Multiple Segmentation of Image Stacks

Jonathan Smets and Manfred Jaeger
Department for Computer Science, Aalborg University, Denmark
jonathansmets@gmail.com, jaeger@cs.aau.dk

Keywords:
Segmentation, Multiple Clustering, Probabilistic Models

Abstract:
We propose a method for the simultaneous construction of multiple image segmentations by combining a recently proposed "convolution of mixtures of Gaussians" model with a multi-layer hidden Markov random field structure. The resulting method constructs for a single image several, alternative segmentations that capture different structural elements of the image. We also apply the method to collections of images with identical pixel dimensions, which we call image stacks. Here it turns out that the method is able to both identify groups of similar images in the stack, and to provide segmentations that represent the main structures in each group.

1 INTRODUCTION

Traditional clustering methods construct a single (possibly hierarchical) partitioning of the data. However, clustering when used as an explorative data analysis tool may not possess a single optimal solution that is characterized as the optimum of a unique underlying score function. Rather, there can be multiple distinct clusterings that each represent a meaningful view of the data. This observation has led to a recent research trend of developing methods for *multiple clustering* (or *multi-view clustering*). The general goal of these methods is to automatically construct several clusterings that represent alternative and complementary views of the data (see (Müller et al., 2012) for a recent overview, and the proceedings of the MultiClust workshop series for current developments).

The perhaps most typical application area for multiple clustering is document data (e.g. collections of news articles or web pages). For example, the standard benchmark WebKB dataset consists of university webpages that can be alternatively clustered according to page-type (e.g. personal homepage or course page), or the different universities the pages are taken from. Turning to image data, previously used benchmark sets are the CMU and the Yale Face Images data, which consists of portrait images of different persons in several poses, and accordingly can be clustered according to persons or poses (Cui et al., 2007; Jain et al., 2008). In this setting, each image is a data-point, and (multiple) clustering means grouping images. When, instead, one views as a data-point a single image pixel, then multiple clustering becomes *multiple image segmentation.*

Relatively few work has been done on finding multiple, alternative image segmentations. Kim and Zabih (2002) developed a quite specific *factorial Markov random field* model in which an image is modeled as an overlay of several layers, and each layer corresponds to a binary segmentation. Qi and Davidson (2009) apply a general multiple clustering approach to a variety of datasets, including images. Their multiple clustering approach falls into the category of iterative multiple clustering, where given an initial (primary) clustering, a single alternative clustering is constructed. Our approach, on the other hand, falls into the category of simultaneous multiple clustering methods, where an arbitrary number of different clusterings is constructed at the same time, and without any priority ordering among the clusterings. Finally, Kato et al. (2003) generate alternative segmentations based on color and texture features, respectively. However, the objective here is not to provide different, alternative segmentations, but to combine the two segmentations into a single one.

It is worth emphasizing that multiple clustering in the sense here considered is different from the construction of *cluster ensembles* (Strehl and Ghosh, 2003). In the latter, numerous clusterings are built in order to overcome the convergence to only locally optimal solutions of clustering algorithms, and to construct out of a collection of clusterings a single consensus clustering. The multiple segmentations in the sense of (Hoiem et al., 2005; Russell et al., 2006) are segmentation analogues of cluster ensembles, not of multiple clusterings in our sense.

In this paper we develop a method for constructing multiple segmentations of images and *image stacks*, which we define as a collection of images with equal pixel dimensions. The most import type of image stacks are the collection of frames in a video sequence. However, we can also consider other such collections of pixel-aligned images. As we will see in the experimental section, multiple clustering of such image stacks can give results that combine elements of clustering at the image and at the pixel level. For the design of our method we build on the *convolution of mixtures of Gaussians* model of (Jain et al., 2008) which we customize for the segmentation setting by combining it with a Markov Random Field structure to account for the spatial dimension of the data.

Our approach is intended as a general method that can be applied to image data of quite different types, and that thereby is a quite general tool for explorative image data analysis. For more specialized application tasks, our general method may serve as a basis, but will presumably require additional modifications and adaptations.

2 THE CONVOLUTIONAL CLUSTERING MODEL

Probabilistic clustering approaches are based on *latent variable models* where a data point $\boldsymbol{x}$ is assumed to be sampled from a joint distribution $P(\boldsymbol{X}, L \mid \boldsymbol{\theta})$ of an observed data variable $\boldsymbol{X}$ and a latent variable $L \in \{1, \ldots, k\}$, governed by parameters θ (throughout this paper we use bold symbols to denote tuples of variables, parameters, etc.; when talking about random variables, then uppercase letters stand for the variables, and lowercase letters for concrete values of the variables). Clustering then is performed by learning the parameters $\boldsymbol{\theta}$, and assigning $\boldsymbol{x}$ to the cluster with index i for which $P(\boldsymbol{X} = \boldsymbol{x}, L = i \mid \boldsymbol{\theta})$ is maximal.

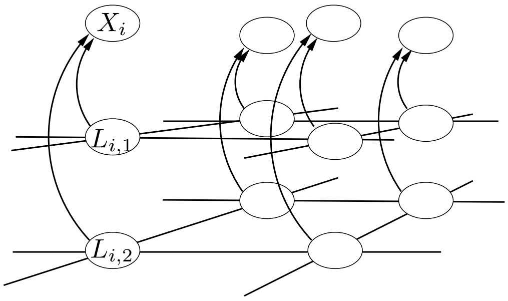


Figure 1: Multi-layer Hidden Markov Random Field

This probabilistic paradigm is readily generalized to multiple clustering models. One only needs to design a model $P(\boldsymbol{X}, \boldsymbol{L} \mid \boldsymbol{\theta})$ containing multiple latent variables $\boldsymbol{L} = L_1, \ldots, L_m$. Then the joint assignment $L_1 = i_1, \ldots, L_m = i_m$ (abbreviated $\boldsymbol{L} = \boldsymbol{i}$) maximizing $P(\boldsymbol{X} = \boldsymbol{x}, L_1 = i_1, \ldots, L_m = i_m \mid \boldsymbol{\theta})$ defines the cluster indices for $\boldsymbol{x}$ in m distinct clusterings. Models for multiple clustering that are based on multiple latent variables include the factorial Hidden Markov Model (Ghahramani and Jordan, 1997), the factorial Markov Random Fields of (Kim and Zabih, 2002), convolution of mixtures of Gaussians (Jain et al., 2008), the latent tree models of (Poon et al., 2010), and the factorial logistic model of (Jaeger et al., 2011).

2.1 The Probabilistic Model

Our model is structurally identical to the factorial Markov Random Field model of (Kim and Zabih, 2002). Figure 1 shows the structure of such a *multi-layer hidden Markov random field*: with each pixel $i \in I$ (I the set of all pixels) are associated m latent variables $\boldsymbol{L}_{i,\bullet} = L_{i,1}, \ldots, L_{i,m}$ and a vector of observed variables $\boldsymbol{X}_i$. For $k = 1, \ldots, m$ the variables $\boldsymbol{L}_{\bullet,k} = L_{1,k}, \ldots, L_{|I|,k}$ take values in the set $\{1, \ldots, n_k\}$, so that the kth segmentation will consist of n_k segments.

For this paper we assume that in the case of single image analysis, $\boldsymbol{X}_i$ is simply the 3-dimensional vector (R_i, G_i, B_i) of *rgb*-values at pixel i. In the case of image stacks with N images, $\boldsymbol{X}_i$ will be a $3 \cdot N$-dimensional vector containing the *rgb*-values of all images in the stack. We denote with $|\,\boldsymbol{X}\,|_i$ the dimension of $\boldsymbol{X}_i$. Though we do not explore this in the current paper, we note that $\boldsymbol{X}_i$ could also contain differently defined observed features of pixel i.

For every $k = 1, \ldots, m$, the latent variables $\boldsymbol{L}_{\bullet,k}$ form a Markov random field with a square grid structure. The distribution of $\boldsymbol{X}_i$ depends conditionally on the latent variables $\boldsymbol{L}_{i,\bullet}$.

The marginal distribution $P(\boldsymbol{L} \mid \boldsymbol{\theta})$ is defined as a product of m Potts models defined by a common temperature parameter T:

$$P(\boldsymbol{L} = \boldsymbol{l} \mid \boldsymbol{\theta}) = P(\boldsymbol{L} = \boldsymbol{l} \mid T) = \frac{1}{Z} \prod_{k=1}^{m} e^{V(\boldsymbol{L}_{\bullet,k} = \boldsymbol{l}_{\bullet,k})/T}$$

where Z is the normalization constant, and

$$V(\boldsymbol{L}_{\bullet,k} = \boldsymbol{l}_{\bullet,k}) = \sum_{i,j:i\sim j} \mathbb{I}(l_{i,k} \neq l_{j,k})$$

with $\mathbb{I}(l_{i,k} \neq l_{j,k}) = 1$ if $l_{i,k} \neq l_{j,k}$, and $= 0$ otherwise.

For the conditional distribution $P(\boldsymbol{X} \mid \boldsymbol{L}, \boldsymbol{\theta})$ the model of Figure 1 implies conditional independence for different pixels of the observed pixel features $\boldsymbol{X}_i$ given the latent pixel variables $\boldsymbol{L}_{i,\bullet}$. Moreover, we assume that the conditional model $P(\boldsymbol{X}_i \mid \boldsymbol{L}_{i,\bullet}, \boldsymbol{\theta})$ is identical for all i. It is defined as the convolution of m mixtures of Gaussians as follows. For $k = 1, \ldots, m$ and $j = 1, \ldots, n_k$ let $\mu_{k,j} \in \mathbb{R}^{|\boldsymbol{X}_i|}$. Writing $\boldsymbol{\mu}_k = \mu_{k,1}, \ldots, \mu_{k,n_k}$, we obtain for every k a distribution for a variable $\boldsymbol{Z}_{i,k}$ defined as a mixture of Gaussians

$$P(\boldsymbol{Z}_{i,k} \mid L_{i,k}, \boldsymbol{\mu}_k) = \sum_{j=1}^{n_k} N(\mu_{k,j}, \mathbf{1}) \mathbb{I}(L_{i,k} = j),$$

where $\mathbf{1}$ stands for the unit covariance matrix. For two distributions $P(\boldsymbol{Y}), P(\boldsymbol{Z})$ of two k-dimensional real random variables $\boldsymbol{Y}, \boldsymbol{Z}$, we denote with $P(\boldsymbol{Y}) * P(\boldsymbol{Z})$ their convolution, i.e., the distribution of the sum $\boldsymbol{X} = \boldsymbol{Y} + \boldsymbol{Z}$. The final model for $\boldsymbol{X}_i$ now is defined as the m-fold convolution:

$$P(\boldsymbol{X}_i \mid \boldsymbol{L}_{i,\bullet}, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_m) = P(\boldsymbol{Z}_{i,1} \mid L_{i,1}, \boldsymbol{\mu}_1) * \cdots * P(\boldsymbol{Z}_{i,m} \mid L_{i,m}, \boldsymbol{\mu}_m).$$

Combining the model for $\boldsymbol{L}$ and $\boldsymbol{X} \mid \boldsymbol{L}$, We now obtain

$$\log(P(\boldsymbol{L} = \boldsymbol{l}, \boldsymbol{X} = \boldsymbol{x} \mid \boldsymbol{\mu}, T)) \approx - 1/T \sum_{k=1}^{m} \sum_{i,j:i\sim j} \mathbb{I}(l_{i,k} \neq l_{j,k}) - \sum_{i \in I} \| \boldsymbol{x}_i - \sum_{k=1}^{m} \mu_{k,l_{i,k}} \|^2 \quad (1)$$

2.2 The Regularization Term

Maximizing the log-likelihood (1) alone is a sound approach to probabilistic multiple segmentation. However, Jain et al. (2008) suggest to add to the likelihood the *regularization term*

$$-\lambda \sum_{\substack{k,k'=1,\ldots,m \\ k \neq k'}} \sum_{\substack{j=1,\ldots,n_k \\ j'=1,\ldots,n_{k'}}} (\mu_{k,j} \cdot \mu_{k',j'})^2 \quad (2)$$

Here $\lambda \geq 0$ is a weight parameter that regulates the strength of the influence of the regularization term. This penalty term is minimized when the means $\boldsymbol{\mu}_k, \boldsymbol{\mu}_{k'}$ corresponding to different segmentations lie in orthogonal subspaces. The rationale given for this regularization term is twofold. First, the likelihood function (1) does not have a unique maximum. Indeed, taking the case $m = 2$, the two solutions $(\mu_{1,1}, \ldots, \mu_{1,n_1}, \mu_{2,1}, \ldots, \mu_{2,n_2}, T)$ and $(\mu_{1,1} + c, \ldots, \mu_{1,n_1} + c, \mu_{2,1} - c, \ldots, \mu_{2,n_2} - c, T)$ $(c \in \mathbb{R}^3)$ define the same distribution, and therefore have the same likelihood score. Second, the likelihood alone does not give an explicit reward for the distinctness, or complementarity, of the resulting multiple clusterings. Following other approaches to multiple clustering, it is hoped that encouraging the means corresponding to different clusterings to lie in orthogonal subspaces will lead to a greater diversity of those clusterings.

We argue that the form and justification for this particular regularization term is slightly flawed, and that it should be replaced by a modified version. First, we note that the non-uniqueness of the optimal solution for (1) is not a real problem as long as two different optimal solutions define the same multiple segmentation. This, however, is exactly the case for the two solutions distinguished by the offset vector c as described above. Second, regularization with (2) is not invariant under simple shifts of the coordinate system: adding a constant vector $\boldsymbol{z}$ to all datapoints $\boldsymbol{x}_i$ should have no effect on the optimal segmentation, which should be characterized by also adding $\boldsymbol{z}$ to all model parameters $\mu_{k,j}$. Since (2) is not invariant under addition of a constant to all $\mu_{k,j}$, this is not the behavior one obtains with this regularization term. We therefore propose to modify (2) so as to reward means $\boldsymbol{\mu}_k, \boldsymbol{\mu}_{k'}$ to lie in orthogonal affine sub-spaces, rather than orthogonal linear sub-spaces. Thus, we propose the following regularization term:

$$-\lambda \sum_{\substack{k,k'=1,\ldots,m\\ k\neq k'}} \quad \sum_{\substack{j,h=1,\ldots,n_k:j<h\\ j',h'=1,\ldots,n_{k'}:j'<h'}}$$

$$\left(\frac{\mu_{k,j}-\mu_{k,h}}{\parallel \mu_{k,j}-\mu_{k,h} \parallel}\cdot\frac{\mu_{k',j'}-\mu_{k',h'}}{\parallel \mu_{k',j'}-\mu_{k',h'} \parallel}\right)^2. \quad (3)$$

Thus, we reward solutions in which normalized difference vectors between the means of different layers are orthogonal, rather than the means themselves. The term (3) now is invariant under adding, respectively subtracting, a constant vector c to all means of two different layers, and hence we again have the non-uniqueness of optimal solutions as for the pure likelihood (1). However, as argued above, we do not see this as a problem.

One small practical problem arises when we define our objective function as the sum of (1) and (3): the likelihood term (1) increases in magnitude linearly with the number of pixels. The regularization term, on the other hand, only increases as a function of the number of layers and the number of segments per layer. The choice of an appropriate tradeoff parameter λ between likelihood and regularization term, thus, would depend on the number of pixels. In order to get a more uniform scale for λ across different experiments, we therefore normalize the regularization term with the factor $\mid I \mid /K$, where K is the number of terms in the sum (3).

We remark that the probabilistic model (1) alone also has some built-in capability to encourage a diversity in the parameters $\boldsymbol{\mu}_k$ for different layers, and hence, in the different segmentations. This is because having two layers with very similar means $\boldsymbol{\mu}_k$ does not allow a much better fit to the data than a single layer with those means. Exploiting the full parameter space of the model to obtain a good fit to the data, thus, will tend to lead to some diversity in the parameters $\boldsymbol{\mu}_k$. For this reason, in our experiments, we also pay particular attention to the case $\lambda = 0$, i.e., segmentation according to the pure probabilistic model (1).

2.3 Clustering Algorithm

We take the model parameter $\beta := 1/T$ and the regularization parameter λ as user-defined inputs that may be varied in an iterative data exploration process. Large values of β mean that high emphasis is put on segmentations with large connected segments and smooth boundaries. Larger values of λ mean that diversity of segmentations as measured by the regularization term (3) is more strictly enforced.

Thus, the only model-parameters we have to fit are the mean vectors $\boldsymbol{\mu}_k$. Our goal, then, is to maximize a score function $S(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_m, \boldsymbol{l})$ which is given as the sum of (1) and (3).

We use a typical 2-phase iterative process for this optimization: in a *MAP*-step we compute for a current setting of the $\boldsymbol{\mu}_k$ the most probable assignment $\boldsymbol{L} = \boldsymbol{l}$ for the latent variables according to the likelihood function (1) (since (3) does not depend on $\boldsymbol{l}$, we can ignore it in this phase). In a *M(aximization)*-step we recompute for the current setting $\boldsymbol{L} = \boldsymbol{l}$ the $\boldsymbol{\mu}_k$ optimizing $S(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_m, \boldsymbol{l})$. This well-known clustering approach (sometimes referred to as *hard EM*) has also been proposed for image segmentation in (Chen et al., 2010).

2.3.1 MAP-step

For the MAP-step we make use of the α-*expansion* algorithm of (Boykov et al., 2001; Kolmogorov and Zabin, 2004; Boykov and Kolmogorov, 2004). This algorithm provides solutions to segmentation problems characterized by an energy function E for segmentations s, which are of the form

$$E(s) = \sum_{i,j:i\sim j} V_{i,j}(s(i), s(j)) + \sum_i D_i(s(i)), \quad (4)$$

where $s(i)$ is the segment label of pixel i, $V_{i,j}$ is a penalty function for discontinuities in s, and D_i is any non-negative function measuring the discrepancy of the label assignment $s(i)$ with the observed data for i. It is shown in (Boykov et al., 2001) that if $V_{i,j}(s(i), s(j))$ is a metric on the label space, then the α-expansion algorithm is guaranteed to find a solution s that is within a constant factor of the globally minimal energy $E()$.

Up to a change of sign (and a corresponding change from a minimization to a maximization objective) our likelihood function (1) has the form (4) for the m-dimensional label space $\times_{k=1}^{m}\{1, \ldots, n_k\}$ (i.e. $s(i) = (l_{i,1}, \ldots, l_{i,m})$), with $V_{i,j}(s(i), s(j)) = \sum_{k=1}^{m} \mathbb{I}(l_{i,k} \neq l_{j,k})$ and $D_i(s(i)) = \parallel \boldsymbol{x}_i - \sum_{k=1}^{m} \mu_{k,l_{i,k}} \parallel^2$.

Furthermore, it is straightforward to see that our $V_{i,j}$ is a metric on the m-dimensional label space.

To use the α-*expansion* algorithm we flatten our m-dimensional label space to a one-dimensional label space with $\prod_{k=1}^{m} n_k$ different labels. Thus, our method has a complexity that is exponential in the number of layers. On the other hand, the α-expansion algorithm in practice is quite efficient as a function of the number of pixels. It is reputed to show a linear complexity in practice (Boykov et al., 2001), which was confirmed by our observations in our experiments.

2.3.2 M-step

The M-step is performed by gradient ascent, leading to a local maximum of the score function given the current segmentation $\boldsymbol{L} = \boldsymbol{l}$.

2.3.3 Implementation

The algorithm is implemented in Matlab, using the α-expansion implementation provided by the gco-v3.0 library available on `http://vision.csd.uwo.ca/code/`.

3 EXPERIMENTS

In all our experiments we construct multiple segmentations with the same number of segments in each layer. We therefore refer to a multiple segmentation with m layers and k segments in each layer as a (m, k)-segmentation.

3.1 Single Images

Our first experiment establishes the baseline result that the segmentation methods works as intended when the input closely fits the underlying modeling assumption. To this end we construct the image shown in Figure 2 (c) as the overlay of the two images (a) and (b), and used our method to construct (2,3)-segmentations from the single input image (c). First setting $\lambda = \beta = 0$, we performed 200 runs of the algorithm with different random initializations. The highest-scoring solution that was found consists of the segmentations (d) and (e). In these figures, the color of the jth segment in the kth layer is set to $\tilde{\mu}_{k,j}$, where $\tilde{\mu}_{k,j}$ is obtained from $\mu_{k,j}$ by applying min-max normalization to re-scale the components of all the mean vectors $\boldsymbol{\mu}_k$ $(k = 1, \ldots, m)$ into the interval $[0..255]$ of proper rgb-values. Essentially the same optimal result was found in 9 out of the 200 runs. In the remaining runs the algorithm converged to local minima, an example of which is shown by (f) and (g). These results were clearly identified by the algorithm as sub-optimal by being associated with significantly lower score function values.

With increasing λ parameter the results in this experiment deteriorated. At $\lambda = 5000$ the "correct" solution was not found in 200 restarts. This is not very surprising, since for this image with $\lambda = \beta = 0$ the correct solution is clearly distinguished as the solution that can achieve a perfect score of 0 on the remaining Euclidean part of the likelihood term (1).

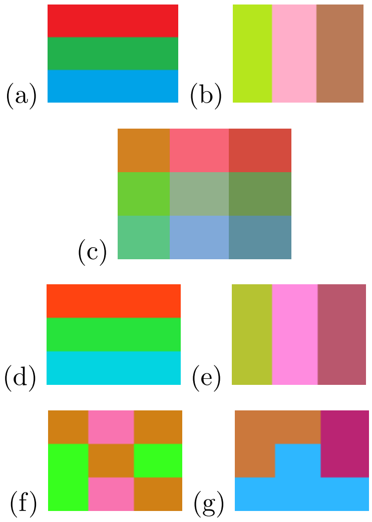


Figure 2: Baseline: overlay image

Next, we perform a series of experiments on the butterflies image by M.C. Escher, shown in Figure 3, which has previously been used in (Qi and Davidson, 2009). The size of this image is 402x401 pixels.

We first compute (2,3)-segmentations with varying values of λ (and $\beta = 0$). Figure 4 shows the highest scoring results (in 20 restarts) obtained for $\lambda = 0, 1000, 10000$. In all cases, essentially the same two segmentations are computed: one that corresponds to the main colors of the three types of butterflies in the image, and one that captures the finer structure of the borders between the butterflies, as well as the shading inside the butterflies. The main effect of the regularization term here is not a difference in the

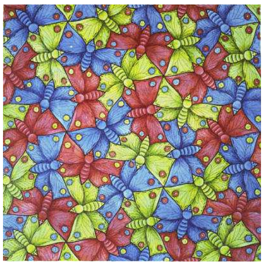


Figure 3: Escher's butterflies

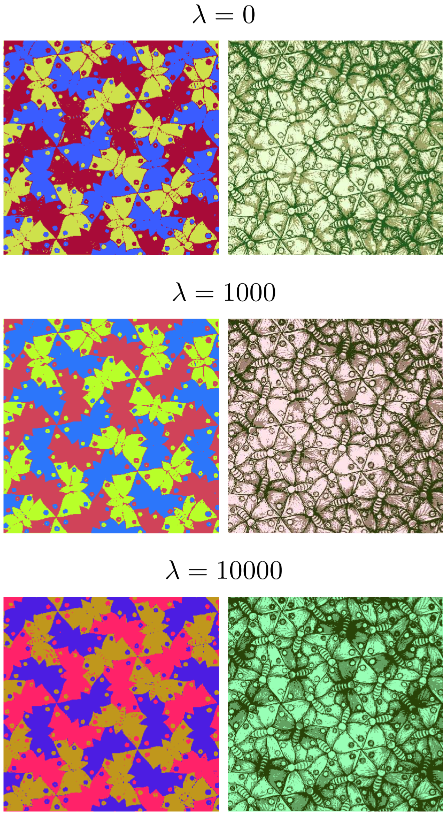


Figure 4: Escher (2,3)-segmentations, varying λ

segmentations, but only a difference in the means associated with the segments: for the high value $\lambda = 10000$, the means in the second segmentation all have a strong green component, whereas the means of the first component only have weak green components. This makes the means of the two components lie in near-orthogonal affine spaces. A similar color-separation does not appear at $\lambda = 0$.

A common way to measure dissimilarity of two clusterings L_1, L_2 is *normalized mutual information*

$$NMI(L_1, L_2) = \frac{MI(L_1, L_2)}{\sqrt{H(L_1)H(L_2)}},$$

where MI is the mutual information and $H()$ the entropy of L_1, L_2, as determined by the empirical joint distribution of L_1, L_2 defined by the cluster assignments of the pixels. Low values of NMI indicate statistical independence, and hence dissimilarity of clusterings. Furthermore, a justification given by Jain et al. (2008) for the regularization term (2) is that it induces a bias towards statistically independent clusterings. This justification carries over to our modified version (3). Therefore, the *NMI* as an evaluation measure is quite consistent with our objective function.

However, while low values of the regularization term can be due to statistical independence of the segmentations, this is not a strict correlation. As discussed above, the increasing weight of the regularization term in Figure 4 only leads to a shift of the mean rgb-vectors without a noticeable change in the segmentations. This leads to an improvement in the value of the regularization term from $8.28 \cdot 10^6$ at $\lambda = 1000$ to $1.82 \cdot 10^6$ at $\lambda = 10000$ (at $\lambda = 0$ no regularization term is computed). However, the NMI values for the three solutions of Figure 4 are $8.4 \cdot 10^{-3}, 5.4 \cdot 10^{-2}, 7.1 \cdot 10^{-2}$ for $\lambda = 0, 1000, 10000$, respectively. Thus, the NMI values are even slightly increasing for larger λ-values.

We note at this point that NMI values have to be used with caution when assessing dissimilarity of image segmentations (rather than other types of data clusterings): NMI is a function only of cluster membership of pixels. However, for segmentations one is perhaps more interested in the borders defined between segments, than in the global grouping of pixels into segments. Figures 5-7 illustrate this issue. Figure 5 shows a modified version of Escher's butterflies in which we have superimposed an additional square grid structure on the butterfly image. Figure 6 is shows a hypothetical (2,4)-segmentation (not computed by our method) of this image. Both segmentations identify the grid structure – the first one dividing the structure according to columns (and background), the second according to rows (and background). For the non-background pixels row and column membership are independent random variables. The mutual information of the two segmentations therefore reduces to $-P(b)\log P(b) - (1 - P(b))log(1 - P(b))$, where $P(b)$ is the probability of background pixels (i.e. the relative image area covered by background). In the limit where the size of the squares is increased, and $P(b) \rightarrow 0$, the mutual information of the two segmentations, thus, goes to zero (and so does the normalized mutual information). This shows that dissimilarity as measured by low mutual information need not correspond to the kind of complementarity we may be looking for in different segmentations. Figure 7 shows the (2,4)-segmentation actually obtained by our method. The result shown is for $\lambda = 0$, but results for higher λ-values are similar. Clearly, we will not obtain segmentations similar to those in Figure 6, since these would score very poorly in

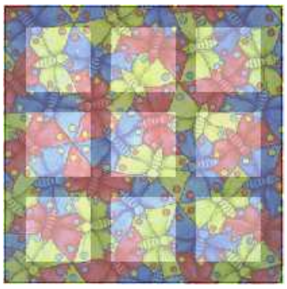


Figure 5: Butterflies with squares

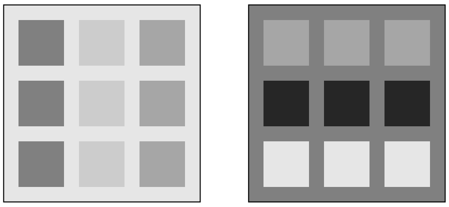


Figure 6: Segmentations with low mutual information

the likelihood term (1), and their low NMI score also would not be reflected in a low value of the regularization term.

We see, thus, that neither need there be a good correspondence between low NMI values and complementarity of segmentations in the intuitive sense, nor does the regularization term necessarily induce a strong bias towards low NMI solutions. Fortunately, as Figure 7 shows, the likelihood score alone is quite successful in producing segmentations that are complementary in an intuitively meaningful sense.

In the next experiment we keep $\lambda = 0$ fixed, and vary $\beta = 1000, 16000$. As the results in Figure 8 show, the effect is quite consistent with expectations: the already fairly smooth first segmentation remains quite stable (even though some further smoothing of the borders occurs), whereas the smoothing of the initially rather fragmented second segmentation leads to an eventual dissolving of the structure, including the elimination of one of the three segments (we note that we here always manually label segmentations as “first” and “second” to facilitate the comparison; the algorithm may return either segmentation with index 1 or 2).

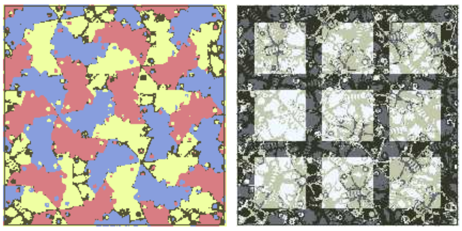
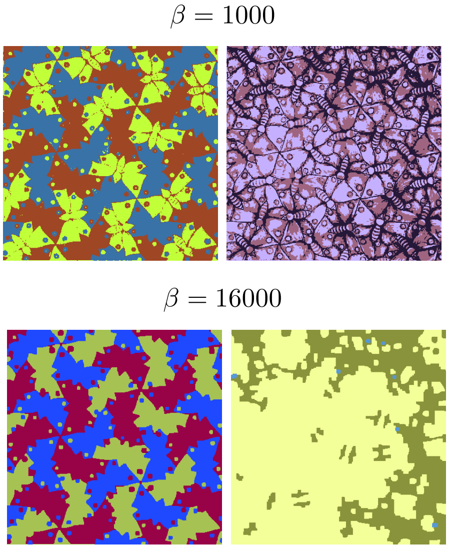


Figure 7: Actual (2,4)-segmentation for butterflies with squares



$\beta = 1000$

$\beta = 16000$

Figure 8: Escher (2,3)-segmentations, varying β

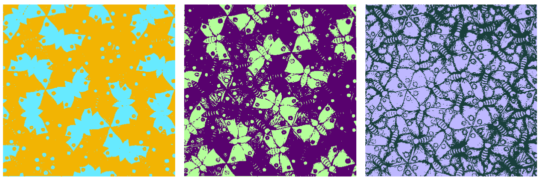


Figure 9: Escher (3,2)-segmentation

Finally, we perform a (3,2)-segmentation with $\lambda = \beta = 0$. The result is shown in Figure 9. The first segmentation again is based on the main underlying color distribution, isolating the blue butterflies from the rest. The last segmentation again represents mostly the border structure and shading. Finally, the segmentation in the middle is mostly identifying the green butterflies, but also represents some structure. Qi and Davidson (2009) present a (2,2)-segmentation for the butterfly image obtained from their iterative clustering method. Their two segmentations are quite similar in nature to the first two in Figure 9.

3.2 Image Stacks

As a first experiment with an image stack, we used the collection of 25 flag-images shown in Figure 10 (each at a resolution of 150 × 75 pixels).

Again setting $\lambda = \beta = 0$, the highest scoring (2,3)-segmentation is shown at the bottom of Figure 10. Here we now depict the different segments using arbitrarily chosen greyscale values. The means $\mu_{k,j}$ characterizing segments now

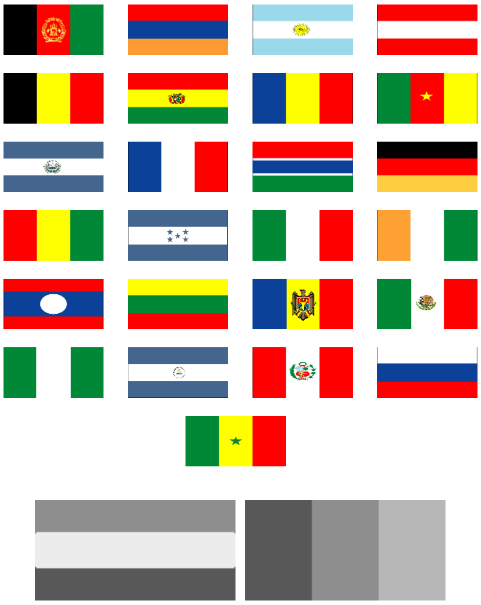


Figure 10: Stack of flag images

are $3 \cdot 25$ dimensional vectors that can be interpreted as an average color sequence for pixels in a segment. Taking for visualization the average over all colors in the sequence typically leads to all segments represented by very similar brownish colors (although, curiously, in this particular case the average colors for the segmentation with the vertical stripes yield a somewhat washed-out looking French flag). The same "correct" solution here was found in 9 out of 50 random restarts.

A second image stack we constructed consists of 10 images each of trains and horses, as shown in Figure 11. We performed (2,3)-segmentation with $\lambda = 0$ and $\beta = 50$. The highest scoring result within 400 runs is shown at the bottom of Figure 11. The method identifies the main structures in the two groups of images also in this somewhat more diverse collection of images. The results in the different runs were relatively stable, with other high-scoring solutions similar to the top-scoring one. Results with lower scores often separated the two groups of images less clearly, or contained segmentations in which on segment was reduced to very few pixels.

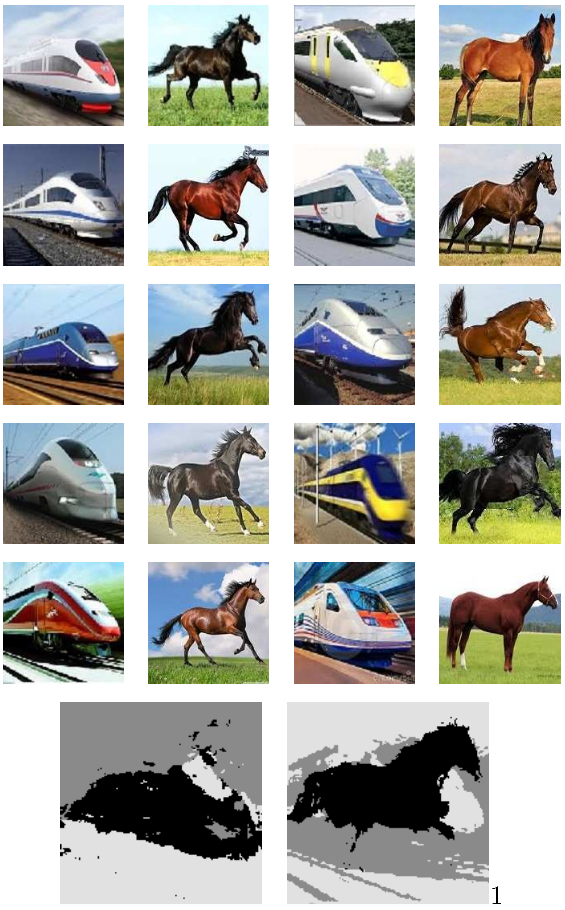


Figure 11: Stack of Horse and Train images

In all our experiments results were quite robust under variations of the λ and β parameters. Good results are typically already obtained at the baseline setting $\lambda = \beta = 0$. Note that $\beta = 0$ means that the Markov random field structure of the model is ignored, and that the MAP step could be implemented in a much simplified manner. In applications where smooth and contiguous segments are required, settings of $\beta > 0$ will be needed. The impact of the λ parameter on the segmentations was rather small. It appears that larger values of λ affected the placement of the mean parameters representing the different segments, but not so much the resulting segmentations themselves.

We close this section with some information on the runtimes of our experiments: a single run of a (2,3) or (3,2)-segmentation of the 402x401 pixel butterfly image takes about 1 minute on average, with an average of about 8 iterations of MAP and M steps until our termination criterion is met that the score improvement in one iteration is less than 2%. The same experiments with the image at twice the resolution take about twice as long. The average runtime for the Horse-Train image stack also is about 1 minute. The higher dimensionality of the feature vector here is offset by the smaller number of pixels at the resolution of 151x151 for the images in the stack. For the Horse-Train stack most of the computation time (about 90%) is taken by the M step, which is more affected by the dimensionality of the feature vector. For the butterfly image, on the other hand, most of the time (approx. 70%) is spent on the MAP step.

4 CONCLUSION

We have introduced a method for constructing multiple segmentations of image stacks by combining the convolution of mixtures of Gaussians model (Jain et al., 2008) with a multi-layer Markov Random field. While novel in this form, the resulting model is a quite straightforward combination of existing components. The main original contribution of this paper is the first dedicated investigation of multiple clustering for image segmentation, and the introduction of (multiple) segmentation of image stacks. We note that the latter is different from cosegmentation (Rother et al., 2006) and standard video segmentation, where also "stacks" of images are segmented simultaneously, but where a separate segmentation is computed for each image (or frame).

We have conducted a range of experiments that demonstrate that the method is able to produce meaningful results in a broad variety of datasets. Applied to single images, it is able to identify the structures of multiple constituent components. Applied to image stacks, it can perform a simultaneous clustering at the image and at the pixel level. All these results were obtained using only the basic rgb pixel features. No task-specific preprocessing or feature engineering was needed to obtain our results. One can thus conclude, that the proposed method provides a useful baseline approach for explorative image analysis.

For more specific application purposes or data analysis objectives, it will be necessary to construct more specific pixel features. One possible such application domain is multiple segmentation of video sequences. The frames of a video can obviously be seen as an image stack. Using only the rgb pixel features our method is not very well adapted to video analysis, since it does not take into account the temporal order of the frames. New pixel features that capture some of the temporal dynamics of the pixel values can be constructed, for example, simply by considering the variance of the pixel's rgb values, or by constructing features that describe the trajectory of the pixel's rgb values in rgb-space. Performing multiple segmentation of video sequences based on such features is a topic for future work.

In this paper we have also tried to evaluate the usefulness of regularization terms along the lines proposed in (Jain et al., 2008) for stimulating diversity in the multiple segmentations. Our results lead to some doubts both with regard to the effectiveness of the regularization term to produce segmentations with low mutual information, and with regard of the usefulness of mutual information as a measure for diversity in image segmentations. On the other hand, our results indicate that the likelihood term (1) alone is quite capable of identifying the most relevant, distinct segmentations.

REFERENCES

Boykov, Y. and Kolmogorov, V. (2004). An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(9):1124–1137.

Boykov, Y., Veksler, O., and Zabih, R. (2001). Fast approximate energy minimization via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(11):1222–1239.

Chen, S., Cao, L., Wang, Y., Liu, J., and Tang, X. (2010). Image segmentation by map-ml estimations. *Image Processing, IEEE Transactions on*, 19(9):2254–2264.

Cui, Y., Fern, X., and Dy, J. (2007). Non-redundant multi-view clustering via orthogonalization. In *Proceedings of Seventh IEEE International Conference on Data-Mining (ICDM 2007)*, pages 133 – 142.

Ghahramani, Z. and Jordan, M. (1997). Factorial hidden Markov models. *Machine Learning*, 29(2-3):245–273.

Hoiem, D., Efros, A., and Hebert, M. (2005). Geometric context from a single image. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, pages 654–661 Vol. 1.

Jaeger, M., Lyager, S. P., Vandborg, M. W., and Wohlgemuth, T. (2011). Factorial clustering with an application to plant distribution data. In *Proceedings of the 2nd MultiClust Workshop: Discovering, Summarizing and Using Multiple Clusterings*, pages 31–42. Online proceedings http://dme.rwth-aachen.de/en/MultiClust2011.

Jain, P., Meka, R., and Dhillon, I. S. (2008). Simultaneous unsupervised learning of disparate clusterings. *Statistical Analysis and Data Mining*, 1(3):195–210.

Kato, Z., Pong, T.-C., and Qiang, S. G. (2003). Unsupervised segmentation of color textured images using a multilayer mrf model. In *Proceedings or the IEEE International Conference on Image Processing (ICIP 2003)*, volume 1, pages 961–964. IEEE.

Kim, J. and Zabih, R. (2002). Factorial Markov random fields. In Heyden, A., Sparr, G., Nielsen, M., and Johansen, P., editors, *Computer Vision – ECCV 2002*, volume 2352 of *Lecture Notes in Computer Science*, pages 321–334. Springer Berlin Heidelberg.

Kolmogorov, V. and Zabin, R. (2004). What energy functions can be minimized via graph cuts? *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(2):147–159.

Müller, E., Günnemann, S., Färber, I., and Seidl, T. (2012). Discovering multiple clustering solutions: Grouping objects in different views of the data. In *Proceedings of 28th International Conference on Data Engineering (ICDE-2012)*, pages 1207–1210.

Poon, L. K. M., Zhang, N. L., Chen, T., and Wang, Y. (2010). Variable selection in model-based clustering: To do or to facilitate. In *Proceedings of the 27th International Conference on Machine Learning (ICML-2010*, pages 887–894.

Qi, Z. and Davidson, I. (2009). A principled and flexible framework for finding alternative clusterings. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD-09)*, pages 717–725.

Rother, C., Minka, T., Blake, A., and Kolmogorov, V. (2006). Cosegmentation of image pairs by histogram matching-incorporating a global constraint into mrfs. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 993–1000. IEEE.

Russell, B., Freeman, W., Efros, A., Sivic, J., and Zisserman, A. (2006). Using multiple segmentations to discover objects and their extent in image collections. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1605–1614.

Strehl, A. and Ghosh, J. (2003). Cluster ensembles — a knowledge reuse framework for combining multiple partitions. *J. Mach. Learn. Res.*, 3:583–617.