Co-Sparse Textural Similarity for Interactive Segmentation[⋆]

Claudia Nieuwenhuis[1], Simon Hawe[2],
Martin Kleinsteuber[2], and Daniel Cremers[2]

[1] UC Berkeley, USA
[2] Technische Universität München, Germany

Abstract. We propose an algorithm for segmenting natural images based on texture and color information, which leverages the co-sparse analysis model for image segmentation. As a key ingredient of this method, we introduce a novel textural similarity measure, which builds upon the co-sparse representation of image patches. We propose a statistical MAP inference approach to merge textural similarity with information about color and location. Combined with recently developed convex multilabel optimization methods this leads to an efficient algorithm for interactive segmentation, which is easily parallelized on graphics hardware. The provided approach outperforms state-of-the-art interactive segmentation methods on the Graz Benchmark.

1 Introduction

The segmentation of natural images is a fundamental problem in computer vision. It forms the basis of many high-level algorithms such as object recognition, image annotation, semantic scene analysis, motion estimation, and 3D object reconstruction.

Despite its importance, the task of unsupervised segmentation is highly ill-posed and admittedly hard to evaluate. Therefore, we focus on *supervised segmentation* where ambiguities are solved by additional user input (scribbles or bounding boxes) and a clearly defined ground truth for performance evaluation is available. One can compute data likelihoods from a given set of scribbles using color texture or location. The simplest way is to compute the color distance of each pixel to the mean color value for each label [17]. More sophisticated approaches use density estimators, e.g. histograms [1,30], mixtures of Gaussians [25,28], or Parzen kernel density estimators [19]. Texture features were integrated in interactive segmentation by learning classifiers [27,26], filter banks [31] or SIFT features [29]. The integration of spatial information [2,19] also improved the performance. While all features carry relevant information, for natural images texture features are particularly relevant, but harder to capture due to

[⋆] This work was supported by the German Academic Exchange Service (DAAD) and the ERC Starting Grant 'Convex Vision'.

D. Fleet et al. (Eds.): ECCV 2014, Part VI, LNCS 8694, pp. 285–301, 2014.

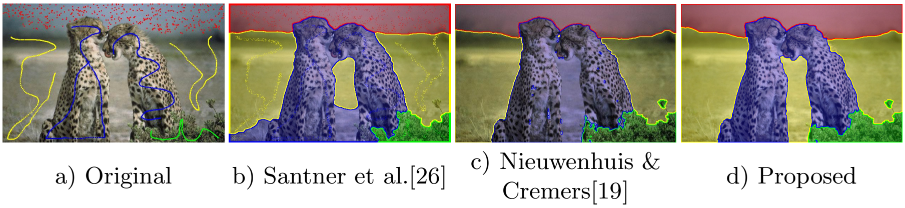

Fig. 1. Leveraging the co-sparse analysis operator for image segmentation yields a simple texture descriptor that ultimately leads to state-of-the-art results on the Graz interactive segmentation database. We compare against b) the texture-based approach by Santner et al. [26], who train a random forest classifier on texture features and c) spatially adaptive color models by Nieuwenhuis and Cremers [19], which locally approximate texture. d) The proposed method based on co-sparsity.

their diversity and spatial extent. To extract textural information from images, methods based on sparse representations are quite successful [13].

Commonly, sparsity is exploited via the synthesis model, aka sparse coding. It assumes that every image patch can be approximated as a linear combination of a few predefined atoms, which form the columns of a dictionary. With this, the textural information is encoded in the set of active dictionary atoms, i.e. the support of the sparse code. Finding this set for a given dictionary, however, requires to solve a costly optimization problem.

In this paper, we propose a more efficient way to obtain textural information by employing the *co-sparse analysis model* [7,18]. In this model, the sparse image representation is determined efficiently by a simple matrix vector product. We derive a novel textural similarity measure for image patches and demonstrate that it can be successfully introduced into image segmentation approaches. To the best of our knowledge, there has not yet been an attempt that employs the co-sparse analysis model for extracting textural information. So far, the model has only been successfully applied to regularize inverse problems such as super-resolution, denoising or depth estimation [5,11]. We refer to [10,24,32] for learning a co-sparse analysis model for natural images. The model has potential impact also for segmentation tasks in other imaging methods, where structure plays a prominent role, e.g. in medical imaging. Figure 1 shows that the proposed measure combined with an efficient convex multilabel approach generates convincing results for supervised segmentation problems, which outperform previous interactive state-of-the-art approaches [26,19].

Contributions

In this paper we present a novel approach for the task of supervised segmentation of natural images, which yields state-of-the-art results on the Graz benchmark for interactive segmentation. In particular, we make the following contributions.

- The co-sparse analysis model is leveraged for image segmentation through a novel texture similarity measure. Until today, this model has only been employed for regularizing inverse problems, such as inpainting or denoising. Showing that it is also useful for analyzing structural similarity (via the proposed novel distance measure) is the main contribution of this paper.

- The proposed algorithm combines the co-sparse analysis model and recent convex relaxation techniques within a single convex optimization problem.
- The method explicitly models the dependence between texture, color and location leading to a space-dependent color and texture model. This accounts for non-iid samples in scribble based probability density estimation.
- We merely require the four images in Figure 2 (which are not part of the benchmark) to train the co-sparse analysis operator for texture recognition and thus avoid over-fitting to specific benchmarks.
- The approach can be efficiently parallelized on graphics hardware with average runtimes of two seconds per image.

The paper is organized as follows. In Section 2, we derive a texture similarity measure from co-sparse analysis. In Section 3, we integrate this likelihood into a variational segmentation scheme, for which we give a convex relaxation and minimization method in Section 4. In Section 5, we present experimental results.

2 Co-Sparse Textural Similarity

The co-sparse analysis model [7,18] is based on the assumption that if $\mathbf{s} \in \mathbb{R}^N$ denotes a vectorized image patch, there exists an analysis operator $\mathbf{O} \in \mathbb{R}^{k \times N}$ with $k > N$ such that $\mathbf{a} := \mathbf{O}\mathbf{s}$ is sparse. We refer to $\mathbf{a} \in \mathbb{R}^k$ as the *analyzed version of* $\mathbf{s}$. Notice that the rows of $\mathbf{O}$ can be interpreted as filters and the analyzed version of $\mathbf{s}$ as the corresponding filter responses. The two major differences to the more commonly known synthesis model are: (i) the sparse code is found via a simple matrix vector multiplication and (ii) the *zero* entries of $\mathbf{a}$ are the informative coefficients describing the underlying signal. Concretely, the textural structure of $\mathbf{s}$ is encoded in its co-support

$$\mathrm{Co}(\mathbf{a}) := \{j \mid a_j = 0\}, \tag{1}$$

where a_j denotes the j-th entry of $\mathbf{a}$. Geometrically, $\mathbf{s}$ is orthogonal to all rows that determine the co-support and thus lies in the intersection of the respective hyperplanes. Thus the co-sparsity of a vector $\mathbf{s}$ increases with the cardinality of

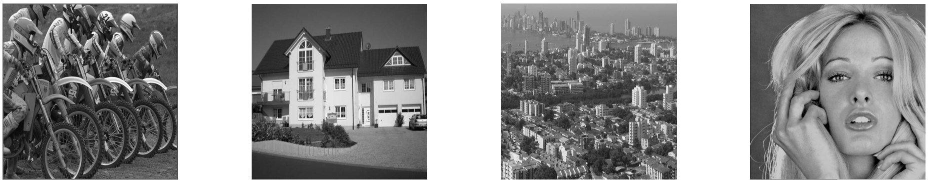

Fig. 2. The four training images used for learning the analysis operator

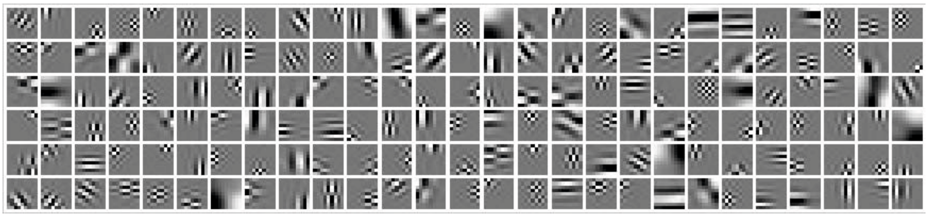

Fig. 3. Sample filters from the co-sparse analysis operator O, which was learned from natural images for 9x9 patches. The samples show that the operator includes low, intermediate and high frequency signals as well as spatially global and local signals.

its co-support $|\operatorname{Co}(\mathbf{a})|$, i.e. with the sparsity of its analyzed version $\mathbf{a}$. A subset of the signals learned by our operator is shown in Figure 3.

A prominent example for an analysis operator is the finite difference operator in image processing. However, the advantage of the low computational complexity of such an analytically given transformation comes at the cost of a poor adaptation to specific signal classes of interest. It is now well-known that for a particular class, sparser signal representations and thus better reconstruction accuracies can be achieved if the analysis operator $\mathbf{O}$ is learned from a representative training set. Here, we employ an analysis operator learned according to the geometric optimization procedure proposed in [10] from patches extracted from natural images. As we only want to gather textural information independent of varying illumination conditions, we follow the simple bias and gain model and use patches $\mathbf{s}$ from a training set $\mathcal{S}$ that have been normalized to zero-mean and unit-norm, i.e. $\sum_i s_i = 0$ and $\|\mathbf{s}\|_2 = 1$. Given the smooth sparsity measure

$$g(\mathbf{a}) := \sum_j \log(1 + \nu a_j^2), \tag{2}$$

where $\nu > 0$ is some constant, the optimal analysis operator aims at minimizing the expected *squared* sparsity

$$\mathbf{O} \in \arg\min_{\widehat{\mathbf{O}}} \tfrac{1}{|\mathcal{S}|} \sum_{\mathbf{s}\in\mathcal{S}} g(\widehat{\mathbf{O}}\mathbf{s})^2. \tag{3}$$

This can be interpreted as a balanced minimization of expectation and variance of the samples' co-sparsity. For regularizing the set of feasible solutions, the Euclidean norm of the rows of $\mathbf{O}$ is restricted to one, and the so-called coherence property and the rank are controlled via two penalty functions. The optimization problem is then tackled using a conjugate gradient method on an appropriate manifold, cf. [10]. We initialize randomly, which - despite the non-convex nature of the optimization problem - in practice leads to an optimal solution [10].

Since our ultimate goal is to discriminate between distinctive textures in natural images,a measure of textural similarity should better distinguish between representative patches, i.e. patches that fit the co-sparse analysis model of natural image patches, while discriminating moderately for "outlier"-patches, i.e.

patches that seldom occur in natural images. This motivates us to measure the textural similarity between two patches via

$$TSM_{\mathbf{O}}(\mathbf{s}_1, \mathbf{s}_2) := \sum_{j=1}^{k} |\mathbb{1}_{\mathrm{Co}(\mathbf{O}\mathbf{s}_1)}(j) - \mathbb{1}_{\mathrm{Co}(\mathbf{O}\mathbf{s}_2)}(j)|, \tag{4}$$

where $\mathbb{1}_A$ is the indicator function of a set, i.e. $\mathbb{1}_A(j) = 1$ if $j \in A$ and zero otherwise. This measure has two desired properties: 1) it distinguishes sensibly between patches that fit the model well, i.e. patches with a large co-support, 2) it does not heavily discriminate between patches that fit the model less.

To identify an "average" textural structure from a set of m patches $\mathcal{S} = \{\mathbf{s}_1, \dots, \mathbf{s}_m\}$ that serves as their textural representative, we provide the following definition. A patch $\mathbf{r} \in \mathbb{R}^N$ is called a *textural representative of* $\mathcal{S}$ if

$$\mathbf{r} \in \arg\min_{\mathbf{z}} \sum_{i=1}^{m} TSM_{\mathbf{O}}(\mathbf{s}_i, \mathbf{z}). \tag{5}$$

So far, we considered truly co-sparse image patches, i.e. patches whose analyzed versions contain many coefficients that are exactly zero. However, this is an idealized assumption and in practice those patches are not truly co-sparse but rather contain many coefficients that are close to zero. To account for this, we introduce the mapping $\iota_\sigma \colon \mathbb{R}^k \to \mathbb{R}^k$ as a smooth approximation of the indicator function of the co-support, which is defined component-wise with a free parameter $\sigma > 0$ as

$$(\iota_\sigma(\mathbf{a}))_j = \exp(-a_j^2/\sigma). \tag{6}$$

In fact, it is easily seen that $\mathbb{1}_{\mathrm{Co}(\mathbf{a})}(j) = \lim_{\sigma \to 0}(\iota_\sigma(\mathbf{a}))_j$ and $\lim_{a_j \to 0}(\iota_\sigma(\mathbf{a}))_j = 1$.

With this approximation of the co-support, the textural similarity measure in (4) of two patches $\mathbf{s}_1$ and $\mathbf{s}_2$ associated with the analysis operator $\mathbf{O}$ and σ is approximated by

$$TSM_{\mathbf{O},\sigma}(\mathbf{s}_1, \mathbf{s}_2) = \|\iota_\sigma(\mathbf{O}\mathbf{s}_1) - \iota_\sigma(\mathbf{O}\mathbf{s}_2)\|_1, \tag{7}$$

with $\|\cdot\|_1$ denoting the ℓ_1-norm. According to Eq. (5), a structural representative $\mathbf{r} \in \mathbb{R}^N$ of a set $\mathcal{S} = \{\mathbf{s}_1, \dots, \mathbf{s}_m\}$ with respect to $TSM_{\mathbf{O},\sigma}$ is

$$\mathbf{r} \in \arg\min_{\mathbf{z}} \sum_{i=1}^{m} TSM_{\mathbf{O},\sigma}(\mathbf{s}_i, \mathbf{z}). \tag{8}$$

Using the well-known fact that the centroid of a cluster with respect to the ℓ_1-distance is the median of all corresponding cluster points, the approximated co-support of the analyzed version of a structural representative fulfills

$$\iota_\sigma(\mathbf{O}\mathbf{r}) = \mathrm{median}(\{\iota_\sigma(\mathbf{O}\mathbf{s}_j)\}_{j=1}^{m}). \tag{9}$$

3 Variational Co-sparse Image Segmentation

In this section, we derive a statistical MAP inference formulation for supervised image segmentation based on the novel proposed textural similarity measure. We explicitly model the dependence of texture and color on the scribble location in the image to account for texture variations within regions, e.g. a sky which is partially covered by clouds. At the same time this model alleviates the issue of spatially non-iid distributed scribble samples for density estimation.

3.1 A Space Variant Texture and Color Distribution

For an image domain $\Omega \subset \mathbb{R}^2$, let $I : \Omega \to \mathbb{R}^d$ denote the input color (or gray scale) image.

The segmentation problem can be solved by computing a labeling $l : \Omega \to \{1, .., n\}$ that indicates, which of the n regions each pixel belongs to, i.e. $\Omega_i := \{x \,|\, l(x) = i\}$. In a statistical MAP framework the labeling l can be computed by maximizing the conditional probability

$$\arg\max_l \mathcal{P}(l \,|\, I) = \arg\max_l \mathcal{P}(I \,|\, l)\ \mathcal{P}(l). \tag{10}$$

In the following, we will model the dependence of color and texture on the image location. We use the image for two sources of information, color and structure. Structure is obtained by computing the gray value image by eliminating the hue and saturation and only keeping the luminance channel. Let $\mathbf{s}_x$ denote a small gray value texture patch centered at pixel x. With the assumption that a pixel color jointly depends on the local structure $\mathbf{s}_x$ given a location x and a label $l(x)$, but is independent of the label of other pixels we obtain

$$\mathcal{P}(I \,|\, l) = \prod_{i=1}^{n} \prod_{x \in \Omega} \mathcal{P}(I(x), \mathbf{s}_x \,|\, l(x) = i, x). \tag{11}$$

In the following, we derive the probability $\mathcal{P}(I(x), \mathbf{s}_x \,|\, l(x) = i, x)$ that a pixel at location x belonging to segment i has color $I(x)$ and texture patch $\mathbf{s}_x$. Assuming independence, we can compute the likelihood of a pixel for belonging to region i as

$$\mathcal{P}(I(x), \mathbf{s}_x, \,|l(x){=}i, x) = \mathcal{P}(I(x)\,|l(x){=}i, x)\mathcal{P}(\mathbf{s}_x\,|l(x){=}i, x). \tag{12}$$

Given the set of scribble samples consisting of location, color, and texture patches for each segment i, i.e.

$$S_i := \big\{(x_{ij}, I_{ij}, \mathbf{s}_{x_{ij}}),\ j = 1, .., m_i\big\} \tag{13}$$

we can estimate the joint distribution from sample data. We use Parzen density estimators [21], since they come with the advantage that they can represent arbitrary kinds of probability densities and provably converge to the true density for infinitely many samples. However, they require independent and identically distributed samples. This assumption may be acceptable for color, but for texture

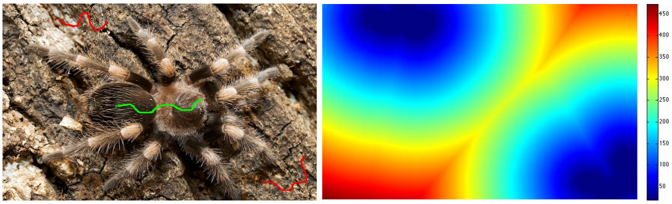

Fig. 4. Estimation of the variance $\rho_{bg}(x)$ of the spatial kernel in (14) for the background region from the red scribbles. The spatial variance is proportional to the distance of each pixel to the closest background scribble point. The larger the minimum distance to the scribbles the larger the uncertainty in the density estimation and the more samples will be taken into consideration.

and location it is clearly violated since patches and scribbles are by no means spatially independent and identically distributed.

As a remedy, in [19] we proposed a spatially varying color distribution using the following Parzen density

$$\mathcal{P}(I(x)\,|\,l(x)=i,x) = \tfrac{1}{m_i}\sum_{j=1}^{m_i} k_{\rho_i(x)}(x-x_{ij})k_\mu(I-I_{ij}). \tag{14}$$

Here k denotes a kernel function with variance indicated as subscript. The idea behind the spatial dependence of $\rho_i(x)$ on x is that each color kernel is weighted by a spatial kernel with location dependent variance in order to account for non-iid samples. An intuitive explanation is that for pixels close to a scribble we only want to use few samples in the direct vicinity of the pixel (and thus a small spatial kernel variance) to estimate the color distribution since we are quite certain what the color should be at that pixel. In contrast, if we are far from all scribbles we use a large number of scribble points (and thus a larger kernel variance) since we are uncertain about the color at the current pixel.

The variance of the spatial kernel $\rho_i(x)$ is therefore adapted to the distance of the current pixel x from the nearest user scribble of this label:

$$\rho_i(x) = \alpha|x - x_{v_i}|_2 \tag{15}$$

where x_{v_i} is the closest scribble location of all pixels in segment i and α a scaling factor, which we set to 1.3. Figure 4 shows the function $\rho_i(x)$ for the spider image and the background region. Thus, the spatial dependence of $\rho_i(x)$ accounts for spatially non-iid samples and at the same time for the level of uncertainty in the estimator.

After the spatially varying color distribution we will now formulate the spatially varying texture distribution $\mathcal{P}(\mathbf{s}_x|l(x)=i,x)$ - see (12). Using a Parzen density estimator in a similar way as in (14) to obtain a texture distribution is only possible for very small patches due to the high dimensionality of the distribution, which would require a prohibitively large amount of samples not

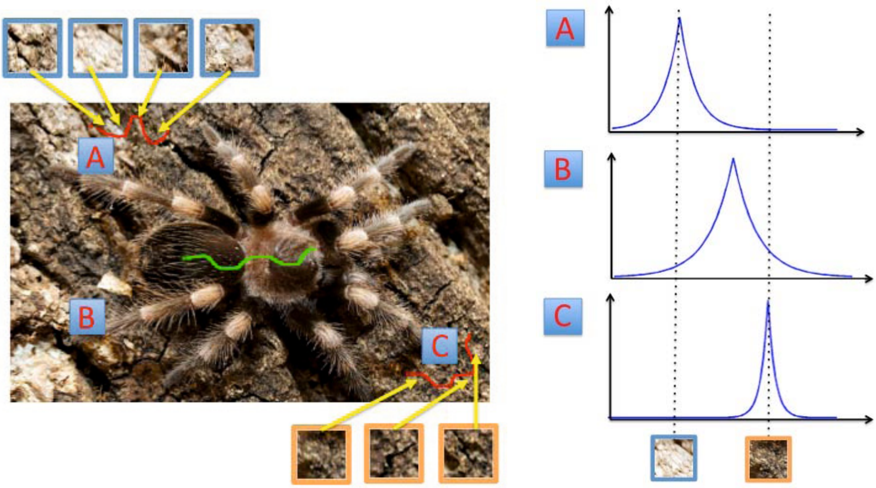

Fig. 5. We exemplarily estimate the spatially varying texture distribution for the background region (red scribbles) at three different locations in the image (A, B and C). The results are shown on the right. The horizontal axis represents the (high-dimensional) texture space with two representative patches below, the vertical axis the corresponding estimated probability. The three distributions are different since we only use sample patches from scribbles, which are close to the current location. If we are close to a scribble (A and C) we only use neighboring background scribble points, but if the closest scribble is far away (B) we use all background scribble samples to estimate the distribution. This results in three different estimated medians in (18) and thus three different peaks in the distribution. This procedure accounts for the spatially non-iid distributed scribble samples.

provided by the user scribbles. For this reason we will formulate a spatially varying texture distribution based on the co-support in equation (1).

As our goal is to extract local textural information in the vicinity of a pixel x, we multiply each patch element-wise with a Gaussian mask to assign more weight to the central pixels prior to normalization to zero-mean and unit-norm according to Section 2. From these patches, we compute the approximated co-support of a textural representative of each set of scribble points according to equation (9), i.e.

$$\mathbf{c}_i = \text{median}(\{\iota_\sigma(\mathbf{O}\mathbf{s}_{x_{ij}})\}_{j=1}^{m_i}). \tag{16}$$

Based on this we assign to each pixel x the a posteriori probability of belonging to class i depending on the corresponding patch as

$$\mathcal{P}(\mathbf{s}_x|l(x)=i, x) = \frac{\exp(-\frac{1}{\beta}\|\mathbf{c}_i - \iota_\sigma(\mathbf{O}\mathbf{s}_x)\|_1)}{\sum_{j=1}^{n} \exp(-\frac{1}{\beta}\|\mathbf{c}_j - \iota_\sigma(\mathbf{O}\mathbf{s}_x)\|_1)}. \tag{17}$$

The parameter $\beta > 0$ controls the variance of the labeling l. It can be interpreted as a measure of how well we trust the similarity measure for deciding to which class x belongs. Large values of β assign a pixel to each of the classes with approximately equal probability, whereas small values of β assign x to the most similar class with very high probability.

We now introduce the spatial variation (i.e. the dependence on scribble location) into the distribution in (17) in order to obtain a spatially varying texture distribution. Avoiding the Parzen density due to prohibitive dimensionality we compute a spatially varying median based on the spatial kernel variance $\rho_i(x)$ in (15). The idea is that we only use the texture samples which are close to the current pixel x with respect to $\rho_i(x)$ to estimate the median:

$$\mathbf{c}_i(x) = \underset{|x-x_{ij}|_2 \leq \rho_i(x)}{\text{median}} (\{\iota_\sigma(\mathbf{O}\mathbf{s}_{x_{ij}})\}_{j=1}^{m_i}). \tag{18}$$

This yields a spatially varying median of co-sparse analyzed texture patches, which we can now introduce into the posterior probability distribution in (17). Figure 5 shows how the spatially varying texture distribution locally adapts to the closer scribble points.

Based on (12) in combination with (14) and (17) we can now compute the joined spatially varying distribution over color and texture, which alleviates the problem of non-iid samples and accounts for variable estimator certainty with respect to the scribble distance.

3.2 Variational Formulation

Based on the segment probabilities $\mathcal{P}\big(I(x), \mathbf{s}_x \big| \, l(x)=i, x\big)$ given in (12), (14) and (17) we now define an energy optimization problem for the task of segmentation. We specify the prior $\mathcal{P}(l)$ in (10) to favor regions of shorter boundary

$$\mathcal{P}(l) \propto \exp\big(-\tfrac{1}{2}\sum_{i=1}^{n} \text{Per}_g(\Omega_i)\big), \tag{19}$$

where $\text{Per}_g(\Omega_i)$ denotes the perimeter of each region Ω_i, i.e. the boundary length, measured in the metric $g : \Omega \to \mathbb{R}^+$ (see (24)).

Instead of maximizing the a posteriori distribution (11), we minimize its negative logarithm, i.e. the energy

$$\mathcal{E} = \sum_{i=1}^{n} \tfrac{\lambda}{2}\text{Per}_g(\Omega_i) - \int_{\Omega_i} \log\big(\mathcal{P}\big(I(x), \mathbf{s}_x \big| \, l(x)=i, x\big)\big) \; dx. \tag{20}$$

The weighting parameter $\lambda \in [0, \infty]$ balances the impact of the data term and the boundary length.

4 Minimization via Convex Relaxation

Problem (20) is the continuous equivalent to the Potts model, whose solution is known to be NP-hard. However, a computationally tractable convex relaxation of this functional has been proposed in [3,4,12,22,34]. For more information and implementation details see [20]. Due to the convexity of the problem the resulting solutions have the following properties: Firstly, the segmentation is independent of the initialization. Secondly, we obtain globally optimal segmentations for the case of two regions and near-optimal – in practice often globally optimal – solutions for the multi-region case. In addition, the algorithm can be parallelized and run on GPUs.

4.1 Conversion to a Convex Differentiable Problem

To apply convex relaxation techniques, we first represent the n regions Ω_i by the indicator function $u \in \mathrm{BV}(\Omega, \{0,1\})^n$, where

$$u_i(x) = \begin{cases} 1, & \text{if } x \in \Omega_i \\ 0, & \text{otherwise} \end{cases} \qquad i \in \{1,..,n\}. \tag{21}$$

Here BV denotes the functions of bounded variation, i.e. functions with a finite total variation. For a valid segmentation we require that the sum of all indicator functions at each location $x \in \Omega$ amounts to one, so each pixel is assigned to exactly one label. Hence,

$$\mathcal{B} = \Big\{ u \in \mathrm{BV}(\Omega, \{0,1\})^n \;\Big|\; \sum_{i=1}^{n} u_i(x) = 1 \; \forall x \in \Omega \Big\}. \tag{22}$$

denotes the set of valid segmentations. To rewrite energy (20) in terms of the indicator functions u_i, we have to rewrite the boundary length prior in (19). The boundary of the set indicated by u_i can be written by means of the total variation. Let $\xi_i \in C_c^1(\Omega, \mathbb{R}^2)$ denote the dual variables and C_c^1 the space of smooth functions with compact support.

Then, following the coarea formula [8] the weighted perimeter of Ω_i is equivalent to the weighted total variation

$$\frac{\lambda}{2}\mathrm{Per}_g(\Omega_i) = \frac{\lambda}{2}\int_\Omega g(x)\,|D\,u_i\,| = \sup_{\xi_i \in \mathcal{K}_g} \int_\Omega \xi_i\, D\,u_i \; = \sup_{\xi_i \in \mathcal{K}_g} -\int_\Omega u_i \,\mathrm{div}\, \xi_i \, dx \tag{23}$$

with $\mathcal{K}_g = \Big\{ \xi \in C_c^1(\Omega, \mathbb{R}^2) \Big| \, |\xi(x)| \leq \frac{\lambda g(x)}{2} \; \forall x \in \Omega \Big\}$, see [34,20]. $D\,u_i$ denotes the distributional derivative of u_i (which is $D\,u_i = \nabla u_i \, dx$ for differentiable u_i). The final transformation in (23) follows from integration by parts and the compact support of the dual variables ξ_i. A commonly used choice for the metric g

$$g(x) = \tfrac{1}{2\gamma} \exp\Big(-\tfrac{|\nabla I(x)|}{\gamma}\Big), \quad \gamma = \tfrac{1}{|\Omega|} \int_\Omega |\nabla I(x)| \, dx, \tag{24}$$

favors boundaries coinciding with strong intensity gradients $|\nabla I(x)|$ and, thus, prevents oversmoothed boundaries. Relaxing the set $\mathcal{B}$ to the convex set $\tilde{\mathcal{B}} = \big\{ u \in \mathrm{BV}(\Omega, [0,1])^n \;\big|\; \sum_{i=1}^n u_i(x) = 1 \; \forall x \in \Omega \big\}$ we finally obtain the convex problem

$$\min_{u \in \tilde{\mathcal{B}}} \sup_{\xi \in \mathcal{K}_g^n} \sum_{i=1}^{n} \int_\Omega -\log\big(\mathcal{P}\big(I(x), \mathrm{s}_x \,\big|\, l(x){=}i, x\big)\big)\, u_i \, dx - \int_\Omega u_i \,\mathrm{div}\, \xi_i \, dx. \tag{25}$$

4.2 Implementation

To solve the relaxed convex optimization problem, we employ a primal- dual algorithm proposed in [22]. Essentially, it consists of alternating a projected gradient descent in the primal variables u_i with projected gradient ascent in the dual variables ξ_i. An over-relaxation step in the primal variables gives rise to auxiliary variables $\bar{u}_i$:

$$\begin{aligned} \xi_i^{t+1} &= \Pi_{\mathcal{K}_g}\Big(\xi_i^t + \tau_\xi \nabla \bar{u}_i^t\Big) \\ u_i^{t+1} &= \Pi_{\tilde{\mathcal{B}}}\Big(u_i^t - \tau_u(-\operatorname{div} \xi_i^{t+1} + f_i)\Big) \\ \bar{u}_i^{t+1} &= u_i^{t+1} + (u_i^{t+1} - u_i^t) = 2u_i^{t+1} - u_i^t \end{aligned} \tag{26}$$

where $f_i(x) := -\log\big(\mathcal{P}\big(I(x), \mathrm{s}_x \,\big|\, l(x)=i, x\big)\big)$, Π denotes the projections onto the respective convex sets and the different τ denote step sizes for primal and dual variables. These are optimized based on [23]. The projections onto $\mathcal{K}_g$ are straightforward, the projection onto the simplex $\tilde{\mathcal{B}}$ is given in [14]. As shown in [22], the update scheme in (26) provably converges to a minimizer of the relaxed problem.

Due to the relaxation we may end up with non-binary solutions $u_i \in \tilde{\mathcal{B}}$. To obtain binary solutions in the set $\mathcal{B}$, we assign each pixel to the label with maximum value u_i, i.e. $l(x) = \arg\max_i u_i(x)$. This operation is known to preserve optimality in case of two regions [4]. In the multi-region case optimality bounds can be computed from the energy difference between the minimizer of the relaxed problem and its reprojected version. Typically the projected solution deviates less than 1% from the optimal energy, i.e. the results are very close to global optimality [20].

5 Experiments and Results

To evaluate the proposed algorithm we apply it to the interactive Graz benchmark [26] for supervised segmentation and compare against state-of-the-art segmentation algorithms. For all experiments we use a patch size of 9×9, and a two times over complete analysis operator, i.e. $k = 2 * 81$, which we have learned from 50 000 randomly extracted patches from the images shown in Figure 2.

Note, that we do not require any training of the operator on the Graz benchmark set but use it as is, avoiding overfitting to specific benchmarks. The parameter σ in (6) required to measure the textural similarity was set to $\sigma = 0.01$.

5.1 Results on the Graz Benchmark

The Graz benchmark consists of 262 scribble-ground truth pairs from 158 natural images containing between 2 and 13 user labeled segments. We used a brush size of 13 pixels in diameter for scribbling as done by Santner et al. [26] and Nieuwenhuis and Cremers [19], set $\lambda = 2000$ and the color kernel variance in

(14) to $\mu = 1.3$ for all experiments. To rank our method, we compare our results with state-of-the-art interactive segmentation algorithms. The Random Walker algorithm by Grady [9] for each pixel computes the probability that a random walker starting from any scribble seed reaches it first based on color and texture edges. In [27,26] Santner et al. train a random forest classifier based on CIELab color as well as Haralick and Local Binary Pattern (LPB) texture features.

Finally, the approach by Nieuwenhuis and Cremers [19] uses spatially varying color distributions which locally represent the texture in the image. Table 1 shows the average Dice-score [6] for all methods. This score compares the overlap of each region Ω_i with its ground truth $\bar{\Omega}_i$

$$dice(\Omega_1, ..\Omega_n) = \frac{1}{n} \sum_{i=1}^{n} \frac{2|\bar{\Omega}_i \cap \Omega_i|}{|\bar{\Omega}_i| + |\Omega_i|}. \tag{27}$$

The results show that our proposed approach outperforms all of the previous approaches. Especially for images, where texture is important to obtain the correct segmentation due to strongly overlapping color distributions in foreground and background, the proposed method shows significant improvements. We show several of these images in Figure 6. For example for the cats, the scorpion, the leopard and the bears image the texture of the animals is the main distinction criterion with respect to the background. The airplane image contains many different textures with similar colors, which are hard to distinguish, and the sign on the wall can only be distinguished from the background by its texture. The ground beneath the walking men changes color due to lighting and can only be recognized by texture as well. For images, where color is sufficient to distinguish between the objects the improvements were minor, which explains the moderate increase of the overall average benchmark score despite substantial improvements for texture based images.

Table 1. Comparison of the average Dice-score (27) to state-of-the-art supervised segmentation approaches by Grady [9], Santner et al. [26] and Nieuwenhuis and Cremers [19] on the Graz benchmark.

Method	Score
Santner et al. [26], Grayscale images, no texture	0.728
Grady [9], Random Walker	0.855
Santner et al. [26], RGB, no texture	0.877
Nieuwenhuis & Cremers [19], space-constant, no texture	0.889
Santner [26], CIELab plus texture	0.927
Nieuwenhuis & Cremers [19], space-varying color (texture approximation)	0.931
Proposed, space-varying color and co-sparse texture	**0.937**

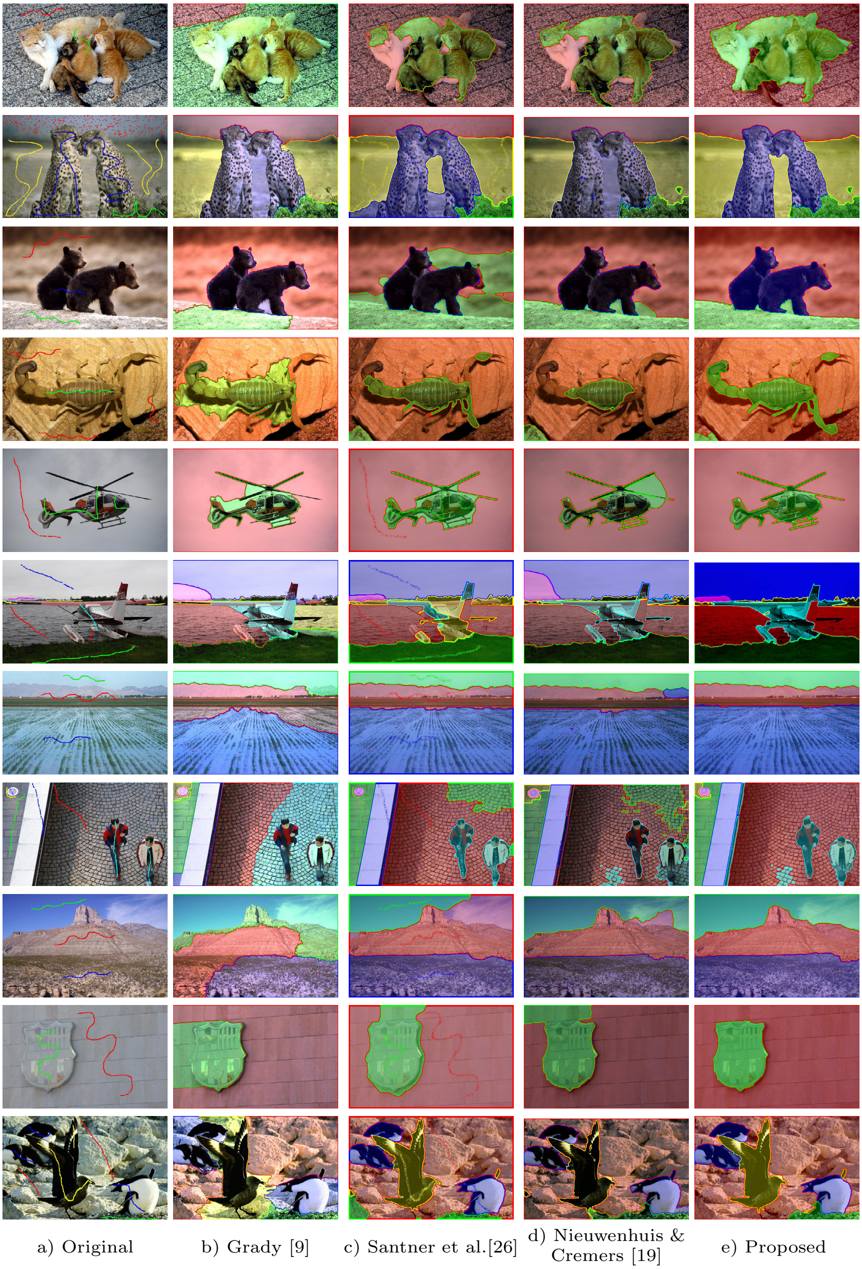

Fig. 6. Comparison of supervised segmentation results based on the proposed co-sparse analysis model to the approaches by Grady [9], Santner et al. [26] and Nieuwenhuis and Cremers [19] on the Graz interactive segmentation benchmark. Note that our model obtains strong improvements especially for those images, where color is insufficient and texture is required to distinguish between objects.

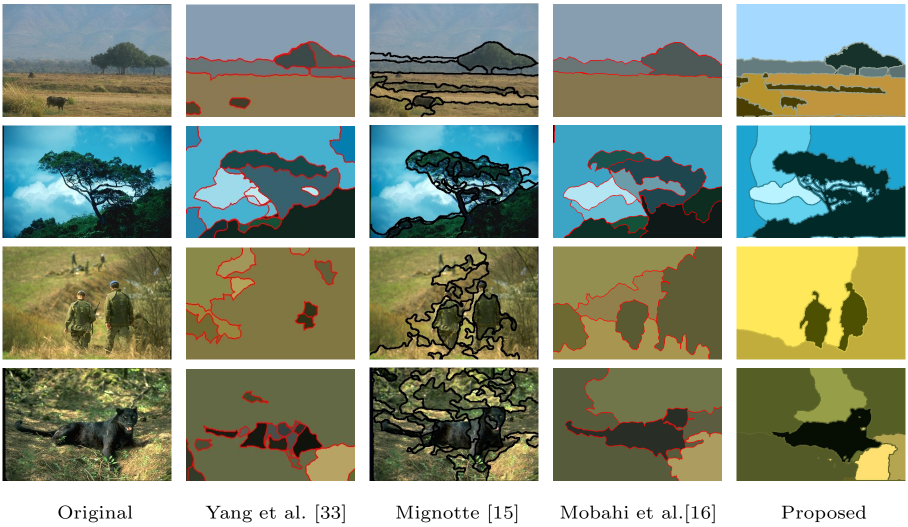

Fig. 7. Application of our method to a few texture based images from the Berkeley segmentation database. To obtain color and texture samples we use simple k-means clustering with a hand-selected number of labels. We compare against texture segmentation methods by Yang et al. [33], Mignotte [15] and Mobahi et al. [16]

5.2 Results on the Berkeley Segmentation Database

In order to compare against other texture segmentation approaches we finally apply our method to a set of images from the Berkeley segmentation database. Since this database does not provide user scribbles we use simple k-means clustering with a hand-selected number of segments to obtain a set of representative samples for each class in color and texture space. Even though this clustering method yields highly suboptimal scribble information we still obtain good results on several images that require texture for correct segmentation, see Figure 7. We compare against the texture based segmentation methods by Mobahi et al. [16], Yang et al. [33] and Mignotte [15].

5.3 Runtimes

The textural similarity analysis is based only on highly parallelizable filter operations. Due to the additional inherently parallel structure of the optimization problem in (26), the algorithm can be easily and efficiently implemented on graphics hardware. The experiments were carried out on an Intel Core i7-3770 3.4 GHz CPU with an NVIDIA Geforce GTX 580 GPU. The average computation time on the Graz Benchmark is 2 seconds, which is along the lines of Santner et al. [26] with 2 seconds and Nieuwenhuis and Cremers [19] with 1.5 seconds.

6 Conclusion

In this paper we introduced co-sparse operator learning for texture recognition into interactive image segmentation. The rows of the learned operator can be interpreted as filters that are trained to deliver sparse filter responses for natural image patches. In contrast to segmentation approaches that use filter banks, we thus do not rely on the typically employed locally windowed filter histograms, but can use an easy-to-implement measure to determine local structural similarity. From this measure, a data likelihood is derived and integrated in a statistical maximum a posteriori estimation scheme in order to combine color, texture, and location information within a spatially varying joint probability distribution. The arising cost functional is minimized by means of convex relaxation techniques. With our efficient GPU implementation of the convex relaxation, the overall algorithm for multiregion segmentation converges within about two seconds. The approach outperforms state-of-the-art methods on the Graz segmentation benchmark.

References

1. Boykov, Y., Jolly, M.: Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. In: IEEE Int. Conf. on Computer Vision (2001)
2. Brox, T., Cremers, D.: On local region models and a statistical interpretation of the piecewise smooth mumford-shah functional. Int. J. of Computer Vision 84, 184–193 (2009)
3. Chambolle, A., Cremers, D., Pock, T.: A convex approach for computing minimal partitions. Tech. rep., TR-2008-05, University of Bonn, Germany (2008)
4. Chan, T., Esedoḡlu, S., Nikolova, M.: Algorithms for finding global minimizers of image segmentation and denoising models. SIAM Journal on Applied Mathematics 66(5), 1632–1648 (2006)
5. Chen, Y., Ranftl, R., Pock, T.: Insights into analysis operator learning: From patch-based sparse models to higher order MRFs. IEEE Trans. on Image Processing 23, 1060–1072 (2014)
6. Dice, L.: Measures of the amount of ecologic association between species. Ecology 26, 297–302 (1945)
7. Elad, M., Milanfar, P., Rubinstein, R.: Analysis versus synthesis in signal priors. Inverse Problems 3(3), 947–968 (2007)
8. Federer, H.: Geometric Measure Theory. Springer (1996)
9. Grady, L.: Random walks for image segmentation. IEEE Trans. on Pattern Analysis and Machine Intelligence 28(11), 1768–1783 (2006)
10. Hawe, S., Kleinsteuber, M., Diepold, K.: Analysis Operator Learning and Its Application to Image Reconstruction. IEEE Trans. on Image Processing 22(6), 2138–2150 (2013)
11. Kiechle, M., Hawe, S., Kleinsteuber, M.: A joint intensity and depth co-sparse analysis model for depth map super-resolution. In: IEEE Int. Conf. on Computer Vision (2013)
12. Lellmann, J., Kappes, J., Yuan, J., Becker, F., Schnörr, C.: Convex multiclass image labeling by simplex-constrained total variation. Tech. rep., HCI, IWR, University of Heidelberg (2008)

13. Mairal, J., Bach, F., Ponce, J., Sapiro, G., Zisserman, A.: Discriminative learned dictionaries for local image analysis. In: Int. Conf. on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
14. Michelot, C.: A finite algorithm for finding the projection of a point onto the canonical simplex of R^n. Journal of Optimization Theory and Applications 50(1), 195–200 (1986)
15. Mignotte, M.: MDS-based segmentation model for the fusion of contour and texture cues in natural images. Computer Vision and Image Understanding (2012)
16. Mobahi, H., Rao, S., Yang, A., Sastry, S., Ma, Y.: Segmentation of natural images by texture and boundary compression. Int. J. of Computer Vision 95 (2011)
17. Mumford, D., Shah, J.: Optimal approximations by piecewise smooth functions and associated variational problems. Communications on Pure and Applied Mathematics 42, 577–685 (1989)
18. Nam, S., Davies, M.E., Elad, M., Gribonval, R.: The Cosparse Analysis Model and Algorithms. Applied and Computational Harmonic Analysis 34(1), 30–56 (2013)
19. Nieuwenhuis, C., Cremers, D.: Spatially varying color distributions for interactive multi-label segmentation. IEEE Trans. on Patt. Anal. and Mach. Intell. 35(5), 1234–1247 (2013)
20. Nieuwenhuis, C., Toeppe, E., Cremers, D.: A survey and comparison of discrete and continuous multi-label optimization approaches for the Potts Model. Int. J. of Computer Vision 104(3), 223–240 (2013)
21. Parzen, E.: On the estimation of a probability density function and the mode. Annals of Mathematical Statistics (1962)
22. Pock, T., Cremers, D., Bischof, H., Chambolle, A.: An algorithm for minimizing the piecewise smooth Mumford-Shah functional. In: IEEE Int. Conf. on Computer Vision (2009)
23. Pock, T., Chambolle, A.: Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In: IEEE Int. Conf. on Computer Vision, pp. 1762–1769 (2011)
24. Ravishankar, S., Bresler, Y.: Learning Sparsifying Transforms. IEEE Transactions on Signal Processing 61(5), 1072–1086 (2013)
25. Rother, C., Kolmogorov, V., Blake, A.: GrabCut: interactive foreground extraction using iterated graph cuts. ACM Transactions on Graphics (Proc. SIGGRAPH) 23(3), 309–314 (2004)
26. Santner, J., Pock, T., Bischof, H.: Interactive multi-label segmentation. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part I. LNCS, vol. 6492, pp. 397–410. Springer, Heidelberg (2011)
27. Santner, J., Unger, M., Pock, T., Leistner, C., Saffari, A., Bischof, H.: Interactive texture segmentation using random forests and total variation. In: British Machine Vision Conference (2009)
28. Tai, Y., Jia, J., Tang, C.: Soft color segmentation and its applications. IEEE Trans. on Patt. Anal. and Mach. Intell. 29(9), 1520–1537 (2007)
29. Tran, T.: Combining color and texture for a robust interactive segmentation algorithm. In: IEEE Int. Conf. Comp. and Comm. Techn., Research, Innov. and Vision for the Future (2010)
30. Unger, M., Pock, T., Cremers, D., Bischof, H.: TVSeg - interactive total variation based image segmentation. In: British Machine Vision Conference (2008)
31. Xiang, S., Nie, F., Zhang, C.: Texture image segmentation: An interactive framework based on adaptive features and transductive learning. In: Narayanan, P.J., Nayar, S.K., Shum, H.-Y. (eds.) ACCV 2006. LNCS, vol. 3851, pp. 216–225. Springer, Heidelberg (2006)

32. Yaghoobi, M., Nam, S., Gribonval, R., Davies, M.E.: Constrained Overcomplete Analysis Operator Learning for Cosparse Signal Modelling. IEEE Transactions on Signal Processing 61(9), 2341–2355 (2013)
33. Yang, A., Wright, J., Ma, Y., Sastry, S.: Unsupervised segmentation of natural images via lossy data compression. Computer Vision and Image Understanding (2008)
34. Zach, C., Gallup, D., Frahm, J.M., Niethammer, M.: Fast global labeling for real-time stereo using multiple plane sweeps. In: Vision, Modeling and Visualization Workshop (VMV) (2008)