Optimizing Parametric Total Variation Models

Strandmark, Petter; Kahl, Fredrik; Overgaard, Niels Christian

Published in:
[Host publication title missing]

DOI:
10.1109/ICCV.2009.5459464

2009

Optimizing Parametric Total Variation Models

Petter Strandmark Fredrik Kahl Niels Chr. Overgaard
Centre for Mathematical Sciences, Lund University, Sweden
{petter,fredrik,nco}@maths.lth.se

Abstract

One of the key factors for the success of recent energy minimization methods is that they seek to compute global solutions. Even for non-convex energy functionals, optimization methods such as graph cuts have proven to produce high-quality solutions by iterative minimization based on large neighborhoods, making them less vulnerable to local minima. Our approach takes this a step further by enlarging the search neighborhood with one dimension.

In this paper we consider binary total variation problems that depend on an additional set of parameters. Examples include:

(i) the Chan-Vese model that we solve globally

(ii) ratio and constrained minimization which can be formulated as parametric problems, and

(iii) variants of the Mumford-Shah functional.

Our approach is based on a recent theorem of Chambolle which states that solving a one-parameter family of binary problems amounts to solving a single convex variational problem. We prove a generalization of this result and show how it can be applied to parametric optimization.

1. Introduction

The Mumford-Shah functional [15] is a widely used functional for image segmentation. As a special case, Chan and Vese proposed in [6] a segmentation method where an image is approximated with a function taking only two values. By minimizing an energy consisting of a smoothness term added to the squared distance between the original and the approximation, a large variety of images can be segmented correctly. However, the exact minimization of this energy functional is a difficult problem and this paper will describe new results on this topic obtained by generalizing recent results by Chambolle [4] and Chan et al. [5].

This paper considers optimization problems over images, where without loss of generality the image $I : \mathbf{R}^2 \supset \Omega \to \mathbf{R}$ is assumed to take values on $[0, 1]$.

Our main contribution is that we show how one can evaluate real-valued functions of the following type:

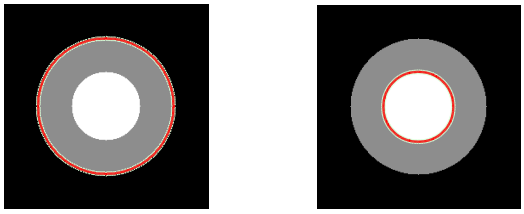
$$m(\boldsymbol{t}) = \min_{\theta,s} E(\theta, s, \boldsymbol{t}), \tag{1}$$

where E is a functional depending on the binary valued function $\theta : \Omega \to \{0, 1\}$, the one-dimensional parameter $s \in \mathbf{R}$ as well as some additional vector of real parameters $\boldsymbol{t}$. The ability to evaluate such functions allows us to efficiently optimize parametric, binary total variation models including several variants of the Mumford-Shah functional. The standard way of solving such problems is by alternating optimization: (1) keep the real parameters fixed and solve for θ, (2) keep θ fixed and solve for the real parameters. By including one additional parameter in the first step, the neighborhood search is enlarged and the risk of getting trapped in local minima is reduced.

Another consequence of (1) is that we can obtain globally optimal solutions to low-order parametric total variation models. We analyze in more detail one of the main problems in this class of segmentation models, namely the Chan-Vese model. We want to approximate the image with a function taking only two values μ_0 and μ_1, by solving the following optimization problem:

$$\begin{aligned} \underset{\theta,\mu_0,\mu_1}{\text{minimize}} \quad & \lambda J(\theta) \\ & + \int_\Omega (1-\theta)(I-\mu_0)^2 + \theta(I-\mu_1)^2 \, d\boldsymbol{x} \\ \text{subject to} \quad & \theta(\boldsymbol{x}) \text{ binary} \\ & 0 \le \mu_0 < \mu_1 \le 1. \end{aligned} \tag{2a}$$

Here $J(\theta)$ is the total variation of θ, $J(\theta) = \int_\Omega |\nabla\theta| \, d\boldsymbol{x}$. When θ is binary, the total variation is the length of the boundary between the two regions defined by θ. The weight $\lambda > 0$ controls how important a short boundary is. The assumption that $\mu_1 > \mu_0$ is without loss of generality and it prevents (2a) from inherently having two optima. We show how to find the optimal segmentation as well as the optimal values of the two parameters μ_0 and μ_1 by a simple branch and bound search over a single dimension.



(a) $E = 614.81$ (b) $E = 600.05$

Figure 1. Segmenting a simple image. The result shown in (a) was obtained after setting $\mu_0 = 0$, $\mu_1 = 1$ and alternating between minimizing θ and updating μ_0, μ_1. In (b), the global minimum is shown.

Related work. If we keep μ_0 and μ_1 fixed and only optimize over θ, problem (2a) becomes equivalent to

$$\underset{\theta(\boldsymbol{x}) \text{ binary}}{\text{minimize}} \quad \lambda J(\theta) + \int_\Omega \theta \left[(I-\mu_1)^2 - (I-\mu_0)^2 \right] d\boldsymbol{x} \,. \quad \text{(2b)}$$

This problem is still non-convex, as the discrete set $\{0, 1\}$ is non-convex. Chan et al. [5] showed that globally optimal solutions can still be obtained by relaxing θ to the interval $[0, 1]$, solving the resulting *convex* problem and then thresholding the result. Several algorithms have been developed to solve this convex minimization problem [2]. If the image is discretized, optimal solutions can also be obtained via graph-cuts, with a suitable J.

On the other hand, if one wants to also optimize over μ_0 and μ_1 simultaneously the problem is no longer convex. In practice, this is solved by alternating between minimizing over θ with μ_0, μ_1 fixed and minimizing μ_0, μ_1 with θ fixed [2, 5, 6]. The latter step is very simple, it just consists of taking the means of the two regions defined by θ [15]. This procedure does not guarantee that the final solution obtained is globally optimal. Indeed, Fig. 1 shows an image where this procedure fails. The result with initial values of $\mu_0 = 0$ and $\mu_1 = 1$ is shown in Fig. 1a, which is only a local optimum, because the segmentation in Fig. 1b has a lower energy.

Another method is of course to perform an exhaustive search over the parameters μ_0 and μ_1, solving (2b) for each possible pair. This is done in [8], where a maximum-flow formulation of (2b) is solved for every pair of the two levels. The size of the graphs are reduced with a method that bears some resemblance to the one described here. An alternative approach was pursued in [13] where branch and bound was applied over μ_0 and μ_1 for a discretized version of (2a). The main advantage with our approach is that we reduce the problem with one dimension, and branch-and-bound is only necessary in the remaining dimension. Another possible advantage is that our work is based on continuous optimization methods and hence metrication errors are smaller as demonstrated in [10]. Moreover, our algorithm is amenable to GPU acceleration [17].

Our work is also applicable to other variants of the Mumford-Shah family of segmentation methods. Alternating minimization is used, for example, in [9] for a discretized version of Mumford-Shah, in [7] for motion segmentation of a parametric optical flow model. In [12], applications of parametric max-flow problems are given including ratio minimization and incorporation of global constraints in the optimization. Instead of solving a series of max-flow problems, we show how the same applications can be solved via a single convex variational problem.

2. Parametric Binary Problems

For any function v, let $v^{(t)}$ denote the function thresholded at t, i.e. $v^{(t)}(\boldsymbol{x}) = 1$ if $v(\boldsymbol{x}) > t$ and 0 otherwise. From now on, we require that our smoothness function J satisfies the following requirements:

1. $J(v)$ is convex and $J(v) \geq 0$.
2. $J(tv) = tJ(v)$ for every $t > 0$.
3. $J(v) = \int_{-\infty}^{\infty} J(v^{(t)}) dt$ (general co-area formula).

For example, the total variation $\int_\Omega |\nabla v| \, d\boldsymbol{x}$ satisfies these three conditions.

We will now define two optimization problems and show that by thresholding the solution to one, we get a solution to the other. Let $f(\boldsymbol{x}, s)$ be a real-valued function such that $f(\boldsymbol{x}, \cdot)$ is continuously strictly increasing for each fixed $\boldsymbol{x} \in \Omega$ and $f(\boldsymbol{x}, z(\boldsymbol{x})) = 0$ for all $\boldsymbol{x}$ and some bounded function z. Let F be any function such that $\partial F / \partial s(\boldsymbol{x}, s) = f(\boldsymbol{x}, s)$ for all $(\boldsymbol{x}, s)$. Consider the following discrete problem:

$$\underset{\theta(\boldsymbol{x}) \text{ binary}}{\text{minimize}} \quad \lambda J(\theta) + \int_\Omega \theta(\boldsymbol{x}) f(\boldsymbol{x}, s) \, d\boldsymbol{x}. \qquad (P_s)$$

We will need the following property of the solutions to (P_s):

Lemma 2.1. *Let s be fixed. Assume $f(\boldsymbol{x}, s) > \hat{f}(\boldsymbol{x}, s)$ for all $\boldsymbol{x} \in \Omega$. Then the solutions θ and $\hat{\theta}$ of (P_s) for $f(\boldsymbol{x}, s)$ and $\hat{f}(\boldsymbol{x}, s)$, respectively, satisfy $\theta(\boldsymbol{x}) \leq \hat{\theta}(\boldsymbol{x})$ for all $\boldsymbol{x}$.*

See [4] for a proof that can be adapted to our case with just a change of notation. The corresponding convex variational problem to (P_s) is:

$$\underset{w(\boldsymbol{x}) \in \mathbf{R}}{\text{minimize}} \quad \lambda J(w) + \int_\Omega F(\boldsymbol{x}, w(\boldsymbol{x})) \, d\boldsymbol{x} \,. \qquad (Q)$$

Problems (P_s) and (Q) are related, as stated by the following theorem:

Theorem 2.2. *A function w solves (Q) if and only if $w^{(s)}$ solves (P_s) for any $s \in \mathbf{R}$.*

Proof. Define w, for every $\boldsymbol{x} \in \Omega$, as follows:

$$w(\boldsymbol{x}) = \sup \{ s \,|\, \exists \theta \text{ solving } (P_s) \text{ with } \theta(\boldsymbol{x}) = 1 \}.$$

If $f(\boldsymbol{x}, s) \leq 0$ for all $\boldsymbol{x}$ it follows that $\theta \equiv 1$ solves (P_s). Similarly, $f(\boldsymbol{x}, s) \geq 0$ implies that $\theta \equiv 0$ is a solution. From this we see that w is bounded, more precisely that $\inf_{\boldsymbol{y}} z(\boldsymbol{y}) \leq w(\boldsymbol{x}) \leq \sup_{\boldsymbol{y}} z(\boldsymbol{y})$. To see this, choose $s' < \inf_{\boldsymbol{y}} z(\boldsymbol{y})$. By definition we have $w(\boldsymbol{x}) \geq s'$ for all $\boldsymbol{x}$.

Let s be fixed. If $s < w(\boldsymbol{x})$, any solution θ of (P_s) must satisfy $\theta(\boldsymbol{x}) = 1$, while if $s > w(\boldsymbol{x})$ we must have $\theta(\boldsymbol{x}) = 0$ (Lemma 2.1). Since $w^{(s)}$ satisfies these requirements, we see that $w^{(s)}$ is a solution to (P_s). By continuity arguments [4] one can show that $w^{(s)}$ also is a solution on the set where $w(\boldsymbol{x}) = s$.

To show that w is the solution to (Q), we start out by letting $s_* < \inf_{\boldsymbol{y}} v(\boldsymbol{y})$ for some v and observing that

$$\int_{s_*}^{\infty} v^{(s)}(\boldsymbol{x}) f(\boldsymbol{x}, s)\, ds = \int_{s_*}^{v(\boldsymbol{x})} f(\boldsymbol{x}, s)\, ds = F(\boldsymbol{x}, v(\boldsymbol{x})) - F(\boldsymbol{x}, s_*).$$

Now we integrate over problem (P_s) for all s:

$$\int_{s_*}^{\infty} \left(\lambda J(v^{(s)}) + \int_{\Omega} v^{(s)}(\boldsymbol{x}) f(\boldsymbol{x}, s)\, d\boldsymbol{x} \right) ds = \lambda J(v) + \int_{\Omega} F(\boldsymbol{x}, v(\boldsymbol{x}))\, d\boldsymbol{x} - \int_{\Omega} F(\boldsymbol{x}, s_*)\, d\boldsymbol{x}\,.$$

We have already shown that $w^{(s)}$ minimizes the integrand for every s. This means that for any function v,

$$\lambda J(v) + \int_{\Omega} F(\boldsymbol{x}, v(\boldsymbol{x}))\, d\boldsymbol{x} \;\geq\; \lambda J(w) + \int_{\Omega} F(\boldsymbol{x}, w(\boldsymbol{x}))\, d\boldsymbol{x}\,.$$

This shows that w is the unique solution of the strictly convex functional in (Q). □

Remark 2.3. Problem (Q) with $F(\boldsymbol{x}, w(\boldsymbol{x})) = \frac{1}{2}(w(\boldsymbol{x}) - I(\boldsymbol{x}))^2$ is a well-studied convex functional in image restoration [19] and we will refer to it as a ROF problem.

Remark 2.4. Chambolle [4] proved Theorem 2.2 for the special case $f(\boldsymbol{x}, s) = s - G(\boldsymbol{x})$ and used the result to approximate the solution to problem (Q) with a series of discrete solutions to (P_s). We will use the result in the other direction, for solving a one-parameter family of discrete problems by thresholding a single ROF solution.

Remark 2.5. *If $f(\boldsymbol{x}, s) = H(\boldsymbol{x})s - G(\boldsymbol{x})$ with $H(\boldsymbol{x}) > 0$, then*

$$F(\boldsymbol{x}, w(\boldsymbol{x})) = \frac{1}{2}\left(\sqrt{H(\boldsymbol{x})} w(\boldsymbol{x}) - \frac{G(\boldsymbol{x})}{\sqrt{H(\boldsymbol{x})}} \right)^2. \quad (3)$$

This is the result we will use for the rest of this paper. We call Problem (Q) with this data term a weighted ROF problem.

2.1. Numerical Method

In our numerical experiments, we will use the following smoothness function:

$$J(\theta) = \sum_{i=0}^{M} \sum_{j=0}^{N} \sqrt{(\theta_{i+1,j} - \theta_{i,j})^2 + (\theta_{i,j+1} - \theta_{i,j})^2}, \quad (4)$$

which can be seen as a discretization of the "true" length of the boundary. Problem (Q) with (3) can be written as

$$\underset{w}{\text{minimize}} \qquad J(w) + \frac{1}{2\lambda} \int_{\Omega} \left(D(\boldsymbol{x}) w(\boldsymbol{x}) - B(\boldsymbol{x}) \right)^2 d\boldsymbol{x}\,, \quad (5)$$

and we solve it by adapting a method in [3, 4]. Introduce a dual field ξ, with which a solution can be found by iterating the following scheme:

$$\begin{cases} w_{i,j}^{(n)} = \left(B_{i,j} + \lambda \left(\operatorname{div} \xi^{(n)} \right)_{i,j} / D_{i,j} \right) / D_{i,j} \\ \xi_{i,j}^{(n+1)} = \dfrac{\xi_{i,j}^{(n)} + (\tau/\lambda)(\nabla w^{(n)})_{i,j}}{\max\{1, |\xi_{i,j}^{(n)} + (\tau/\lambda)(\nabla w^{(n)})_{i,j}|\}} \end{cases}, \quad (6)$$

where τ is the step-length. The initial condition can be set to $\xi^{(0)} = \mathbf{0}$ and $w^{(0)} = B$. A suitable stopping criterion when $D \equiv 1$ is also derived in [4]; if $\bar{w}$ is the true solution we have the following error bound:

$$||w^n - \bar{w}||^2 \leq \lambda J(w) - \lambda \sum_{i,j} \xi_{i,j}^n \cdot (\nabla w^n)_{i,j}\,. \quad (7)$$

We divide this error bound with the number of pixels in order to get a bound independent of the size of the image being processed.

3. Two-Phase Mumford-Shah Functional

We now return to the original problem (2a). To formulate our optimization problem, we perform the following change of variables:

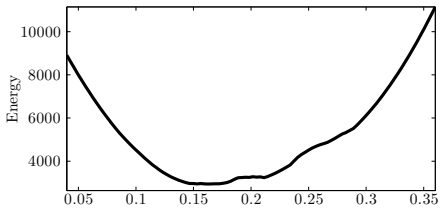
$$\begin{cases} \delta = \mu_1 - \mu_0 \\ \nu = \mu_1^2 - \mu_0^2 \end{cases} \iff \begin{cases} \mu_1 = \dfrac{\nu + \delta^2}{2\delta} \\ \mu_0 = \dfrac{\nu - \delta^2}{2\delta}\,. \end{cases} \quad (8)$$

We can now rewrite the energy in (2a) as

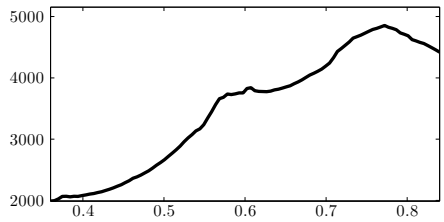
$$E(\theta, \nu, \delta) = \lambda J(\theta) + \int_{\Omega} \theta(\boldsymbol{x})(\nu - 2\delta I(\boldsymbol{x})) + \left(\frac{\nu - \delta^2}{2\delta} - I(\boldsymbol{x}) \right)^2 d\boldsymbol{x}\,. \quad (9)$$

Let the function $m(\delta)$ previously introduced in (1) denote the minimum energy possible given a fixed δ, $m(\delta) = \min_{\theta,\nu} E(\theta, \nu, \delta)$. The set of parameters (μ_0, μ_1) has two degrees of freedom and when we evaluate $m(\delta)$ we optimize over one degree of freedom while keeping the other one fixed.

(a) ROF solution (b) Thresholded



(c) $\delta = 0.2$, $\nu^* = 0.16$



(d) $\delta = 0.6$, $\nu^* = 0.36$

Figure 2. Finding the globally optimal energy for a fixed δ by thresholding the solution to problem (Q).

3.1. Optimization with Fixed Difference

The result in Theorem 2.2 implies that $m(\delta)$ can be evaluated by thresholding the solution to the real-valued problem (Q) for all ν and evaluating the energy. This is because evaluating $m(\delta)$ amounts to solving

$$\min_{\nu} \left(\min_{\theta} \; E(\theta, \nu, \delta) \right) . \tag{10}$$

The inner problem is a special case of (P_s) with $f(\boldsymbol{x}, s) = s - 2\delta I(\boldsymbol{x})$. To see this, recall that the last term of $E(\theta, \nu, \delta)$ does not depend on θ. From Remark 2.5 we see that it can be solved with (6). After the solution to (Q) is obtained, the solution is thresholded and E evaluated for all ν. To summarize, computing $m(\delta)$ consists of the following steps:

1. Solve problem (Q) with $F(\boldsymbol{x}, s) = \frac{1}{2}(s - 2\delta I(\boldsymbol{x}))^2$.
2. For each ν, threshold the solution w at ν and evaluate the resulting energy (9).
3. The pair (ν^*, θ^*) with the lowest energy is the global solution to problem (2a) with δ fixed.

Step one is a standard ROF problem, for which there exist fast minimization methods, see [16] for an overview and [17] for a GPU implementation . Our simple MATLAB implementation performed one (6)-iteration in about 27 ms for $\lambda = \delta = 0.5$. The number of iterations required until convergence is strongly dependent on λ and δ. The second step does not need as much attention as it is a very fast procedure and can trivially be parallelized.

Figure 2 shows an example where δ has been fixed to 0.2 and 0.6, respectively. The graphs show that the energy has a lot of local minima as ν varies. The thresholding process finds the global minimum quickly. It is also interesting to note that the graph of the energy looks entirely different for different δ, which suggest that the minimum energy is a complicated function with respect to (δ, ν), and therefore nontrivial to minimize.

Figure 3 shows $m(\delta)$ evaluated on the entire interval $[0, 1]$ for ten images. Note that $m(\delta)$ is often very flat around the global optimum, which has two consequences: (i) it will be difficult to find the optimum δ^* with certainty, but (ii) one evaluation of $m(0.5)$ is often enough to find a good solution, close to the global solution.

3.2. Optimization with Varying Difference

It is also possible to solve problem (2a) along another degree of freedom. We can define another function

$$\hat{m}(\nu) = \min_{\theta, \delta} E(\theta, \nu, \delta), \quad \theta(\boldsymbol{x}) \text{ binary.} \tag{11}$$

We set $s = -\delta$ and see that computing this function means solving

$$\min_{s} \left(\min_{\theta(\boldsymbol{x})} \lambda J(\theta) + \int_{\Omega} \theta(\boldsymbol{x})(2I(\boldsymbol{x})s + \nu) \, d\boldsymbol{x} \right) . \tag{12}$$

The procedure for calculating $\hat{m}(\nu)$ is the same as the one described in the previous section, with the first step replaced by:

1'. Solve problem (Q) with $F(\boldsymbol{x}, s) = \frac{1}{2}(2I(\boldsymbol{x})s + \nu)^2$.

The resulting minimization problem can be written on the form (5). Therefore, this step can be performed with the method described in Section 2.1.

Figure 4 shows an example with the "camera man" image. In the experiment ν was fixed to 0.55 and 0.75. This resulted in two very different energy curves for the same image.

3.3. Obtaining a Lower Bound

We have a method to compute $m(\delta)$; the next logical step is to minimize it. To be able to prove a lower bound, we need a way to obtain a lower bound for m on an interval $[\delta_1, \delta_2]$. A good lower bound is an essential part of the branch and bound paradigm.

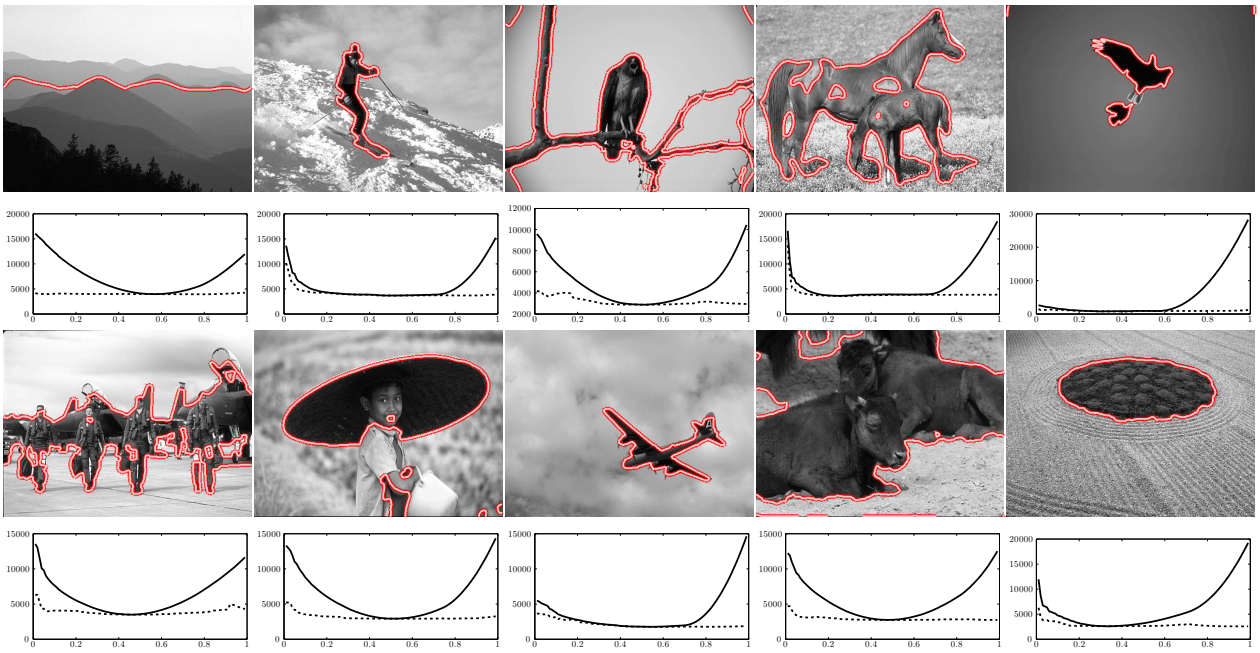
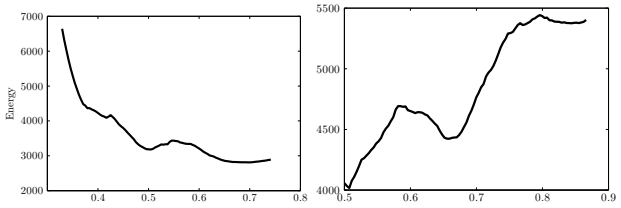


Figure 3. The function $m(\delta)$ for 10 images. Note that m is very flat near the optimum. The dashed line shows the energy after subsequent optimization of μ_0, μ_1. The weight $\lambda = 0.5$ was used for these images from [14].



(a) $\nu = 0.55$ (b) $\nu = 0.75$

Figure 4. "Camera man" image. Energies (y-axis) for problem (2a) on the curve $\mu_1^2 - \mu_0^2 = \nu$, each obtained by thresholding all possible δ (x-axis).

Finding a lower bound for $m(\delta)$ amounts to finding a lower bound for the energy $E(\theta, \nu, \delta)$ defined in (9) for any $\delta \in [\delta_1, \delta_2]$. This energy can be bounded from below on the interval by:

$$E^{\text{bound}}_{\delta_1,\delta_2}(\theta,\nu) = \lambda J(\theta) + \int_\Omega \theta(\boldsymbol{x})(\nu - 2\delta_2 I(\boldsymbol{x}))\, d\boldsymbol{x} + \min_{\delta\in[\delta_1,\delta_2]} \int_\Omega \left(\frac{\nu-\delta^2}{2\delta} - I(\boldsymbol{x})\right)^2 d\boldsymbol{x}\,. \quad (13)$$

It follows that the minimum of $E^{\text{bound}}_{\delta_1,\delta_2}$ is a lower bound to the minimum of m on $[\delta_1, \delta_2]$. The last term does not depend on θ and can be computed by choosing δ such that $\frac{\nu-\delta^2}{2\delta}$ is as close to the mean of the image as possible. Finding the lower bound therefore amounts to solving (P_s) with $f(\boldsymbol{x}, s) = s - 2\delta_2 I(\boldsymbol{x})$ for every s and computing the minimum of the resulting energies. Just like before, every solution can be obtained by thresholding the solution to (Q). Denote the obtained lower bound $m_{\text{bound}}(\delta_1, \delta_2)$.

3.4. Global Optimization

The lower bound can be used to perform a branch and bound search on the interval $[0, 1]$, splitting each subinterval until it can either be discarded or contains the optimum. However, obtaining a useful bound for even moderately large intervals is hard because m is flat (see Fig. 3). Since every calculated bound and every evaluation of m require a solution to (Q), it is essential that previous solutions can be reused. The number of (6)-iterations can then be kept to a minimum. For memory reasons, we also want to keep the number of cached solutions as low as possible.

For these reasons, we propose the following method to search for the optimum δ^*: A *feasible region* $[\delta_L, \delta_H]$ is maintained, known to contain the optimal value. This region is initially set to $[0, 1]$. The goal is to shrink the feasible region from both ends, i.e. to provide new regions $[\delta_L^{(n+1)}, \delta_H^{(n+1)}] \subset [\delta_L^{(n)}, \delta_H^{(n)}]$ containing δ^*, with the limit of the lengths equal to 0. The algorithm consists of three main steps: two for shrinking the interval from both endpoints using lower bounds and one to search the remaining feasible interval after good candidates to the optimal energy E^*. Good candidates are necessary for the bounds to be useful; fortunately, good candidates are found very quickly in practice.

The algorithm iterates three main steps, each associated with a cached dual field ξ for speeding up the (6)-iterations.

The two bounding steps also store step lengths t_L, t_H which controls the size of the interval to be removed. The steps are detailed in the following list:

1. Try to shrink the interval from above
   - Using the cached dual field ξ_H, solve problem (3) with $G(\boldsymbol{x}) = 2(\delta_H + t_H)I(\boldsymbol{x})$.
   - Evaluate $m_{\text{bound}}(\delta_H, \delta_H + t_H)$ by thresholding the solution.
   - If the bound is greater than the currently best energy, discard the interval by setting $\delta_H \leftarrow \delta_H + t_H$. Otherwise, replace t_H by a smaller step; we used $0.8t_H$.
2. Similarly, try to shrink the interval from below.
3. Choose δ inside the feasible interval from $\left\langle \frac{1}{2}, \frac{1}{4}, \frac{3}{4}, \frac{7}{8}, \frac{5}{8}, \frac{3}{8}, \frac{1}{8}, \frac{1}{16}, \ldots \right\rangle$ and evaluate $m(\delta)$.

Because the sequence of evaluated δ is dense in $[0, 1]$, m will eventually be evaluated arbitrarily close to the optimal value. We also have $m_{\text{bound}}(\delta, \delta + t) \to m(\delta)$ as $t \to 0$. From these observations, it is not hard to show that the algorithm is convergent.

3.5. Results

Because $m(\delta)$ typically is very flat (Fig. 3), the interval cannot be made very small without substantial computational effort. But an approximate localization of the global optimum can be computed and proved in reasonable time. Figure 5 shows the cumulative number of (6)-iterations required to localize the global optimum for the "camera man" image. The computation of the first bound for $m(\delta)$ required 401 iterations, while the total number of iterations required to compute the bounds for every subinterval was 1151. The search for the optimal point within the feasible interval required 302 iterations.

It should be noted that even a single evaluation of m at e.g. $\delta = 0.5$ is enough for most images in practice, due to the flatness of m and the fact that the solution will be optimal in the s-direction, which typically has lots of local minima as shown in Fig. 2. Also, after the optimal solution for a particular δ is obtained, μ_0 and μ_1 are updated before evaluating the energy. In fact, the first evaluation of $m(0.5)$ during the test in Fig. 5 resulted in a solution that could not be further improved.

4. Ratio Minimization

Problem (P_s) appears in [12] as "parametric max-flow", where it is used, among other things, to minimize a ratio of two functionals. A similar method is used in [11], where instead a sequence of convex problems is solved. We shall

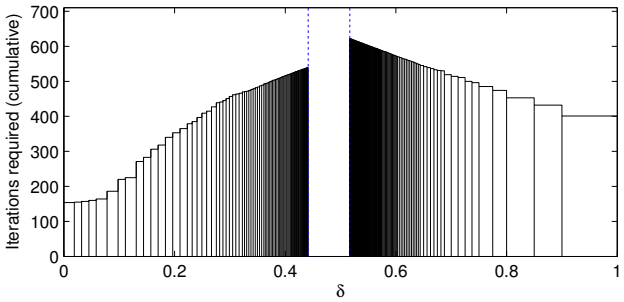


Figure 5. Iterations required to shrink the interval for the "camera man" image. A precision of 0.001 was used for the ROF problems. As the intervals grow small, the cached dual field ξ can be reused, allowing the total number of iterations to stay reasonable.

see how one can solve some problems of the same type by solving a single convex minimization problem. The ratio of two functionals P and Q with $Q(\theta) > 0$ is considered:

$$R(\theta) = \frac{P(\theta)}{Q(\theta)}. \tag{14}$$

Let $s^* = R(\theta^*)$ be the optimal value of $R(\theta)$. We see that $P(\theta^*) - s^*Q(\theta^*) = 0$ and

$$\begin{aligned} \min_\theta P(\theta) - sQ(\theta) \geq 0 \quad &\Longleftrightarrow \quad s \leq s^* \\ \min_\theta P(\theta) - sQ(\theta) \leq 0 \quad &\Longleftrightarrow \quad s \geq s^*. \end{aligned} \tag{15}$$

This means that we can solve problems for different values of s until we have come arbitrarily close to the optimal value s^* using bisections. This is done in [12] with repeated max-flow problems.

With the result in Theorem 2.2, we are able to minimize functionals of the following form:

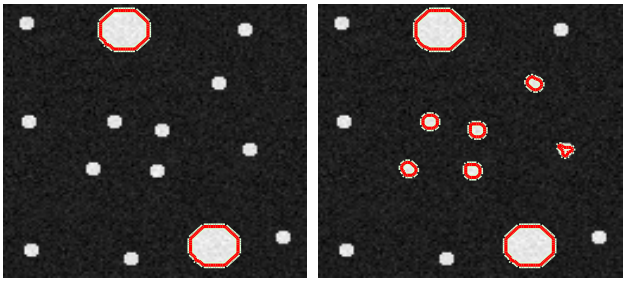
$$\frac{P(\theta)}{Q(\theta)} = \frac{\lambda J(\theta) + \int_\Omega \theta(\boldsymbol{x}) g(\boldsymbol{x})\, d\boldsymbol{x}}{\int_\Omega \theta(\boldsymbol{x}) h(\boldsymbol{x})\, d\boldsymbol{x} + K}, \quad h(x) > 0. \tag{16}$$

To compute the minimizer of $P(\theta)/Q(\theta)$, we formulate the problem of finding the minimizer of $P(\theta) - sQ(\theta)$, which amounts to solving

$$\underset{\theta(\boldsymbol{x}) \text{ binary}}{\text{minimize}} \quad \lambda J(\theta) + \int_\Omega \theta(\boldsymbol{x}) \left(h(\boldsymbol{x})(-s) + g(\boldsymbol{x}) \right). \tag{17}$$

By solving (Q) once and thresholding the result at $-s$ we find minimizers to $P(\theta) - sQ(\theta)$. Searching for s^* is now reduced to thresholding the solution w at different levels and evaluating an energy, which can be performed very fast. Define $E_s(\theta) = P(\theta) - sQ(\theta)$ and

1. Start with $s_{\text{min}}, s_{\text{max}}$
2. $s \leftarrow (s_{\text{min}} + s_{\text{max}})/2$.
3. $\theta \leftarrow w^{(-s)}$.



(a) Minimizing $P(\theta)$ (b) Minimizing $P(\theta)/Q(\theta)$

Figure 6. Minimizing a ratio of two functions. $Q(\theta) = \int_\Omega \theta(\boldsymbol{x})h(\boldsymbol{x})\,d\boldsymbol{x}$, where $h(\boldsymbol{x})$ is chosen to be larger in the center of the image domain, which makes $\theta(\boldsymbol{x}) = 1$ more favorable.

4. If $E_s(\theta) > 0$ set $s_{\text{max}} \leftarrow s$. Otherwise, set $s_{\text{min}} \leftarrow s$.

5. Repeat from step 2.

This scheme will rapidly converge to s^*. Figure 6 shows an example. More examples of ratio minimization are found in [12].

4.1. Constrained Optimization

It is interesting to note that the minimizing a ratio of two functionals bears some similarities to constrained minimization. Consider the following problem, where in addition to an energy functional, the area of the resulting foreground (where $\theta(\boldsymbol{x}) = 1$) is also required to be larger than a predetermined minimum value:

$$\begin{aligned} \underset{\theta}{\text{minimize}} \quad & E(\theta, \mu_0, \mu_1) \\ \text{subject to} \quad & \theta(\boldsymbol{x}) \text{ binary} \\ & \int_\Omega \theta(\boldsymbol{x})\,d\boldsymbol{x} \geq A\,. \end{aligned} \tag{18}$$

The dual function $d(s)$ [1] of this problem is:

$$\min_\theta \left(E(\theta, \mu_0, \mu_1) + s\left(A - \int_\Omega \theta(\boldsymbol{x})\,d\boldsymbol{x} \right) \right). \tag{19}$$

For any $s \geq 0$ we have $d(s) \leq E^*$, where E^* is the optimal energy for (18). The best lower bound is given by $d^* = \max_{s\geq 0} d(s)$, which can be computed by thresholding a solution to (Q), since computing $d(s)$ is equivalent to solving

$$\underset{\theta}{\text{minimize}} \quad E(\theta, \mu_0, \mu_1) + \int_\Omega \theta(\boldsymbol{x})\, s\,d\boldsymbol{x}\,, \tag{20}$$

followed by an evaluation of (19). However, since θ is constrained to be binary, strong duality does not generally hold [1], that is, $d^* < E^*$ in general.

5. Gaussian Distributions

Many variants of the two-phase Mumford-Shah functional have been used for image segmentation. For example, ultrasound images can be segmented using a maximum-likelihood formulation with the assumption that the image pixels are Rayleigh distributed [20]. To emphasize the difference to (2a), we will instead use a model where all pixels have equal expected values. This problem has previously been treated in [18] with local methods. Consider the following image model, were the image pixels comes from two Gaussian distributions with zero mean and different variance:

$$I(\boldsymbol{x}) \sim \begin{cases} \mathrm{N}(0, \sigma_1^2), & \theta(\boldsymbol{x}) = 1 \\ \mathrm{N}(0, \sigma_0^2), & \theta(\boldsymbol{x}) = 0\,. \end{cases} \tag{21}$$

Given that $\theta(\boldsymbol{x}) = i$, the log-likelihood for the image pixel is

$$\ell_i(I(\boldsymbol{x})) = \log\left(\frac{1}{\sigma_i\sqrt{2\pi}} \exp\left(\frac{I(\boldsymbol{x})^2}{2\sigma_i^2} \right) \right). \tag{22}$$

Given an observed image, we want to recover θ, so we want to solve the following minimization problem:

$$\begin{aligned} \underset{\theta,\sigma_0,\sigma_1}{\text{minimize}} \quad & \lambda J(\theta) + \int_\Omega \theta(\boldsymbol{x})\left[-\ell_1(I(\boldsymbol{x}))\right] \\ & + \big(1 - \theta(\boldsymbol{x})\big)\big[-\ell_0(I(\boldsymbol{x}))\big]\,d\boldsymbol{x}\,. \end{aligned} \tag{23}$$

Following the same approach as in Section 3.1, we remove the term which does not depend on θ. After rearranging the factors inside the logarithms of the functional, we obtain:

$$\lambda J(\theta) + \int_\Omega \theta(\boldsymbol{x}) \left(\log\frac{\sigma_1}{\sigma_0} + I(\boldsymbol{x})^2 \left(\frac{1}{2\sigma_1^2} - \frac{1}{2\sigma_0^2} \right) \right) d\boldsymbol{x}\,.$$

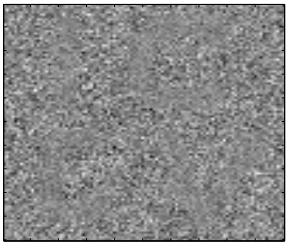
This suggests the following change of variables:

$$\begin{cases} r = \log\frac{\sigma_1}{\sigma_0} \\ t = \frac{1}{2\sigma_1^2} - \frac{1}{2\sigma_0^2} \end{cases} \iff \begin{cases} \sigma_1 = \frac{\sqrt{-2t(e^{2r}-1)}}{2t} \\ \sigma_0 = \frac{\sqrt{-2t(e^{2r}-1)}}{2te^r}\,. \end{cases}$$

We can now solve problem (23) for t = constant. First, we solve a ROF problem with $(w(\boldsymbol{x}) + I(\boldsymbol{x})^2 t)^2$ as data term. Then we threshold the solution at all possible levels r, evaluating the energy in (23) and choosing the lowest energy. A result with this segmentation model can be seen in Fig. 7.

6. Conclusion

We have shown that the two-phase, binary Mumford-Shah functional can be effectively optimized by solving a continuous problem followed by thresholding. The method works if the difference between the two levels is fixed. To solve the general case, we give a branch and bound-like



(a) Original image with $\sigma_1 = 6$ and $\sigma_0 = 10$

(b) Resulting segmentation. Recovered σ: 6.07 and 10.35

Figure 7. Segmentation of an image composed of two Gaussian distributions with zero mean.

method which is shown to perform quite efficiently. It is interesting to note that a single evaluation of $m(\delta)$, which is optimal along one dimension, often seems to be enough to find the global optimum. The two main contributions of this paper are:

1. We have defined $m(\delta)$ in (1) and shown how it can be computed efficiently.

2. This allows us to solve the problem in (2a) using branch and bound in a single dimension.

In addition, we have briefly discussed some connections to the parametric max-flow problems in [12] and optimization under constraints. We have also extended the numerical method given in [3, 4] to the weighted ROF problem (Q).

The source code used for the experiments in this paper is available at http://www.maths.lth.se/~petter

Acknowledgments

This work has been funded by the Swedish Research Council (grant no. 2007-6476), by the Swedish Foundation for Strategic Research (SSF) through the programme Future Research Leaders, and by the European Research Council (GlobalVision grant no. 209480).

References

[1] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, March 2004.

[2] X. Bresson, S. Esedoḡlu, P. Vandergheynst, J. Thiran, and S. Osher. Fast global minimization of the active contour/snake model. *Journal of Mathematical Imaging and Vision*, 28(2):151–167, June 2007.

[3] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20:89–97, 2004.

[4] A. Chambolle. Total variation minimization and a class of binary MRF models. In *EMMCVPR05*, pages 136–152, 2005.

[5] T. Chan, S. Esedoglu, and M. Nikolova. Algorithms for finding global minimizers of image segmentation and denoising models. *SIAM Journal on Applied Mathematics*, 66(5):1632–1648, 2006.

[6] T. Chan and L. Vese. Active contours without edges. In *IEEE Transactions on Image Processing*, volume 10, pages 266–277, February 2001.

[7] D. Cremers and S. Soatto. Motion competition: A variational framework for piecewise parametric motion segmentation. *Int. Journal Computer Vision*, 63(3):249–265, 2005.

[8] J. Darbon. A note on the discrete binary Mumford-Shah model. In A. Gagalowicz and W. Philips, editors, *MIRAGE*, volume 4418 of *Lecture Notes in Computer Science*, pages 283–294. Springer, 2007.

[9] L. Grady and C. Alvino. Reformulating and optimizing the Mumford-Shah functional on a graph - a faster, lower energy solution. In *European Conf. Computer Vision*, pages 248–261, Marseille, France, 2008.

[10] M. Klodt, T. Schoenemann, K. Kolev, M. Schikora, and D. Cremers. An experimental comparison of discrete and continuous shape optimization methods. In *European Conf. Computer Vision*, pages 332–345, Marseille, France, 2008.

[11] K. Kolev and D. Cremers. Continuous ratio optimization via convex relaxation with applications to multiview 3d reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Miami, Florida, 2009.

[12] V. Kolmogorov, Y. Boykov, and C. Rother. Applications of parametric maxflow in computer vision. In *ICCV*, pages 1–8. IEEE, 2007.

[13] V. Lempitsky, A. Blake, and C. Rother. Image segmentation by branch-and-mincut. In *European Conf. Computer Vision*, pages 15–29, Marseille, France, 2008.

[14] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, volume 2, pages 416–423, 2001.

[15] D. Mumford and T. Shah. Optimal approximation by piecewise smooth functions and associated variational problems. In *Comm. on Pure and Applied Mathematics*, 1989.

[16] T. Pock. *Fast Total Variation for Computer Vision*. PhD thesis, Graz University of Technology, January 2008.

[17] T. Pock, M. Unger, D. Cremers, and H. Bischof. Fast and exact solution of total variation models on the GPU. In *CVPR Workshop on Visual Computer Vision on GPUs*, 2008.

[18] M. Rousson and R. Deriche. A variational framework for active and adaptative segmentation of vector valued images. In *In Proc. IEEE Workshop on Motion and Video Computing*, pages 56–62, 2002.

[19] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[20] A. Sarti, C. Corsi, E. Mazzini, and C. Lamberti. Maximum likelihood segmentation with Rayleigh distribution of ultrasound images. *Computers in Cardiology, 2004*, pages 329–332, Sept. 2004.