A Fast Approach for Pixelwise Labeling of Facade Images

Björn Fröhlich, Erik Rodner and Joachim Denzler
Chair for Computer Vision, Friedrich-Schiller University of Jena
{bjoern.froehlich,erik.rodner,joachim.denzler}@uni-jena.de
http://www.inf-cv.uni-jena.de

Abstract

Facade classification is an important subtask for automatically building large 3d city models. In the following we present an approach for pixelwise labeling of facade images using an efficient Randomized Decision Forest classifier and robust local color features. Experiments are performed with a popular facade dataset and a new demanding dataset of pixelwise labeled images from the LabelMe project. Our method achieves high recognition rates and is significantly faster for training and testing than other methods based on costly feature transformation techniques.

1. Introduction

Visual facade classification is the task of estimating the position and size of various structural (e.g. window, door) and non-structural elements (e.g. sky, road, building) in a given image of a building or street scene. This recognition task has gained interest in the last years [6], which is mainly due to the growing need to store the appearance of buildings in large 3d city models [6]. For example, an efficient representation of already labeled images with a grammar based compression scheme [11] allows to reduce each facade image to few parameters. Furthermore, by incorporating a large amount of prior knowledge, the recognition of facade elements also allows to estimate the rough 3d structure of buildings [6].

Previous works on facade classification mostly regard the facade classification problem as multiple object detection tasks [7] resulting in bounding boxes of some structural elements. In contrast, we try to estimate the category label of each pixel, which is often called semantic segmentation [13]. To the best of our knowledge, this paper presents the first approach to estimate such a dense and detailed description of facade images.

Our algorithm is based on classifying local color features as proposed by [3]. In contrast to [3], we show how to use the concept of a *Randomized Decision Forest* (RDF) [1] to significantly speed up the learning and recognition process. Due to the high performance of this discriminative classifier, we can skip computationally intensive transformations of local features such as bag-of-features or fisher kernel estimation. The use of a RDF for semantic segmentation was previously investigated [4, 13] These approaches utilize simple color histogram features or pixel differences and do not use current local descriptors which provide invariance properties to illumination changes [14].

We perform experiments on the small eTRIMS dataset [8], which is especially designed for facade classification, and a large dataset obtained from the LabelMe project [12].

The remainder of the paper is organized as follows. First we explain the pipeline of a semantic segmentation approach as proposed by [3]. A brief review of RDF is given in Sect. 3. The *Sparse Logistic Regression* classifier (SLR), which was used to compare the performance of our RDF approach, is presented in Sect. 4. Experiments in Sect. 5 show the detailed results of our approach. A summary of our findings and a discussion of future research directions conclude this paper.

2. Pixelwise Labeling with Local Features

Our approach to semantic segmentation is based on the framework of Csurka et al. [3] but omits a costly feature transformation by using an efficient RDF classifier. An overview of all involved steps can be found in Fig. 1. First of all local features $\mathbf{x}_i \in \mathbb{R}^n$ are computed for each image by dense sampling of feature points with a pixel spacing of τ pixels.

Each of these features is classified independently, which results in a sparse probability map for each category. Afterwards the probability at a single pixel is computed by smoothing the sparse probability map with a Gaussian filter. For each position there is more than

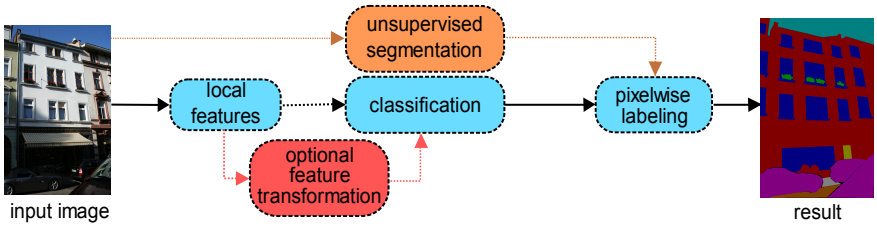

Figure 1. Overview of semantic segmentation using local features as proposed by [3].

one feature available which differs in the used scale. Therefore probability maps are computed for each level with varying standard deviation of the Gaussian filter. To get the final result for each category all probability maps are averaged.

To integrate the result of unsupervised segmentation techniques, such as mean shift [2], we follow [3] and label each region with the category corresponding to the maximum average probability.

The selection of an appropriate local descriptor plays a fundamental role. For our experiments we used the Opponent-SIFT descriptor, which was one of the best color descriptors evaluated in [14]. Additionally, we tested some other descriptors like color moments, which generally results in lower recognition rates.

3. Randomized Decision Forest

A *Randomized Decision Forest* is a discriminative classifier that can handle a large set of features without issues due to the curse of dimensionality. Standard decision tree approaches suffer from severe over-fitting problems. A RDF overcomes these problems by generating an ensemble (forest) of T decision trees (e.g. $T = 5$). During the classification, the overall probability of a class κ given a feature vector $\mathbf{x}_i$ can be obtained by simple averaging of the posterior probabilities $p_\tau(\cdot)$ estimated by each tree of the ensemble:

$$p(y_i = \kappa \mid \mathbf{x}_i) = \frac{1}{T} \sum_{\tau=1}^{T} p_\tau(y_i = \kappa \mid \mathbf{x}_i) \ . \quad (1)$$

In contrast to Boosting, the RDF approach uses two types of randomization to learn the ensemble. The first type of randomization is Bootstrap Aggregating [1], where each tree is trained with a random fraction of the training data. Additionally, to reduce training time and to incorporate randomization into the building process of a tree, the search for the most informative split function in each inner node is done using only a random fraction of all features [5].

4. Sparse Logistic Regression Classifier

In addition to a RDF, we also tested a logistic regression approach to compare its performance. Instead of using a multinomial logistic regression classifier, we follow [3] by applying the idea of a one-vs-all technique to a binary logistic regression classifier:

$$p(\tilde{y}_i = 1 | \mathbf{x}_i, \mathbf{w}) = \left(1 + \exp(-\mathbf{w}^T \mathbf{x}_i)\right)^{-1} \ , \quad (2)$$

where $\mathbf{w}$ is the weight vector and $\tilde{y}_i$ the corresponding label of a single binary classification problem. The weight vector $\mathbf{w}$ is estimated in the training step for each binary subproblem by maximizing the sum of the logarithmic likelihoods [9]. Additionally a Gaussian prior is used for $\mathbf{w}$, which speeds up the estimation and improves numerical behavior . [3] uses an adaption to benefit from the sparsity of the transformed features. This extension is called *Sparse Logistic Regression* classifier.

Additional Feature Transformation Csurka et al. [3] propose to transform feature vectors by a subsequent transformation technique, similar to the bag-of-features idea for a single local feature.

Therefore a Gaussian mixture model (GMM) with 1024 Gaussians is estimated and the final feature vector consists of the soft votes of each Gaussian given a local feature. For speeding up the estimation of the GMM we apply PCA to reduce the 384-dimensional local features to a dimension of 100.

5. Experiments

We experimentally evaluated our approach to illustrate the advantages and disadvantages of all involved steps. In the following we empirically validate the following hypotheses: (1) RDF and SLR with additional feature transformation have comparable performances. (2) RDF do not benefit from additional feature transformation. (3) The dataset obtained from LabelMe images is much more difficult than the eTRIMS dataset

and might be better suited for further experimental evaluation in the field of facade image classification.

We use two different performance measures: the *overall recognition rate* is the percentage of all correctly classified pixels and the *average recognition rate* is the average of the recognition rate for each class. All recognition rates are averages of tests with 10 different random splits of the data into training and test sets. For evaluating computation times we use a Intel®Core™2 Duo CPU 6600 with 2.4GHz.

5.1 Datasets

eTRIMS The eTRIMS database [8] contains 60 pixelwise labeled images with eight different classes. These classes are typical objects which can appear in images of house facades (buildings, cars, doors, pavements, roads, sky, vegetation, windows). There is no given split into training and test data. Therefore we use 40 randomly selected images for training and 20 for testing the algorithms. We use $\tau = 20$ for training and $\tau = 5$ for testing due to the small size of the test set.

LabelMeFacade[1] Due to the small number of images available in the eTRIMS database, we generated a similar database using LabelMe [12] which contains a huge number of images with labeled polygons. We extracted images which contain buildings, windows, sky and a limited number of unlabeled regions (maximally 20% covering of the image). This resulted in 945 images. The pixelwise labeled images are created by utilizing the eTRIMS categories and a simple depth order heuristic. We split this dataset into 100 images for training and 845 images for testing and use for both $\tau = 20$.

6. Evaluation

For testing our introduced algorithms we use three different combinations of training and test datasets, which are listed in Tab. 1. The parameters of the SLR and the Gaussian filters for estimating the dense probability maps are optimized using the eTRIMS test set, which we use as a validation set for testing our approach with the LabelMeFacade dataset.

First of all we tested the following combinations: RDF or SLR classifier with or without additional feature transformation. The first four rows of Tab. 1 demonstrate that the RDF classifier gives better results without using transformed features. As opposed to this property, SLR benefits from additional transformation. This might be due to a higher dimensional feature space which is easier to separate by a linear classifier. Especially the high run time of SLR without sparse features estimated using a GMM makes this variant impractical for large data sets. On the contrary the RDF uses a random fraction of all features and is thus not able to handle sparse feature vectors.

As shown in Tab. 1 the results of the best two combinations are comparable, but the training and test computation time of the GMM and the SLR classifier is much higher than the run time of the RDF approach. In Tab. 1 we show the computation times of the classifiers without local feature computation, estimation of the GMM and feature transformation. Fig. 2 presents some result images of our RDF approach.

The recognition rates using the LabelMeFacade dataset are in general lower than the recognition rates for the eTRIMS dataset. For the LabelMe dataset the average recognition rates using the LabelMe or the eTRIMS training set are comparable, but the overall recognition rate differs a lot. This might be due to the different appearance of the images in the eTRIMS and the LabelMe dataset. All eTRIMS pictures cover the whole facade of exactly one house in each image. The LabelMe dataset contains street scenes with many houses, different viewpoints and more complex situations and classes like pavement, road and vegetation are more prevalent than in the eTRIMS dataset.

Available publications using the eTRIMS database concentrate only on deriving an efficient description of already labeled images [11] or detection of single structural elements (e.g. windows). Therefore, it is not possible to directly compare with others.

7. Conclusions and Further Work

In this paper we demonstrated an approach to pixelwise labeling of facade images based on a *Randomized Decision Forest* and color based local features. In our experiments we have shown that this approach tends to similar results in comparison with [3] but with significantly less time used for the learning process (≈ 5 times faster) and for classification (≈ 12 times faster with a large number of local features). For evaluation we introduced a new database based on LabelMe [12].

We also tested a Markov random field approach similar to the one presented by [10] but we were not able to increase the recognition performance significantly. Therefore it would be interesting to develop special models explicitly incorporating properties of facade images like symmetry and periodicity.

[1] This database is publicly available at `http://www.inf-cv.uni-jena.de/labelmefacade`.

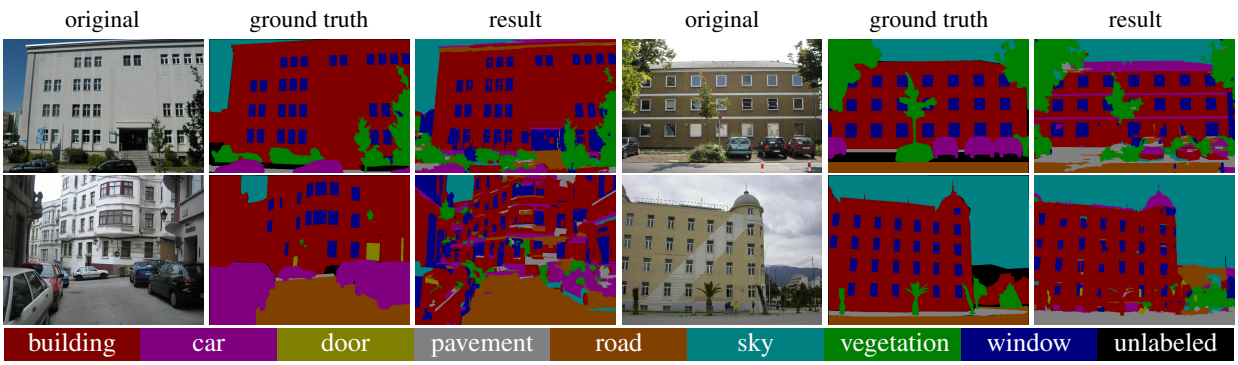

Figure 2. Example images of the eTRIMS (first row) and LabelMeFacade database (second row) and results of our approach based on a Randomized Decision Forest.

Table 1. Recognition rates and computation times of our experiments with different classifiers and feature transformations. (*) Training time measured without the estimation of the GMM, (‡) testing time for each image including additional feature transformation

training/ test set	feat. transform.	classifier	average rec. rate	overall rec. rate	train-t^*	test-$t^‡$
eTRIMS/ eTRIMS	none	RDF	63.68% (±1.25)	**68.86%** (±1.36)	**1m 17s**	**11.8s**
	PCA+GMM	SLR [3]	**65.51%** (±1.34)	68.72% (±1.40)	5m 57s	2m 27s
	PCA+GMM	RDF	44.07% (±1.30)	32.33% (±1.74)	2m 38s	1m 20s
	none	SLR	55.81% (±2.38)	66.85% (±0.77)	∼ 5h	1m 21s
LabelMeF/ LabelMeF	none	RDF	**44.08%** (±0.45)	**49.06%** (±0.52)	**2m 59s**	**5.2s**
	PCA+GMM	SLR [3]	42.81% (± 0.89)	48.46% (±1.58)	4m 45s	11.4s
eTRIMS/ LabelMeF	none	RDF	**43.95%** (±0.35)	39.45% (±0.64)	**1m 17s**	**5.2s**
	PCA+GMM	SLR [3]	41.11% (±1.04)	**40.08%** (±1.43)	5m 57s	11.4s

References

[1] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[2] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *PAMI*, 24(5):603–619, 2002.

[3] G. Csurka and F. Perronnin. A simple high performance approach to semantic segmentation. In *BMVC*, pages 213–222, 2008.

[4] M. Dumont, R. Marée, L. Wehenkel, and P. Geurts. Fast multi-class image annotation with random subwindows and multiple output randomized trees. In *VISAPP*, volume 2, pages 196–203, 2009.

[5] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Maching Learning*, 63(1):3–42, 2006.

[6] L. J. V. Gool, G. Zeng, F. V. den Borre, and P. Müller. Towards mass-produced building models. In *Photogrammetric Image Analysis*, pages 209–220, 2007.

[7] J.-E. Haugeard, S. Philipp-Foliguet, F. Precioso, and J. Lebrun. Extraction of windows in facade using kernel on graph of contours. In *SCIA*, pages 646–656, 2009.

[8] F. Korč and W. Förstner. etrims image database for interpreting images of man-made scenes. Technical report, Dept. of Photogr., University of Bonn, 2009.

[9] B. Krishnapuram and A. J. Hartemink. Sparse multinomial logistic regression: Fast algorithms and generalization bounds. *PAMI*, 27(6):957–968, 2005.

[10] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. In *ICCV*, pages 1–8, 2007.

[11] N. Ripperda and C. Brenner. Evaluation of structure recognition using labelled facade images. In *DAGM*, pages 532–541, 2009.

[12] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: A database and web-based tool for image annotation. *IJCV*, 77(1-3):157–173, 2008.

[13] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *CVPR*, pages 1–8, 2008.

[14] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. *PAMI*, (in press), 2010.