Global Similarity with Additive Smoothness for Spectral Segmentation

Dahl, Vedrana Andersen; Dahl, Anders Bjorholm

Published in:
Scale Space and Variational Methods in Computer Vision

Link to article, DOI:
10.1007/978-3-030-22368-7_28

Publication date:
2019

Document Version
Early version, also known as pre-print

Link back to DTU Orbit

Citation (APA):
Dahl, V. A., & Dahl, A. B. (2019). Global Similarity with Additive Smoothness for Spectral Segmentation. In *Scale Space and Variational Methods in Computer Vision* (pp. 357-368). Springer. Lecture Notes in Computer Science, Vol.. 11603 https://doi.org/10.1007/978-3-030-22368-7_28

Global Similarity with Additive Smoothness for Spectral Segmentation

Vedrana Andersen Dahl[0000−0001−6734−5570] and
Anders Bjorholm Dahl[0000−0002−0068−8170]

Department of Applied Mathematics and Computer Science,
Technical University of Denmark, Kongens Lyngby, Denmark
{vand,abda}@dtu.com

Abstract. Faithful representation of pairwise pixel affinities is crucial for the outcome of spectral segmentation methods. In conventional affinity models only close-range pixels interact, and a variety of subsequent techniques aims at faster propagation of local grouping cues across long-range connections. In this paper we propose a general framework for constructing a full-range affinity matrix. Our affinity matrix consists of a global similarity matrix and an additive proximity matrix. The similarity in appearance, including intensity and texture, is encoded for each pair of image pixels. Despite being full-range, our similarity matrix has a simple decomposition, which exploits an assignment of image pixels to dictionary elements. The additive proximity enforces smoothness to the segmentation by imposing interactions between near-by pixels. Our approach allows us to assess the advantages of using a full-range affinity for various spectral segmentation problems. Within our general framework we develop a few variants of full affinity for experimental validation. The performance we accomplish on composite textured images is excellent, and the results on natural images are promising.

Keywords: image segmentation, spectral methods, affinity matrix

1 Introduction

Spectral clustering is fundamental in several popular methods for unsupervised image segmentation [13, 2, 7]. The outcome of spectral segmentation depends largely on pairwise pixel affinities. In general, there are two terms contributing to affinities: a similarity term (how alike are the pixels values) and a proximity term (how spatially close are the pixels).

In the original formulation of normalized cuts [13] similarity and proximity are both modelled using a Gaussian similarity function spanning a certain scale. The two contributions are multiplied, so only pairs of pixels that are within the distance range and within the intensity range get assigned non-zero affinity. The reasoning behind is that close-by pixels with similar intensity value are likely to belong to one object. An argument in favor of short-range interactions is the desire to reduce computational cost of the eigenvalue problem by using sparse matrices.

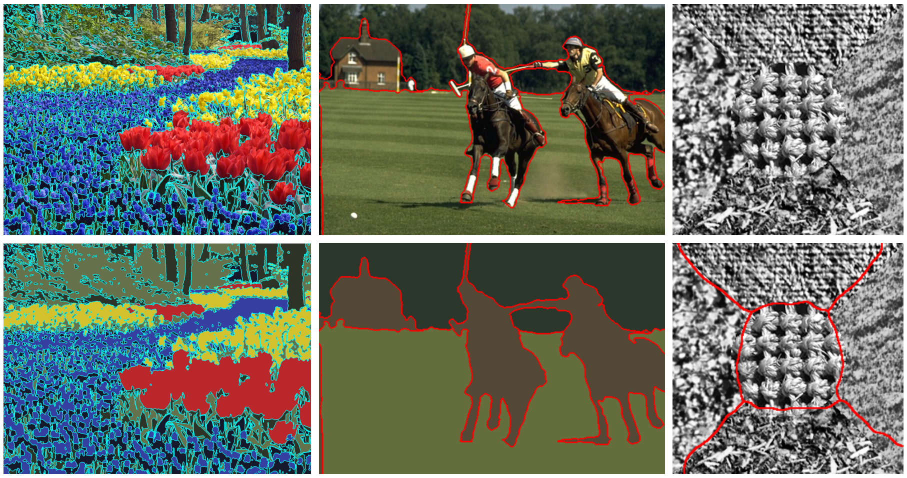

Fig. 1. The left column shows spectral segmentation of an image of size 2400 × 3200 using full-range affinity, middle column is an image from the Berkeley Segmentation Dataset [1] and right is a composed texture from [12]. All images are segmented using no training data.

Since longer range interactions generally make segmentation better, multiscale model [2] combines both coarse and fine-level details, but still on a fixed neighborhood radius. Another popular approach is to use superpixels to propagate grouping cues across larger image regions [7, 16, 17]. Such aggregating methods depend on the quality of initial oversegmentation.

Instead of purely merging regions, the method proposed in [6] utilizes initial oversegmentation to propagate local affinities in the affinity matrix. In a single optimization step all pairwise affinities are estimated, yielding impressive segmentation results. The approach is however still based on propagating, and not directly modelling full-range affinities.

Further challenge involves modelling pairwise similarity for images with regions characterized by cues like boundary and texture. In the intervening contour approach [10, 2] the pairwise pixel affinity is extended with a term which measures the magnitude of image edges on the straight line connecting two pixels. Furthermore, textons are extracted from the image to define texture cue, but still only close-by pixels are connected.

In our approach, pairwise pixel similarity and pairwise pixel proximity are treated more independently. We model the full-range pairwise similarity, so pairs of similar pixels have positive affinity regardless of their distance. Pixel similarity is defined in terms of a dictionary assignment: two pixels are similar it they link to the same dictionary element (or two similar dictionary elements, or if they link to the dictionary elements in the similar way). This allows for a compact representation of the full-range similarity using a biadjacency matrix, which also makes the computation of eigenvectors efficient. As for the proximity, we incorporate it as an additive smoothness term modelled by the Gaussian similarity function. The reasoning behind is that pixels likely to belong to one region are either similar or close-by. This construction has great advantages when segmenting images containing textured regions. Close-by pixels in highly textured regions need not be similar, but, due to the repetitive nature of the texture, similar pixels will be found in a larger neighborhood, creating many unconnected networks within the textured region. Additive smoothness contributes with additional links, connecting those networks. The resulting general framework is capable of handling both textured and not-textured images.

The main contribution of our paper is an efficient and versatile method for approximating a full affinity matrix. This is useful for segmenting an image into global regions of homogeneous appearance, and may be included in more specialized segmentation engines. Our approach is rather different from other spectral methods, both in the problem statement and in the proposed solution. The aim of this paper is to introduce this novel affinity matrix and to provide insight in its properties and use.

The key contributions of our paper are:

1. For modelling full-range similarity we propose an efficient representation of a similarity matrix based on dictionary assignment.
2. For spectral segmentation we propose using a full-range similarity matrix and additive proximity. Our method has linear time complexity and can handle images with millions of pixels, see Fig. 1, left column.
3. For segmenting natural images we propose similarity matrix based on dictionary containing image patches and normalized image patches, see Fig. 1, middle column.
4. For segmenting textured images we propose similarity matrix based on dictionary containing image patches and SIFT features, see Fig. 1, right column.

2 Method

Our approach follows the common framework of spectral segmentation. To set the stage, we start this section with a brief review of a spectral segmentation using a conventional affinity matrix. After that, we cover the basic principles of constructing our novel full-range similarity matrix, describe few of its variants, and discuss its use.

2.1 Conventional affinity matrix

For an image I containing n pixels, an affinity matrix $\mathbf{W}$ is an $n \times n$ matrix representing a graph build upon the image. Matrix elements $w_{ii'}$ represent a weight of the edge between pixels i and i'. Dividing nodes of this graph in k partitions, using some optimality criteria, provides image segmentation. A popular objective function is normalized cut [13], which can be conveniently rewritten using the graph Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{W}$, where $\mathbf{D} = \mathrm{diag}(\mathbf{W}\mathbf{1}_{n\times 1})$ is a diagonal degree matrix. Relaxation of this discrete NP hard optimization problem yields the solution as the smallest k generalized eigenvectors of the generalized eigenproblem $\mathbf{Lu} = \lambda\mathbf{Du}$. Real-valued eigenvectors lead to a new k-dimensional representation of the image pixels which enhances the clustering properties in the data. Typically k-means algorithm is used for detecting clusters of pixels in this new representation. For a detailed coverage of spectral segmentation we refer to [15]. It also shows that the solution equivalent to normalized cut may be found as largest k eigenvectors of the matrix $\mathbf{D}^{-1/2}\mathbf{W}\mathbf{D}^{-1/2}$ multiplied by $\mathbf{D}^{-1/2}$, which is the approach we choose in practice.

Constructing affinity matrix is a crucial step in spectral segmentation. Conventional approach is to define pixels affinities based on their intensities and positions in the image grid. In [13] and [2] pairwise affinity is defined as a product of intensity similarity term and spatial proximity term

$$w_{ii'} = \begin{cases} \exp\left(-\frac{d_I(i,i')^2}{2\sigma_I^2} - \frac{d_X(i,i')^2}{2\sigma_X^2}\right) & \text{if } d_X(i,i') < r \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where $d_I(i,i') = |I_i - I'_i|$ is the absolute intensity difference and $d_X(i,i') = \|X_i - X'_i\|_2$ is the spatial distance between pixels, r is the radius of the interaction and σ_I and σ_X are parameters of the Gaussian similarity functions. Approach is that close-by pixels with similar intensity value are likely to belong to one object.

2.2 Full-range affinity matrix

Instead of defining affinity as a multiplication of the similarity with the short-range proximity we will define full-range similarity and add the proximity. By replacing multiplication with addition we replace a similar-*and*-close assumption with the similar-*or*-close assumption.

Dictionary assignment. The initial step of our method involves constructing a dictionary and assigning every image pixel to a dictionary element. In the simplest version a dictionary is constructed by collecting pixel intensities from the image and clustering those in m clusters using k-means algorithm. All image pixels are then assigned to the nearest intensity and this assignment is denoted $D(i)$, where $i \in \{1,\ldots,n\}$, $D(i) \in \{1,\ldots,m\}$.

Biadjacency matrix. Dictionary assignment can be represented as a biadjacency matrix $\mathbf{B}$, where an element b_{ij} takes a value 1 if $D(i) = j$ and 0 otherwise. An $n \times n$ matrix $\mathbf{S} = \mathbf{B}\mathbf{B}^{\mathsf{T}}$ is the simplest variant of our full-range similarity matrix. An important property of $\mathbf{S}$ is

$$s_{ii'} = \sum_{j=1}^{m} b_{ij} b_{i'j} = \begin{cases} 1 & \text{if } D(i) = D(i') \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

We say that $\mathbf{S}$ encodes global binary similarity on the basis of dictionary assignment. This full-range similarity and its extensions are fundamental for our method.

A few variants of a full-range similarity. For the simplest segmentation problems the similarity matrix given by (2) might suffice, but generally we want a better description of pixel similarity, and here we describe a few possibilities. Since all variants of full-range similarity may be used in the final segmentation model, we will denote them all by $\mathbf{S}$.

Pixel similarity weighted by dictionary similarity. One extension of the full affinity incorporates an $m \times m$ matrix $\mathbf{A}$ containing pairwise similarity between dictionary elements, computed in a conventional way using Gaussian similarity function as explained in connection with similarity term from Eq. (1). In a weighted similarity matrix $\mathbf{S} = \mathbf{B}\mathbf{A}\mathbf{B}^{\mathsf{T}}$ an element

$$s_{ii'} = \sum_{j=1}^{m} \sum_{j'=1}^{m} a_{jj'} b_{ij} b_{i'j'} = a_{D(i)D(i')} \tag{3}$$

is a similarity between dictionary element $D(i)$, where pixel i is assigned to, and a dictionary element $D(i')$, where pixel i' is assigned to. As such, this version of $\mathbf{S}$ approximates the conventional full pairwise similarity matrix, and the quality of the approximation depends on how well the dictionary represents regions present in the image. To our knowledge this type of a full affinity approximation has not been used in spectral image segmentation.

Dictionary of features and patches. Another extension concerns the dictionary construction. When image regions are characterized by the texture, we want a better representation of pixel similarity. Instead of clustering pixel intensities, we have an option of creating clusters according to any local feature, for example a SIFT feature. In another choice, we extract a quadratic $M \times M$ patch around each image pixel, and use the collected pixel intensities as a feature vector. The dictionary is constructed by clustering patches from the image using k-means algorithm with Euclidian distance.

Dictionary of patch pixels. Using image patches as features opens for another possibility. Instead of linking each pixel to one of m dictionary elements, we can, in an approach similar to [3] and [4], define links between n image pixels and mM^2 dictionary pixels. Each assignment $D(i)$ now contributes with M^2 links between the image and the dictionary. Since every pixel takes one of M^2 positions in an patch, rows of $\mathbf{B}$ now sum to M^2. The motivation for this approach is a desire for a better localization of the boundaries between segments.

A final similarity-proximity model Full-range similarity matrix is a key ingredient to our affinity model for spectral segmentation. However, using similarity alone we risk a very fragmented segmentation, as it ignores the spatial position of image pixels.

Proximity matrix. To enforce the smoothness of the segmentation we define a proximity term which links the close-by pixels according to a Gaussian similarity function of spatial distances. Proximity matrix $\mathbf{P}$ has elements

$$p_{ii'} = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{d_X(i,i')^2}{2\sigma^2}\right) ,$$

where σ is a smoothing radius. For σ small compared to image dimensions, matrix $\mathbf{P}$ will in effect be a sparse matrix.

Affinity matrix. We want to incorporate the proximity matrix to our final model without reducing the range of interactions between pixels. Therefore we add the proximity links to full-range similarity matrix, leading to the affinity matrix

$$\mathbf{W} = \mathbf{S} + \alpha\mathbf{P}$$

where $\mathbf{S}$ is one of the variants of the similarity matrix and $\mathbf{P}$ is the proximity matrix weighted by α.

Normalization. In spectral clustering, normalizing the affinity matrix will favor more balanced segmentation. This is important if degrees of the graph nodes vary considerably. In case of our proximity matrix $\mathbf{P}$, the degree of the nodes is constant except at the boundary, so we only need to normalize $\mathbf{S}$. In similarity matrix $\mathbf{S}$, a pixel is linked to all other pixels belonging to the same dictionary element. This property allows us to define a normalized similarity matrix

$$\mathbf{S} = \mathbf{B}\,\mathrm{diag}(\mathbf{B}\mathbf{1}_{m\times 1})^{-1}\mathbf{B}^{\mathsf{T}}.$$

Eigendecomposition. To calculate leading eigenvectors of $\mathbf{W}$ using power methods such as Lanczos solver [5] we do not need an explicit representation of $\mathbf{W}$. It is sufficient to define the result of the matrix-vector computation $\mathbf{y} = \mathbf{W}\mathbf{x}$ for every $\mathbf{x}$. With minor modifications, depending on the variant of the similarity matrix, we have

$$\mathbf{y} = \mathbf{B}\,\mathrm{diag}(\mathbf{B}\mathbf{1}_{m\times 1})^{-1}\mathbf{B}^{\mathsf{T}}\mathbf{x} + \alpha\mathbf{P}\mathbf{x}\,.$$

Computational efficiency. The similarity term can be efficiently computed by multiplying from right to left. Since $\mathbf{B}$ is $n \times m$ binary and sparse matrix with n non-zero entries, the first and the last multiplication is $\mathcal{O}(n)$. The time complexity of the multiplication in the middle does not depend on n, yielding time complexity of $\mathcal{O}(m)$ for diagonal matrix $\mathrm{diag}(\mathbf{B1})^{-1}$. Had we used a formulation from Eq. (3) multiplication with $\mathbf{A}$ would give complexity of $\mathcal{O}(m^2)$.

The proximity term can also be efficiently computed by arranging values of $\mathbf{x}$ in an image grid and filtering this image-like construction with a one-dimensional Gaussian kernel in each direction. The time complexity of this operation is $\mathcal{O}(nr)$, where r is the size of the kernel. Assuming that dictionary size m and kernel size r do not depend on the image size n, the running time for computing $\mathbf{W}\mathbf{x}$ is linear with respect to n.

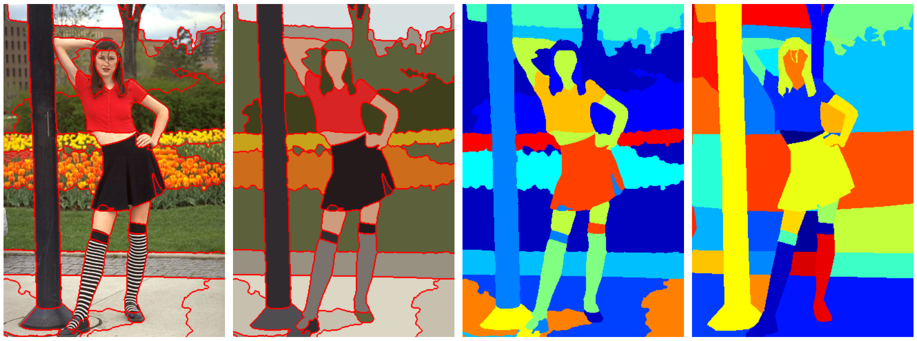

Fig. 2. Example demonstrating the global property of the spectral segmentation when employing our full-range affinity. All instances of grass, sky, skin and orange tulips belong to the same regions. Left: the original image with segments found using our method outlined in red. Middle left: average colors within segments. Middle right: segments in false colors. Right: manual segmentation from BSDB shown with false colors [1] lacks the global property.

The time complexity of the Lanczos method is $O(ln)+O(lf(n))$ where l is the maximal number of matrix-vector computations required and $f(n)$ is the cost of each matrix-vector computation [13]. Consequently, the total running time for the eigendecomposition of our full-range affinity matrix is $O(ln)$.

The maximal number of matrix-vector computations depends on the ability of the affinity matrix to propagate clustering cues. In our experience, a few hundred iterations is usually enough, regardless of the size of the image. We conclude that in practice our algorithm has linear time complexity.

3 Results

We have carried out a number of experiments in order to investigate the properties of our novel full range affinity matrix.

Since the self-similarity is most evident in composite images, we validate the quality of our method on segmenting Brodatz texture mosaics from [12]. To demonstrate the potential of our method in a more realistic setting, we include segmentations examples of natural images from the Berkeley Segmentation Dataset [11, 1]. In order to retain a focus on our contribution, we show those results without incorporating additional cues e.g. an edge term or a shape prior.

Global property. As a consequence of a full-range similarity, all image regions with similar texture and color belong to the same segment despite being disconnected. An example of this is shown in Fig. 2, where an image of the girl is segmented using 14 labels. The flowers, grass and bushes in the background, despite being separated, are labeled together due to the similar color and texture. The same is seen for the skin, sky, stockings and the paved curb.

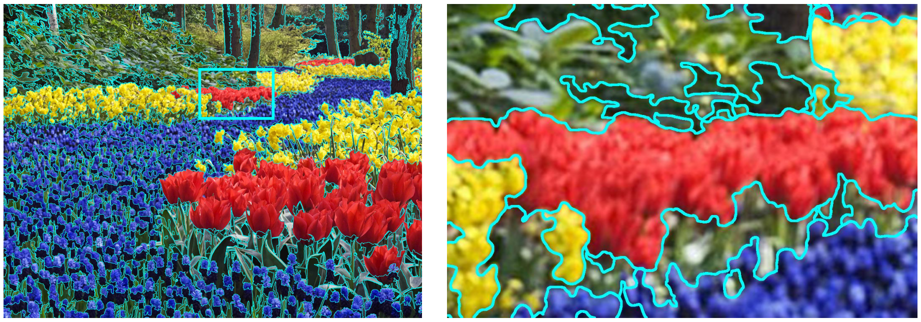

Fig. 3. Segmentation of a very large image. Left is a 4800 × 6400 image with segmentation boundaries in cyan. Right is a 750 × 1125 cutout indicated by a rectangle in the left image.

Efficiency. Our affinity matrix allows for an efficient decomposition, so we can process large images. In Fig. 3 we show an image containing more than 30 million pixels segmented in 5 segments. The dictionary used for this segmentation is based on 3 × 3 patches grouped in 200 clusters. Smoothness parameters are $\sigma = 1$ and $\alpha = 15$.

Textured images. We have made a quantitative evaluation of our method on a Randen texture dataset [12], with images composed of 5 - 16 different textures, allowing us to evaluate our method against ground truth. In this experiment, we base the dictionary on dense SIFT features grouped into 500 clusters. The smoothness parameters are $\sigma = 8$ and $\alpha = 4$. The segmentation on subset of images from the dataset is shown in the Fig. 4.

Quantitative results for the texture segmentation images are shown in Tab. 1, compared against other results reported for the dataset. It is important to note that the methods we compare against are all *trained* on examples of textured images, which gives them a great advantage. Our method takes a number of classes as input, but no other information about the textures. In other words, our method performs pixel clustering (unsupervised learning) while methods we compare to perform pixel classification (supervised learning). Still we obtain comparable results for most of the images. We compare to supervised methods because no unsupervised results have been reported for the Randen dataset.

Natural images. Segmentation of natural images from the Berkeley Segmentation Dataset [11, 1] in Fig. 5 is included to illustrate additional properties and the potentials of our method. These experiments have been carried out using 3 × 3 image patches clustered into dictionaries of 1000 elements. Segmenting into 2 to 6 regions was attempted, and we show the most meaningful result. Since global property is not part of the manually segmented reference data in the Berkeley Segmentation Dataset (see Fig. 2) we did not quantify our method against reference data.

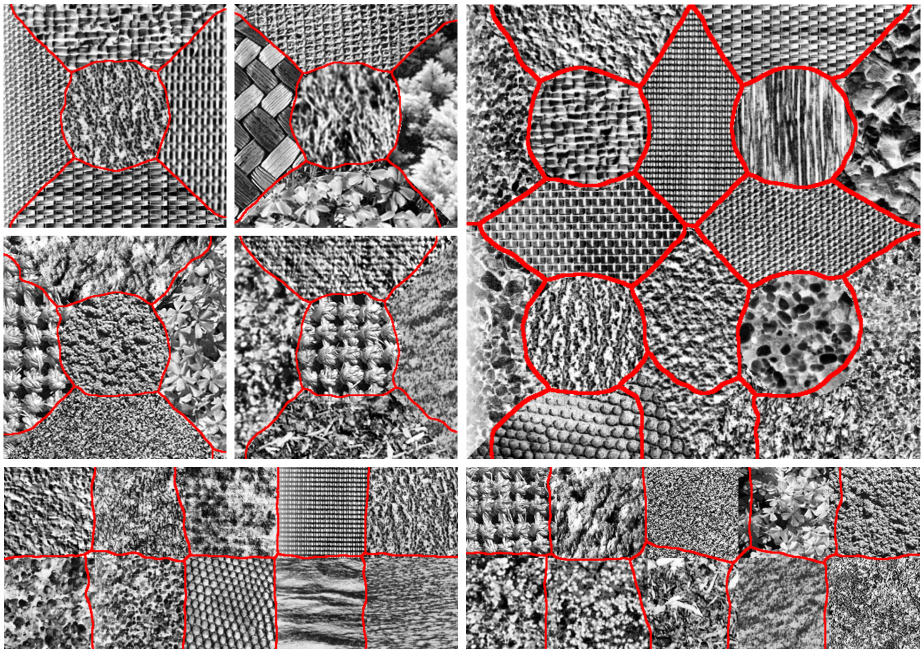

Fig. 4. Texture segmentation examples. Images 1 (top left), 2 (top middle), 3 (middle left), 4 (middle), 6 (top right), 8 (bottom left) and 9 (bottom right) from the Randen dataset [12].

	Trained methods				Our
Image	[12]	[8]	[9]	[14]	(without
no.	1999	2007	2008	2014	training)
1	7.20	3.37	1.61	2.00	1.68
2	18.90	16.05	16.42	3.24	4.43
3	20.60	13.03	4.15	4.01	6.74
4	16.80	6.62	3.67	2.55	6.58
5	17.20	8.15	4.58	1.26	9.29
6	34.70	18.66	9.04	6.72	10.06
7	41.70	21.67	8.80	4.14	31.57
8	32.30	21.96	2.24	4.80	2.55
9	27.80	9.61	2.04	3.90	7.99
10	0.70	0.36	0.17	0.42	0.24
11	0.20	1.33	0.60	0.61	0.26
12	2.50	1.14	0.78	0.70	0.75

Table 1. Comparison with Randen et al. [12], Lillo et al. [8], Mairal et al. [9], and Skretting and Engang [14] (best reported method). Reported number is the error rate. Unlike our method, the methods we compare to use training data.

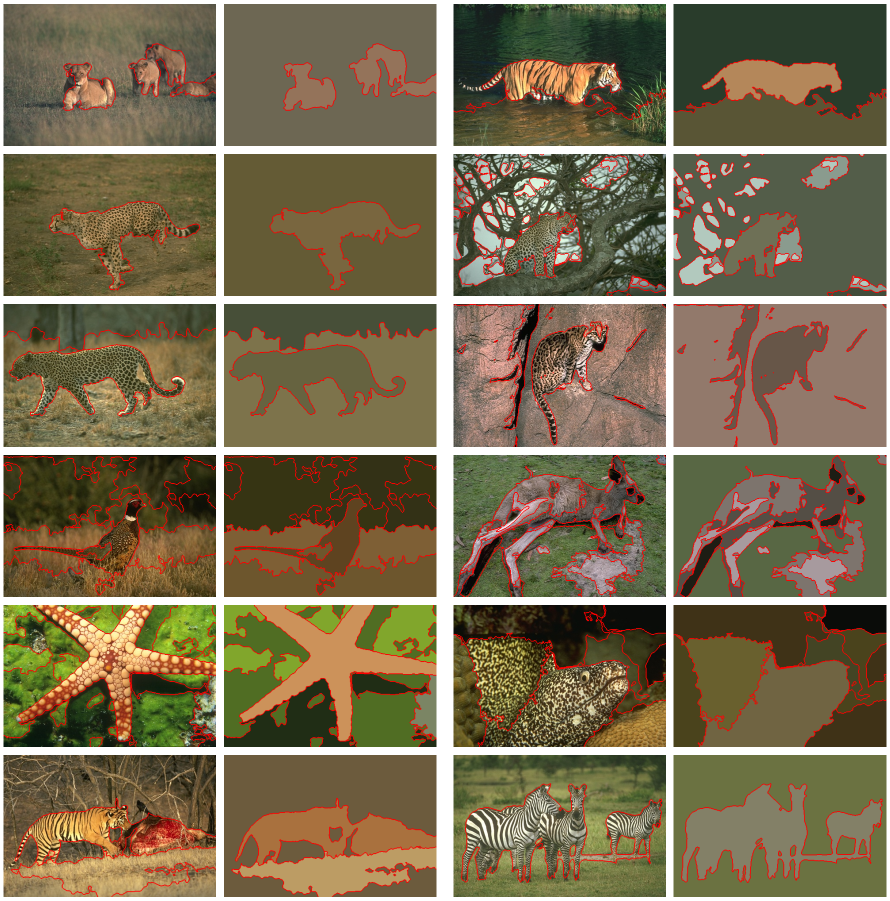

Fig. 5. Segmentation results for a selection of images from the Berkeley Segmentation Dataset.

Smoothness parameters. Influence of the model parameters on the segmentation results is shown in Fig. 6. We show the effect of changing the smoothing radius for the proximity matrix σ and the weight for proximity matrix α. For this experiment we use textured images and the dictionary based only on image patches without using SIFT features. This is because we also want to demonstrate the influence of the patch size M on the results.

4 Conclusion

The contribution of this paper is a general framework for the construction of the global affinity matrix using full-range similarity and additive smoothness. The full-range affinity matrix is constructed employing a dictionary and a bi-

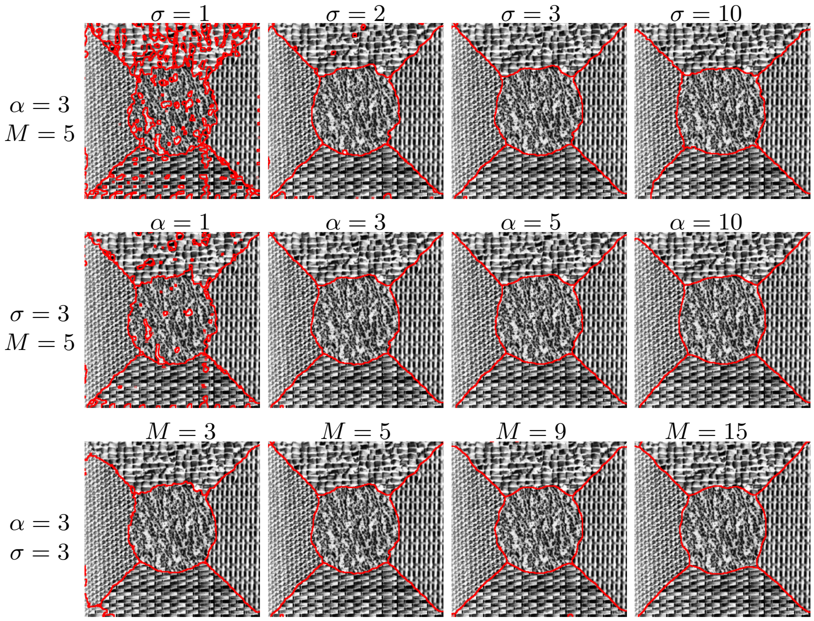

Fig. 6. The influence of the model parameters on the segmentation results. Parameters are: smoothing radius for proximity σ, proximity weight α and patch size M.

partite graph representation linking pixels in the image to elements/pixels in the dictionary.

Our approach is extremely versatile. Within the same general framework we apply our method for unsupervised spectral segmentation of the grayscale textured images and the rgb natural images. Our approach is also very efficient. Unlike other spectral methods, we obtain a linear time complexity for segmentation using our affinity matrix. This is because we do not represent the affinity matrix explicitly, and therefore it does not need to be sparse. We employ Lanczos algorithm that iteratively updates eigenvectors using an implicit matrix vector multiplication.

Our empirical investigation shows how the method partitions an image into regions with similar texture and color despite these being spatially separated. In addition we obtain impressive results on images composed of Brodatz textures, where no training data is used and only the number of segments is given to the algorithm.

In conclusion, we see our presented approach for spectral segmentation as an interesting alternative to conventional methods, appropriate for segmenting large images using a full-range affinity.

References

1. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**(5), 898–916 (2011)
2. Cour, T., Benezit, F., Shi, J.: Spectral segmentation with multiscale graph decomposition. In: Conference on Computer Vision and Pattern Recognition. vol. 2, pp. 1124–1131. IEEE (2005)
3. Dahl, A.B., Dahl, V.A.: Dictionary based image segmentation. In: Scandinavian Conference on Image Analysis, pp. 26–37. Springer (2015)
4. Dahl, V.A., Dahl, A.B.: A probabilistic framework for curve evolution. In: International Conference on Scale Space and Variational Methods in Computer Vision. pp. 421–432. Springer (2017)
5. Golub, G., Van Loan, C.F.: Matrix Computations, 642 pp. Johns Hopkins Univ. Press, Baltimore, Md (1996)
6. Kim, T.H., Lee, K.M., Lee, S.U.: Learning full pairwise affinities for spectral segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **35**(7), 1690–1703 (2013)
7. Li, Z., Wu, X.M., Chang, S.F.: Segmentation using superpixels: A bipartite graph partitioning approach. In: Conference on Computer Vision and Pattern Recognition. pp. 789–796. IEEE (2012)
8. Lillo, A.D., Motta, G., Storer, J., et al.: Texture classification based on discriminative features extracted in the frequency domain. In: International Conference on Image Processing. vol. 2, pp. II–53. IEEE (2007)
9. Mairal, J., Bach, F., Ponce, J., Sapiro, G., Zisserman, A.: Discriminative learned dictionaries for local image analysis. In: Conference on Computer Vision and Pattern Recognition. pp. 1–8. IEEE (2008)
10. Malik, J., Belongie, S., Leung, T., Shi, J.: Contour and texture analysis for image segmentation. International Journal of Computer Vision **43**(1), 7–27 (2001)
11. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: International Conference on Computer Vision. vol. 2, pp. 416–423. IEEE (2001)
12. Randen, T., Husoy, J.H.: Filtering for texture classification: A comparative study. IEEE Transactions on Pattern Analysis and Machine Intelligence **21**(4), 291–310 (1999)
13. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **22**(8), 888–905 (2000)
14. Skretting, K., Engan, K.: Energy minimization by α-erosion for supervised texture segmentation. In: Image Analysis and Recognition, pp. 207–214. Springer (2014)
15. Von Luxburg, U.: A tutorial on spectral clustering. Statistics and computing **17**(4), 395–416 (2007)
16. Wang, X., Li, H., Bichot, C.E., Masnou, S., Chen, L.: A graph-cut approach to image segmentation using an affinity graph based on ℓ_0-sparse representation of features. In: International Conference on Image Processing. pp. 4019–4023. IEEE (2013)
17. Wang, X., Tang, Y., Masnou, S., Chen, L.: A global/local affinity graph for image segmentation. IEEE Transactions on Image Processing **24**(4), 1399–1411 (2015)