Real-Time Detection of Landscape Scenes

Sami Huttunen[1], Esa Rahtu[1], Iivari Kunttu[2],
Juuso Gren[2], and Janne Heikkilä[1]

[1] Machine Vision Group, University of Oulu, Finland
firstname.lastname@ee.oulu.fi
http://www.cse.oulu.fi/MVG
[2] Nokia Corporation, Tampere, Finland
firstname.lastname@nokia.com

Abstract. In this paper we study different approaches that can be used in recognizing landscape scenes. The primary goal has been to find an accurate but still computationally light solution capable of real-time operation. Recognizing landscape images can be thought of a special case of scene classification. Even though there exist a number of different approaches concerning scene classification, there are no other previous works that try to classify images into such high level categories as landscape and non-landscape. This study shows that a global texture-based approach outperforms other more complex methods in the landscape image recognition problem. Furthermore, the results obtained indicate that the computational cost of the method relying on Local Binary Pattern representation is low enough for real-time systems.

Keywords: computational imaging, scene classification, image categorization.

1 Introduction

Knowledge of the scene type provides important information in a number of applications that deal with consumer photographs and digital cameras. Generally, determining the scene type is the starting point of further image analysis and search in large image collections [1,5]. On the other hand, it can already guide the online image capture process in a camera device [3,10].

In this paper we study different approaches that can be used in recognizing landscape scenes. The detection of landscape scenes is a difficult problem given the fact that several landscape scenes have similar objects as non-landscape scenes, and vice versa. Furthermore, illumination conditions are equally unpredictable for both cases. Due to the computational restrictions set by the target applications the primary goal of our work has been to find an accurate but still computationally light solution capable of real-time operation.

The results of our work can be utilized when developing a fast method for separating the landscape and non-landscape scenes. This kind of classification can serve as a preprocessing step for speeding-up image retrieval in large databases and improving accuracy, or for performing automatic image annotation [5]. In

A. Heyden and F. Kahl (Eds.): SCIA 2011, LNCS 6688, pp. 338–347, 2011.

Fig. 1. Example of landscape classification

addition to image retrieval applications, camera settings may be adjusted automatically depending on the scene type, so that the best possible representation can be achieved [3,10].

Definition of *landscape* and *non-landscape* images is not totally straightforward. In this work we assume that if there are no distinct and easily separable objects present in a natural scene, the image is classified as landscape. From the photographic point of view, this requirement would mean that as much of the scene as possible should be in focus. As a result of the aforementioned restrictions, the landscape category would contain sunset, beach, mountain, etc., subcategories. On the other hand, it is obvious that all the images taken indoors should be classified as non-landscape. In this case, the non-landscape branch would consist of indoor scenes and other images containing man-made objects at relatively close distance (Fig. 1).

Recognizing landscape images can be thought of a special case of scene classification which aims at labeling an image into a set of different semantic categories. Even though there exist a number of different approaches concerning scene classification, to the best of our knowledge, there are no any other works concentrating on classifying images into the *landscape* and *non-landscape* categories. The previous works differ by the number of the scene classes, the image representations, and the classification method. The most methods so far have aimed at classifying into a small number of scene categories, including indoor/outdoor [8,14,15,16], city/landscape [17], and subsets of urban and natural scenes [9,13,17]. It can be noticed that none of these categorizations is directly applicable in our problem.

A common approach in image categorization is to use local features [11,18] combined with the bag-of-words (BOW) representation [4] and the Support Vector Machine (SVM) classifier. In this approach the image is then expressed by a histogram of visual word occurrences which can be used in training a classifier.

Another common way to categorize images is to compute low-level features, such as color and texture, which are further processed with a classifier engine for inferring high-level information about the image. These methods assume that the type of scene can be directly described by the color or texture properties of the image. In fact it has been shown that low-level features can give very comparable results on many scene classification tasks [2,15,16]. The work done in [15] employs low-level color and texture features whereas [16] concatenates

the histograms in the Ohta color space with texture and frequency features. Later [14] has introduced an indoor/outdoor classification technique based on edge analysis.

We show with extensive experiments that a global texture-based approach competes with or outperforms other more complex methods in the landscape image recognition problem. Especially the computational cost of the method relying on Local Binary Pattern [12] representation is minimal compared to the GIST [13] and BOW methods investigated in this paper.

The rest of the paper is organized as follows. Section 2 gives a detailed description of the different features and classifier used in this study. The experimental results are presented in Sect. 3. Finally, the conclusions are summarized in Sect. 4.

2 Methods for Landscape Scene Recognition

There are two main elements in a typical image classification system. The first one is responsible for the computation of the feature vector representing an image whereas the second part is the classifier, the algorithm that classifies an input image into one of the predefined categories based on the feature vector. In this section we describe two approaches for landscape/non-landscape image classification. We begin with the image representation models followed by the classifier engine.

2.1 Global Features

Here we present two different approaches based on global description of image content.

GIST. One of the most well known global approaches in scene categorization is the GIST descriptor that was initially proposed in [13]. The main idea of this approach is to develop a low dimensional representation of the scene, which does not require any form of segmentation. The authors propose a set of perceptual dimensions (naturalness, openness, roughness, expansion, ruggedness) that represent the dominant spatial structure of a scene. They show that these dimensions may be reliably estimated using spectral and coarsely localized information.

To compute the color GIST description, the image is first divided into a 4×4 grid on which orientation histograms are extracted. Most of the works using the GIST descriptor resize the image as a preliminary stage, producing a small square image whose width typically ranges from 32 to 256 pixels. In our work, the images are rescaled to 240×240 size irrespective of their original aspect ratio. This is sufficient due to the low dimensionality of the descriptor, in other words, it does not represent the details of an image.

Local Binary Pattern (LBP). The discrete occurrence histogram of the LBP patterns computed over an image or a region of image is shown to be a very

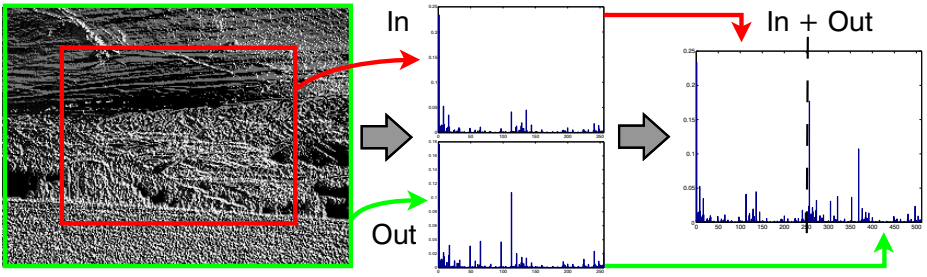


Fig. 2. The LBP histograms are computed in the center (*in*) and on the boundary areas (*out*) separately. The final image representation is then concatenation of these two histograms (*in+out*).

powerful texture feature. In LBP [12], the original 3×3 neighborhood is thresholded by the value of the center pixel. The values of the pixels in the thresholded neighborhood are multiplied by the weights given to the corresponding pixels. Finally, the values of the eight pixels are summed to obtain the number of a single texture unit.

When we think about landscape images depicting natural scenes usually the center of the image does not contain any distinctive objects. Therefore it is reasonable to utilize this information by computing the histograms in the center and on the boundary areas surrounding the center separately (Fig. 2). The final image representation is then concatenation of these two histograms providing us with a 512 bins long representation. From here onward it is referred as $\mathrm{LBP}_{\mathrm{io}}$, and the basic version of the LBP is annotated by $\mathrm{LBP}_{\mathrm{b}}$.

2.2 Local Features

A common approach in image categorization is to use some local features combined with the bag-of-words (BOW) representation which describes an image as an orderless collection of local features [4]. The basic idea of these approaches is that a set of local image patches is sampled either densely, randomly, or using a keypoint detector. After the sampling, a vector of visual descriptors is computed on each image patch independently (Fig. 3). There is a large number of different methods that can be used for describing the image patch content. One of the most popular approaches is to use SIFT-based descriptors [11,18] but also histograms or moments can be considered [18]. Regardless of the choice of the method, the resulting collection of descriptors is vector quantized and the global word histogram obtained is used as a characterization of the image.

In this study, the descriptors (see Table 1) were extracted using dense sampling with a step size of 10 pixels and default scale defined in the binary implementation [18]. For more information about the descriptors and their implementation details, please refer to [18]. The descriptor quantization was done by k-means clustering resulting in a vocabulary of 1000 words. To be independent of the

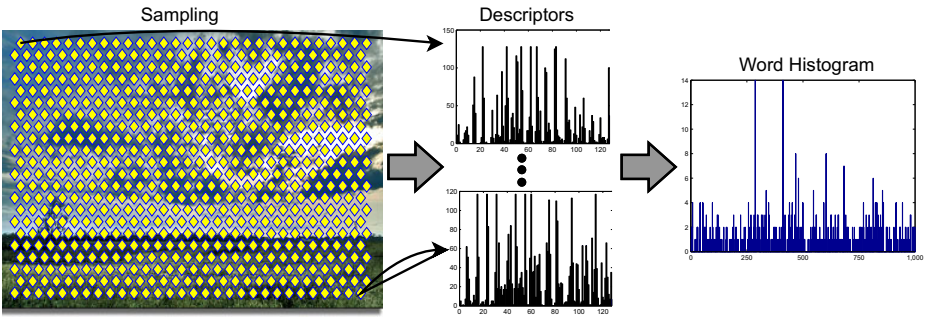


Fig. 3. The stages of the bag-of-words approach. First, the sample points are picked from the image. Then, for every point a color descriptor is computed over the area around that point. All the descriptors are subsequently vector quantized against a codebook of prototypical descriptors. This results in a fixed-length feature vector representing the image.

Table 1. Local descriptors[a]

Type	Descriptors	
SIFT	rgsift	csift
	opponentsift	rgbsift
	hsvsift	sift
	huesift	
Histogram	huehistogram	nrghistogram
	transformedcolorhistogram	opponenthistogram
	rgbhistogram	
Moment	colormomentinvariants	colormoments

[a] For details on the descriptors, see [18].

total number of descriptors in an image, the sum of the final feature vector was normalized to 1.

2.3 Classification

The Support Vector Machine (SVM) is widely used in scene classification and therefore it is selected as a classifier in this work. Even though the linear SVM is light in terms of computational burden, based on our preliminary evaluations we employ the Radial Basis Function (RBF) kernel in this study. In our application the classification step is carried out only once per image, thus its effect on overall time cost is minimal. When computing the kernels the distance function is Chi-squared with LBPs and local features whereas the GIST features are compared with the L2 norm.

3 Experimental Results

Comparative evaluation has been carried out between the methods described in Sect. 2. Combinations of the features were not considered because such kind of approaches would be too complex in view of the practical applications.

3.1 Image Sets

The images used for training and testing of the SVM classifier were downloaded from the PASCAL Visual Object Classes (VOC2007) database [6] and the Flickr site [7]. All the images mentioned below were manually labeled and resized to QVGA (320×240) resolution apart from the GIST, which uses 240×240 images.

Training dataset. The combined training and validation database contains 1115 landscape images and 2617 non-landscape images. Approximately 20 % of the training images were used for validation of the SVM classifier.

Testing dataset. Testing database contains 912 landscape images and 2140 non-landscape images. As with the training images, most of the landscape images come from the Flickr database and the non-landscape images originate mainly from the VOC2007 collection.

3.2 Evaluation Criteria

The classification task will be evaluated by the precision/recall curve, and the principal quantitative measure used is the average precision (AP). In addition, the performance will be evaluated by the Receiver Operating Characteristic (ROC) curve. In this case the measure used is the area under curve (AUC).

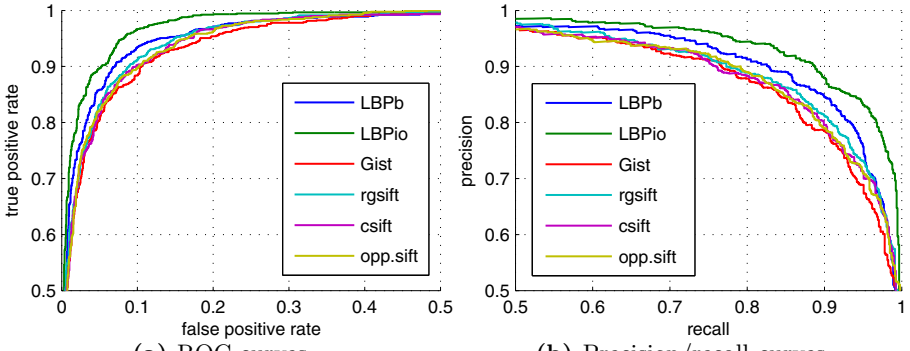
Furthermore we report the true positive and false positive rates (TPR and FPR, respectively) of the different approaches when the threshold for the SVM decision value is set to zero. In our case the definitions for the test images are as follows:

- False positive (FP): non-landscape classified as landscape
- True positive (TP): landscape classified as landscape

3.3 Results

The precision/recall and ROC curves are illustrated in Fig. 4. For clarity, only the best performing methods are included in the figures but Table 2 summarizes all the results in a numerical form. It can be seen that the LBP based approaches perform best both in terms of AUC and AP. It is worth noting that the $\mathrm{LBP}_{\mathrm{io}}$ approach, which concatenates the histograms computed in the image center and boundary area, gives better performance than $\mathrm{LBP}_{\mathrm{b}}$.

Figure 5 contains a collection of sample images when using the $\mathrm{LBP}_{\mathrm{io}}$ representation. When looking at the false positive images (Fig. 5c) it can be seen that most of the images contain smooth areas around some object.



(a) ROC curves **(b)** Precision/recall curves

Fig. 4. Results of the best performing methods

Table 2. Summary of the results

		Classification				Execution time (s)	
		AUC	AP	TPR	FPR	Descriptor	Total
Global	LBP_{io}	0.982	0.958	0.882	0.040	0.001	0.005
	LBP_{b}	0.972	0.939	0.862	0.055	0.001	0.003
	GIST	0.963	0.924	0.809	0.050	NA	>0.029
Local descriptors + BOW	rgsift	0.969	0.934	0.814	0.045	0.340	2.699
	csift	0.966	0.926	0.825	0.052	0.350	2.712
	opponentsift	0.966	0.922	0.828	0.052	0.340	2.694
	rgbsift	0.960	0.918	0.804	0.047	0.330	2.744
	hsvsift	0.959	0.915	0.804	0.050	0.340	2.494
	sift	0.956	0.901	0.806	0.059	0.120	0.595
	huesift	0.954	0.902	0.791	0.067	0.290	1.046
	colormomentinvariants	0.926	0.857	0.737	0.067	1.410	1.444
	transformedcolorhistogram	0.924	0.851	0.692	0.061	0.100	0.147
	opponenthistogram	0.909	0.825	0.689	0.079	0.090	0.140
	rgbhistogram	0.903	0.805	0.683	0.087	0.070	0.118
	colormoments	0.897	0.811	0.697	0.084	1.340	1.376
	huehistogram	0.863	0.717	0.525	0.071	0.180	0.223
	nrghistogram	0.861	0.727	0.601	0.102	0.080	0.119

3.4 Computational Cost

In order to evaluate the computational cost of the different image representations the preliminary performance analysis was conducted on a regular Windows PC (Core 2 Duo 3.2GHz, 4GB RAM).

Our own LBP C code implementation was evaluated with Visual Studio 2010 Profiler whereas the execution times of the different color descriptors were obtained using the binaries publicly available [18]. The results are shown in Table 2

Fig. 5. Classification examples with LBP_{io} representation. (a) Landscape classified as landscape, (b) non-landscape classified as non-landscape, (c) non-landscape classified as landscape, and (d) landscape classified as non-landscape.

and they include the time spent on descriptor computation as well as the total time for SVM classification. It should be noted that the most time consuming part of the bag-of-words based methods is the word histogram computation.

Unfortunately the GIST descriptor codes [13] were available only for MATLAB and therefore its performance could not be studied thoroughly in these experiments. However, since the GIST descriptor is computed using several filters corresponding to different orientations and scales, its computational cost is likely to be higher than that of LBP.

3.5 Real-Time Implementation

Based on the results presented in Table 2, it is obvious that the LBP histogram is the best choice when building a real-time system. On the other hand, selection between the two different LBP representations depends mainly on the requirements set by the target platform. Our current real-time implementation coded in C relies on the basic LBP_b, which gives reasonable results with lower memory consumption.

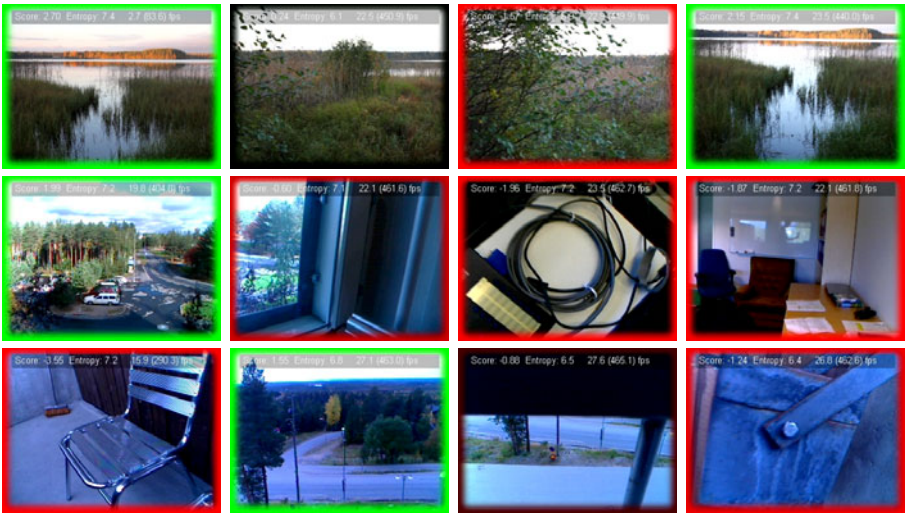


Fig. 6. Video frames with the classification results. Green boundaries are used for the landscape and red boundaries for the non-landscape frames. If the boundary is black, selection between the classes cannot be done reliably.

When we take a closer look at the profiler results of the final system, the results indicate that most of the time is spent on the SVM classifier since LBP histogram computation takes only one third of the overall processing time. The total execution time for one QVGA frame is about 3 ms which guarantees real-time performance even on more constrained platforms.

Test videos. In order to evaluate the performance of the LBP_{b} approach in real-time scenarios we captured several video sequences with different cameras (Canon EOS 5D Mark II, Logitech QuickCam Fusion, Nokia N95). All the videos were resized to QVGA resolution but no other pre-processing steps were applied.

The results of the sequences are shown in Fig. 6. To illustrate how the detection of landscape scenes works with the given frames, we use green boundaries for the landscape and red boundaries for the non-landscape frames. If the boundary is black, the decision value of the classifier is close to zero and therefore selection between the classes cannot be done reliably.

4 Conclusion

In this paper we have studied different approaches that can be used in automatic landscape scene recognition. Due to the computational restrictions set by the target devices the primary goal of our work has been to find an accurate but still computationally light solution capable of real-time operation.

We have shown with extensive experiments that a global texture-based approach outperforms other more complex methods in the landscape image recognition problem. It appears that the local features are too distinctive for the given

task. The results obtained clearly indicate that the computational cost of the method relying on the Local Binary Pattern (LBP) representation is low enough for real-time systems. It should be noted that the LBP operates on gray scale images, which means that the use of color information is not needed.

References

1. Bianco, S., Ciocca, G., Cusano, C., Schettini, R.: Improving color constancy using indoor–outdoor image classification. IEEE TIP 17(12), 2381–2392 (2008)
2. Bosch, A., Munoz, X., Marti, R.: Which is the best way to organize/classify images by content? Image and Vision Computing 25(6), 778–791 (2007)
3. Chung, D., Kim, S., Bae, J., Lee, S.: Photographic expert-like capturing by analyzing scenes with representative image set. In: Casasent, D.P., Hall, E.L., Röning, J. (eds.) Proc. SPIE, vol. 7252 (2009)
4. Csurka, G., Dance, C., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: Proc. Workshop on Statistical Learning in Computer Vision, ECCV, vol. 1, p. 22 (2004)
5. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Image retrieval: Ideas, influences, and trends of the new age. ACM Computing Surveys 40(2), 5:1–5:60 (2008)
6. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results (2007), `http://www.pascal-network.org/challenges/VOC/voc2007/workshop/`
7. Flickr: Flickr homepage (2010), `http://www.flickr.com/search/?q=landscape`
8. Kim, W., Park, J., Kim, C.: A novel method for efficient indoor-outdoor image classification. Journal of Signal Processing Systems 61, 251–258 (2010)
9. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. Proc. IEEE CVPR 2, 2169–2178 (2006)
10. Lipowezky, U., Vol, I.: Indoor-outdoor detector for mobile phone cameras using gentle boosting. In: Proc. IEEE CVPR Workshops (CVPRW), pp. 31–38 (2010)
11. Lowe, D.: Distinctive image features from scale-invariant keypoints. IJCV 60(2), 91–110 (2004)
12. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. Pattern Recognition 29(1), 51–59 (1996)
13. Oliva, A., Torralba, A.: Modeling the shape of the scene: A holistic representation of the spatial envelope. IJCV 42(3), 145–175 (2001)
14. Payne, A., Singh, S.: Indoor vs. outdoor scene classification in digital photographs. Pattern Recognition 38(10), 1533–1545 (2005)
15. Serrano, N., Savakis, A., Luo, A.: A computationally efficient approach to indoor/outdoor scene classification. Proc. IEEE ICPR 4, 146–149 (2002)
16. Szummer, M., Picard, R.W.: Indoor-outdoor image classification. In: Proc. IEEE Workshop on Content-Based Access of Image and Video Database, pp. 42–51 (1998)
17. Vailaya, A., Figueiredo, M.A.T., Jain, A.K., Zhang, H.J.: Image classification for content-based indexing. IEEE TIP 10(1), 117–130 (2001)
18. van de Sande, K.E.A., Gevers, T., Snoek, C.G.M.: Evaluating color descriptors for object and scene recognition. IEEE TPAMI 32(9), 1582–1596 (2010)