomposition [30]. We calculate the RMSE of our localization estimation from the model on the same validation set and on a separated test set, denoted by "V" and "T" in the tables, respectively. To compare the reduction in the number of features, we use the compression ratio [43]. The compression ratio is defined as:

$$\text{Compression ratio} := \frac{\text{number of original features}}{\text{number of selected features}}.$$

Tables 1 and 2 show the appearance-based feature type (column 1), the total number of features (column 2), the optimum number of features along with the compression ratio on the validation set (column 3), the localization RMSE obtained using the total number of features (column 4), the localization RMSE from the optimum number of features selected by the backward sequential elimination (denoted as "BE") implemented on the test set (column 5), and the localization RMSE from validation set (column 6), the maximum localization error, i.e., $e_{\max}$ taken over the test set (column 7), the BIC indexes of the model using the total number of features (column 8), and the BIC indexes from the optimum number of features (column 9). For Case 2, the FFT and SP features are tested with the data in two situations when the yaw angles are varying and when they are fixed (denoted as (fixed) in Table 2) to gauge the effect of yaw angles.

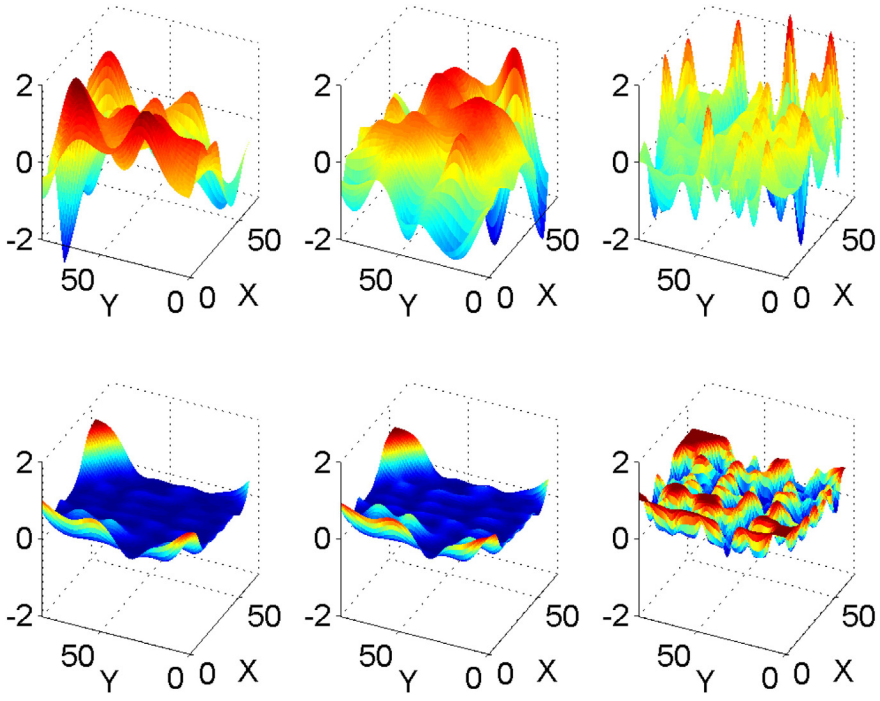

Fig. 5. GP models for each of the first three FFT features from the outdoor data set. The first row shows the means and the second row shows the variances of the GP models.

5.4.2. Performance among features

For Case 1, the SP shows the lowest localization RMSE with 4.8 compression ratio. For Case 2, the histogram shows the lowest localization RMSE with 2.7 compression ratio. The predicted trajectory that utilizes the histogram for Case 2 is shown in Fig. 7. For all experiments, BIC indexes before and after the feature selection show the significant improvement that makes the selected model less likely susceptible to over-fitting.

From the RMSE on the test data set, all feature types seem to be robust to the varying yaw angles.

5.4.3. Effect of localization noise

To investigate the effect of noisy sampling positions on the method, we added fictitious localization noise generated by a Gaussian white noise process with standard deviation of 0.3048 m (i.e., 1 ft) to the sampled locations of Case 1, which is denoted by (noisy) in Table. 1. As expected, the results show degradation when noisy sampling positions are used due to the sampling uncertainty in the GP learning and prediction processes.

5.4.4. Comparison of Cases 1 and 2

Note that sampling positions of Case 1 (non-noisy data) were recorded exactly while those of Case 2 were noisy due to the uncertainty in the GPS unit. On the other hand, it is clear that Case 2 has the larger RMSE due to the larger scale of the surveillance site compared to Case 1. Together, the results of Case 1 are shown to outperform those of Case 2.

Table 1
The localization performance for Case 1.

Feature type	# of features		RMSE (m)			e_{max} (m)	BIC index	
	Total	Opt	All	BE			All $\times 10^3$	Opt $\times 10^3$
		V	T	T	V	T		
FFT	128	77 (1.7)	1.48	1.68	0.94	7.2	16.2	9.5
FFT (noisy)	128	20 (6.4)	1.78	1.89	0.61	7.1	16.7	2.2
Hist	156	9 (17.3)	1.69	1.71	0.55	7.1	18.5	2.2
Hist (noisy)	156	50 (3.1)	1.58	1.64	1.14	6.0	20.5	6.4
SP	72	15 (4.8)	0.67	0.85	0.27	3.4	22.2	4.1
SP (noisy)	72	62 (1.2)	1.63	1.45	0.59	5.8	9.1	7.9

5.4.5. Comparison with the BOW

We compare the performance between our proposed GP-based approach and the BOW in Table 3. Table 3 shows the feature types (column 1), the total number of features (column 2), the optimum selected number of features from the angle-varying data set (column 3) and the fixed angle data set (column 4), the localization RMSE of the GP-based method from the angle-varying data set (column 5) and the fixed angle data set (column 6), the localization RMSE of the BOW from the angle-varying (column 7) and the fixed angle

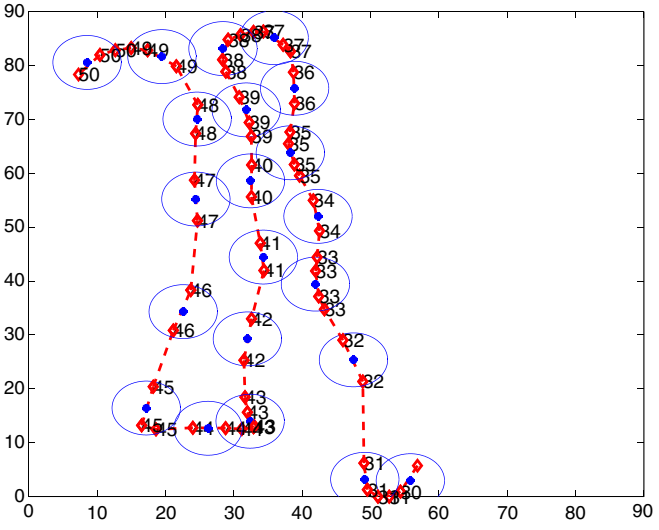

Fig. 6. Training data set assignment for the BOW. The test points, the training points (with new labels), and the 5 m radii are plotted in blue dots, red diamonds, and blue circles, respectively. The training points that do not belong to any test groups are eliminated. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 2
The localization performance for Case 2.

Feature type	# of features		RMSE (m)			e_{max} (m)	BIC index	
	Total	Opt	All	BE			All $\times 10^3$	Opt $\times 10^3$
		V	T	T	V	T		
FFT	128	41 (3.1)	14.5	10.9	4.3	58.5	28.4	8.1
FFT (fixed)	128	25 (5.12)	13.7	13.6	4.8	56.7	28.4	4.7
Hist	156	58 (2.7)	7.5	6.9	3.7	42.4	33.8	11.4
SP	72	35 (2.1)	21.08	18.72	7.86	63.4	15.6	7.1
SP (fixed)	72	28 (2.6)	14.52	14.74	13.19	51.9	15.6	5.5

Table 3
The localization performance comparison between the proposed approach and the BOW in Case 2. The RMSE is from the test set.

Feature type	Number of features			RMSE (m)			
	Total	Optimum		GP		BOW	
		Unfixed	Fixed	Unfixed	Fixed	Unfixed	Fixed
FFT	128	41	25	10.97	13.68	–	–
Histogram	156	58	–	6.91	–	–	–
SP	72	35	28	18.72	14.74	–	–
SURF	–	–	–	–	–	23.15	22.07

(column 8) data sets. Since the performance of the BOW highly depends on the clustering results, we run the BOW with different sizes of clusters, and the one that yields the highest classification percentage (80–90%) is chosen to calculate the RMSE using the error defined in Eq. (15). As discussed, the change in yaw angle does not show significant effect on the SURF. Table 3 shows that our approach outperforms the BOW.

In summary, we achieve significant reduction in the number of features while improving the RMSE when applied to the validation set. Furthermore, we maintain approximately the same RMSE levels when applied to a new test data set.

6. Conclusion and future works

This paper has presented a novel approach to use vision data for the robot localization. The predictive statistics of vision data is learned in advance and used in order to estimate the position of a vehicle, equipped just with an omnidirectional camera in both indoor and outdoor environments. The multivariate GP model is used to model a collection of selected visual features.

The locations are estimated by maximizing the likelihood function without fusing combining vehicle dynamics with measured features in order to evaluate the proposed scheme alone. Hence, we believe that the localization performance will be further improved when vehicle dynamics are fused together via Kalman filtering or particle filtering.

A limitation of the current approach arises from the fact that, after the initial training phase, learning is discontinued. If the environment changes, it is desirable that the localization routines adapt to the changes in the environment. Thus, a future research direction is to develop a localization scheme that is adaptive to changes in the environment.

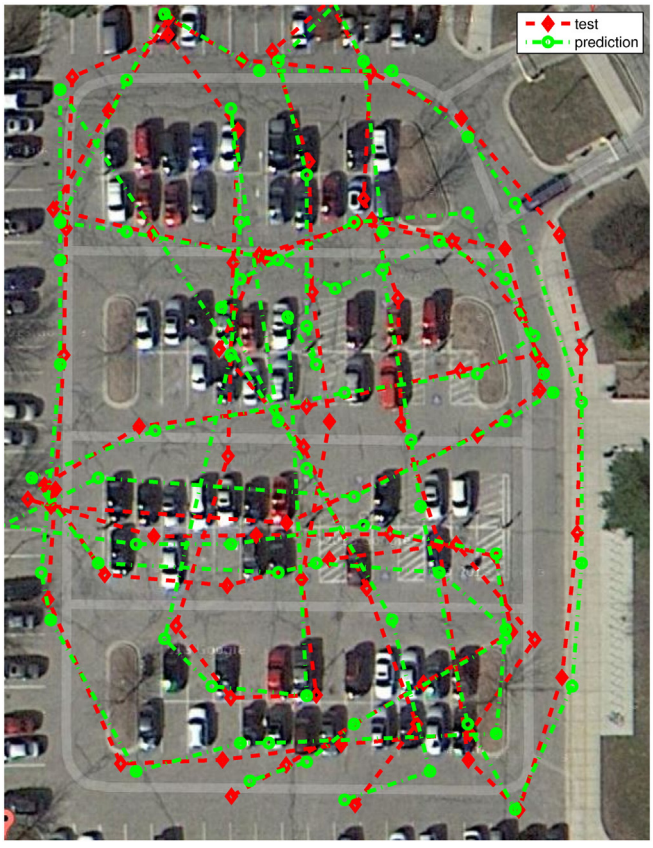

Fig. 7. Prediction result for Case 2 with the histogram. The test path and the prediction are plotted over the Google Map image in red diamonds and green dots, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Acknowledgment

This work has been supported by the National Science Foundation through CAREER award CMMI-0846547. Mr. Do has been supported by the Vietnam Education Foundation (G-3-10180) fellowship. The authors would like to thank Mr. Alexander Robinson from Thornapple Kellogg High School and Ms. Tam Le from the Department of Computer Science and Engineering, Michigan State University for their contributions in the preparation of the experiments.

References

[1] S. Choi, M. Jadaliha, J. Choi, S. Oh, Distributed gaussian process regression under localization uncertainty, J. Dyn. Syst. Meas. Control. vol. 137 (no. 3) (2015) 031002.
[2] M. Jadaliha, Y. Xu, J. Choi, N.S. Johnson, W. Li, Gaussian process regression for sensor networks under localization uncertainty, IEEE Trans. Signal Process. 61 (2) (2013) 223–237.
[3] G.N. DeSouza, A.C. Kak, Vision for mobile robot navigation: a survey, IEEE Trans. Pattern Anal. Mach. Intell. 24 (2) (2002) 237–267.
[4] F. Bonin-Font, A. Ortiz, G. Oliver, Visual navigation for mobile robots: a survey, J. Intell. Robot. Syst. 53 (3) (2008) 263–296.
[5] B. Guo, M.S. Nixon, Gait feature subset selection by mutual information, IEEE Trans. Syst. Man Cybern. Syst. Hum. 39 (1) (2009) 36–46.
[6] H. Peng, F. Long, C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, IEEE Trans. Pattern Anal. Mach. Intell. 27 (8) (2005) 1226–1238.
[7] G. Jang, S. Lee, I. Kweon, Color landmark based self-localization for indoor mobile robots, Proceedings of the IEEE International Conference on Robotics and Automation, vol. 1 2002, pp. 1037–1042.
[8] D. Scharstein, A. Briggs, Real-time recognition of self-similar landmarks, Image Vis. Comput. 19 (11) (2001) 763–772.
[9] W.Y. Jeong, K.M. Lee, Visual slam with line and corner features, Proceeding of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE 2006, pp. 2570–2575 (1em plus 0.5em minus 0.4em).
[10] P. Blaer, P. Allen, Topological mobile robot localization using fast vision techniques, Proceedings of the IEEE International Conference on Robotics and Automation, vol. 1 2002, pp. 1031–1036.
[11] E. Royer, M. Lhuillier, M. Dhome, J.-M. Lavest, Monocular vision for mobile robot localization and autonomous navigation, Int. J. Comput. Vis. 74 (3) (2007) 237–260.
[12] V. Dovgalecs, R. Mégret, Y. Berthoumieu, Multiple feature fusion based on co-training approach and time regularization for place classification in wearable video, Adv. Multimed. 1 (2013).
[13] S. Thrun, Bayesian landmark learning for mobile robot localization, Mach. Learn. 33 (1) (1998) 41–76.
[14] B. Ferris, D. Fox, N. Lawrence, WiFi-SLAM using Gaussian process latent variable models, Proceedings of the 20th International Joint Conference on, Artificial Intelligence 2007, pp. 2480–2485.
[15] I. Vallivaara, J. Haverinen, A. Kemppainen, J. Roning, Simultaneous localization and mapping using ambient magnetic field, Proceeding of the IEEE International

Conference on Multisensor Fusion and Integration for Intelligent Systems Sep. 2010, pp. 14–19.
[16] G. Wells, C. Venaille, C. Torras, Vision-based robot positioning using neural networks, Image Vis. Comput. 14 (10) (1996) 715–732.
[17] J. Kosecka, L. Zhou, P. Barber, Z. Duric, Qualitative image based localization in indoors environments, Proceedings of the IEEE Computer Society Conference on Computer Vision and, Pattern Recognition, vol. 2 2003, pp. II-3–II-8.
[18] A. Torralba, K. Murphy, W. Freeman, M. Rubin, Context-based vision system for place and object recognition, Proceedings of the IEEE International Conference on Computer Vision 2003, pp. 273–280.
[19] N. Ohnishi, A. Imiya, Appearance-based navigation and homing for autonomous mobile robot, Image Vis. Comput. 31 (6) (2013) 511–532.
[20] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2 2006, pp. 2169–2178.
[21] A. Brooks, A. Makarenko, B. Upcroft, Gaussian process models for indoor and outdoor sensor-centric robot localization, IEEE Trans. Robot. 24 (6) (2008) 1341–1351.
[22] E. Menegatti, T. Maeda, H. Ishiguro, Image-based memory for robot navigation using properties of omnidirectional images, Robot. Auton. Syst. 47 (4) (2004) 251–267.
[23] E. Menegatti, M. Zoccarato, E. Pagello, H. Ishiguro, Image-based Monte Carlo localisation with omnidirectional images, Robot. Auton. Syst. 48 (1) (2004) 17–30.
[24] G. Wells, C. Torras, Assessing image features for vision-based robot positioning, J. Intell. Robot. Syst. 30 (1) (2001) 95–118.
[25] N. Vlassis, R. Bunschoten, B. Krose, Learning task-relevant features from robot data, Proceedings of the IEEE International Conference on Robotics and Automation, vol. 1 2001, pp. 499–504.
[26] T. Schairer, B. Huhle, P. Vorst, A. Schilling, W. Straser, Visual mapping with uncertainty for correspondence-free localization using Gaussian process regression, Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2011, pp. 4229–4235.
[27] T. Botterill, S. Mills, R. Green, Speeded-up bag-of-words algorithm for robot localisation through scene recognition, Image and Vision Computing New Zealand, 2008. IVCNZ 2008. 23rd International Conference Nov. 2008, pp. 1–6.
[28] J.C.A. Fernandes, J.A.B.C. Neves, Using conical and spherical mirrors with conventional cameras for 360 panorama views in a single image, Proceedings of the IEEE International Conference on Mechatronics 2006, pp. 157–160.
[29] C. Demonceaux, P. Vasseur, Y. Fougerolle, Central catadioptric image processing with geodesic metric, Image Vis. Comput. 29 (12) (2011) 840–849.
[30] E.P. Simoncelli, W.T. Freeman, The steerable pyramid: a flexible architecture for multi-scale derivative computation, 2nd IEEE International Conference on Image Processing, vol. 3, no. 3 1995, pp. 444–447.
[31] M. Nixon, A.S. Aguado, Feature Extraction & Image Processing, Academic Press, 2008. (1em plus 0.5em minus 0.4em).
[32] E. Hadjidemetriou, M.D. Grossberg, S.K. Nayar, Multiresolution histograms and their use for recognition, IEEE Trans. Pattern Anal. Mach. Intell. 26 (7) (2004) 831–847.
[33] T.T. Herbert Bay, Andreas Ess, L.V. Gool, Speeded-up robust features (SURF), Comput. Vis. Image Underst. 110 (3) (2008) 346–359.
[34] C. Kanan, G.W. Cottrell, Color-to-grayscale: does the method matter in image recognition? PLoS One 7 (1) (2012) e29740.
[35] D. Higdon, J. Gattiker, B. Williams, M. Rightley, Computer model calibration using high-dimensional output, J. Am. Stat. Assoc. 103 (482) (2008) 570–583.
[36] C.E. Rasmussen, Gaussian Processes for Machine Learning, MIT Press, 2006. (1em plus 0.5em minus 0.4em).
[37] Y. Xu, J. Choi, Adaptive sampling for learning Gaussian processes using mobile sensor networks, Sensors 11 (3) (March 2011) 3051–3066.
[38] J.-H. Kim, Estimating classification error rate: repeated cross-validation, repeated hold-out and bootstrap, Comput. Stat. Data Anal. 53 (11) (2009) 3735–3745.
[39] Y. Dupuis, X. Savatier, P. Vasseur, Feature subset selection applied to model-free gait recognition, Image Vis. Comput. 31 (8) (2013) 580–591.
[40] S. Konishi, G. Kitagawa, Bayesian information criteria, Inf. Criteria Stat. Model. (2008) 211–237.
[41] J.A. Hartigan, M.A. Wong, Algorithm as 136: a k-means clustering algorithm, Appl. Stat. (1979) 100–108.
[42] J.M. Keller, M.R. Gray, J.A. Givens, A fuzzy k-nearest neighbor algorithm, IEEE Trans. Syst. Man Cybern. (no. 4) (1985) 580–585.
[43] D. Salomon, Data Compression: The Complete Reference, Springer Science & Business Media, 2004. (1em plus 0.5em minus 0.4em).