

Improving Foreground Segmentations with Probabilistic Superpixel Markov Random Fields

Alexander Schick*

Martin Bäuml†

Rainer Stiefelhagen*†

*Fraunhofer IOSB

†Karlsruhe Institute of Technology

`alexander.schick@iosb.fraunhofer.de`

`{martin.baeuml, rainer.stiefelhagen}@kit.edu`

Abstract

We propose a novel post-processing framework to improve foreground segmentations with the use of Probabilistic Superpixel Markov Random Fields. First, we convert a given pixel-based segmentation into a probabilistic superpixel representation. Based on these probabilistic superpixels, a Markov random field exploits structural information and similarities to improve the segmentation. We evaluate our approach on all categories of the Change Detection 2012 dataset. Our approach improves all performance measures simultaneously for eight different basis foreground segmentation algorithms.

1. Introduction and related work

Segmentation of an image into foreground and background is arguably one of the most important pre-processing steps in many computer vision applications. The goal of change detection, or foreground segmentation, is the separation of the dynamic foreground from the presumably static background. A good segmentation of the relevant image regions can greatly improve the performance of applications building on top of it. For example, people detection is much easier and computationally more efficient when static background is reliably removed.

Because of the importance of change detection, there is a large body of literature and a great number of variations. However, most algorithms can be classified by how they model the background and how they compute the distance of a frame to the background model. The background model is usually described on pixel level, *e.g.* by a simple mean value, by storing a set of samples [4], or by one or multiple Gaussians [9, 16, 17]. Other approaches use self-organizing neural maps [11] or non-parametric density estimation methods [8]. More comprehensive overviews can be found in [5, 6, 14].

It is common to post-process a segmentation to improve results and comparative evaluations can be found in [6, 13].

Typical post-processing methods include noise removal, morphologic operators, median filtering, but also higher-level methods such as saliency or optical flow analysis. Post-processing generally improves the results and can alleviate the differences between algorithms [6]. However, the methods and their parameters must, in general, be chosen carefully depending on the sequence [13].

Independent of the segmentation algorithm or post-processing method, most approaches are based on pixels. But pixels are a result of the sensors we use, not meaningful units by themselves. In addition, they are very susceptible to noise. *Superpixels*, on the other hand, are a higher-level image representation that partitions an image into meaningful regions. Their key property is that they align well with object boundaries. They are more robust to noise than pixels and serve well to represent objects in the image. Therefore, they can be used as atomic primitives in image processing applications. We use an improved variation of the superpixel segmentation algorithm SLIC [2] that was proposed in [15]. [15] generally maintains a lattice-like structure [12] due to the initial rasterization. This means that the superpixels conform to a grid with known and fixed neighborhoods, like pixels in an image. This property is only weakly guaranteed, but we can nevertheless exploit it in the Markov random field (MRF). While there are segmentation approaches based on superpixels [3, 10], we are unaware of superpixels being used in a post-processing framework.

In the remainder of this paper, we will introduce a novel post-processing framework based on Probabilistic Superpixel Markov Random Fields (PSP-MRF) to improve a given foreground segmentation. We show evaluations of eight benchmark algorithms for all categories of the Change Detection 2012 dataset before concluding with a discussion.

2. Probabilistic Superpixel Markov Random Fields

We will first introduce the concept of probabilistic superpixels before incorporating them in a Markov random field.

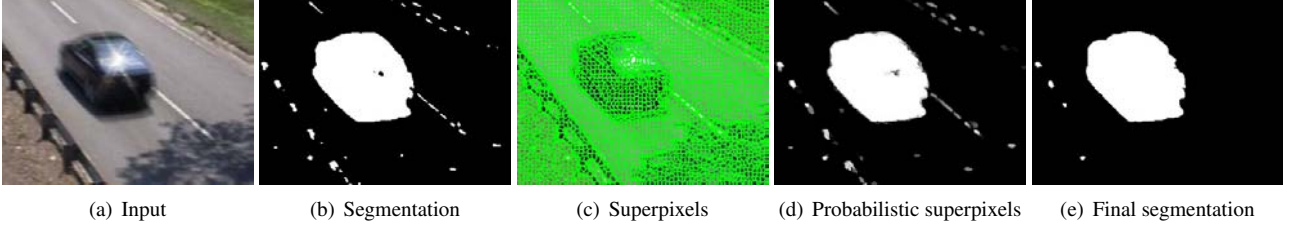


Figure 1. Visualization of the processing pipeline. Based on the input, the superpixels are computed. Combining them with the segmentation leads to probabilistic superpixels with white regions indicating a high foreground probability. The final segmentation shows a clear improvement.

2.1. Probabilistic superpixels

We will now introduce the term *probabilistic superpixel* and explain how to compute them. For a more visual explanation, we refer to Figure 1. A probabilistic superpixel gives the probability that its pixels belong to a certain class. In this paper, we consider two classes: foreground and background. Therefore, a probabilistic superpixel gives the probability of its pixels belonging to the foreground.

Let F be the foreground segmentation of image I and \mathfrak{S} its superpixel segmentation. Let $S \in \mathfrak{S}$ be a superpixel with pixels $p \in S$ and $|S|$ its size. Let $F(p)$ be 1 if pixel p belongs to the foreground and 0 otherwise. Then, the probability of superpixel S belonging to the foreground is given by

$$p(S) = \frac{\sum_{p \in S} F(p)}{|S|}. \quad (1)$$

Probabilistic superpixels have several advantages. Their shape only depends on the image, not on the foreground segmentation. Because they accumulate foreground pixels, they are therefore able to restore the original shape of the objects even if the foreground segmentation contains errors. In addition, they allow to transform any binary segmentation into a probabilistic one. Further, they can also be applied to non-binary inputs and extended to multiple classes. Finally, probabilistic superpixels fit very well into probabilistic frameworks, like Markov random fields, as we will show now.

2.2. Markov random fields

Superpixels usually provide an over-segmentation of the image, and in general foreground objects consist of more than one superpixel. Assuming that nearby superpixels which are similar in appearance also jointly belong to either foreground or background, we make use of neighborhood relationships between superpixels in order to improve the segmentation. Let us denote the neighborhood of superpixel S as \mathcal{N}_S . Note that because of the lattice structure of the superpixels, and considering vertical, horizontal and diagonal connections, each superpixel has exactly eight neighbors. Let $\mathcal{N} = \{(S, T) \mid S \in \mathfrak{S}, T \in \mathcal{N}_S, id(S) < id(T)\}$ be the ordered set of all neighboring superpixels in the image.

Following [7], we define an energy function

$$E(\mathbf{f}_{\mathfrak{S}}) = \sum_{S \in \mathfrak{S}} U_S(f_S) + \sum_{(S, T) \in \mathcal{N}} V_{ST}(f_S, f_T) \quad (2)$$

over the domain of all foreground/background-labelings $\mathbf{f}_{\mathfrak{S}} \in \{0, 1\}^{|\mathfrak{S}|}$. The segmentation problem can now be seen as an energy minimization problem, where we seek the $\mathbf{f}_{\mathfrak{S}}^*$ that minimizes E .

The *unary* term $U_S(f_S)$ captures the likelihood of superpixel S belonging to either foreground or background, and we define it as

$$U_S(f_S) = -\ln(\sigma(p(S), \theta))f_S - \ln(1 - \sigma(p(S), \theta))(1 - f_p), \quad (3)$$

where $p(S)$ is given by Eq. (1), and $\sigma(p(S), \theta) = \min(1, \frac{p(S) \cdot \theta}{0.5})$ is a linear mapping function of the superpixel probability.

The relationship between two neighboring superpixels S and T is modeled by a *pairwise* term

$$V_{ST}(f_S, f_T) = |f_S - f_T| (\lambda_1 + \lambda_2 \cdot e^{-\beta \|\mu_S - \mu_T\|}). \quad (4)$$

It consists of the *Ising prior* $\lambda_1 |f_S - f_T|$ and an *edge-sensitive term*, which gives a higher penalty if the mean colors μ_S and μ_T (computed from the original input image) of neighboring superpixels S and T are not similar. Parameter β models the expected color differences and is estimated for each image as $\beta = (2 * \mathbb{E}(\|\mu_S - \mu_T\|))^{-1}$.

We find the labeling for the superpixels by minimizing E , i.e. $\mathbf{f}^* = \arg \min_{\mathbf{f}} E$, which can be done efficiently via graph cuts [7]. Since the number of superpixels is significantly lower than the number of original pixels (by a factor of ~ 25 in our experiments), the graph cut computation does not present a computational bottleneck.

3. Experimental setup

Our post-processing framework works on existing foreground segmentations. To cover a wide range of algorithms, we used pre-computed results of benchmark algorithms made available in the Change Detection 2012

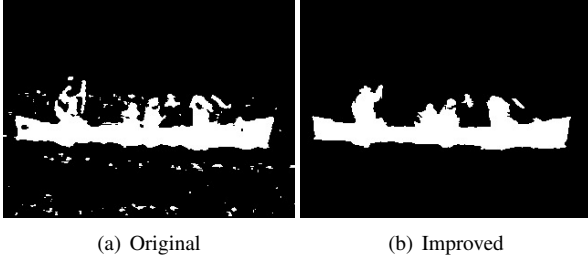


Figure 2. Original and improved segmentation ("canoe"). Our approach reduces noise and fills holes, while maintaining the shape of the objects due to the superpixels.

dataset [1]. The algorithms are Euclidean distance [6] (EURL), GMM KaewTraKulPong [9] (GMM_Kaew), GMM Stauffer and Grimson [16] (GMM_Stau), GMM Zivkovic [17] (GMM_Zivk), KDE [8], Mahalanobis distance [6] (MAHA), SOBS [11], and ViBe [4].

Please note that these results were already post-processed with a 5x5 median filter. We tried to improve them further by applying common post-processing methods (*e.g.* various combinations of morphologic operators). While this led to an improvement of some performance measures (*e.g.* precision), it was at the cost of decreasing other performance measures (*e.g.* recall). We will show in Section 4 that our method simultaneously improves all performance measure of all benchmark algorithms.

We applied our post-processing framework to all categories and evaluated the results with the tools made available by [1]. For evaluation, we used the performance measures used in the Change Detection 2012 challenge [1]: recall, specificity, false positive rate (FPR), false negative rate (FNR), percentage of bad classification (PBC), F-measure, and precision. We first segmented each input frame into superpixels. Then, based on the existing foreground segmentation, we computed the probabilistic superpixels which were then used as input for the MRF. The result of the MRF is the final post-processed segmentation that we used in the evaluation. For the superpixel segmentation, we chose a superpixel size of 25 pixels to capture smaller objects and compactness parameter $\alpha = 0.9$ as recommended in [15]. For the MRF, we chose $\lambda_1 = 0.3$, $\lambda_2 = 3$, and $\theta = 0.35$. The parameters were kept constant for all benchmark algorithms and for all categories.

4. Evaluation

Table 1 shows the summary results over all categories for each benchmark algorithm and for the improved segmentation with the proposed framework. The proposed PSP-MRF achieves the best results for all algorithms and in all performance measures. It equally improves recall and precision which also leads to an improvement for the F-measure. Both the false positive and false negative rates

are reduced and consequently also the PBC, while the specificity is slightly improved. Note that the span of some of the performance measures is quite narrow, *e.g.* the span of the F-measure is only 0.12 over all benchmark algorithms.

A closer analysis (Table 2) of the improvements shows that the categories "dynamic background" and "camera jitter" benefit the most which is due to the neighboring relations modeled by the MRFs. The "shadow" category also benefits strongly from our improvements mainly due to an increased recall rate without a loss in precision. The "baseline" category benefits the least from the PSP-MRFs, mainly because the results were already very good due to easy video sequences. Due to space limitations, the detailed results of all algorithms are available for download ¹.

Figures 1, 2, and 3 show qualitative examples of the effect of PSP-MRFs. Noisy regions can be reduced even if they form relatively large segments. Holes in the segmentation are also closed. One of the biggest benefits, however, is the fact that the shape of the objects is maintained after pre-processing because the superpixels capture the meaningful object boundaries in the images.

In our prototype implementation, this post-processing framework achieves up to ten frames per second. There are many aspects that can be parallelized, and we are confident that with careful optimization a true real-time implementation is possible.

5. Conclusion

We proposed a novel post-processing framework to improve foreground segmentations based on Probabilistic Superpixel Markov Random Fields. We evaluated our method on all categories of the Change Detection 2012 dataset for eight benchmark algorithms and showed continuously better results for all performance measures.

In future work, we want to further investigate the effects of the underlying superpixel segmentation and incorporate temporal relationships between frames into the Markov random field.

6. Acknowledgments

This work was partially supported by the FhG Internal Programs under Grant No. 692026 and by the German Federal Ministry of Education and Research (BMBF) under Contract No. 01ISO9052E. The views expressed herein are the authors responsibility and do not necessarily reflect those of BMBF.

¹<http://cvhci.anthropomatik.kit.edu/projects/pspmrf>

Algorithm	Recall	Specificity	FPR	FNR	PBC	F-Measure	Precision
Eucl [6]	0.7048	0.9692	0.0308	0.0169	4.3465	0.6111	0.6223
Eucl [6] + PSP-MRF	0.7252	0.9699	0.0301	0.0162	4.1821	0.6350	0.6509
GMM _{K_{ae}w} [9]	0.5072	0.9947	0.0053	0.0291	3.1051	0.5904	0.8228
GMM _{K_{ae}w} [9] + PSP-MRF	0.5221	0.9948	0.0052	0.0284	3.0252	0.6043	0.8359
GMM _{Stau} [16]	0.7108	0.9860	0.0140	0.0202	3.1046	0.6623	0.7012
GMM _{Stau} [16] + PSP-MRF	0.7302	0.9869	0.0131	0.0196	2.9623	0.6779	0.7202
GMM _{Ziko} [17]	0.6964	0.9845	0.0155	0.0193	3.1504	0.6596	0.7079
GMM _{Ziko} [17] + PSP-MRF	0.7213	0.9855	0.0145	0.0181	2.9520	0.6821	0.7284
KDE [8]	0.7442	0.9757	0.0243	0.0138	3.4602	0.6719	0.6843
KDE [8] + PSP-MRF	0.7655	0.9760	0.0240	0.0128	3.3419	0.6899	0.7039
Maha [6]	0.7607	0.9599	0.0401	0.0110	4.6631	0.6259	0.6040
Maha [6] + PSP-MRF	0.7784	0.9602	0.0398	0.0100	4.5365	0.6524	0.6395
SOBS [11]	0.7882	0.9818	0.0182	0.0094	2.5642	0.7159	0.7179
SOBS [11] + PSP-MRF	0.8037	0.9830	0.0170	0.0089	2.3937	0.7372	0.7512
ViBe [4]	0.6821	0.9830	0.0170	0.0176	3.1178	0.6683	0.7357
ViBe [4] + PSP-MRF	0.7113	0.9837	0.0163	0.0161	2.9018	0.7006	0.7733

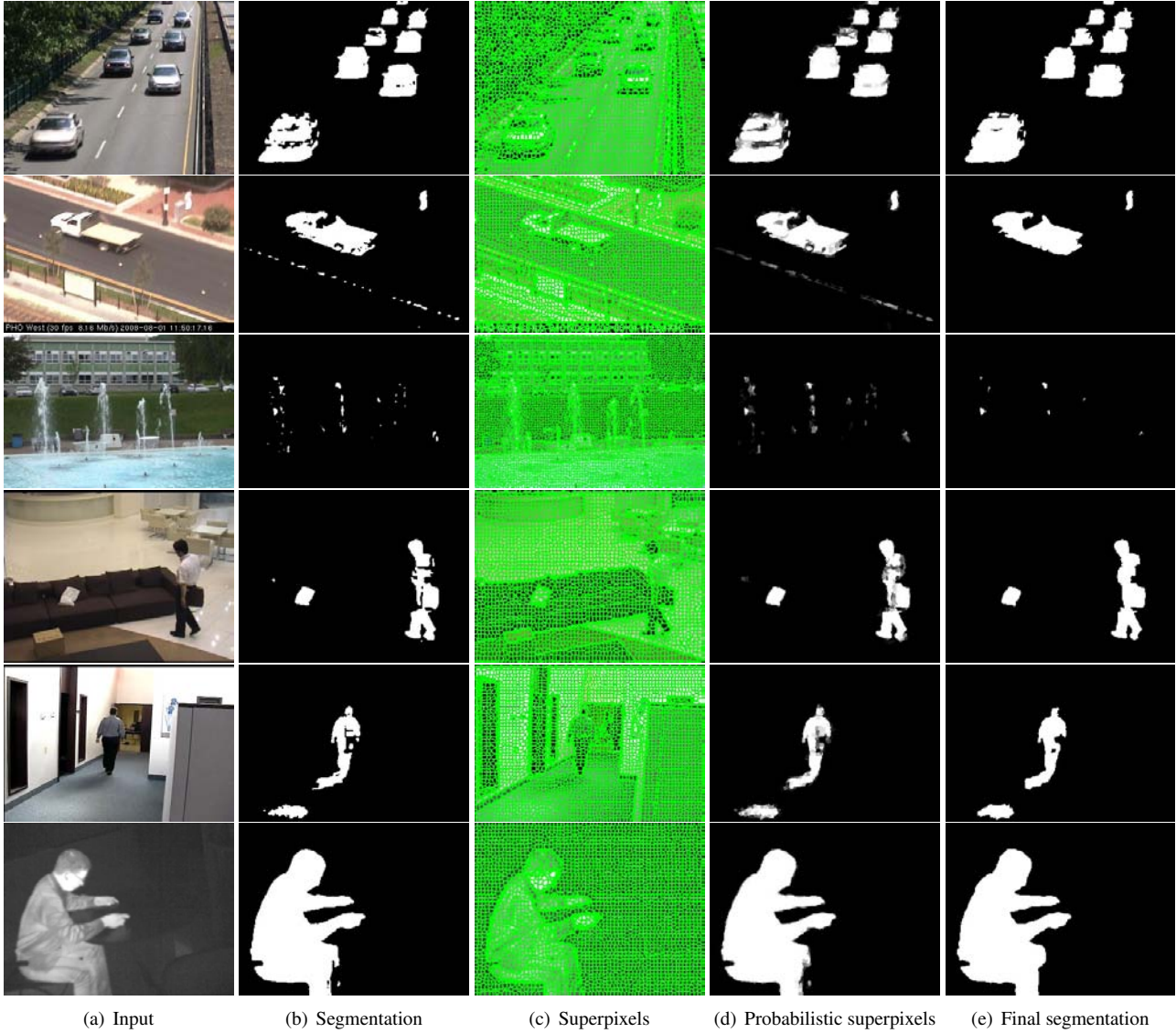
Table 1. Evaluation results for eight benchmark algorithms averaged over all categories of the Change Detection 2012 dataset. Each double row shows the results of the original segmentation algorithm and the improved segmentation generated by the proposed Probabilistic Superpixel Markov Random Field (PSP-MRF).

Category	Recall	Specificity	FPR	FNR	PBC	F-Measure	Precision
Baseline	0.9193	0.9980	0.0020	0.0026	0.4332	0.9251	0.9313
Baseline (PSP-MRF)	0.9319	0.9978	0.0022	0.0021	0.4127	0.9289	0.9261
Dyn. Back.	0.8798	0.9843	0.0157	0.0009	1.6367	0.6439	0.5856
Dyn. Back. (PSP-MRF)	0.8955	0.9859	0.0141	0.0006	1.4514	0.6960	0.6576
Camera Jitter	0.8007	0.9787	0.0213	0.0075	2.7479	0.70860	0.6399
Camera Jitter (PSP-MRF)	0.8211	0.9825	0.0175	0.0064	2.2781	0.7502	0.7009
Interm. Object Motion	0.7057	0.9507	0.0493	0.0183	6.1324	0.5628	0.5531
Interm. Object Motion (PSP-MRF)	0.7010	0.9530	0.0470	0.0200	6.0594	0.5645	0.5727
Shadow	0.8355	0.9836	0.0164	0.0083	2.3318	0.7717	0.7219
Shadow (PSP-MRF)	0.8736	0.9829	0.0171	0.0067	2.2414	0.7907	0.7281
Thermal	0.5888	0.9956	0.0044	0.0188	2.0983	0.6834	0.8754
Thermal (PSP-MRF)	0.5991	0.9962	0.0038	0.0173	1.9189	0.6932	0.9218

Table 2. Detailed evaluation of the SOBS algorithm for all categories of the Change Detection 2012 dataset. Results for other algorithms will be made available. Each double row shows the results of SOBS and the improved segmentation generated by the proposed Probabilistic Superpixel Markov Random Field (PSP-MRF).

References

- [1] <http://www.changedetection.net/>. In *Workshop on Change Detection 2012*.
- [2] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. SLIC Superpixels. Technical report, École Polytechnique Fédérale de Lausanne, 2010.
- [3] A. Ayvaci and S. Soatto. Motion Segmentation with Occlusions on the Superpixel Graph. In *Workshop on Dynamical Vision*, pages 727–734, 2009.
- [4] O. Barnich and M. Van Droogenbroeck. ViBe: A universal background subtraction algorithm for video sequences. *Transactions on Image Processing*, 20(6):1709–1724, June 2011.
- [5] Y. Benezeth, P. Jodoin, B. Emile, H. Laurent, and C. Rosenberger. Review and evaluation of commonly-implemented background subtraction algorithms. In *ICPR*, 2008.
- [6] Y. Benezeth, P.-M. Jodoin, B. Emile, H. Laurent, and C. Rosenberger. Comparative study of background subtraction algorithms. *Journal of Electronic Imaging*, 19(3):1–12, 2010.
- [7] Y. Boykov and G. Funka-Lea. Graph Cuts and Efficient N-D Image Segmentation. *IJCV*, 70(2):109–131, 2006.
- [8] A. Elgammal, D. Harwood, and L. Davis. Non-parametric



(a) Input (b) Segmentation (c) Superpixels (d) Probabilistic superpixels (e) Final segmentation

Figure 3. Additional examples of the processing pipeline. Each row shows one example from each category. The categories (and videos) are from top to bottom: baseline (highway), camera jitter (boulevard), dynamic background (fountain01), intermittent object motion (sofa), shadow (cubicle), and thermal (library). The columns are identical to Figure 1: input image, input segmentation based on SOBS, superpixel segmentation, probabilistic superpixels, and final segmentation.

Model for Background Subtraction. In *ECCV*, 2000.

- [9] P. KaewTraKulPong and R. Bowden. An Improved Adaptive Background Mixture Model for Real-time Tracking with Shadow Detection. In *European Workshop on Advanced Video Based Surveillance Systems*, pages 1–5, 2001.
- [10] L. Lu and G. Hager. Dynamic Foreground / Background Extraction from Images and Videos using Random Patches. In *NIPS*, pages 929–936, 2006.
- [11] L. Maddalena and A. Petrosino. A Self-Organizing Approach to Background Subtraction for Visual Surveillance Applications. *Transactions on Image Processing*, 17(7):1168–1177, July 2008.
- [12] A. P. Moore, S. J. D. Prince, J. Warrell, U. Mohammed, and G. Jones. Superpixel Lattices. In *CVPR*, 2008.
- [13] D. H. Parks and S. S. Fels. Evaluation of Background Subtraction Algorithms with Post-processing. In *Advanced Video and Signal Based Surveillance*, 2008.
- [14] M. Piccardi. Background subtraction techniques: a review. In *Systems, Man and Cybernetics*, pages 3099–3104, 2004.
- [15] A. Schick, M. Fischer, and R. Stiefelhagen. Measuring and Evaluating the Compactness of Superpixels. Manuscript submitted for publication. 2012.
- [16] C. Stauffer and W. Grimson. Adaptive background mixture models for real-time tracking. In *CVPR*, 1999.
- [17] Z. Zivkovic and A. S. Group. Improved Adaptive Gaussian Mixture Model for Background Subtraction. In *ICPR*, 2004.