Machine Vision and Applications (2013) 24:1043–1053
DOI 10.1007/s00138-012-0480-y

ORIGINAL PAPER

Large-scale gaussian process multi-class classification for semantic segmentation and facade recognition

Björn Fröhlich · Erik Rodner · Michael Kemmler · Joachim Denzler

Received: 22 February 2012 / Revised: 28 November 2012 / Accepted: 17 December 2012 / Published online: 4 January 2013

Abstract This paper deals with the task of semantic segmentation, which aims to provide a complete description of an image by inferring a pixelwise labeling. While pixelwise classification is a suitable approach to achieve this goal, state-of-the-art kernel methods are generally not applicable since training and testing phase involve large amounts of data. We address this problem by presenting a method for large-scale inference with Gaussian processes. Standard limitations of Gaussian process classifiers in terms of speed and memory are overcome by pre-clustering the data using decision trees. This leads to a breakdown of the entire problem into several independent classification tasks whose complexity is controlled by the maximum number of training examples allowed in the tree leaves. We additionally propose a technique which allows for computing multi-class probabilities by incorporating uncertainties of the classifier estimates. The approach provides pixelwise semantics for a wide range of applications and different image types such as those from scene understanding, defect localization, and remote sensing. Our experiments are performed with a facade recognition application that shows the significant performance gain achieved by our method compared to previous approaches.

Keywords Large scale classification · Gaussian processes · Random decision forest · Semantic segmentation · Facade recognition · Scene interpretation

B. Fröhlich (✉) · E. Rodner · M. Kemmler · J. Denzler
Friedrich Schiller University, Jena, Germany
e-mail: bjoern.froehlich@uni-jena.de

E. Rodner
ICSI, UC Berkeley, USA

1 Introduction

Semantic segmentation can be regarded as one of the most difficult visual recognition problems, since it requires turning each pixel of an image into a suitable category label. Due to the very general problem description, semantic segmentation approaches can be used in nearly every application that requires a precise labeling. Especially in the context of facade recognition, semantic segmentation has been found to be an useful tool. In contrast to the direct categorization of objects in a street scene [12], the general framework of semantic segmentation can be often augmented with additional information about the special task at hand. For instance, the consecutive nature of images drawn from a sequence can be exploited to enhance classification accuracy [45] or to infer a 3D reconstruction of streets [43]. In the work of [36], prior information regarding the composition of facade parts, such as the relative location of windows and doors, is enforced using shape grammars.

Irrespective of the kind and amount of prior information used, the semantic segmentation step remains a crucial part in most facade recognition approaches. Usually, this task is solved in a supervised manner by learning a classifier on local patches with training examples obtained from pixelwise labeled images [9,11,30,31]. To cope with the large amount of training data, previous works use piecewise linear classifiers as classification techniques such as logistic regression [9], random decision forests [12,30,36] or boosting [16,31,43,45]. In this area, the use of non-linear and non-parametric learning machines, such as Gaussian process (GP) classifiers [24], is limited due to their computational demands and their need for a large memory capacity.

In this paper, we demonstrate how to perform inference for GP classification with tens of thousands of training examples occurring in supervised semantic segmentation

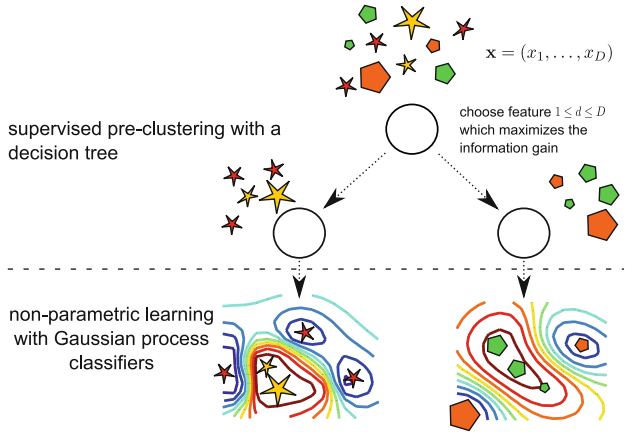


Fig. 1 An outline of our approach: a decision tree is used to cluster the data in a supervised manner and a GP classifier is used to separate classes in each leaf node

scenarios. Our approach is based on pre-clustering the available training set with decision trees and learning a GP classifier for each leaf of the tree (cf. Fig. 1). In contrast to large-margin-based learners, such as support vector machines [28], GP classifiers implicitly allow us to calculate the uncertainty of an estimate, which is particularly useful to derive suitable multi-class probabilities or for novelty detection. The resulting combined classifier offers to go beyond the restrictions of piecewise linear classifiers. Due to the large variability of local features belonging to different object categories, the ability to discriminate classes in a non-linear way is especially important for semantic segmentation tasks. Furthermore, our approach is adaptive and allows for handling the trade-off between accuracy and computation time.

In comparison to other semantic segmentation methods like conditional Markov random fields (CRF), our proposed method models the image in multiple local features, which are all analyzed in a continuous probabilistic framework. Furthermore, the output of our framework can be used as unary term in different CRF methods. With this, our probabilistic framework is not in a direct competition with CRF methods, but with typical classification methods like support vector machines and logistic regression.

With the rich and meaningful representation of a pixelwise labeled image, a whole bunch of applications is directly available. In the following work we concentrate on facade recognition, which, for example, allows for automatically generating facade models used for 3D city modeling [14].

1.1 Related work on semantic segmentation and facade recognition

Semantic segmentation is abstract name for all methods trying to label any sort of image pixelwise. The goal is to separate an image into homogeneous areas, where each region represents an instance of one of the trained classes. Due to the high need of computational resources, this research topic got important in the second half of the last decade. Csurka et al. [9] presented a straight forward approach very similar to ours. The main parts are unsupervised segmentation, feature extraction, feature classification and region labeling.

Further approaches focused on improving these local results using conditional Markov random fields (CRF) [15, 19,44]. Yang and Förstner [47] present an approach to label facades using a CRF, in which the unary potentials are computed by applying a random forest classifier. A subsequent work of the same authors [46] improves this method by considering a hierarchical CRF that exploits region segmentations on multiple image scales.

Another way to improve the results is by applying model-based approaches with hard coded prior knowledge. Teboul et al. [36] uses the so-called shape grammars. For this, the authors of [36] propose to use a simple random decision forest (RDF) for an initial result which will be improved by optimizing the labels with respect to a given grammar. In [35] the authors advanced the solving of the optimization problem in speed and accuracy. Normally, model-based methods tend to much better results than non-model-based methods like ours, with the precondition that the analyzed images are in a similar scenario as in the model encoded. Instead of that, classical semantic segmentation approaches like ours are much more flexible and tends also to good results on more general images.

In contrast to all those works, we focus on the essential part of accurately classifying local patches without any contextual knowledge. In our experiments, we show that we are even able to outperform previous CRF approaches. We expect that adding a CRF model to our approach, which is beyond the scope of this paper, would further improve the recognition performance.

1.2 Related work on efficient GP classification

In the last years, a large amount of scientific effort has been spent to develop fast inference techniques for GP regression and classification [24]. Most of them usually rely on conditional independence assumptions [4] with respect to a small set of predefined variables which might either be part of the training dataset [37] or learned during training [33]. A separate branch of methods is based on decomposition techniques, where the original large-scale problem is broken down into a collection of smaller problems. Next to simple Bagging strategies [7], unsupervised kd-trees neglecting label information during clustering were recently proposed [29] for GP regression. As a supervised alternative,

Broderick et al. [3] combined a Bayesian decision tree with GP classifiers. The approach of Urtasun et al. [40] performs GP regression by selecting training examples from a local neighborhood of a test point. The paper also compares the local approach to global ones using a pre-clustering technique. Whereas, their local approach allows reducing boundary effects, our pre-clustering method leads to a logarithmic rather than a linear computation time during learning with respect to the number of training examples.

Another important direction for fast inference with Gaussian process models is Bayesian committee machines as introduced by Tresp et al. [38]. The underlying idea is also to decompose the training set into several subsets and to learn a regressor or classifier for each set independently. However, unlike our approach, each partition is used for classifying test examples. Tresp et al. [38] also study fast GP classification with time-consuming approximate inference techniques instead of relying on GP regression as done in this work. Especially in the context of visual classification tasks it has been shown that, despite its improper noise model, GP regression directly applied to the labels is often sufficient [26].

There is also a large number of related papers concerning large-scale learning with support vector machines (SVM). For example, Tsang et al. [39] improves the core vector machine formulation of SVM by considering enclosing balls of fixed radius and presenting corresponding approximation techniques. In contrast to our approach, they do not focus on speeding up the prediction time necessary to classify a new example. An approach highly related to ours is proposed in Chang et al. [5], where SVM are accelerated using a decomposition derived from a decision tree. In their setting, standard SVMs are employed resulting in a classifier which produces hard decisions. In the context of scene recognition, Fröhlich et al. [13] recently proposed a GP-based method relying on a pre-clustering via random decision forests. However, this approach is solely based on a-posteriori estimates of the predictive distribution, neglecting available uncertainty values.

1.3 Outline of the paper

The remainder of this paper is organized as follows. First of all, we describe the basic principles of the semantic segmentation approach used. Section 3 reviews Gaussian processes for classification tasks and proposes a method to obtain suitable probabilities from the one-vs.-all method of [18]. Our tree-based acceleration technique for inference with Gaussian processes is presented in Sect. 4. We perform experiments for facade recognition applications as a special case of semantic segmentation and evaluate them in Sect. 5.

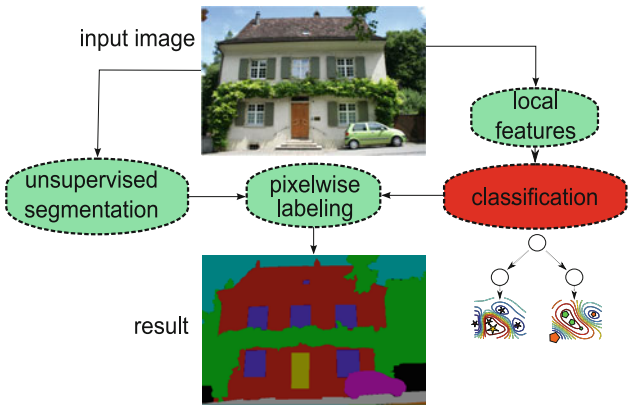


Fig. 2 Overview of semantic segmentation using local features

A summary of our findings and a discussion of future research directions conclude this paper.

2 Semantic segmentation framework

As described above, semantic segmentation is concerned with assigning class labels (or probabilities) to each pixel of a given image. Csurka et al. [9] proposed a simple, but powerful framework for tackling this task. Relying on a bottom-up methodology, their approach combines an initial unsupervised over-segmentation of a given image with pixelwise classification results.

It has been recently shown on an empirical basis [12] that a time-consuming feature transformation step from the original framework [9] can be bypassed. Using a random decision forest, training and prediction time is considerably reduced.

The whole processing pipeline of this approach is depicted in Fig. 2. It mainly includes four steps:

1. *Unsupervised segmentation* an over-segmentation is obtained using an image segmentation algorithm.
2. *Local feature extraction* to capture color and texture information in a local neighborhood, feature descriptors are computed on an equally spaced grid for various scales.
3. *Pixelwise classification* labels are softly assigned to each grid point using a probabilistic classifier. To generate dense probability maps, all grid-based classification results are convolved with a Gaussian filter. The final probability map is generated by averaging all maps obtained for different scales.
4. *Combination of over-segmentation and probability map* one deterministic class label is assigned to each cluster segment by choosing the category with maximum average probability within that region.

A detailed description of our experimental setup can be found in Sect. 5.2.

3 Gaussian process classification

In the following, we briefly review Gaussian process (GP) regression and classification. We concentrate on the main model assumptions and the resulting prediction equation. For a presentation of the full Bayesian treatment, we refer to Rasmussen and Williams [24].

3.1 Basic principles of GP priors

Given n training examples $\mathbf{x}_i \in \mathbb{R}^D$ denoting input feature vectors and corresponding binary labels $y_i \in \{-1, 1\}$, we need to predict the label y_* of an unseen example $\mathbf{x}_*$. Therefore, a learner has to find the intrinsic relationship between inputs $\mathbf{x}$ and labels y. It is often assumed that the desired mapping can be modeled by $y = f(\mathbf{x}) + \varepsilon$, where f is a latent function (which is not observed during training) and ε denotes a noise term.

One common modeling approach is to assume that f belongs to some parametric family and to learn the parameters which best describe the training data. However, the main benefit of the GP framework is its ability to model the underlying function f directly, i.e. without any fixed parameterization, by assuming that f is a sample of a specific distribution. Defining a distribution over functions in a non-parametric manner can be done with a Gaussian process, which is a special stochastic process.

3.2 Bayesian framework for regression and classification with GP

To use the modeling ideas described in the previous section, we formalize and correctly specify the two main modeling assumptions for regression and classification with Gaussian processes:

1. The latent function f is a sample from a GP prior

   $$f \sim \mathcal{GP}(\mathbf{0}, \mathcal{K}(\cdot, \cdot))$$

   with zero mean and covariance or kernel function $\mathcal{K}$:

   $$\mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R} \ .$$

2. Labels y are conditionally independent given latent function values $f(\mathbf{x})$ and are described using some noise model $p(y \mid f(\mathbf{x}))$.

The Gaussian process prior enables to model the correlation between labels using the similarity of inputs, which is described by the kernel function. It is, thus, possible to model the assumption of smoothness, i.e. that similar inputs should lead to similar labels.

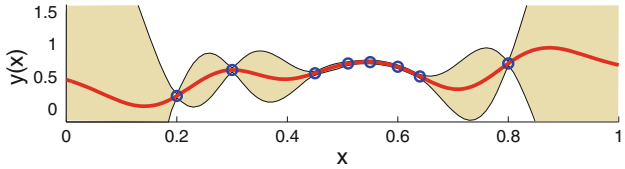


Fig. 3 Gaussian process regression applied to a small one-dimensional example. Training points are shown as *blue circles* and the predictive mean is plotted in *red color*. The *shaded area* highlights the confidence interval derived from the predictive variance (color figure online)

For classification purposes, sigmoid functions are often employed as noise models [24]. In contrast, we follow Kapoor et al. [18] and use zero mean Gaussian noise with variance σ_n^2:

$$p(y \mid f(\mathbf{x})) = \mathcal{N}(y \mid f(\mathbf{x}), \sigma_n^2), \tag{1}$$

which is the standard assumption for GP regression. The advantage of this label regression approach is that tractable predictions for unseen points $\mathbf{x}_*$ are possible, without using approximate inference methods [24].

Let $\mathbf{K}$ be the kernel matrix with pair-wise kernel values of the training examples $\mathbf{K}_{ij} = \mathcal{K}(\mathbf{x}_i, \mathbf{x}_j)$ and $\mathbf{k}_*$ be kernel values $(\mathbf{k}_*)_i = \mathcal{K}(\mathbf{x}_i, \mathbf{x}_*)$ corresponding to test example $\mathbf{x}_*$. The most likely outcome $\bar{y}_*$ given input $\mathbf{x}_*$ and labeled training data can then be predicted analytically using the following equation:

$$\bar{y}_*(\mathbf{x}_*) = \mathbf{k}_*^T (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{y} \ . \tag{2}$$

with $\mathbf{y} \in \{-1, 1\}^n$ being the vector of the binary labels of all training examples. This prediction equation is equivalent to kernel ridge regression, but with a clear probabilistic meaning. For example, the GP framework allows for predicting the standard deviation σ_*^2 of the estimation by:

$$\sigma_*^2(\mathbf{x}_*) = \mathcal{K}(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^T (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{k}_* + \sigma_n^2 \ . \tag{3}$$

Please note that standard support vector machines lack this intrinsic probabilistic formulation and that the associated optimization objective does not give rise to an uncertainty estimate. An example of the result of GP regression is given in Fig. 3.

3.3 Multi-class classification

In the previous section, GP classification is restricted to binary tasks. However, by applying the one-vs.-all strategy in combination with a majority voting scheme, multi-class classification problems can be solved without much additional computational effort [18]. Let $\mathbf{y}^m \in \{-1, 1\}^n$ be the vector of binary labels corresponding to class $m \in \{1, \ldots, M\}$ derived from the multi-class label vector $\mathbf{y}$ by:

$$y_i^m = 2\, \delta(y_i = m) - 1, \tag{4}$$

where $\delta(a) = 1$ if and only if a is true. The final predicted category is the one that achieves the highest predictive posterior mean given by the corresponding binary problem:

$$\bar{y}^{\text{multi}}(\mathbf{x}_*) = \underset{m=1\ldots M}{\text{argmax}}\ \bar{y}^{m*}(\mathbf{x}_*) \tag{5}$$

$$= \underset{m=1\ldots M}{\text{argmax}}\ \mathbf{k}_{*T}\,(\mathbf{K} + \sigma_{n^2}\mathbf{I})^{-1}\mathbf{y}^m. \tag{6}$$

3.4 Probability calibration

Due to additional smoothing applied to the probability maps, the semantic segmentation framework presented in Sect. 2 requires that the classifier predicts benign probabilities for each class. The one-vs.-all approach of [18] only offers a hard classification decision as given in Eq. (6). To derive probability estimates for each class, we could squash the posterior means using a softmax function [24]. However, this strategy completely ignores the uncertainty of the estimate and hides the fact that the one-vs.-all decision is also probabilistic in its nature.

We propose to take the whole posterior distribution

$$\mathcal{N}(\bar{y}^m_*(\mathbf{x}_*), \sigma^2_*(\mathbf{x}_*))$$

of each random variable y^m_* into account, so that the probability of class m achieving the maximum score can be expressed by

$$p(y^{\text{multi}}(\mathbf{x}_*) = m) = p\left(\max_{m'=1\ldots M}\ y^{m'}_* = y^m_*\right). \tag{7}$$

Unfortunately, it does not seem to be possible to derive a closed-form solution for the probability on the right hand side of Eq. (7) for a multi-class scenario with $M > 2$. Therefore, we use a simple Monte-Carlo technique and sample Z times, e.g. $Z = 200$, from all M Gaussian distributions $\mathcal{N}(\bar{y}^m_*(\mathbf{x}_*), \sigma^2_*(\mathbf{x}_*))$ and estimate the probability of each class m by

$$p(y^{\text{multi}}(\mathbf{x}_*) = m) \approx \frac{Z_m}{Z}, \tag{8}$$

with Z_m denoting the number of times where the draw from y^m was the maximum value. A large variance σ^2_*, i.e. a high uncertainty of the estimate, leads to a nearly uniform distribution $p(y^{\text{multi}}(\mathbf{x}_*) = m)$, whereas a zero variance results in a distribution which is equal to one for the class which corresponds to the highest posterior mean.

An alternative would be to directly use a multi-class classification approach with Gaussian processes, but this has to be paid with time-consuming approximation techniques like Laplace approximation [24].

4 Large-scale GP classification with tree-based models

In the following, we show how to speed up learning and classification with our tree-based Gaussian process approach. The main advantage is that we are able to directly handle the trade-off between accuracy and computation time, which allows for using our approach in very different semantic segmentation scenarios with varying requirements.

4.1 Learning

The major shortcoming of GP-based models is their runtime and memory complexity. Since the inversion of $\mathbf{K}$ is required for computing Eqs. (2) and (3) the runtime (the needed memory) scales cubically (quadratically) in the number of training examples n. This fact often renders GP models unsuitable for large-scale problems, where tens or hundreds of thousand training examples are given. To circumvent this problem, many techniques have been proposed to speed-up the inference process using conditional independence assumptions [4], kernel matrix approximation [42] or efficient decomposition of the problem into several sub-tasks [3,29].

For the latter point, deterministic decision trees [2] have been found particularly useful in large-scale classification problems [5,13] due to their ability to efficiently cluster the input space in a supervised manner. Starting by a root node which contains the whole training set, the input space is divided using a simple classifier, e.g. a decision stump [17]. This directly induces a clustering of the training data and new child nodes are associated with the resulting subsets. This procedure is repeated until a stopping criterion is met.

By utilizing the above-mentioned tree decomposition, powerful classifiers such as GP classifiers or SVMs [41] can be trained in each leaf node. The training complexity hence solely depends on the amount of data arriving at the leaves of the tree. For large-scale applications, it is hence necessary to avoid leaf nodes which contain many training examples. As proposed in [5] and [13], this constraint can be directly encoded in the termination criterion of the decision tree. In this approach, leaf nodes exceeding a number ℓ of training examples are only allowed if they are homogeneous, i.e. all training examples share the same label. Since classification is only required on inhomogeneous leaf nodes, the runtime complexity is $\mathcal{O}(\ell^3)$ for each node. It can be shown that the overall runtime complexity hence reduces to $\mathcal{O}(n \log n + n\ell^2)$ (building a tree and additional $\mathcal{O}\left(\frac{n}{\ell}\right)$ Gaussian process classifiers) for the whole training step including the calibration method proposed in Sect. 3.4 (see Table 1). The parameter ℓ thus enables a trade-off between accuracy and efficiency (arriving at the full GP classifier for $\ell = n$).

To avoid overfitting, the standard decision tree can be replaced by a random decision forest (RDF [1]). This

Table 1 Computational complexity of all presented methods

	Training	Classification
GP	$\mathcal{O}(n^3)$	$\mathcal{O}(n^2)^\dagger$
Decision Tree	$\mathcal{O}(n \log n)$	$\mathcal{O}(\log n)$
DT-GP	$\mathcal{O}(n \log n + n\ell^2)$	$\mathcal{O}(\log n + \ell)$
DT-GP CALIBRATED	$\mathcal{O}(n \log n + n\ell^2)$	$\mathcal{O}(\log n + \ell^2)^\dagger$

For the sake of simplicity, balanced trees are assumed
n the number of training examples, ℓ the maximum number of examples in inhomogeneous leaf nodes
† Time includes the calculation of the uncertainty

architecture is based on multiple trees, each of which is trained on a randomly drawn training subset. Moreover, a further randomization can be introduced using a random feature subset for each node. The resulting classifier is known for its high stability with respect to input and label noise [1]. For a complexity assessment regarding the combination of a GP classifier and RDF, we refer to [13].

In the following, we use the acronyms DT-GP and RDF-GP to refer to GP classifiers augmented by decision trees and random decision forests, respectively.

4.2 Prediction

Classifying a new test example with DT-GP is straightforward. The test example first finds its path through the decision tree by checking the decision stumps in each inner node. Finally, the GP classifier associated with the resulting leaf node is evaluated and returns the classification result as well as scores for each category. For RDF-GP, the randomized version of our approach, the scores returned by each tree in the forest are summed up and the class with the highest score is returned as a classification result. In total, this yields an asymptotic runtime of $\mathcal{O}(\log n + \ell^2)$ for each test example (see Table 1).

5 Experiments

The results of the following experiments can be summarized as follows:

1. Tree-based Gaussian process classifiers can outperform previously used machine learning methods for semantic segmentation tasks.
2. The behavior of DT-GP and RDF-GP strongly depends on the amount of label-noise and the intra-class variance of the classification task.
3. Probability calibration can improve the classification results for semantic segmentation.
4. Our method outperforms the approaches of [47] and [46] that exploit structure information of facades with conditional Markov random fields.

5.1 Experimental datasets

For our experiments, we follow [12] and use the eTRIMS [20], LabelMeFacade and the Paris [35] databases. The eTRIMS database contains 60 and LabelMeFacade 945 pixelwise labeled images. The split in 100 training images and 845 testing images for LabelMeFacade is the same as used in [12]. For the eTRIMS dataset, we use the same split proposed by [47] where they use ten different random splits into 40 images for training and 20 for testing. In the Paris dataset, we use the same split as introduced by the authors of [35] in 20 training images and 84 for testing. A detailed description of the experimental setup is presented in the next section.

5.2 Experimental setup

In our framework, we utilize mean shift [8] as unsupervised segmentation method and Opponent-SIFT [27] for extracting local features. For classification, any classifier which can handle the large number of training examples can be used. In our setup, we apply the combined classifiers introduced in Sect. 4 (DT-GP and RDF-GP). For the eTRIMS dataset, we compute local features on five different scales on a 5×5 pixel grid leading to 19,275 training examples and 1,633,240 examples for testing. A higher number of training and test examples (22,976 and 3,140,040) was derived from the LabelMeFacade dataset using a 20×20 pixel grid. This large number of examples for training cannot be handled by a standard Gaussian process classifier, but by a DT-GP classifier. Note that due to the imbalanced nature of the databases, both training sets are restricted to have equal numbers of training examples for each category arriving at above numbers.

For RDF learning, we use the following settings. At each node, the data are split by decision stumps optimized by employing the mutual information criterion. The maximal depth of each tree is 10 and the number of trees is 5. As shown in [13], the choice of the parameter highly depends on the desired recognition performance and computational speed. As a trade-off between both criteria, we are using 500 examples as the minimum number of examples in each leaf.

For evaluation, we use two different performance measures. Whereas the overall recognition rate denotes the fraction of correctly classified results, the average recognition rate computes the mean of all class-specific recognition rates such that all categories have the same impact on the performance.

Table 2 Recognition rates of our experiments with different classifiers in comparison to previous work

Dataset	Approach	Average recognition rate	Overall recognition rate
eTRIMS	Yang and Förstner [47] (CRF)	49.75 %	65.80 %
	Yang and Förstner [46] (HCRF)	61.63 %	69.00 %
	RDF [12]	63.68 % (±1.25)	68.86 % (±1.36)
	SLR [12]	65.57 % (±2.47)	71.18 % (±2.69)
	DT-GP	72.13 % (±0.65)	74.96 % (±0.25)
	DT-GP CALIBRATED	**72.36 %** (±0.55)	**75.05 %** (±0.35)
	RDF-GP	67.88 % (±2.19)	65.95 % (±1.08)
	RDF-GP CALIBRATED	66.71 % (±0.35)	63.59 % (±0.53)
LabelMeF	RDF [12]	44.08 % (±0.45)	49.06 % (±0.52)
	SLR [12]	42.81 % (±0.89)	48.46 % (±1.58)
	DT-GP	43.52 % (±1.04)	42.63 % (±1.02)
	DT-GP CALIBRATED	41.86 % (±1.34)	43.52 % (±2.10)
	RDF-GP	**51.47 %** (±0.09)	40.32 % (±0.09)
	RDF-GP CALIBRATED	51.11 % (±0.09)	**51.10** % (±1.13)
Paris	Teboul et al. [34] (RDF)	55.00 %	52.57 %
	Teboul et al. [34] (grammar-based)	77.00 %	82.14 %
	Teboul et al. [35] (grammar-based)	84.14 %	84.21 %
	DT-GP	58.38 %(±0.56)	62.25 % (±0.88)
	DT-GP CALIBRATED	57.68 % (±0.77)	61.86 % (±0.81)
	RDF-GP	**63.20 %**(±0.20)	**66.44 %** (±0.42)
	RDF-GP CALIBRATED	62.25 % (±0.04)	65.86 % (±0.07)

In contrast to [12], we used random splits of training and testing for the eTRIMS dataset to allow for fair comparison with [47] and [46]

The bold values represent the best results without contextual knowledge

Our approach is compared to the methods of [35,46,47] and standard classifiers, like sparse logistic regression and random decision forests [12]. Note that we do not use any conditional random field models or any other method used to incorporate local context information as done in several other related work [10,16,23]. However, we believe that those methods would benefit from integrating the output of our non-linear classifier as an unary term respectively as initialization for a grammar model [34–36]. A comparison with standard GP without tree decomposition was done in [13] and it turned out that the performance is comparable. For semantic segmentation, this comparison is not possible, due to the large number of training examples.

5.3 Results and evaluation

The results of the experiments are listed in Table 2. Along with DT-GP, we give an overview of the results from [12] for the sparse logistic regression (SLR) and the random decision forest (RDF) for the eTRIMs and the LabelMeFacade dataset and an overview of the results from [34,35] for the Paris dataset.

In our experiment, the runtimes for the DT-GP (7.8 s per image) and for the RDF-GP with five trees (19.14 s per image) were longer as for the simple RDF (2.10 s per image). But this fits very well to our expectation and the experimental evaluation results from [13]. Please note that, as mentioned above, it is not possible to apply standard Gaussian processes to this high amount of features on current hardware.

For the eTRIMS database, the DT-GP classifier clearly achieves a higher average recognition rate compared to SLR, RDF, and RDF-GP. On the LabelMeFacade dataset, RDF-GP leads to the best recognition rates while the deterministic variant DT-GP does not improve upon state-of-the-art classifiers used in [12].

This leads us to the question why DT-GP and RDF-GP exhibit such an opposite behavior in their recognition performance. By taking a closer look on the data, the following can be noticed: there are severe differences in the amount of label-noise between both datasets. Whereas the eTRIMS database was manually labeled with care focusing on consistency, the LabelMeFacade dataset was derived by combining annotations of several non-experts [12] who often missed to label several important parts of a facade. As stated by [1] random decision forests avoid overfitting and are thus robust to the shortcomings of the LabelMeFacade dataset, which contains high label-noise. The DT-GP uses all available information in the training data in a deterministic manner to build a supervised pre-clustering.

Since eTRIMS provides nearly perfect ground-truth data, a suitable partitioning of the feature space can be successfully estimated.

In the Paris dataset, our approaches outperform the basic randomized decision forest approach from [34] significantly. However, the shape grammars from Teboul et al. [34,35] tend to significantly better results than the Gaussian Process approaches. This is of course due to the hard coded information in the shape grammar, which also could be used to improve our results, but this is not the focus of current paper.

The numbers in Table 2 also validate the third hypothesis, i.e. semantic segmentation with the LabelMeFacade dataset benefits from the soft decision calculated by the method

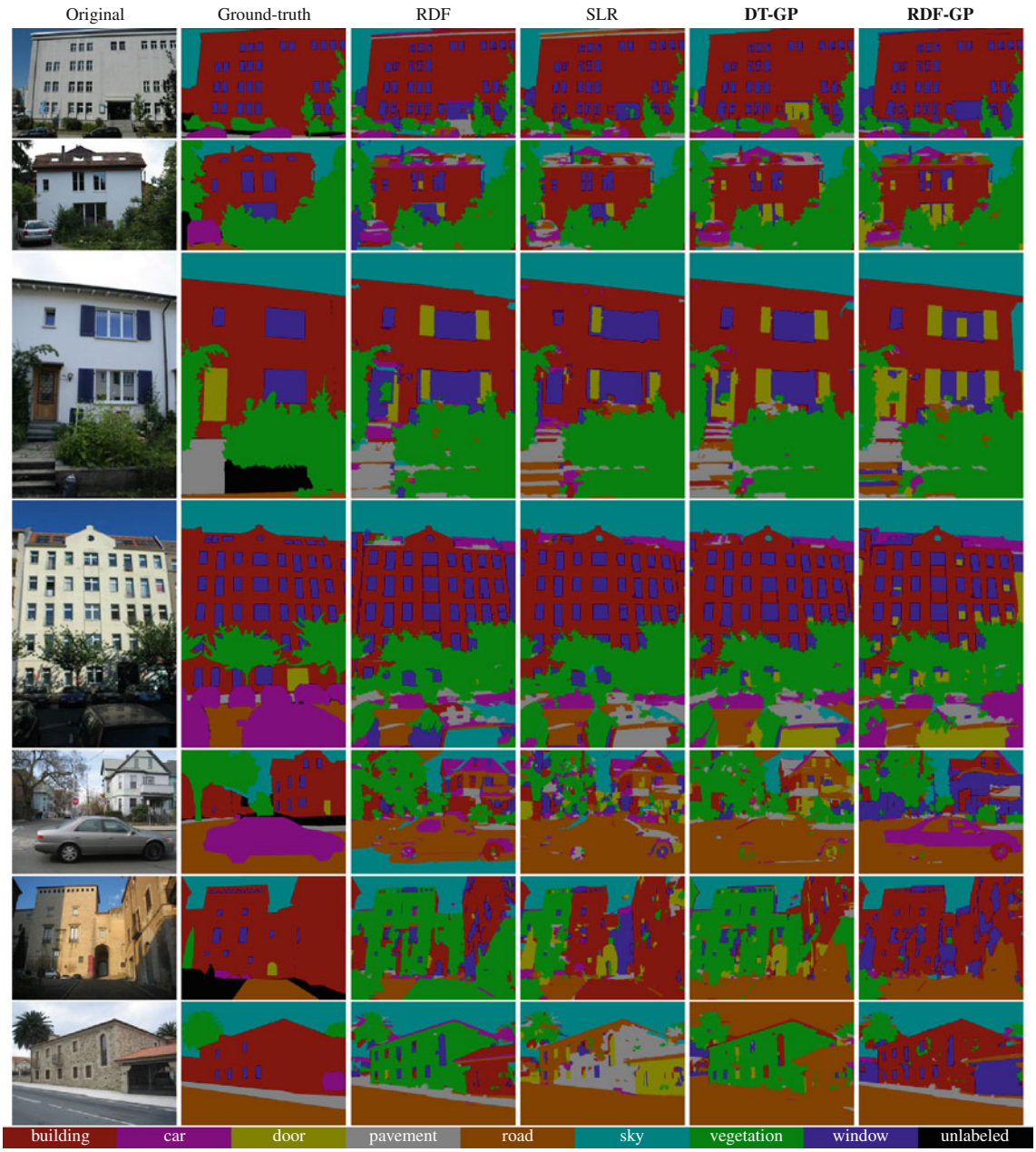
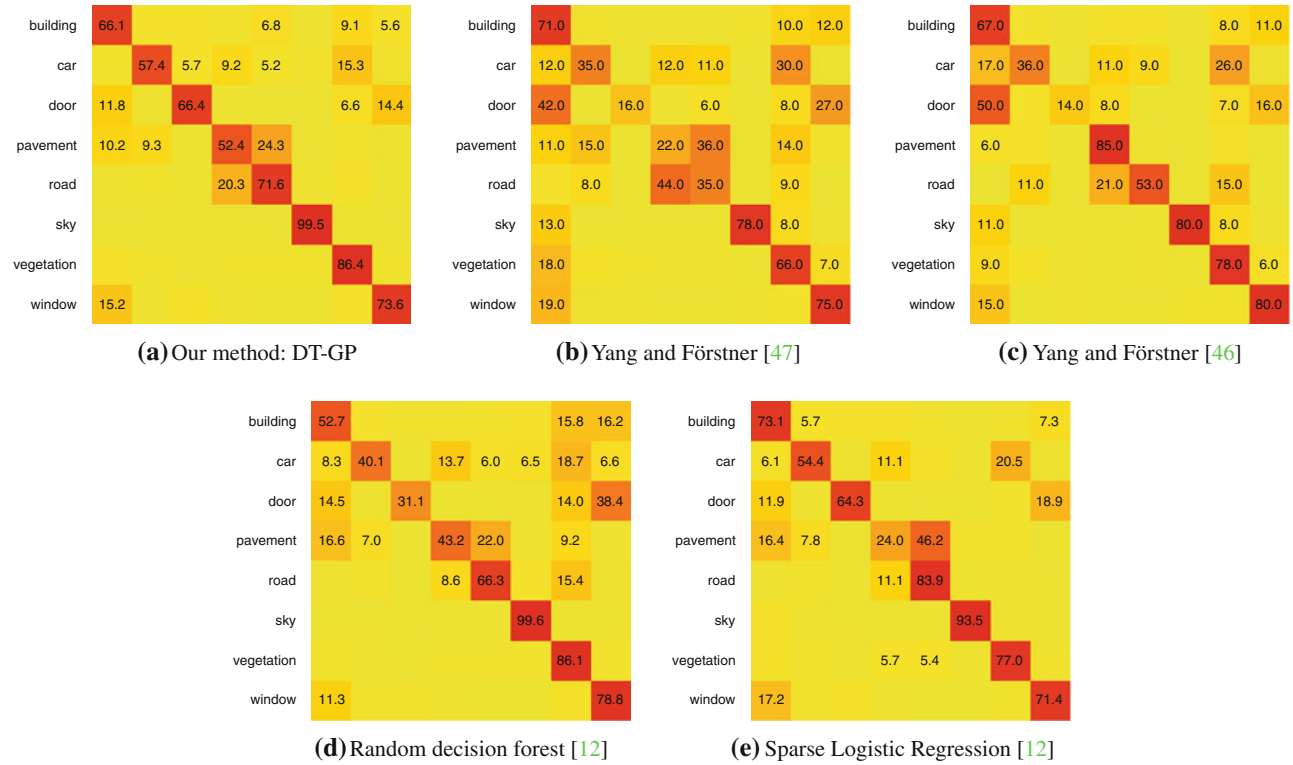


Fig. 4 Example images from eTRIMS (*first four rows*) and LabelMeFacade database (*last three row*) and corresponding results obtained by random decision forest (RDF) [12], sparse logistic regression (SLR) [12], decision trees augmented by Gaussian processes (DT-GP), and random decision forest augmented by Gaussian processes (RDF-GP). DT-GP and RDF-GP/LabelMe use the proposed probability calibration. Note the correct recognition of the door in the first row by DT-GP which was not labeled in the ground-truth data. Furthermore, the results shown in *row five* demonstrate the disadvantages of our completely local classifier in complex scenes



(a) Our method: DT-GP **(b)** Yang and Förstner [47] **(c)** Yang and Förstner [46]

(d) Random decision forest [12] **(e)** Sparse Logistic Regression [12]

Fig. 5 Average confusion matrices achieved by our methods and the approaches of [47] and [46] on the eTRIMS dataset. Values are only displayed above 5 %

presented in Sect. 3.4. While there is no significant difference on the eTRIMS dataset, a clear performance boost is apparent for the LabelMeFacade database, where the overall recognition rate is increased by 10.68 % from uncalibrated to calibrated RDF-GP.

Another interesting observation, which can be seen in the results of the eTRIMS dataset, is that we outperform the approaches of [47] and [46], which exploit the structure of facades by utilizing a (hierarchical) conditional Markov random field. Those techniques can also be used to enhance our results, however, they are beyond the scope of the current paper.

Unfortunately, in some cases pure recognition rates do not allow us to make sufficient statements about visual quality of the resulting segmentations. For this purpose, Fig. 4 contains a few images from both datasets along with their ground-truth data and the resulting segmentations calculated by [12] and our approach. Figure 5 shows the result for the same training and testing split of the different methods as confusion matrices. It can be seen that especially the discrimination between door and window benefits from the incorporation of the DT-GP method into the semantic segmentation framework. The matrices also highlight cases that are still difficult to differentiate, such as pavement and road or window and building.

6 Conclusions and further work

In this work, we presented an approach to semantic segmentation that allows accurate prediction for very large datasets. Our method employs a fast Gaussian process (GP)-based classifier which relies on a pre-clustering of the input space using decision trees. We additionally proposed a fast method for generating probabilistic outputs in the multi-class setting without resorting to costly inference methods [24]. We validated our approach on different challenging facade image datasets and compared it to existing work. The results clearly show that a significant performance boost is achieved using our tree-based GP framework. Furthermore, our probability calculation method can provide an additional performance benefit.

Semantic segmentation with a predefined list of categories is in general ill-posed in its nature, since there are always some regions in the image which belong to unknown categories or where no decision can be made even by human annotators. One idea for further work is to use the estimation uncertainty given by the GP classifier to mark such image areas or to identify outliers in the training data using leave-one-out estimates [24].

Another direction for future research is semi-supervised methods, which use all information available in only partially

annotated images. In our case, the combination of semi-supervised extensions of random forests [22] and Gaussian process classifiers [21] seems to be promising.

Facade recognition clearly benefits from additional prior knowledge, such as periodicity and typical structure. Our approach is completely local and it would be interesting to model the dependencies with conditional Markov random fields [10,16,23,46] and do inference based on our estimated probability maps. However, facades have a structure that cannot be completely modeled with standard CRF models, which are mostly restricted to pair-wise dependencies between pixels or regions. A solution would be to use grammar techniques [25,32,35,36] or to incorporate topological constraints using the minimal perturbation idea of [6].

Apart from facade recognition, we are planning to evaluate our methods on remote sensing data and common datasets of semantic segmentation like MSRC21 and Pascal VOC [9].

Acknowledgments This work was partially supported by the Graduate School on Image Processing and Image Interpretation funded by the state of Thuringia/Germany.

References

1. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
2. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: Classification and Regression Trees. Chapman and Hall, London (1984)
3. Broderick, T., Gramacy, R.B.: Treed gaussian process models for classification. In: Classification as a Tool for Research, Studies in Classification, Data Analysis and Knowledge Organization, pp. 101–108 (2010)
4. Candela, Q.J., Rasmussen, C.E.: A unifying view of sparse approximate gaussian process regression. J. Mach. Learn. Res. **6**, 1939–1959 (2005)
5. Chang, F., Guo, C.Y., Lin, X.R., Lu, C.J.: Tree decomposition for large-scale SVM problems. J. Mach. Learn. Res. **11**, 2935–2972 (2010)
6. Chen, C., Freedman, D., Lampert, C.: Enforcing topological constraints in random field image segmentation. In: Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'11) (2011)
7. Chen, T., Ren, J.: Bagging for gaussian process regression. Neurocomputing **72**(7–9), 1605–1610 (2009)
8. Comaniciu, D., Meer, P.: Mean shift: a robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Mach. Intell. **24**(5), 603–619 (2002)
9. Csurka, G., Perronnin, F.: An efficient approach to semantic segmentation. IJCV **95**(2), 198–212 (2011)
10. Domke, J.: Crossover random fields. J. Mach. Learn. Res. (2009)
11. Dumont, M., Marée, R., Wehenkel, L., Geurts, P.: Fast multi-class image annotation with random subwindows and multiple output randomized trees. In: Proceedings of the 4th International Conference on Computer Vision, Theory and Applications (VISAPP), vol. 2, pp. 196–203 (2009)
12. Fröhlich, B., Rodner, E., Denzler, J.: A fast approach for pixelwise labeling of facade images. In: Proceedings of the International Conference on Pattern Recognition (ICPR'10), pp. 3029–3032 (2010)
13. Fröhlich, B., Rodner, E., Kemmler, M., Denzler, J.: Efficient gaussian process classification using random decision forests. Pattern Recogn. Image Anal. **21**, 184–187 (2011)
14. Gool, L.J.V., Zeng, G., den Borre, F.V., Müller, P.: Towards mass-produced building models. In: Photogrammetric Image Analysis, pp. 209–220 (2007)
15. Gould, S., Rodgers, J., Cohen, D., Elidan, G., Koller, D.: Multi-class segmentation with relative location prior. Int. J. Comput. Vis. **80**(3), 300–316 (2008). doi:10.1007/s11263-008-0140-x
16. Huang, Q.X., Han, M., Wu, B., Ioffe, S.: A hierarchical conditional random field model for labeling and segmenting images of street scenes. In: Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1953–1960. IEEE, New York (2011)
17. Iba, W., Langley, P.: Induction of one-level decision trees. In: Proceedings of the International Conference of Machine Learning (ICML'92) (1992)
18. Kapoor, A., Grauman, K., Urtasun, R., Darrell, T.: Gaussian processes for object categorization. Int. J. Comput. Vis. **88**(2), 169–188 (2010)
19. Kohli, P., Ladicky, L., Torr, P.: Robust higher order potentials for enforcing label consistency. In: IEEE Conference on Computer Vision and Pattern Recognition, 2008. CVPR 2008, pp. 1–8 (2008). doi:10.1109/CVPR.2008.4587417
20. Korč, F., Förstner, W.: etrims image database for interpreting images of man-made scenes. Technical report, Department of Photography, University of Bonn (2009). http://www.ipb.uni-bonn.de/projects/etrims_db/
21. Lawrence, N.D., Jordan, M.I.: Semi-supervised learning via gaussian processes. In: Advances in Neural Information Processing Systems, pp. 753–760 (2005)
22. Leistner, C., Saffari, A., Santner, J., Bischof, H.: Semi-supervised random forests. In: Proceedings of the 2009 International Conference on Computer Vision (ICCV'09), pp. 506–513 (2009)
23. Rabinovich, A., Vedaldi, A., Galleguillos, C., Wiewiora, E., Belongie, S.: Objects in context. In: Proceedings of the 2007 International Conference on Computer Vision (ICCV'07), pp. 1–8 (2007)
24. Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning). MIT Press, Cambridge (2005)
25. Ripperda, N., Brenner, C.: Evaluation of structure recognition using labelled facade images. In: Proceedings of the DAGM, pp. 532–541 (2009)
26. Rodner, E., Hegazy, D., Denzler, J.: Multiple kernel gaussian process classification for generic 3d object recognition from time-of-flight images. In: Proceedings of the International Conference on Image and Vision Computing (2010)
27. van de Sande, K., Gevers, T., Snoek, C.: Evaluating color descriptors for object and scene recognition. PAMI **32**, 1582–1596 (2010)
28. Schölkopf, B., Smola, A.: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, Cambridge (2001)
29. Shen, Y., Ng, A., Seeger, M.: Fast gaussian process regression using kd-trees. In. Advances in Neural Information Processing Systems, pp. 1225–1232 (2006)
30. Shotton, J., Johnson, M., Cipolla, R.: Semantic texton forests for image categorization and segmentation. In: Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'08), pp. 1–8 (2008)
31. Shotton, J., Winn, J.M., Rother, C., Criminisi, A.: Textonboost: joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: Proceedings of the European Conference of Computer Vision (ECCV'06), pp. 1–15 (2006)
32. Simon, L., Teboul, O., Koutsourakis, P., Paragios, N.: Random exploration of the procedural space for single-view 3d modeling of buildings. Int. J. Comput. Vis. **93**, 253–271 (2011)
33. Snelson, E., Ghahramani, Z.: Sparse gaussian processes using pseudo-inputs. In: Advances in Neural Information Processing Systems (2006)

34. Teboul, O.: Shape Grammar Parsing: Application to Image-Based Modeling. PhD thesis, Ecole Centrale de Paris (2011)
35. Teboul, O., Kokkinos, I., Koutsourakis, P., Simon, L., Paragios, N.: Shape grammar parsing via reinforcement learning. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2313–2319 (2011)
36. Teboul, O., Simon, L., Koutsourakis, P., Paragios, N.: Segmentation of building facades using procedural shape priors. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2010)
37. Tipping, M.E.: Sparse bayesian learning and the relevance vector machine. J. Mach. Learn. Res. **1**, 211–244 (2001)
38. Tresp, V.: A bayesian committee machine. Neural Comput. **12**, 2719–2741 (2000)
39. Tsang, I.W., Kocsor, A., Kwok, J.T.: Simpler core vector machines with enclosing balls. In: Proceedings of the 24th international conference on Machine learning, pp. 911–918 (2007)
40. Urtasun, R., Darrell, T.: Sparse probabilistic regression for activity-independent human pose inference. In: Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'08) (2008)
41. Vapnik, V.N.: The Nature of Statistical Learning Theory. Springer, Berlin (1995)
42. Williams, C.K., Seeger, M.: Using the nyström method to speed up kernel machines. In: Advances in Neural Information Processing Systems, pp. 682–688 (2001)
43. Xiao, J., Fang, T., Zhao, P., Lhuillier, M., Quan, L.: Image-based street-side city modeling. ACM Trans. Graph. **28**(5) (2009)
44. Xiao, J., Hays, J., Ehinger, K., Oliva, A., Torralba, A.: Sun database: Large-scale scene recognition from abbey to zoo. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3485–3492 (2010). doi:10.1109/CVPR.2010.5539970
45. Xiao, J., Quan, L.: Multiple view semantic segmentation for street view images. In: Proceedings of 12th IEEE International Conference on Computer Vision, pp. 686–693 (2009)
46. Yang, M.Y., Forstner, W.: A hierarchical conditional random field model for labeling and classifying images of man-made scenes. In: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 196–203 (2011). doi:10.1109/ICCVW.2011.6130243
47. Yang, M.Y., Förstner, W.: Regionwise classification of building facade images. In: Photogrammetric Image Analysis. Lecture Notes in Computer Science vol. 6952, pp. 209–220. Springer, Berlin (2011)

Author Biographies

Björn Fröhlich earned the Diploma degree in Computer Science from the Friedrich Schiller University of Jena in the year 2009. He is currently a holder of a scholarship in the Graduate School on Image Processing and Image Interpretation from the Free State of Thuringia (Germany) and a Ph.D. student at the Chair of Computer Vision, Institute of Computer Science, Friedrich Schiller University in Jena. His research interests are focused on object recognition and image segmentation.

Erik Rodner earned the Diploma degree in Computer Science with honours in 2007 from the University of Jena, Germany. He received his Ph.D. in 2011 with summa cum laude for his work on learning with few examples, which was done under supervision of Joachim Denzler at the Computer Vision Group of the University of Jena. Erik is currently continuing his research as a postdoctoral researcher at UC Berkeley and the International Computer Science Institute. His research interests include kernel methods, visual object discovery, domain adaptation, scene understanding, as well as exploring every National and State park in California.

Michael Kemmler received the Diploma degree in Computer Science with honors in 2009 from the Friedrich Schiller University of Jena, Germany. As a Ph.D. student at the Jena Graduate School for Microbial Communication, he pursued his studies under the supervision of Joachim Denzler at the Computer Vision Group of the University of Jena. His research interests are in the area of machine learning, object recognition and bioinformatics, including kernel methods, visual image and scene classification as well as bacterial classification.

Joachim Denzler earned the degrees "Diplom-Informatiker", "Dr.-Ing.," and "Habilitation" from the University of Erlangen in the years 1992, 1997 and 2003, respectively. Currently, he holds a position as full time professor for Computer Science and is head of the Chair for Computer Vision, Faculty of Mathematics and Informatics, Friedrich Schiller University of Jena. His research interests comprise active computer vision, object recognition and tracking, 3D reconstruction, and plenoptic modeling, as well as computer vision for autonomous systems. He is author and coauthor of over 90 journal papers and technical articles. He is a member of the IEEE, IEEE Computer Society, DAGM, and GI. For his work on object tracking, plenoptic modeling, and active object recognition and state estimation, he was awarded the DAGM best paper awards in 1996, 1999 and 2001, respectively.