EVALUATION OF TEXTURE ENERGIES FOR CLASSIFICATION OF FACADE IMAGES

Martin Drauschke, Helmut Mayer

Institute for Applied Computer Science, Bundeswehr University Munich, Neubiberg, Germany
martin.drauschke@unibw.de, helmut.mayer@unibw.de

Commission III

KEY WORDS: Terrestrial Building Images, Feature Extraction, Texture Analysis, Classification

ABSTRACT:

Texture is one of the most important features for object detection and recognition. In many applications, it is derived from the responses of texture filters. In this paper, we evaluate the potential of seven texture filter banks for the pixel-based classification of terrestrial facade images. Particularly, we analyze features from Gabor, Haar, Root Filter Set, and Walsh filters as well as filters that have been proposed by Laws (1980), Leung and Malik (2001), and Schmid (2001). We determine texture energies similar to the approach of Laws (1980) using the proposed filter banks, and then we classify the derived feature vectors by three different methods: maximum a posteriori probabilities (MAP), linear discriminant analysis (LDA) and random forest (RF), respectively. In all three cases, we obtained best classification results with Haar, Laws, and Walsh filters.

1 INTRODUCTION

Pattern analysis is an important part of visual inspection processes. Treisman (1986) showed that humans can easily group visible patterns. The grouping of patterns may lead to a meaningful image segmentation, which can be an essential step for object detection and recognition. The texture around pixels is often used to segment image regions and to classify them (Galun et al., 2003; Winn et al., 2005). Then, texture is understood as the spatial arrangement of gray values and is discriminative by their specific radiometric and geometric appearance.

Regarding terrestrial facade images, the most dominant objects are the building itself, the ground, vegetation, and the sky. They can often be distinguished very well by textural information. Pixels showing sky lie often in a homogeneous gray to blue area or belong to circular patterns of clouds. Leafs of trees and shrubs point to all directions, hence edges in all directions can be found in the corresponding image parts. Contrarily, man-made objects such as buildings or pavements show strong edges in horizontal and vertical direction (in 3D). Especially these distinctive properties are used for facade segmentation or for automatic image rectification of facade images (Hernández and Marcotegui, 2009; Kalantari et al., 2008).

Recently, Korč and Förstner (2009) published an image data set showing urban buildings in their environment. It allows benchmarking of facade image classification, and therefore the repeatable comparison of different approaches. Most of the images of this data set show facades in Switzerland and Germany. Fig. 1 demonstrates the variability of the object data including a chain house in Basel (Switzerland) with two floors and a small front garden, an apartment house in Heidelberg (Germany) with five floors and a partly occluded ground floor, and a freestanding museum in Bonn (Germany). The manual annotation with four classes presents *building* in red, *ground* in brown, *sky* in blue, and *vegetation* in green. Pixels in black represent *background*, i. e. they do not belong to any of the four classes, and therefore, we do not consider them in our evaluation.

Texture segmentation and texture-based object detection and recognition is commonly performed analyzing the responses of texture filters. In the last thirty years, several texture filter banks,

Figure 1: Examples from benchmark data set, published by (Korč and Förstner, 2009). Bottom right shows the manual annotation of the image from bottom left with *building* red, *ground* brown, *sky* blue, and *vegetation* green.

i. e. set of filters, have been proposed and evaluated on many different images, often with the purpose of segmenting synthetic and natural scenes into uniformly textured regions. So far, not much effort on benchmarking different filters has been spent with respect to real scenes, especially facade images. Shao and Förstner (1994) evaluate the potential of Gabor filters on aerial imagery, Galun et al. (2003) segment landscape images on the basis of edge-based texture filters, Martin et al. (2004) use a subset of the root filter set (RFS) for their segmentation scheme on a collection of natural images, many of them showing animals, and Lazaridis and Petrou (2006) derive their texture features from the Walsh transform for analyzing medical data.

In this paper, we evaluate several texture filters with respect to

a pixel-based classification of facade images using the benchmark data set of Korč and Förstner (2009). The data set shows perspectively deformed scenes under different illumination conditions. Compared to the textured material data set of Dana et al. (1999), which are well accepted for benchmarking texture categorization with classes such as bread, bricks, stones, straw, textiles, and weaves etc, the scale of the typical facade image textures is different. Thus, these accepted data sets cannot be used for validating texture-based facade image interpretation.

Since the domain of facade image interpretation is very challenging, the benchmark data set of (Korč and Förstner, 2009) has been used for comparing the results of very powerful interpretation tools such as grammars (Ripperda and Brenner, 2009) or Markov Random Fields (Korč and Förstner, 2008). For a comparison of texture filters, it is not necessary to integrate such complex tools. Instead, we determine texture energies as proposed by Laws (1980), where the filter responses are accumulated over a pixel's neighborhood. So, the filter responses are smoothed and the classification results do not vary too much between adjacent pixels. Other common strategies are clustering or histogram analysis (Varma and Zisserman, 2002; Salahuddin et al., 2009). Yet, they seem to be rather (semi-) global approaches which can only be used for texture analysis, if one texture covers a large image part. As facade images also show relatively small objects such as shrubs and trees, we favor a local texture analysis in the form of texture energies.

Many Conditional Random Field (CRF) approaches have integrated texture information. Schnitzspan et al. (2008) state that the discriminative power of the unary potentials of the CRF is the key to the performance of the CRF. Interpreting the unary potentials as pixel-based classification we take this information as an additional motivation for our evaluation of texture filters.

In the following Section, we discuss the general workflow of our evaluation. We also present the considered texture filter banks and the determination of the texture energy. The learning step is described in Section 3, where we describe the feature and model selection for classification. In section 4, we show our results and discuss the effect of each filter bank with respect to the classification of facade images. We finally end up with conclusions.

2 GENERAL WORKFLOW

Our evaluation consists of two major steps, the training and the test phase. Therefore, we separated the benchmark data into two equally sized sets of images. One half is used for training the classifier, i. e. the selection of the most appropriate features and the determination of the best representative or discriminative model.

In this paper, we compare seven different texture filter banks with respect to pixel-based classification of facade images. All employed filter banks have been used recently in different applications. The filter banks are noted $\mathcal{F}_i$ and have a cardinality of N_i filters. In Tab. 1, we give an overview on the texture filter banks used.

We consider three different classification schemes in our evaluation. First, we classify the pixels by applying the maximum a posteriori classification (MAP) with Gaussian mixture models. Therefore, we select the best N_i^* filters from each filter bank by choosing the sets of filters that minimize the classification errors on the training data. Second, we perform a linear discriminant analysis (LDA), where the feature space is reduced by a projection into a smaller subspace. There, we determine the class membership by nearest mean. And third, we choose the modern classification technique of a random forest (RF) which consists of randomly designed decision trees with probabilities for class memberships in their leaves (Breiman, 2001; Teboul et al., 2010).

	Name	N_i	Size of filters
$\mathcal{F}_1$	Gabor	40	15×15 to 51×51
$\mathcal{F}_2$	Haar	16	4×4
$\mathcal{F}_3$	Laws	25	5×5
$\mathcal{F}_4$	Leung & Malik (LM)	48	49×49
$\mathcal{F}_5$	Root (RFS)	38	49×49
$\mathcal{F}_6$	Schmid	13	49×49
$\mathcal{F}_7$	Walsh	36	6×6

Table 1: Texture filter banks used in our evaluation.

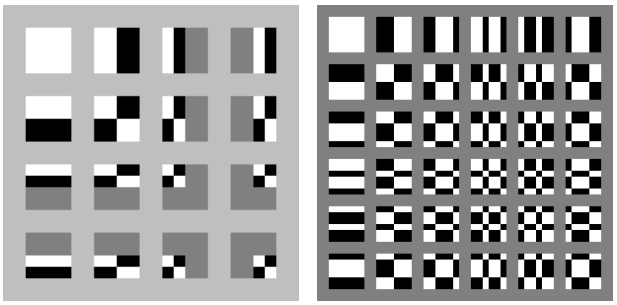

Figure 2: Visualization of set with 16 Haar filters (left) and set of 36 Walsh filters (right). White filter parts represent the value 1, black parts represent the value -1.

2.1 Texture Filter Banks

Two filter banks of our evaluation have their origin in the mathematical field of functional analysis. The Haar filters are derived from a wavelet transform of the signal, and the Walsh filters are derived from a generalized Fourier transform, cf. (Petrou and Bosdogianni, 1999). The filter banks as shown in Fig. 2 are the basis images of these two transforms. Both are defined iteratively and have been discretized for digital image processing. We realized both filter banks with 16 and 36 sufficiently diverse filters, respectively, but both filter banks could easily be extended by additional filters. As demonstrated in Fig. 2, both filter sets prefer the detection of horizontal and vertical lines, and chess-board-like repetitive structures. In recent applications, the Haar transform filters have been integrated into face detection algorithms (Viola and Jones, 2001). The Walsh transform filters have been applied in medical image analysis by Lazaridis and Petrou (2006) and in classification of building parts by Bochko and Petrou (2007).

Laws (1980) proposed a filter bank of five one-dimensional filters, which can be combined to 25 two-dimensional filters. The

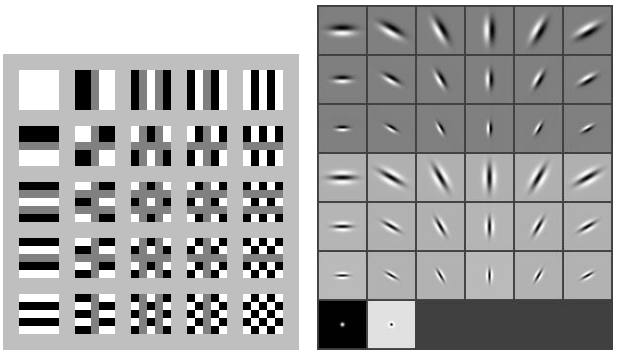

Figure 3: Visualization of set with 25 Laws filters (left) and set of 38 filters from the Root filter set (RFS) (right). The gray values in this visualization correspond to positive and negative filter entries, respectively.

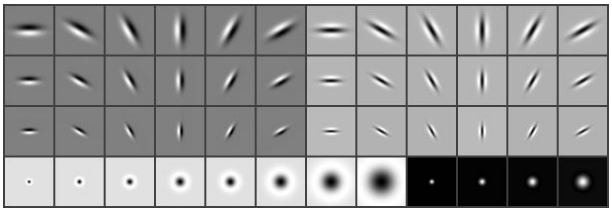

Figure 4: Visualization of set with 48 Leung & Malik (LM) filters.

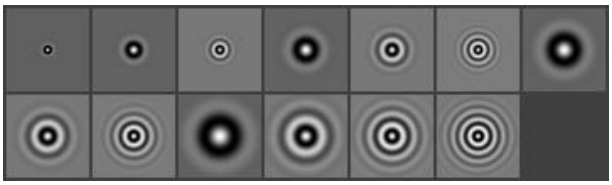

Figure 5: Visualization of set with 13 Schmid filters.

original filters are one smoothing operator and the first four derivatives. Realized filters are shown in Fig. 3, left. (Chang et al., 1999) included these filters into their evaluation of texture segmentation algorithms. They did not perform too bad although the filters are not illumination and rotation invariant.

In the last years, several filter banks have been proposed, which consist of Gaussian, Laplacian of Gaussian, and Gaussian derivative filters. Often applied filter banks are the ones proposed by Leung and Malik (2001), Schmid (2001), and the Root Filter Set (RFS). Subsets of the RFS filters have been used by Varma and Zisserman (2002); Winn et al. (2005); Salahuddin et al. (2009). We generated all three filter banks by the procedures published by the Oxford vision group on the internet[1]. The filters have been visualized in Figs. 3 to 5.

The last filter bank in our evaluation are the Gabor filters as defined in (Shao and Förstner, 1994). Similar to the Haar filters, Gabor filters consist in a wavelet transform of the image, and the magnitudes of the complex-valued filter responses are considered as the filter output. We realized a Gabor filter bank with filters at eight orientations and five scales, cf. Fig. 6. Gabor filters have been intensively used in various applications, e. g., (Fogel and Sagi, 1989; Jain and Farrokhnia, 1991; Dunn et al., 1994), but have recently lost their popularity.

2.2 Texture Energies

Although we could have performed the evaluation on color images, we preferred to work with gray valued images to reduce the complexity of the calculations. Since most of the textured structures are detectable in gray valued images, the loss of information due to the combination of the color channels seems to be acceptable. Our gray valued images reflect the luminance of a color pixel, defined as the Y-value in the YIQ color space, cf. (Foley et al., 1996). This procedure is a commonly used transform of RGB images into gray valued images in computer graphics.

[1] http://www.robots.ox.ac.uk/~ vgg/research/texclass/filters.html, last accessed on 28th May 2010

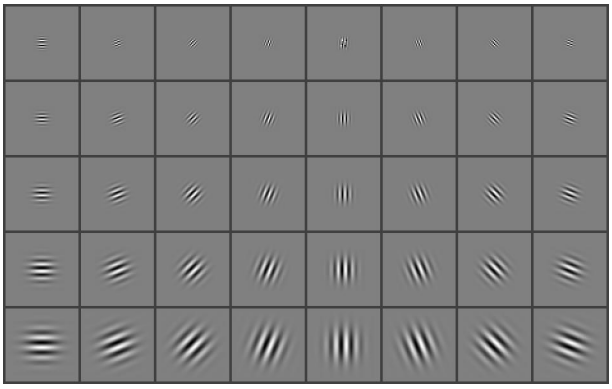

Figure 6: Visualization of a set with 40 Gabor filters.

We denote the gray valued image with g. It is convolved with each filter $F_n \in \mathcal{F}_i$. Thus, we derive a vector valued image $\mathbf{f} = [f_n]$ of filter responses with

$$f_n(x, y) = g(x, y) * F_n(x, y). \tag{1}$$

For Gabor filters, we obtain complex filter responses. We determinate the magnitude of the complex numbers and save these real values in f_n. Finally, we derive the vector valued energy image $\mathbf{e} = [e_n]$ by considering the texture filter responses in a pixel's neighborhood:

$$e_n(x, y) = \sum_{(u,v)\in S_r(x,y)} f_n(u, v) \tag{2}$$

with

$$(u, v) \in S_r(x, y) \Leftrightarrow \sqrt{(u - x)^2 + (v - y)^2} \leq r. \tag{3}$$

Thus, $S(x, y)$ describes a circular neighborhood around the pixel (x, y) with a radius r. In our experiments, we used $r = 7$.

3 TRAINING PHASE

3.1 Training Data

The 30 images used for training contain over 11 million pixels. Since we work with a real valued feature vector, the data for training a classifier is very large. For an ordinary computer, the learning of the classifier would exceed the memory, if the number of samples is not reduced. Furthermore, the four classes *building*, *ground*, *sky*, and *vegetation* do not appear equally frequent in the images. Hence, we are able to adjust the ratio of classes when selecting only a fraction of the pixels for training in each image. In all training images, we randomly select 1000 pixels representing one class, and less if the class is not that frequent in the image. So, we obtain a little less than 120 000 samples for training. Fig. 7 presents in the first column an image used for training and those pixels selected for training. The other two columns show all pixels that have been annotated as *building* or *vegetation*, respectively, and the selected pixels to these classes. If a pixel does not belong to one of the defined classes, it is labeled as *background*, and it is not further considered for training.

3.2 Feature Selection for MAP-Classification

The classification of an image pixel (x, y) is based on the texture energy values e_n which form an N_i-dimensional feature vector. Since many of the features might be not appropriate for recognizing class-specific patterns, we perform a feature selection to improve the classification and to reduce the complexity of the calculations.

Using the filter bank $\mathcal{F}_i$ with N_i elements, we obtain N_i texture energy features, and there are $2^{N_i} - 1$ possibilities to select non-empty subsets of these features. As an alternative to this exhaustive search, (Guyon and Elisseeff, 2003) propose a ranking of the features based on an independently defined measure, e. g. the correlation coefficient, etc. We would like to choose the most appropriate features on the basis of the classification's error rate.

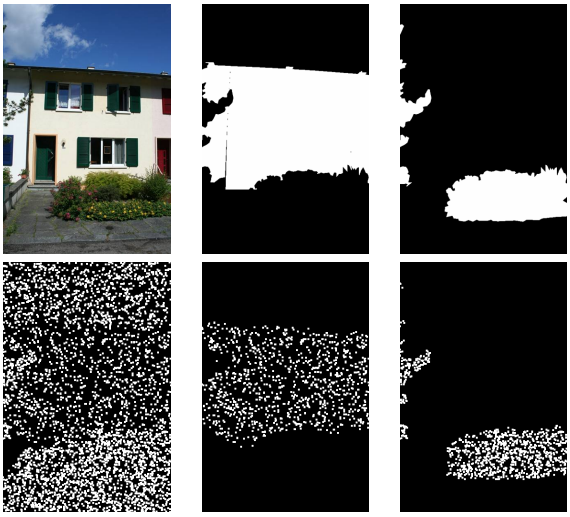

Figure 7: Random selection of pixels for training. For visualization purpose, the selected pixels have been dilated.

	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_3$	$\mathcal{F}_4$	$\mathcal{F}_5$	$\mathcal{F}_6$	$\mathcal{F}_7$
N_i	40	16	25	48	38	13	36
N_i^*	24	6	7	23	33	12	10
r	0.35	0.26	0.27	0.37	0.37	0.50	0.29

Table 2: Minimum training error rate r and number of selected features.

Instead of choosing the features by a random selection as it is done by Breiman (2001), we prefer to select our features by a greedy strategy. It is very fast with determining a subset of features which is mostly near the optimum.

We formulated our greedy strategy as an iterative forward selection, starting with the empty set of features, selecting one feature in each iteration, and terminating when the set of selected features is the complete set of N_i features. Thus, we avoid stopping in local minima when evaluating the set of selected features. In the beginning of the k-th iteration, we have already selected $k-1$ features, and there are $N_i - (k-1)$ features still to select. For each such set extension, we train the classifier and evaluate its performance on the training data. We extend the set of selected features by that new feature which minimizes the training error rate r_k together with the previously selected features. In total, this strategy needs only $\frac{N_i(N_i+1)}{2}$ training steps.

For each iteration k, we determine the training error rates r_k depending on the increasing set of selected features. We determine the set of most appropriate features with cardinality $N_i^* \leq N_i$ as the tuple of features which yields the minimum training error rate $r = \min r_k$. Tab. 2 shows the number of selected features in comparison to the complete set of features. For the three filter banks with Haar, Laws, and Walsh filters, the set of the most appropriate features is very small. For instance, the six most appropriate Haar features are the filter responses of the filters with indices 16, 4, 1, 13, 5 and 8 (selection has been done in this order). The indices also correspond to the row wise order of the filters in Fig. 2.

3.3 Training the Classifiers

MAP: We learn a classifier in each iteration when selecting the most appropriate features. It is later used for evaluation on the test images.

Let $[e_n]_{(k)}$ be the set of selected features in the k-th iteration.

building:	$p(c_1) = 0.585$
ground:	$p(c_2) = 0.111$
sky:	$p(c_3) = 0.166$
vegetation:	$p(c_4) = 0.138$

Table 3: Prior probabilities of the classes.

Then we perform the MAP classification by

$$\arg\max_j p(c_j \mid [e_n]_{(k)}) \propto \arg\max_j p([e_n]_{(k)} \mid c_j) p(c_j). \quad (4)$$

The likelihood functions $p([e_n]_{(k)} \mid c_j)$ are estimated in the form of Gaussian mixture models (GMM) with two normal distributions for each class c_j. Particularly, we applied the EM algorithm as realized by Franc and Hlaváč (2004). The choice for GMMs is based on the empirical observation that there usually exist several patterns for each class. For instance, there are plastered and brick-built facades, homogeneous and cloudy skies, leafs and differently structured bark for trees, and asphalt and tiled streets. While more Gaussians for each GMM would approximate the likelihood functions probably better, we then might have to deal with an overfitted classifier, or with numerical problems.

The a priori probabilities $p(c_j)$ of the classes are derived from all annotated pixels of the training images (cf. Tab. 3).

LDA: For some of the texture filter banks we work in feature spaces with a high dimensionality even after selecting the most appropriate features by the greedy strategy. For comparison, we also learn the model of linear discriminant analysis (LDA). In LDA, the samples are projected into a subspace, which is efficient for discrimination: The inner-class scatter is minimized while the between-class scatter is maximized (Duda et al., 2001). In the LDA-projected subspace, we classify the projected samples based on the smallest Mahalanobis distance to the class-specific means.

RF: The number of features is not relevant for the complexity of constructing a random forest (RF). Such a forest consists of T binary decision trees of depth d. For all inner nodes, we randomly select a feature and a threshold. At each leaf, we obtain a histogram of the class occurrences which we normalize to get probabilities. In the test phase, we apply each decision tree to a test sample, and we obtain T probabilities for the occurrence of each class. As done by Teboul et al. (2010), we determine the average of these probabilities. Due to the exponentially growing number of nodes in the RFs, we did not search for the best fitting parameters, but we selected similar ones as Teboul et al. (2010) and chose $T = 15$ and $d = 12$.

4 TEST PHASE

In the test phase, we work on the other half of the benchmark images. We compute the gray valued luminance images, convolve each image with the texture filters, and determine the texture energy as described in sec. 2. The classification models including the feature reduction step for MAP and LDA are tested on all pixels which have a valid class, i. e. they have not been annotated as *background* in the ground truth. For our evaluation, the differences between the determined and the real class are counted for all three classification techniques separately.

Fig. 8 visualizes classification results. We present pixels classified as *building* in red, *ground* in gray, *sky* in blue, and *vegetation* in green. Again, non-classified *background* pixels are visualized in black. The first two rows show the MAP results using features from Gabor (top left), Laws (top right), RFS (bottom left) and

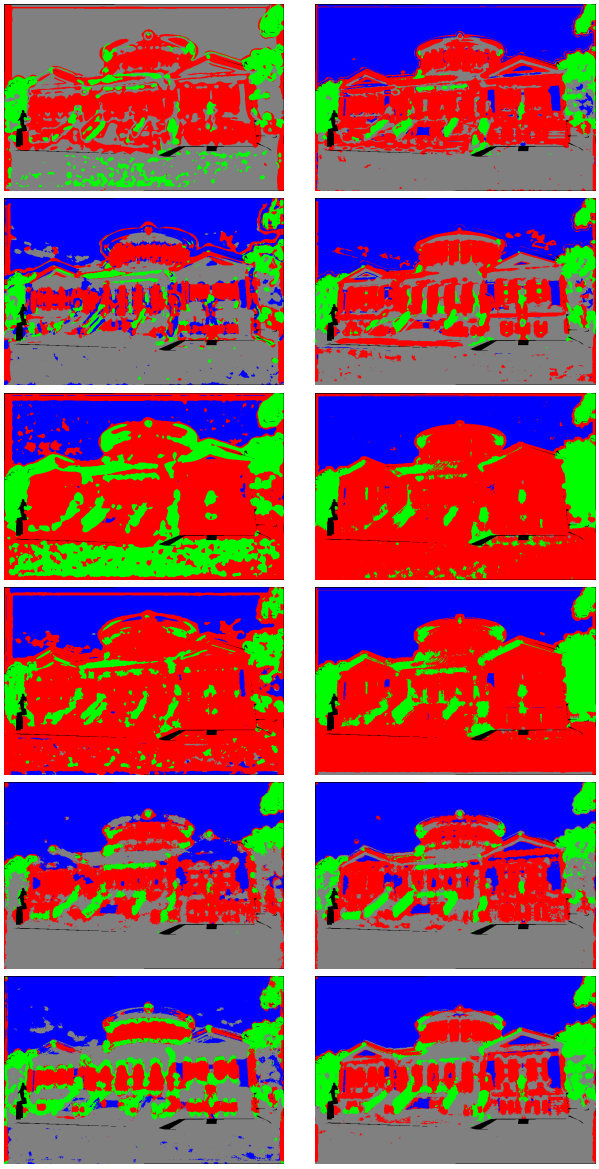

Figure 8: Visualization of results of texture classification of the Academic Art Museum in Bonn, Germany (original image and manual annotation is shown in fig. 1), for further information see text.

Walsh (bottom right). Then the third and fourth row show the LDA results (same order of filters), and the last two rows show the RF results (same order of filters).

The results of the classification using Haar and Leung-Malik filters are visualized in fig. 9, second to fourth row. Again, the MAP classification results are shown on the left, the LDA results on the right. In the figure's first row, we present the original image and its manual annotation.

The quality inspection of the results of Figs. 8 and 9 shows that the LDA yields smoother results, i. e. adjacent pixels often have the same class, but the LDA classifier has difficulties with separating *facade* from *ground* and *sky*. The good results of LDA may be caused from the high occurrence of the two classes *facade* and *vegetation*. The incorrect results at windows are often due to the reflectance of vegetation and sky in the window panes. If buildings, streets and the sky have a homogeneous texture, the misclassifications can only be solved by considering additional features or context, e. g., using Markov Random Fields as proposed by Heesch and Petrou (2007) or Korč and Förstner (2008), or high-level scene interpretation tools as developed by Hartz and Neumann (2007), but this is not the scope of this paper.

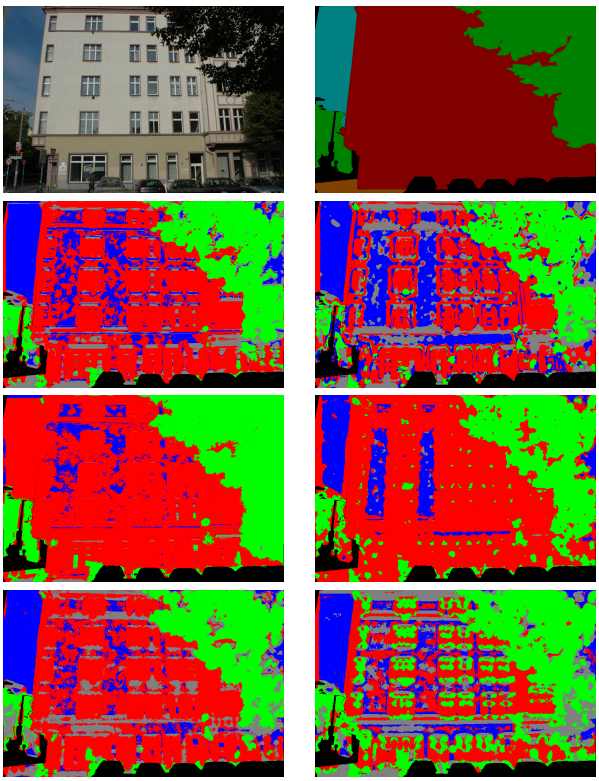

Figure 9: Visualization of results of texture classification of an apartment building in Berlin, Germany (top row shows image and the ground truth. Second to last row show results of MAP, LDA and RF classification using Haar features (left) and features from LM filters (right).

In Tab. 4, we present the classification error rates. The error rate was determined for each image separately, and then used to calculate the mean and the standard deviation over all images. Since other objects such as cars might appear in the test images and their pixels have been annotated as *background*, the number of valid pixels in the test images varies largely.

Gabor filters have the worst performance of all filters for MAP and LDA, and perform only fair RF. Since we have chosen a sufficiently large and varied set of Gabor filters, we consider them to be inadequate for facade image interpretation. Additionally, not only the cardinality of the Gabor filter set is large, but also the filters themselves are relatively large. Hence, the calculations are more time-consuming than for other filters. The root filter set and the LM filters (except for LDA) and the Schmid filters also behave significantly worse than other filter sets. The best filter sets are Haar, Laws and Walsh. They are best with respect to classification performance as well as to computational complexity. All three sets comprise filters enabling the detection of horizontally and vertically structured patterns which appear very frequent in facade images.

5 SUMMARY

We evaluated the performance of seven texture filter banks with respect to pixel-based classification of facade images. The filter sets considered are Gabor, Haar, Laws, Leung & Malik, Schmid, Walsh, and the root filter set. We determined texture energy values and used three classification schemes: maximum a posteriori

filter	MAP	LDA	RF
Gabor	0.497 ± 0.07	0.391 ± 0.09	0.373 ± 0.08
Haar	0.329 ± 0.07	0.316 ± 0.12	0.332 ± 0.08
Laws	0.320 ± 0.09	0.301 ± 0.10	0.319 ± 0.09
LM	0.445 ± 0.06	0.317 ± 0.08	0.479 ± 0.08
RFS	0.446 ± 0.06	0.344 ± 0.07	0.480 ± 0.08
Schmid	0.461 ± 0.05	0.353 ± 0.12	0.536 ± 0.08
Walsh	0.345 ± 0.07	0.325 ± 0.11	0.344 ± 0.09

Table 4: Error rates over all four classes.

classification with Gaussian mixture models, linear discriminant analysis with closest mean, and random forests. The best classification results are obtained for Haar, Laws, and Walsh filters.

The results show that texture is not sufficient: Other features such as color values or context from a pixel's neighborhood should be integrated. This could help to deal the many misclassifications in homogeneous image parts: sky, ground, and plastered facade often have a similar texture in luminance images.

References

Bochko, V. and Petrou, M., 2007. Recognition the Structural Parts of a Building using Support Vector Machines. In: PRIP'07.

Breiman, L., 2001. Random Forests. Machine Learning 45, pp. 5–32.

Chang, K. I., Bowyer, K. W. and Sivagurunath, M., 1999. Evaluation of Texture Segmentation Algorithms. In: CVPR'99, pp. I: 294–299.

Dana, K. J., van Ginneken, B., Nayar, S. K. and Koenderink, J. J., 1999. Reflectance and Texture of Real-World Surfaces. ACM Transactions on Graphics 18(1), pp. 1–34.

Duda, R. O., Hart, P. E. and Stork, D. G., 2001. Pattern Classification. John Wiley and Sons.

Dunn, D., Higgins, W. E. and Wakeley, J., 1994. Texture Segmentation using 2-D Gabor Elementary Functions. PAMI 16(2), pp. 130–149.

Fogel, I. and Sagi, D., 1989. Gabor Filters as Texture Discriminator. Biological Cybernetics 61(2), pp. 102–113.

Foley, J. D., van Dam, A., Feiner, S. K. and Hughes, J. F., 1996. Computer Graphics: Principles and Practice. 2nd ed. in C edn, Addison-Wesley.

Franc, V. and Hlaváč, V., 2004. Statistical Pattern Recognition Toolbox for Matlab. Technical report, CMP, Czech Technical University, http://cmp.felk.cvut.cz/cmp/software/stprtool/.

Galun, M., Sharon, E., Basri, R. and Brandt, A., 2003. Texture Segmentation by Multiscale Aggregation of Filter Responses and Shape Elements. In: ICCV'03, pp. II: 716–723.

Guyon, I. and Elisseeff, A., 2003. An Introduction to Variable and Feature Selection. JMLR 3, pp. 1157–1182.

Hartz, J. and Neumann, B., 2007. Learning a Knowledge Base of Ontological Concepts for High-Level Scene Interpretation. In: ICMLA, pp. 436–443.

Heesch, D. and Petrou, M., 2007. Non-Gibbsian Markov Random Field Models for Contextual Labelling of Structured Scenes. In: BMVC'07.

Hernández, J. and Marcotegui, B., 2009. Morphological Segmentation of Building Facade Images. In: ICIP 2009.

Jain, A. K. and Farrokhnia, F., 1991. Unsupervised Texture Segmentation using Gabor Filters. Pattern Rec. 24(12), pp. 1167–1186.

Kalantari, M., Jung, F., Paparoditis, N. and Guedon, J.-P., 2008. Robust and Automatic Vanishing Points Detection with their Uncertainties from a Single Uncalibrated Image, by Planes Extraction on the Unit Sphere. In: 21st ISPRS Congress, IAPRS 37 (3a), pp. 203–208.

Korč, F. and Förstner, W., 2008. Interpreting Terrestrial Images of Urban Scenes using Discriminative Random Fields. In: 21st ISPRS Congress, IAPRS 37 (B3a), pp. 291–296.

Korč, F. and Förstner, W., 2009. eTRIMS Image Database for Interpreting Images of Man-Made Scenes. Technical Report TR-IGG-P-2009-01, IGG University of Bonn.

Laws, K. I., 1980. Rapid Texture Identification. In: Image Processing for Missile Guidance, SPIE 238, pp. 376–380.

Lazaridis, G. and Petrou, M., 2006. Image Registration using the Walsh Transform. Image Processing 15(8), pp. 2343–2357.

Leung, T. and Malik, J., 2001. Representing and Recognizing the Visual Appearance of Materials using Three-dimensional Textons. IJCV 43(1), pp. 29–44.

Martin, D. R., Fowlkes, C. C. and Malik, J., 2004. Learning to Detect Natural Image Boundaries using Local Brightness, Color, and Texture Cues. PAMI 26(5), pp. 530–549.

Petrou, M. and Bosdogianni, P., 1999. Image Processing: The Fundamentals. Wiley.

Ripperda, N. and Brenner, C., 2009. Evaluation of Structure Recognition using Labelled Facade Images. In: DAGM 2009, LNCS 5748, pp. 532–541.

Salahuddin, M., Drew, M. S. and Li, Z.-N., 2009. A Fast Method for Classifying Surface Textures. In: ICASSP 2009, pp. 1077–1080.

Schmid, C., 2001. Constructing Models for Content-based Image Retrieval. In: CVPR'01, pp. 39–45.

Schnitzspan, P., Fritz, M. and Schiele, B., 2008. Hierarchical Support Vector Random Fields: Joint Training to Combine Local and Global Features. In: ECCV'08, pp. 527–540.

Shao, J. and Förstner, W., 1994. Gabor Wavelets for Texture Segmentation. In: Symposium on Spatial Information from Digital Photogrammetry and Computer Vision, IAPRS 30 (3).

Teboul, O., Simon, L., Koutsourakis, P. and Paragios, N., 2010. Segmentation of Building Facades using Procedural Shape Priors. In: CVPR'10.

Treisman, A., 1986. Features and Objects in Visual Processing. Scientific American 225, pp. 114–125.

Varma, M. and Zisserman, A., 2002. Classifying Images of Materials: Achieving Viewpoint and Illumination Independence. In: ECCV'02, pp. III: 255–271.

Viola, P. and Jones, M., 2001. Rapid Object Detection using a Boosted Cascade of Simple Features. In: CVPR'01, pp. 511–518.

Winn, J., Criminisi, A. and Minka, T., 2005. Object Categorization by Learned Universal Visual Dictionary. In: ICCV'05.