Computer Vision Winter Workshop 2008, Janez Perš (ed.)
Moravske Toplice, Slovenia, February 4–6
Slovenian Pattern Recognition Society, Ljubljana, Slovenia

Vegetation segmentation for boosting performance of MSER feature detector

Dušan Omerčević[1], Roland Perko[1], Alireza Tavakoli Targhi[2], Jan-Olof Eklundh[2], and Aleš Leonardis[1]

[1]Faculty of Computer and Information Science,
University of Ljubljana, Slovenia
{dusan.omercevic, roland.perko, ales.leonardis}@fri.uni-lj.si

[2]Computational Vision and Active Perception Laboratory,
Royal Institute of Technology (KTH), Sweden
{att, joe}@nada.kth.se

Abstract *In this paper, we present a new application of image segmentation algorithms and an adaptation of the image segmentation method of Tavakoli et al. to the problem of vegetation segmentation. While the traditional goal of image segmentation is to provide a figure/ground segmentation for object recognition or semantic segmentation to assist humans, we propose to use image segmentation in order to boost performance of local invariant feature detectors. In particular, we analyze the performance of MSER feature detector and we show that we can prune all features detected on vegetation to gain a 67% speed-up while accuracy of image matching does not decrease. The image segmentation method of Tavakoli et al. that we adapt to the problem of vegetation segmentation is based on singular value decomposition (SVD) of local image patches, where the sum of the smaller singular values describes the high frequency part of the patch. The results of the automatic segmentation of vegetation show that the average overlap between manual and automatic vegetation segmentation is 33% and that the automatic procedure for vegetation segmentation can prune 25% of MSER features, resulting in 33% faster image retrieval.*

1 Introduction

The approach to image matching based on local invariant features has recently been applied to many computer vision problems, including image retrieval [10, 12], object recognition [6], wide baseline matching [8], building panoramas [1], image based localization [16] and video data mining [15]. In this approach, local invariant features are detected independently in each image and then the features of one image are matched against the features of other images by comparing respective feature descriptors. The matched features can subsequently be used to indicate presence of a particular object, to vote for a particular image, or as tentative correspondences for epipolar geometry estimation.

The most important parameter influencing the success of this approach to image matching is the type of local invariant features used. A good feature should be repeatedly and accurately detected, so that the respective image regions overlap exactly the same scene area in two images. In addition to this, the feature detection should be computationally efficient, the number of features detected per image area should be similar for all types of scenes, and, image regions of all sizes should be detected [9]. In several recent surveys (e.g. [9, 4]), maximally stable extremal regions (MSER) [8] have been selected as the most suitable type of local invariant features for many types of applications (e.g., image retrieval and wide baseline matching).

Problem statement. To the joy of city dwellers, there are plenty of trees, flowers, and other vegetation even in urban environments, and therefore also in images of such environments. We have experimentally observed that the density of MSER features detected on vegetation is much larger than the density of MSER features on other types of scenes. MSER features detected on vegetation correspond to leaves, overlapping leaves and branches, or background (e.g., sky) enclosed by leaves and branches (see Fig. 1 for examples). Such image structures have very flexible shape that is very sensitive to viewpoint and scene changes (e.g., wind). Once the features are detected, they are subsequently matched by comparing descriptions of their visual content. Even the most successful feature descriptors (e.g. SIFT descriptor [6]) can provide invariance to affine and photometric transformations only, and cannot provide invariance to deformations of shape of underlying image structures. Based on all of these, we posit a hypothesis that the efficiency of applications using MSER features would substantially improve if we would prune features detected on vegetation. To verify our claim we have made a proof-of-concept experiment (see Sec. 2) in which we compared success of image retrieval once using original query images and the other time using query images with vegetation manually segmented out. The proof-of-concept experiment has shown that efficiency of the application has substantially improved, while the accuracy of image matching has actually increased.

Our contribution. The first contribution of this paper is a new application of image segmentation algorithms. While traditionally the goal of image segmentation was to provide

Figure 1: Vegetation segmentation for boosting performance of MSER feature detector: (a) Input image and (b) the detected MSER features (depicted as ellipses fitted to the detected maximally stable extremal regions). Please note, that out of 5195 detected MSER features, 2130 are located on vegetation. In (c) and (d) two image sections of the same size are shown. The section in (c) includes lots of vegetation and within this section 777 MSER features are detected. The section in (d) incudes much less vegetation and within this section only 200 features are detected. For clarity sake, only every fourth MSER feature is shown.

a figure/ground segmentation for object recognition [13] or semantic segmentation to assist humans [20], we propose to use image segmentation in order to boost performance of local invariant feature detectors. In particular, we have analyzed the performance of MSER feature detector and we have found out that we can prune all features detected on vegetation to gain a substantial speed-up while accuracy of image matching might even increase. The second contribution of this paper is an adaption of the image segmentation method of [18] to the problem of vegetation segmentation. The image segmentation method of [18] is based on singular value decomposition (SVD) of local image patches, where the sum of the smaller singular values describes the high frequency part of the patch and is, in our adaption of the method, used to prune MSER features detected on vegetation.

Organization of the paper. First, we present a proof-of-concept experiment in Sec. 2. Then in Sec. 3 we describe the image segmentation method of [18] and our adaptation of this method to the problem of vegetation segmentation. The performance of the approach is evaluated in Sec. 4 by conducting state-of-the-art image retrieval approaches on a large image set. Finally, Sec. 5 concludes the paper.

2 Proof-of-concept experiment

To verify our claim, that the MSER feature detector could be improved by pruning features detected on vegetation, we did a proof-of-concept experiment using the publicly available Ljubljana urban image data set[1]. First, we manually segmented out the vegetation from the query images. By comparing the number of MSER features detected on query images with vegetation included or segmented out, we found out that 40% of all MSER features are detected on vegetation. Then we performed the image retrieval task using the method of [12] on all 48 query images, once using all MSER features detected and a second time with MSER features detected on vegetation excluded. The success of image retrieval was measured using the scoring defined in [12]. The exclusion of MSER features detected on vegetation resulted in 67% faster image retrieval and, as is evident from Fig. 5, the exclusion of MSER features detected on vegetation did not decrease the accuracy of image retrieval, but on the contrary, it actually produced slightly better results over the method in [12]. When we compared the results of image retrieval using the other two methods used in [12], i.e., aug-

[1] Ljubljana urban image data set is available online at http://vicos.fri.uni-lj.si/LUIS34/

mented k-NN matching[2] and 1-NN matching, we noticed that the accuracy of image retrieval using these two methods was substantially improved.

3 Automatic segmentation of vegetation

Vegetation can be well characterized using just texture information and can therefore be detected by one of the unsupervised segmentation methods based on texture cues. There are several standard methods for image segmentation which perform well in general. For automatic segmentation of vegetation we have considered Normalized Cuts of [14], Level Set method [5], Jesg [3], Mean Shift [2], and the algorithm of [18, 17].

Normalized Cuts method uses the difference of offset Gaussian with different scale and orientation as features for texture segmentation. This and other methods that use information of gradient or edge filters as a texture descriptor, fire strongly on brightness edges of non-texture regions while detecting the texture regions. Therefore, they are not suitable for our application since we have both textured and non textured regions together in the most outdoor images. Fig. 2(c) shows the example of Normalized Cuts segmentation result. Moreover, the Normalized Cuts method is quite slow. For instance it took more than 5 minutes to produce segmentation shown in Fig. 2(c) which is unacceptable for our application. *Level Set* [5] is another technique used for image segmentation. It uses the information of a gradient vector flow with an edge-preserving property. This helps preventing snakes from passing over weak boundaries. As it is shown in Fig. 2(b), also this method is very sensitive to brightness edges and it is not fast enough for our purposes. Other examples of proven segmentation methods are Jesg [3] and Mean Shift [2]. These methods are very general and try to segment the whole image into several segments using the information about texture or about texture combined with color. Both of these methods depend on several parameters which need tuning for each image to perform well. The segmentation results of these methods on our test image are given in Fig. 2(a) and (d).

Regions of vegetation are highly textured regions in an image and can therefore be detected by classifying the texture properties of local image patches. Image regions holding high spatial frequencies are very likely to contain vegetation. We adapted the algorithm of [18, 17] for this purpose. The key idea is to investigate the singular values of matrices formed directly from grayvalues of local image patches. More specifically, the grayvalues of a square patch around a pixel are put into a matrix of the same size as the original patch. The texture descriptor is computed as the sum of some singular values of this matrix. The largest singular value encodes the average brightness of the patch and is thus not useful as a texture description. However, the smaller singular values encode high frequency variations characteristic of visual texture. To extract this information a matrix W is formed by the grayvalues in a $w \times w$ neighborhood centered at a pixel. The singular values of this matrix are computed and sorted in decreasing order, $\{\|\sigma_1\|, \|\sigma_2\|, ..., \|\sigma_w\|\}$. Then the *Texture-transform*, Γ, at each pixel is defined as the average of the smallest singular values,

$$\Gamma(l) = \sum_{k=l}^{w} \sigma_k \quad , \quad 1 \leq l \leq w \ . \tag{1}$$

To avoid that the transform reacts to the brightness of the local image patch, the largest few singular values are ignored.

For the application of vegetation segmentation we use this method, as it only produces a scalar value for each pixel position. Therefore, the binarization step for segmentation is done just with a single threshold value. In addition, the sliding window singular value decomposition can be implemented very efficiently in CPU and GPU [17] in real-time. To reduce the computational cost even more, the descriptor is not computed at all pixel sites, but at every n^{th} pixel in horizontal and vertical directions. Since the descriptor tends to vary slowly this does not cause significant loss in accuracy. The only parameters to be set are the size of the window w, the sparsity of the descriptor extraction n and the number of used singular values that describe the high frequency part of texture. Other advantage of the Texture-transform is its robustness in illumination changes. On top of that, an important reason which motivate us to select the Texture-transform as a texture descriptor instead of some other method is that it responses to small scale structures inside of a region as opposed to texture arising from brightness edges induced for instance by object boundaries. This is in contrast with many of existing texture descriptors, namely most of edge filter texture descriptors such as Gabor-like filters [7], Markov random fields [19] and local binary pattern [11] texture descriptors. Also computational demands of these algorithms make them unsuitable for our target application as an image preprocessing tool enabling better performance of MSER feature detector.

For the Texture-transform in our application we selected a window size of $w = 32$, calculating the function each 8^{th} pixel and set $l = 20$ to consider the 12 smallest singular values. Then we scale the Texture-transform to interval $[0, 1]$ and set the threshold value to 0.3. The binary image is then median filtered to get rid of small regions and noise.

The results of the proposed Texture-transform are given in Fig. 3. The Texture-transform is shown as a grayscale image, which is then binarized manually and also with the proposed fixed threshold. When compared to the segmentation results presented in Fig. 2, it is clear that none of the other state-of-the-art segmentation methods is able to clearly segment only vegetation regions.

4 Experimental results

We have evaluated the increase in performance of MSER feature detector, due to vegetation segmentation, on a challenging publicly available image data set, the Ljubljana urban image data set collected for the purpose of image based localization [12]. This image data set consists of 612 reference images of an urban environment. For each of the

[2] For the sake of clarity, we will use the term 3-NN matching and $r_{max} = 26°$ for the augmented k-NN matching method of [12].

(a) Jesg [3] (b) Level Set [5] (c) Normalized Cuts [14] (d) Mean Shift [2]

Figure 2: Comparison of other well-known segmentation methods shown for the input image in Fig. 1. For all methods we used the original source code from the authors homepage. We have shown the best results according to application of vegetation segmentation.

(a) (b) (c)

Figure 3: Illustration of the Texture-transform used for vegetation segmentation: (a) The result of the Texture-transform using the image shown in Fig. 1 and (b) the thresholded and median-filtered Texture-transform binary mask superimposed on the original image. For this example the threshold was set manually. In (c) the segmentation result is given using the automatic threshold procedure.

48 query images between 10 and 15 best matching reference images were manually selected that share the most of the scene with the query image, and these were taken as groundtruth reference images. In addition, we have manually segmented out vegetation from the query images. It has turned out that in 28 query images at least some vegetation is present, with 17% of pixels belonging to vegetation on average. On pixels belonging to vegetation 40% of all MSER features were detected.

Vegetation segmentation. We have compared the performance of the presented method for automatic segmentation of vegetation by measuring the overlap between manually and automatically acquired segmentation masks. We have defined the overlap between segmentation masks as a ratio between the intersection and union of the two masks. The results of the automatic segmentation show that the average overlap over all 48 images between manual and automatic vegetation segmentation is 33%. An example of the difference between manual and automatic segmentation of vegetation is shown in the first column of Fig. 4.

Image retrieval results. The evaluation of the effect of removing regions with vegetation on accuracy of image retrieval was evaluated by retrieving n most similar reference images once using original query images, once using query images with vegetation manually segmented out, and once using query images with vegetation automatically segmented out. The image retrieval score was calculated by counting how many of the groundtruth reference images were among them. As in [12], the score was upper bounded to 5 because of a weak boundary between reference images selected and those not selected as the groundtruth reference images but still having at least part of the scene in common with the query image. Image retrieval was performed by first detecting MSER features that were subsequently described by the SIFT descriptor. Local feature descriptors detected on the query image were individually matched with features detected in the reference images using either Meaningful Nearest Neighbors [12], 1-NN matching or 3-NN matching method. As in [12], the 3-NN matching was additionally optimized to give best matching results, by using only the nearest neighbors that are within radius (r_{max}) of 26°. When employing the concept of Meaningful Nearest Neighbors, only features that are substantially more similar to the query feature than the rest of features detected in the reference images are used as weighted votes for the respective reference image. For 1-NN the similarity of a reference image to a query image was measured by simply counting the number of nearest neighbors voting for a particular reference image, while for 3-NN matching the number of nearest neighbors voting for a particular reference image was divided by the square root of the number of features detected in that reference image, to account for considerable variation of number of features detected in different reference images.

As already stated, 40% of MSER features are detected on vegetation. When we have pruned these MSER features using manual segmentation, the speed of image retrieval increased by 67%. The automatic procedure for vegetation segmentation pruned 25% of MSER features, which resulted in 33% faster image retrieval.

The results presented in Fig. 5 show that the accuracy of image retrieval has actually increased when vegetation is segmented out from the query images. Already [10] argued that large scale image retrieval should more or less put the right images at the top of the images returned by the query. Therefore we will concentrate our attention in analysis of

Figure 4: An example of improved image retrieval results when vegetation is segmented out and 1-NN matching strategy is used. The first column shows query images with corresponding masks while in the second to sixth column, the five best matching images are shown for the original query image (first row), for a query image with vegetation segmented out manually (second row), and automatically (third row). With vegetation present, all five retrieved images are incorrect. With vegetation excluded, the accuracy improves substantially.

results to the leftmost values in plots of Fig. 5, where the results are given for the case when just 5 reference images are retrieved. Here we can see that the performance of image retrieval when using Meaningful Nearest Neighbors is almost identical, while for the 1-NN and 3-NN the increase in accuracy is substantial. When we compare the results of automatic versus manual segmentation, we can see that also automatic segmentation of vegetation provides some increase in accuracy, though it is not as pronounced as it is the case when vegetation is segmented out manually. An example of improved image retrieval results when vegetation is segmented out and 1-NN matching strategy is used, is given in Fig. 4.

In experiments presented in this section we have extensively used the approximate nearest neighbors search method of [12]. Despite the fact that we were using high-dimensional SIFT descriptors, this method enabled us to perform experiments ten-times faster than if we would be using the exhaustive search. We have repeated all the experiments of this section also using the exact nearest neighbors search and we did not notice any degradation in accuracy of image retrieval incurred by the use of this approximate nearest neighbors search method.

5 Conclusions

In this paper, we have presented a new application of image segmentation algorithms and we have adapted the image segmentation method of [18] to the problem of vegetation segmentation. While traditionally the goal of image segmentation was to provide a figure/ground segmentation for object recognition or semantic segmentation to assist humans, we have proposed to use image segmentation in order to boost performance of local invariant feature detectors. In particular, we have analyzed the performance of MSER feature detector and we have found out that we can prune all features detected on vegetation to gain a 67% speed-up while accuracy of image matching might even increase.

The image segmentation method of [18] that we have adapted to the problem of vegetation segmentation have enabled us to prune 25% of MSER features resulting in 33% faster performance of image retrieval. It is evident from only 33% overlap of segmentation masks acquired manually and by automatic procedure, that we are quite far away from a perfect method for automatic vegetation segmentation. Our future research is directed not only towards better methods for vegetation segmentation, but also towards methods to detect other types of scenes where detectors of local invariant features perform poorly (e.g., cobblestones).

When we compared the results of image retrieval using the Meaningful Nearest Neighbors, 1-NN, and 3-NN matching we have found out that the results of image retrieval using Meaningful Nearest Neighbors were only slightly different when vegetation was excluded, while for the 1-NN and 3-NN matching methods the performance was substantially better with vegetation excluded. This finding further supports the claim given in [12], that elementary methods for matching local invariant features such as threshold-based matching and nearest neighbor(s) matching are inadequate and the more sophisticated methods (e.g., Meaningful Nearest Neighbors of [12]) should be used for image matching based on local invariant features.

Acknowledgement

This research has been supported in part by the following funds: Research program Computer Vision P2-0214 (RS) and EU FP6-511051-2 project MOBVIS.

References

[1] Matthew Brown and David G. Lowe. Automatic panoramic image stitching using invariant features. *International Journal of Computer Vision*, 74(1):59–73, August 2007.

[2] Dorin Comaniciu and Peter Meer. Mean shift analysis and applications. *IEEE International Conference on Computer Vision*, 2:1197–1203, 1999.

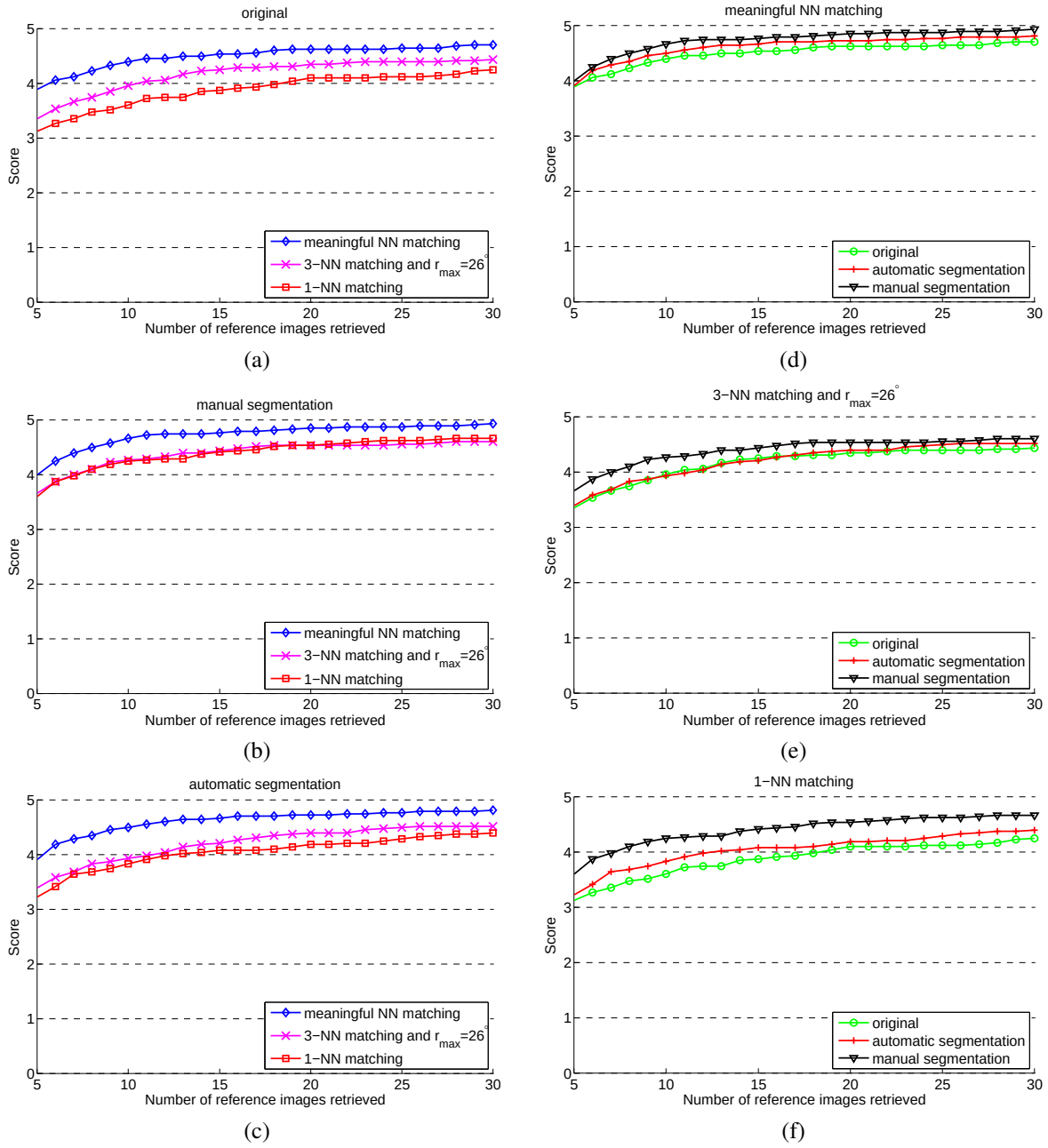

Figure 5: Image retrieval results shown for three different matching strategies using the original query images, using the query image with manual segmentation of vegetation, and using the query images with vegetation segmented out automatically. On the left the different matching strategies are compared using the different types of segmentation, while on the right the different types of segmentation are compared using different matching strategies.

[3] Yining Deng and B. S. Manjunath. Unsupervised segmentation of color-texture regions in images and video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(8):800–810, 2001.

[4] Fritz Fraundorfer and Horst Bischof. A novel performance evaluation method of local detectors on non-planar scenes. In *Workshop Proc. Empirical Evaluation Methods in Computer Vision (CVPR)*, volume 3, pages 33–43, 2005.

[5] Chunming Li, Jundong Liu, and Martin D. Fox. Segmentation of edge preserving gradient vector flow: An approach toward automatically initializing and splitting of snakes. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 162–167, 2005.

[6] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, November 2004.

[7] Jitendra Malik, Serge Belongie, Thomas K. Leung, and Jianbo Shi. Contour and texture analysis for image segmentation. *International Journal of Computer Vision*, 43(1):7–27, 2001.

[8] Jiri Matas, Ondrej Chum, Martin Urban, and Tomas Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*,

22(10):761–767, September 2004.

[9] Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, Jiri Matas, Frederik Schaffalitzky, Timor Kadir, and Luc Van Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1-2):43–72, 2005.

[10] David Nistér and Henrik Stewénius. Scalable recognition with a vocabulary tree. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2161–2168, 2006.

[11] Timo Ojala and Matti Pietikainen. Unsupervised texture segmentation using feature distributions. *Pattern Recognition*, 32(3):477–486, March 1999.

[12] Dušan Omerčević, Ondrej Drbohlav, and Aleš Leonardis. High-Dimensional Feature Matching: Employing the Concept of Meaningful Nearest Neighbors. In *IEEE International Conference on Computer Vision*, 2007.

[13] Xiaofeng Ren, Charless C. Fowlkes, and Jitendra Malik. Figure/ground assignment in natural images. In *European Conference on Computer Vision*, volume 2, pages 614–627, 2006.

[14] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8), August 2000.

[15] Josef Sivic and Andrew Zisserman. Video Google: A text retrieval approach to object matching in videos. In *IEEE International Conference on Computer Vision*, volume 2, pages 1470–1477, 2003.

[16] Ulrich Steinhoff, Dušan Omerčević, Roland Perko, Bernt Schiele, and Aleš Leonardis. How computer vision can help in outdoor positioning. In *European Conference on Ambient Intelligence*, volume 4794, pages 124–141. Springer LNCS, November 2007.

[17] Alireza Tavakoli Targhi, Mårten Björkman, Eric Hayman, and Jan-Olof Eklundh. Real-time texture detection using the lu-transform. In *Workshop Proc. Computation Intensive Methods for Computer Vision (ECCV)*, 2006.

[18] Alireza Tavakoli Targhi and Azad Shademan. Clustering of singular value decomposition of image data with applications to texture classification. In *VCIP*, pages 972–979, 2003.

[19] Manik Varma and Andrew Zisserman. Texture classification: are filter banks necessary? In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages II: 691–698, 2003.

[20] Julia Vogel and Bernt Schiele. Semantic modeling of natural scenes for content-based image retrieval. *International Journal of Computer Vision*, 72(2):133–157, April 2007.