Image Segmentation with One Shape Prior – A Template-Based Formulation

Siqi Chen[a], Daniel Cremers[b], Richard J. Radke[a]

[a]*Department of ECSE, Rensselaer Polytechnic Institute, Troy, NY, USA*
[b]*Department of Computer Science, Technical University of Munich, Germany*

Abstract

Image segmentation with one shape prior is an important problem in computer vision. Most algorithms not only share a similar energy definition, but also follow a similar optimization strategy. Therefore, they all suffer from the same drawbacks in practice such as slow convergence and difficult-to-tune parameters. In this paper, by reformulating the energy cost function, we establish an important connection between shape-prior based image segmentation with intensity-based image registration. This connection enables us to combine advanced shape and intensity modeling techniques from segmentation society with efficient optimization techniques from registration society. Compare with the traditional regularization-based approach, our framework is more systematic and more efficient, able to converge in a matter of seconds. We also show that user interaction (such as strokes and bounding boxes) can easily be incorporated into our algorithm if desired. Through challenging image segmentation experiments, we demonstrate the improved performance of our algorithm compared to other proposed approaches.

Keywords: Image segmentation, shape prior.

1. Introduction

Energy minimization is widely considered to be an efficient approach to the image segmentation problem. However, intensity-based data energy terms alone are typically inadequate, and therefore regularization terms are very important in order to obtain the desired result. Typical regularization terms are based on length and curvature, which tend to be weak and sometimes cannot describe the underlying segmentation problem. In many applications, we may have *a priori* knowledge about the shape of the object we are looking for, and incorporating such shape prior knowledge into the segmentation problem can drastically improve the segmentation performance. Here we focus on the case when a single shape prior template is available, an area of recent research interest [4, 11, 13, 26, 19, 29].

Most regularization-based algorithms for segmentation with a single shape prior follow a similar strategy, and they generally suffer from similar drawbacks when applied in practice(see Section 2). In this paper, we reformulate the regularization-based single-shape-prior segmentation problem and develop a template-based approach to produce a segmentation. The contributions of our new template-based reformulation are:

1. To the best of our knowledge, we are the first to connect single shape-prior based image segmentation to intensity-based image registration, by optimizing only over the transformation space of the shape template.
2. We show the intensity model used in our work can take more general forms. For example, it can be either predefined or estimated from the user input, such as strokes or bounding box. It can also be refined iteratively from the intermediate segmentation result.
3. We propose an efficient and systematic approach to optimize the energy function. Our approach first rapidly computes the optimal global similarity transformation with an efficient frequency-domain algorithm. If desired, we can further refine the segmentation by locally deform the shape template to minimize the cost function. With our current unoptimized C++ code, our algorithm converges within a few seconds on an ordinary computer. In contrast, iterative regularization-based methods are typically much slower.

This paper is organized as follows. In Section 2, we review representative regularization-based approaches for the problem, and show how they can all be formulated in the same framework. We then point out the unavoidable drawbacks of this framework that motivate us to rethink the underlying formulation. In Section 3, we describe our new template-based formulation along with its advantages over the standard regularization-based formulation. We also make the connection between this new formulation with the intensity-based image registration problem. Due to this new formulation, we are able to design an extremely efficient and powerful approach described in Section 4. Section 5 discusses different kinds of intensity models that can be incorporated into our framework. We then show the experimental results in Section 6 and conclude in Section 7.

2. Regularization-based formulations

For the problem of image segmentation with one shape prior, most previous work defines an energy function in the following

Email addresses: siqichensc@gmail.com (Siqi Chen), daniel.cremers@in.tum.de (Daniel Cremers), rjradke@ecse.rpi.edu (Richard J. Radke)

way:

$$E(\mathbf{C},T) = E_{data}(\mathbf{C},I) + \lambda \cdot D(\mathbf{C},T(\mathbf{C}_{ref})) \quad (1)$$

where I is an input image $I:\Omega\rightarrow\mathbb{R}$, $\mathbf{C}$ is a shape representation, T is a shape transformation and D is a shape distance measure. The most common shape representations $\mathbf{C}$ used in the literature include parametric shape representations: $[0,l(\mathbf{C})]\rightarrow\mathbb{R}$ [16, 26], signed distance functions (SDFs) $\phi:\Omega\rightarrow\mathbb{R}$ [6, 10, 4], binary characteristic functions $u:\Omega\rightarrow\{0,1\}$ [11, 13], or very recent work on relaxed characteristic functions $u:\Omega\rightarrow[0,1]$ [19, 29]. Typical transformations T discussed in the shape-prior segmentation is limited to parametric global transformation, include rigid, similarity or more general projective transformations [21]. Various shape distance measures D have been proposed; Cremers et al. [8] gives a quick review of different shape distance measures when the shape is represented as a SDF. As for the data term E_{data}, most approaches use a region-based Maximum Likelihood (ML) model [31] or a boundary-based gradient model [3]. It is also possible to combine these two models together.

When the shape is represented parametrically, as in [16, 26], there are many shape descriptors that are invariant to certain geometric transformations T. For example, curvature (either integral or differential) is an invariant shape measure with respect to rigid transformations [16], and tangent angle is an invariant shape measure with respect to translations [26]. However, the shape distances D based on these shape descriptors are not usually invariant to these geometric transformations, since correspondence between the shapes affects these distances. Therefore, instead of optimizing over the geometric transformations T, a shape correspondence or a diffeomorphism $m:[0,l(\mathbf{C})]\rightarrow[0,l(\mathbf{C}_{ref})]$ is optimized instead where $l(\cdot)$ is the Euclidean length.

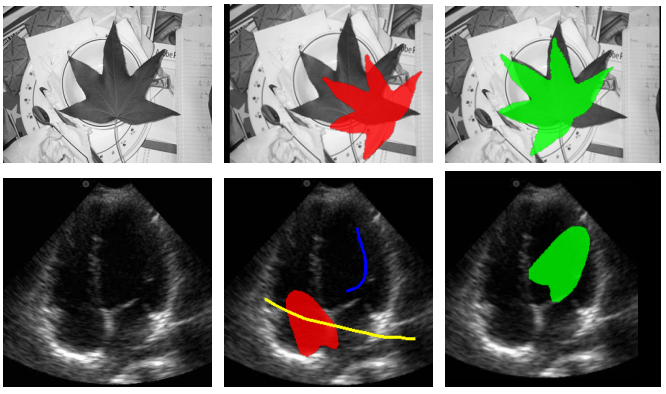
These different approaches not only share similar energy functions, but also follow similar optimization strategies. The typical optimization strategy employed is the following alternating procedure:

1. Fix T and update $\mathbf{C}$. This becomes a standard image segmentation problem and common optimization methods include level-set-based gradient descent [6, 10, 4, 16, 5], discrete graph cuts [11, 13] or the recent convex continuous cut method [19, 29]. The drawback of level-set gradient-based methods include numerical issues, slow convergence and local optimality at convergence. Discrete and continuous cut methods are both more efficient and more powerful, so that a global optimum can be recovered. However, only a specific class of energies can be optimized.
2. Fix $\mathbf{C}$ and update T. This is typically solved by gradient-based optimization methods such as gradient descent on the parameters of the transformation.
3. Go to Step 1 and iterate until convergence.

An exceptional approach that differs from this framework was proposed by Schoenemann and Cremers [26], which solves both steps together. They showed that a global optimum can be obtained when the transformation is restricted to translation and the shape is represented parametrically. Rotation can be incorporated in an exhaustive search manner. However, the computational complexity is extremely high even in the 2D rigid transformation case, which makes the approach quite difficult to be applied to more general similarity transformations and 3D applications.

Due to the local optimality of the second step, the overall energy minimization cannot guarantee global optimality. Also, the alternating strategy of optimizing rigid pose and shape can be slow to converge. Other than the inherent local optimality and slow convergence issues, we observe in practice that the regularization-based framework often needs careful tuning (e.g., gradient descent step size, order of parameter update), which makes the approach difficult to be applied in a general setting. This is also pointed out by other researchers [8, 10].

These shortcomings limit the usage of shape-prior-based segmentation approaches in some applications such as medical imaging, where there is an urgent need to get fast, robust, automatic, and ideally optimal solutions. To better illustrate these shortcomings, Figure 1 illustrates segmentation results of leaf and ultrasound cardiac test images using the recently proposed method of continuous cuts combined with gradient descent on Lie Groups [19], with a large λ. The algorithm converges to a local optimum in approximately 30 seconds with 100 iterations in both cases. In contrast, our method is able to converge to a near-global optimum in as little as 3 seconds as illustrated in Section 6. Since these drawbacks are inherent in the regularization-based formulation, the main objective of our work is to reformulate the energy function (1) in order to avoid these drawbacks.



(a) (b) (c)

Figure 1: Segmentation results for two test images with the method proposed in [19], with a large λ. (a) Input image. (b) Initial position of shape template (in red) overlaid on the original image. (c) The segmentation result (in green). User strokes are incorporated similarly to our method, as discussed in Section 5.1. This figure is best viewed in color.

We note that there is also much work on image segmentation with multiple shape priors [27, 8]. The typical approach is to first build a shape distribution model from training shapes and then fit this model to the image to estimate the segmentation. We do not intend to compare these two problems in this work since they each have their own application area. However, we believe that the problem of segmentation with one shape

prior is related to the segmentation with multiple shape priors. For example, by assuming that the underlying distribution of the training shapes is a Gaussian distribution, the single shape prior studied in our work can be obtained from the training shapes as the mean shape [6, 23]. We also show a preliminary example of a straightforward (though not very efficient) extension of our algorithm to the case of multiple shape priors in Section 6.

3. The template-based formulation

In this paper, we choose the binary characteristic partition $u : \Omega \to \{0,1\}$ as the shape representation for its simplicity. The transformation imposed on the shape template is denoted as T. The exact form of the transformation (similarity, rigid or deformable) will be discussed later in Section 4. We follow the typical framework of using a Maximum Likelihood model as the data energy term:

$$\begin{aligned} E_{data}(u) &= -\int_\Omega \log P_{in} \cdot u \, d\mathbf{x} - \int_\Omega \log P_{out} \cdot (1-u) \, d\mathbf{x} \\ &= \int_\Omega (\log P_{out} - \log P_{in}) \cdot u \, d\mathbf{x} + \text{constant} \\ &= \int_\Omega Q \cdot u \, d\mathbf{x} + \text{constant} \end{aligned} \tag{2}$$

where $P_{in}(\mathbf{x})$ and $P_{out}(\mathbf{x})$ are the probabilities that pixel $\mathbf{x}$ belongs to the object and the background respectively. The exact form of $P_{in}(\mathbf{x})$ and $P_{out}(\mathbf{x})$ can be very general, as we discuss later in Section 5. The log-likelihood map $Q(x)$ describes our confidence that a certain pixel belongs to the object or to the background. The lower $Q(x)$ is, the more likely pixel x belongs to the object, and vice versa. A standard approach is to use parametric distribution, i.e., $P_{in}(\mathbf{x}) = P(\mathbf{x}|\theta_{in})$ and $P_{out}(\mathbf{x}) = P(\mathbf{x}|\theta_{out})$ where θ_{in} and θ_{out} denote the parameters of the intensity distributions, such as Gaussian and Laplace distribution. For example, when both P_{in} and P_{out} are Gaussian distributions with mean intensities M_{in} and M_{out} and the same variance, then

$$\begin{aligned} Q(\mathbf{x}) &= \log\left(e^{\frac{-(I(\mathbf{x})-M_{In})^2}{2\sigma^2}}\right) - \log\left(e^{\frac{-(I(\mathbf{x})-M_{Out})^2}{2\sigma^2}}\right) \\ &= (I(\mathbf{x}) - M_{out})^2 - (I(\mathbf{x}) - M_{in})^2 \end{aligned} \tag{3}$$

Instead of composing this data term with a λ-balanced regularization term as in (1), the reference shape u_{ref} serves as a template in our formulation. That is, we implicitly constrain that $u = T(u_{ref})$. Therefore, we end up with the following energy function:

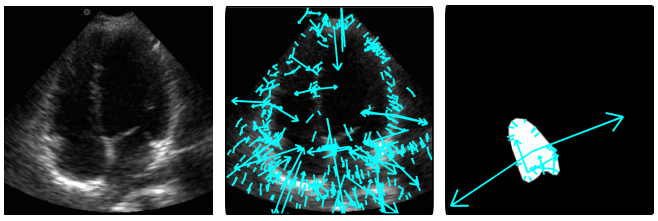
$$E(u) = E_{data}(T(u_{ref})) = \int_\Omega Q \cdot u_{ref}(T(\mathbf{x})) \, d\mathbf{x} \tag{4}$$

The image segmentation problem thus corresponds to minimizing (4) only over the geometric transformation parameters T. This template-based formulation seems at first sight to be a trivial improvement; however, it offers many advantages compared with the traditional regularization-based formulation (1). One obvious advantage is that there is no balance parameter λ, which is also equivalent to setting λ to infinity[1]. One might argue that by setting λ to infinity is too restrictive since it will over emphasize the shape regularization term. However, as we will discuss in Section 4, by allowing the transformation space to include deformable transformation, this limitation is largely avoided. A more important implication of this framework is that it shows clearly the connection between shape-prior based image segmentation and intensity-based image registration as we will discuss next.

3.1. Connection to the registration problem

This template-based reformulation of image segmentation with one shape prior is clearly related to image registration. That is, the segmentation problem is equivalent to registering the binary shape template u_{ref} and the image log-likelihood map Q when the registration metric is standard correlation. The image registration problem has been heavily studied, and many algorithms have been proposed so far. We refer the readers to the survey [32] for a complete review.

Registration methods based on features such as SIFT [15, 1] are very successful in many applications when distinctive features of the same object in different scenes can be detected. However, it is not straightforward to apply these techniques in segmentation problems. Fig. 2 illustrates the SIFT features ([15]) detected on an ultrasound cardiac test image and a clean shape template. Due to the cluttered background and noise, the SIFT algorithm fails to match any of the features between them. Furthermore, feature-based registration methods do not provide any optimality guarantee regarding the estimated transformation.



(a) (b) (c)

Figure 2: The SIFT features (b) of the image (a) and the shape template (c). The SIFT algorithm fails to match any of the features between them.

Another major category of registration is intensity-based registration. To efficiently optimize (4), most intensity-based registration algorithms tend to apply various forms of gradient-based strategies (e.g., gradient-descent or Levenberg-Marquardt) to a cost metric. This is a powerful approach when the transformation is non-rigid or deformable, as we discuss later in Section 4.3. However, when the transformation is limited to the global similarity transform, the registration cost function is non-convex with respect to the similarity transformation parameters (translation, rotation and scaling). Therefore, it is difficult to efficiently locate the optimum solution with gradient-based local optimization techniques. Fig. 3 illustrates an attempt to

[1] To the best of our knowledge, this case has not been studied before.

segment an image based on registering the log-likelihood map to the binary shape template with the correlation metric under similarity transform. Using gradient descent optimization with a multi-resolution framework (3 levels), the algorithm converges to a local optimum in approximately 15 seconds. In contrast, our method is able to achieve a near-global optimum in 3 seconds (see Section 6).

(a) (b) (c)

Figure 3: The segmentation result of the leaf image with multi-resolution gradient descent method. (b) Shape template. (c) The segmentation result in green.

A less popular registration method is frequency-domain approaches, or template matching [20, 30, 14]. Although it has certain drawbacks when applied to the image registration application [32], it is an ideal approach to our similarity transformation estimation step as we will discuss next. However, our segmentation framework differs from standard frequency-domain registration approaches in several aspects.

1. Standard correlation metric has many disadvantages in image registration application. For example, it is not invariant to changes in image intensities such as those caused by changing lighting conditions across the image sequence. Therefore, normalized cross-correlation is typically used in the template matching [14]. However, in our application, correlation is the correct registration metric, derived directly from the Maximum Likelihood model (2).
2. While in typical registration application, two images are considered to be roughly in the same intensity range, in our segmentation application, this is no longer true. The shape template image is binary image $u : \Omega \to \{0,1\}$, while the log-likelihood map Q ranges from $-\infty$ to ∞. The different intensity range makes some algorithms, such as Fourier-Mellin transfrorm [20], not directly applicable to our segmentation problem.

4. Efficient optimization

Inspired by frequency-domain image registration approaches [20, 30], we will first develop an efficient algorithm to search for the globally optimum solution when the only transformation T is a translation vector $\mathbf{t}$. We then discuss how to incorporate the rotation angle r and scaling factor s. We also show how deformable registration techniques can also be applied to our segmentation framework, which greatly reduces the dependence on an accurate reference shape model. We assume all the transformation parameters ($\mathbf{t}$, r and s) are bounded, i.e., $\mathbf{t} \in [-N,N] \times [-N,N]$, $r \in [-\pi,\pi]$ and $s \in [0,N]$ by assuming that I is an $N \times N$ image.

4.1. Translation only

Let's first consider the simplest case where the transformation is a translation $\mathbf{t}$. Then the discretized energy model (4) can be simply written as:

$$E(\mathbf{t}) = \sum_{\mathbf{x}} Q(\mathbf{x}) \cdot u_{ref}(\mathbf{x}-\mathbf{t}) \tag{5}$$

and the segmentation problem is stated as finding the best translation vector $\mathbf{t}$ such that the inner product of the shifted u and the log-likelihood map Q is the maximum. As we mentioned, a gradient-based method will give a local optimum. Since $\mathbf{t}$ is bounded, exhaustive search will guarantee the global optimum; the question is how to make it efficient.

Since the correlation metric used in (5) has a dual in the Fourier domain, we can apply an extremely efficient search method based on the Fast Fourier Transform (FFT) [7], inspired by frequency-domain image registration approaches [20]. From the basic facts of signal processing, the Fourier shift theorem shows that (5) can also be written as:

$$E(\mathbf{t}) = \mathscr{F}^{-1}(\mathscr{F}_Q \cdot \mathscr{F}^*_{u_{ref}}) \tag{6}$$

where $\mathscr{F}$, $\mathscr{F}^{-1}$ and $\mathscr{F}^*$ denote the Fourier transform, the inverse Fourier transform and the complex conjugate of the Fourier transform respectively.

We note that the Fourier shift theorem requires a circular shift instead of the linear shift commonly encountered in registration. Since the shape template u consists of a flat background, and the ground truth object sits within the image domain with all detail situated away from the edges, a linear shift will be equivalent to a circular shift, and the above derivation will hold exactly.

The benefit of converting from the spatial domain to the frequency domain is obvious. The computational complexity of the exhaustive evaluation drops from $O(N^4)$ in the spatial domain to $O(N^2 \log N)$ in the frequency domain, thanks to the efficient FFT [7]. The reduced complexity makes the algorithm extremely fast, requiring only a fraction of a second (including both FFT and finding the extrema) on a 1000x1000 image to obtain the global optimum. One possible further speed up is to use a coarse-to-fine search strategy, where a coarse translation space is first evaluated and the shape template is positioned to the best location. This can be accomplished by subsampling both the shape template and the likelihood map and apply the Fourier shift theorem as in (6). Then a finer search strategy is applied around the current shape template position. We observe in practice that this corase-to-fine strategy is slightly faster.

Fig. 4 illustrates one experiment with Fig. 4a showing the noisy image I that we want to segment. The log-likelihood map Q is obtained from I with a two-phase parametric Gaussian model with mean intensities 0 and 255 and the same variance as in 3. Fig. 4b shows the shape prior template u, which is translated from the ground truth. Fig. 4c shows the corresponding $E(\mathbf{t})$ where the white spot clearly indicates the global minimum. The optimal segmentation result is illustrated in Fig. 4d.

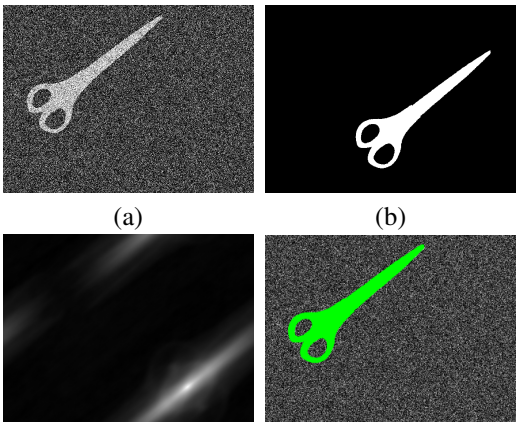


Figure 4: Segmentation result with translation only. (a) The input image I. (b) The prior shape template u_{ref}. (c) The corresponding energy map $E(\mathbf{t})$. (d) The final segmentation result in green. This figure is best viewed in color.

4.2. Rotation and scaling

The FFT-based algorithm can guarantee the global optimum of (4) when only translation is considered. However, it is not straightforward to extend this method to include rotation and scaling. We first discuss when the translation is known and the only transformation considered is a rotation and uniform scaling. Inspired by the work of image registration using the log-polar transform [30], we develop a similar algorithm appropriate for our image segmentation purpose.

Consider the log-polar coordinate system (a,b), where a denotes log radial distance from the center and b denotes angle. We assume that the center is pixel $(0,0)$. Therefore, any point (x,y) in Cartesian space can be represented in log-polar coordinates:

$$a = \log\sqrt{x^2+y^2}, \quad b = \tan^{-1}\frac{y}{x} \tag{7}$$

It is easy to show that isotropic scaling (sx,sy) in the Cartesian space transforms to a linear shift of the a axis by $\log s$ in the log-polar space. Similarly, the rotation $(x\cos r + y\sin r, -x\sin r + y\cos r)$ in the Cartesian space transforms to a circular shift of the b axis by r in the log-polar space. Therefore, the rotation and scaling transformations map to simple translations in the log-polar space and we can employ a similar FFT-based technique to recover the scale s and rotation r. Due to the linear shift of the scale in the log-polar space, the Fourier shift theorem will not hold exactly. In such cases, a window filter function (like a Hamming window) along the a axis should be employed before the Fourier transform to reduce edge effects, and this may reduce the accuracy of the scale estimation to be near-optimal.[2] This implies that when the center of the transformation is fixed, the rotation and scale parameters can be optimized near-globally.

Fig. 5 illustrates the segmentation result when only rotation and scaling are considered. We also show the log-polar image of the log-likelihood map Q and the shape template u_{ref} in Fig. 5c,d. The complexity for recovering scaling and rotation is the same as the translation-only case, which is $O(N^2 \log N)$, if the log-polar image is also $N \times N$.

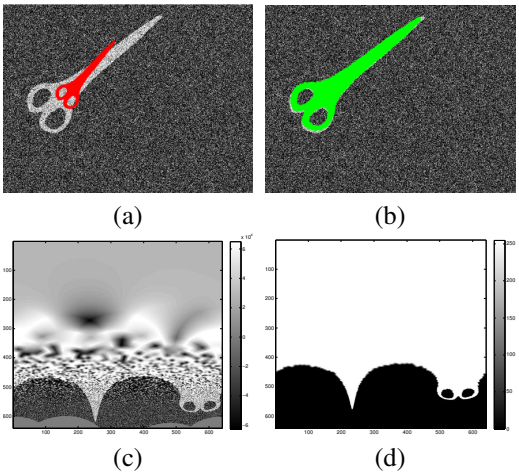


Figure 5: Segmentation result with rotation and scaling only. (a) The image I and the shape template in red. (b) The segmentation result in green. (c) The log-polar image of Q. (d) The log-polar image of u. This figure is best viewed in color.

4.3. Deformable transformation

So far the transformation is restricted to the global similarity transformation where the final segmentation result is simply the translated, rotated and the scaled version of the shape reference template. Although fast and efficient, it is clearly limited in practice, where the target segmentation typically has certain degree of deformable transformation compared to the reference shape. Fortunately, by reformulating our segmentation problem we are able to apply deformable registration techniques to our segmentation application.

The nature of the local deformation of the shape can vary significantly, therefore difficult to describe via global parametrized transformations. Instead, we have chosen an free-form deformation (FFD) model, based on B-Splines, to deform the shape template by manipulating the underlying mesh of control points. The cubic B-splines function is used in this paper, since it has a local support and guarantees C^1 continuity at control points and C^2 continuity everywhere else. The registration is solved by minimizing the correlation metric (4), with respect to the control lattice deformation, and the rest sample points are interpolated from the control points using cubic B-splines. We briefly outline the main components in our algorithm, and refer the interested readers to many excellent work on FFD registration, such as [24, 17].

The control points of the underlying mesh domain $X \times Y$ over the original shape template domain $N \times N$ is defined as,

$$P = (P^x_{m,n}, P^y_{m,n}); (m,n) \in [1,X] \times [1,Y], (x,y) \in [1,N] \times [1,N]$$

thus the deformed position of any pixel $\mathbf{x} = (x,y)$ is defined by a tensor product of cubic B-splines:

$$T(\mathbf{x},P) = \sum_{k=0}^{3}\sum_{l=0}^{3} B_k(u)B_l(v)P_{i+k,j+l} \tag{8}$$

[2] However, we did not observe this frequently in our experiments.

where $i = \lfloor \frac{x}{X} \rfloor - 1$, $j = \lfloor \frac{y}{Y} \rfloor - 1$, $u = \frac{x}{X} - \lfloor \frac{x}{X} \rfloor$, $v = \frac{y}{Y} - \lfloor \frac{y}{Y} \rfloor$.

$P_{i+k,j+l}$ are the coordinates of the sixteen controls points in the neighborhood of pixel $\mathbf{x}$ where $(k,l) \in [0,3] \times [0,3]$, and $B_k(u)$ represents the k^{th} basis function of cubic B-Splines:

$$
\begin{aligned}
B_0(u) &= \frac{(1-u)^3}{6} \\
B_1(u) &= \frac{(3u^3 - 6u^2 + 4)}{6} \\
B_2(u) &= \frac{(-3u^3 + 3u^2 + 3u + 1)}{6} \\
B_3(u) &= \frac{u^3}{6}
\end{aligned} \tag{9}
$$

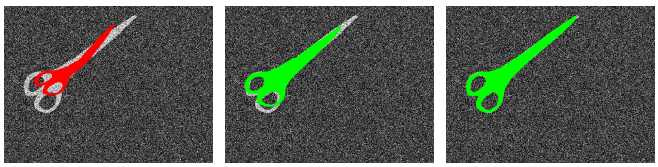
The control points P are the parameters of the B-spline FFD and the resolution of the control points mesh determines the degree of the deformable deformation. A fine resolution of the mesh enables the modeling of highly local nonrigid deformations, but with the cost of higher computational complexity. To find the optimal transformation, we iteratively minimize the following cost function (4) with respect to the control points P:

$$
E(P) = \int_{\Omega} Q \cdot u_{ref}(T(\mathbf{x}, P))\, d\mathbf{x}
$$

where $T(\mathbf{x}, P)$ is defined in (8). The iterative algorithm can then be summarized in Algorithm 1. To update the control points, we use the limited-memory Broyden, Fletcher, Goldfarb and Shanno (LBFGS) algorithm, and the control point mesh grid spacing is 8×8.

Algorithm 1 Deformable transformation estimation

Input: Shape prior template and the Image I.
Pre-computation: Compute the log-likelihood Q from I.
Initialize the control points P
while Not converged **do**
1. Calculate the gradient vector, $\nabla E = \frac{\partial E(P)}{\partial P}$
2. Update control points P
end while
B-spline interpolation to generate deformed shape template.



(a) (b) (c)

Figure 6: Deformable transformation estimation. (a) The input image I and the prior shape template in red. (b) The intermediate segmentation result after 20 iterations. (c) The converged segmentation result. This figure is best viewed in color.

Fig. 6 illustrates one deformable transformation estimation example. The input shape template in red is overlaid with the given image in Fig. 6a. The green shape in Fig. 6b is the intermediate result in the iterative update process. The final converged segmentation result is illustrated in Fig. 6c. The whole process takes 50 iterations, around 20 seconds to converge, which is slower than global similarity transformation estimation.

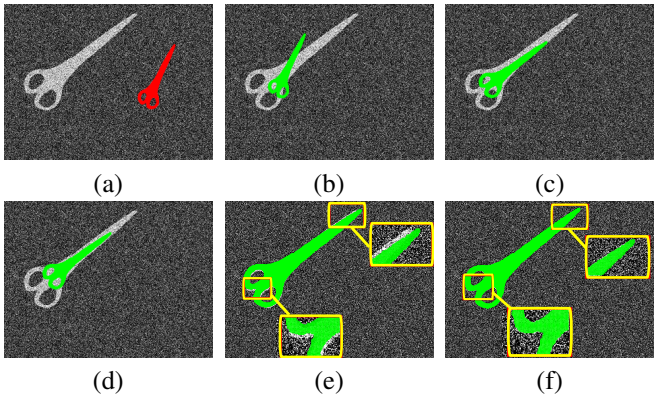
4.4. The full transformation estimation algorithm

In real-world applications, it is important to consider a full transformation space including both similarity transformation ($\mathbf{t}$,r,s) and deformable transformation. Joint optimizing all of them is intractable. Instead, we believe that before applying a higher-order transformation such as deformation, the accurate and efficient estimation of the simpler similarity transformation is critical. It serves as both a near-optimal solution and an accurate and fast initialization for further deformation estimation. This issue is similar to the challenge of appropriately separating transformation and deformation parameters.

Therefore, in the full algorithm, we first alternately update the translation $\mathbf{t}$ and the transformation pair (s, r), as described in Algorithm 2. When the similarity transformation estimation converges to the optimal ($\mathbf{t}$,r,s), we apply the deformation estimation as discussed in Secton 4.3. The input shape template u_{ref} is also the initial shape estimate u_0.

Algorithm 2 Shape prior image segmentation algorithm

Input:Shape prior template u_0 and the image I.
Pre-computation:Compute the log-likelihood Q from I.
while Not converged **do**
1.Compute translation $\mathbf{t}$ from u_i and Q. (Section 4.1)
2.Update u_i with $\mathbf{t}$ to $u_{i\mathbf{t}}$
3.Compute s and r from $u_{i\mathbf{t}}$ and Q. (Section 4.2).
4.Update $u_{i\mathbf{t}}$ with s and r to u_{i+1}
end while
Deformable transformation estimation. (Section 4.3).



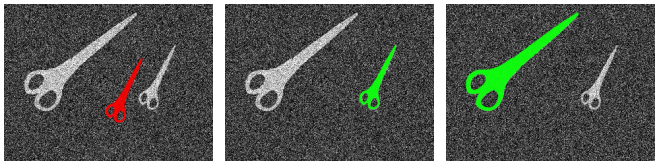
(a) (b) (c)
(d) (e) (f)

Figure 7: Full similarity transformation result. (a) Initialization u_0. (b) u_{0t}. (c) u_1. (d) u_{1t}. (e) u_2. (f) u_3 (final result). This figure is best viewed in color.

Fig. 7 illustrates an experiment of the full similarity transformation estimation without deformation estimation. Fig. 7b-c show the results after the translation and rotation/scaling estimation in the first iteration. Fig. 7d-e show the results after the translation and rotation/scaling estimation in the second iteration. Finally, Fig. 7f shows the result after the third iteration, which obtains the global optimum. The whole segmentation (with image size 640x480) takes 3 seconds on an ordinary

computer, and further speed-up is likely by optimizing our C++ code. Zoom-in results show the accuracy our algorithm can achieve.

By neglecting the window-filtering effect, each individual update of $\mathbf{t}$ and (r,s) can obtain the global optimum. However, the overall similarity transformation estimation can not guarantee global optimum for all four parameters. For example, Fig. 8 shows one such experiment. A noise-corrupted image with two scissors (one big and one small) is provided in Fig. 8a. Due to the unnormalized ML energy model of Eq. 4, it is easy to check that the energy of the big scissor segmentation of Fig. 8c is higher than the energy of the small scissor segmentation of Fig. 8b. However, given the initial shape template in Fig. 8a, our algorithm can only converge to the local optimal result in Fig. 8b, while Fig. 8c illustrates the true global optimum.

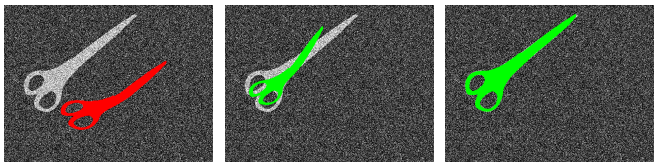


(a) (b) (c)

Figure 8: Locally optimal result. (a) I and u. (b) The segmentation result, a local optimum. (c) The global optimum.

It is straightforward to show that although global optimality is not guaranteed, our similarity transformation estimation leads to the concept of "partial optimality", introduced in [28, 12]. The partial optimum (x^*, y^*) of a function $f(x,y)$ is defined with respect to its whole x-section and y-section at (x^*, y^*), while a local optimum is defined with respect to only a small neighborhood at (x^*, y^*). Wendell and Hurter [28] showed that a partial optimum solution is "almost always" a local optimal solution, but not vice versa. However, rare counter-examples do exist and we refer the reader to [28] for a detailed discussion. Although we can not conclude theoretically that partial optimality is a stronger condition than local optimality due to these counter-examples, we observe in practice that partial optimum has a better chance of achieving global optimum. We could also incorporate multi-start [12] into our approach. This simply amounts to running the algorithm with different initializations, since it has higher chances of obtaining a global optimum result as illustrated in Fig. 10a–b. This practical performance is very desirable in our algorithm since the output of the similarity transformation estimation is used as the input for the final deformation refinement. Therefore, the better the similarity estimation is, the better the final deformation estimation will be.

Fig. 9 illustrates a complete experiment where similarity transformation and deformable transformation are both considered, as in Algorithm 2. Fig. 9a shows the input image and the initial reference shape template. Fig. 9b shows the converged result of the similarity transformation estimation after 4 iterations. It also serves as the input to the deformable transformation estimation process, where the final result is shown in Fig. 9c.



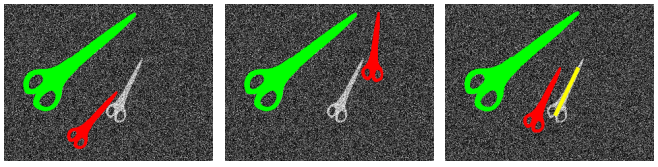
(a) (b) (c)

Figure 9: Full transformation result. (a) Initialization u_0. (b) Converged result of the similarity estimation. (c)Final segmentation result after deformation estimation. This figure is best viewed in color.

We note here that the Fourier spectrum itself is translation invariant and its conversion to the log-polar domain maps the rotation and scaling to simple translation. Therefore, a non-iterative algorithm can be designed to first estimate the rotation and scaling using the corresponding spectrum of the original images after proper filtering to remove edge effects as discussed in Section 4.2. Then translation is estimated with another FFT after the original images are compensated with the estimated rotation and scaling factors. This framework is very popular in the image registration literature and is known as the Fourier-Mellin transfrorm [20]. However, it is not directly applicable to our image segmentation problem. As we mentioned in Section 3.1, in image registration applications, the intensity distributions of two images are similar to each other; therefore their corresponding spectra are also similar. However, in our segmentation application, with one binary 0-1 image (shape template) and another float image ranging from $-\infty$ to ∞, the Fourier spectra are drastically different, which makes the phase-correlation method in the spectral domain not applicable.

5. Intensity modeling

Up to this point, the log-likelihood map Q was simply estimated as the standard two-phase Gaussian model given in (3) and fixed during the iteration. However, the intensity model of the foreground $P_{in}(\mathbf{x})$, background $P_{out}(\mathbf{x})$ and hence the log-likelihood map $Q(\mathbf{x})$ can take many other general forms.



(a) (b) (c)

Figure 10: (a)-(b) Two initializations (in red) and segmentation results (in green) of Fig. 8a. Global optimality is achieved in both cases. (c) User provided background stroke (in yellow) and global optimum result. This figure is best viewed in color.

5.1. Interactive segmentation

Interactive image segmentation is very popular and useful as demonstrated recently [2, 22]. User-provided strokes and bounding box not only serve as interactive hard constraints that the segmentation is required to satisfy, but also provided a method

to estimate the intensity distributions of both the object and the background.

Assume that $\mathbf{O}$ and $\mathbf{B}$ denote the subset of pixels *a priori* known to be a part of "object" and "background" in the image, which can be obtained from user input. Naturally, the subsets satisfy $\mathbf{O} \cap \mathbf{B} = \emptyset$. In the case of bounding box, the region outside the bounding box is defined to be the background region $\mathbf{B}$, and $\mathbf{O} = \emptyset$. The log-likelihood map Q can then be defined as:

$$Q(\mathbf{x}) = \begin{cases} R_1, & \text{if } \mathbf{x} \in \mathbf{B} \\ R_2, & \text{if } \mathbf{x} \in \mathbf{O} \\ log(P(\mathbf{x}|\text{“bkg”})) - log(P(\mathbf{x}|\text{“obj”})) & \text{if } \mathbf{x} \notin \mathbf{O} \cup \mathbf{B} \end{cases}$$

where $R_1 = 1 + \max_{\mathbf{x} \notin \mathbf{O} \cup \mathbf{B}} Q(\mathbf{x})$, $R_2 = -1 - \min_{\mathbf{x} \notin \mathbf{O} \cup \mathbf{B}} Q(\mathbf{x})$. $P(\mathbf{x}|\text{“bkg”})$ and $P(\mathbf{x}|\text{“obj”})$ are histograms of the pixels $\mathbf{x}$ in $\mathbf{O}$ and $\mathbf{B}$ respectively. If either $\mathbf{O}$ or $\mathbf{B}$ is empty, then standard Gaussian model can be used. Setting $Q(\mathbf{x})$ to be large positive and negative values for pixels in $\mathbf{O}$ and $\mathbf{B}$ acts like a hard constraint where the final segmentation must satisfy. This can effectively prune some partial optimum solutions as illustrated in Fig. 10c. Without the user-provided background stroke (in yellow), the segmentation is trapped into the partial optimum (Fig. 8b).

5.2. Gaussian Mixture Model

Aside from simple parametric and non-parametric distributions, we can also incorporate more complex intensity models such as the Gaussian Mixture Model (GMM) [22] into our framework. Each GMM, one for the foreground and one for the background, is defined to be a full-covariance Gaussian mixture with K components each, $K = 5$ in our experiment, totaling 2K components. The probability density of each intensity value $\mathbf{x}$ is contributed by all K components $k_{in}, k_{out} \in 1, ...K$, from either the background or the foreground model. Therefore, the intensity model can be described as:

$$P_{in}(\mathbf{x}) = \sum_{k_{in}} \pi_{in}(k_{in}) P(\mathbf{x}|\theta_{in}(k_{in}))$$

$$P_{out}(\mathbf{x}) = \sum_{k_{out}} \pi_{out}(k_{out}) P(\mathbf{x}|\theta_{out}(k_{out}))$$

where $\pi_{in}(k)$ and $\pi_{out}(k)$ are the k^{th} foreground and background component weights respectively. Similarly, $\theta_{in}(k)$ and $\theta_{out}(k)$ are the mean and the covariance matrix of the k^{th} foreground and background component.

In Algorithm 2, intensity models are pre-computed before the estimation of transformation parameters and are fixed during the iteration. If the intensity model is not accurate, then the segmentation result will be sub-optimal. Also they are not able to model complex intensity distributions, such as Fig. 11. With GMM, the intensity modeling works iteratively. This has the advantage of allowing automatic refinement of the intensity model, as newly labeled foreground and background pixels from $u_{ref}(T(\mathbf{x}))$ are used to refine the GMM parameters. The algorithm of shape prior image segmentation with GMM is illustrated in Algorithm 3. Notice that we add an extra step of intensity model update in the iterative similarity estimation.

Algorithm 3 Shape prior image segmentation with GMM

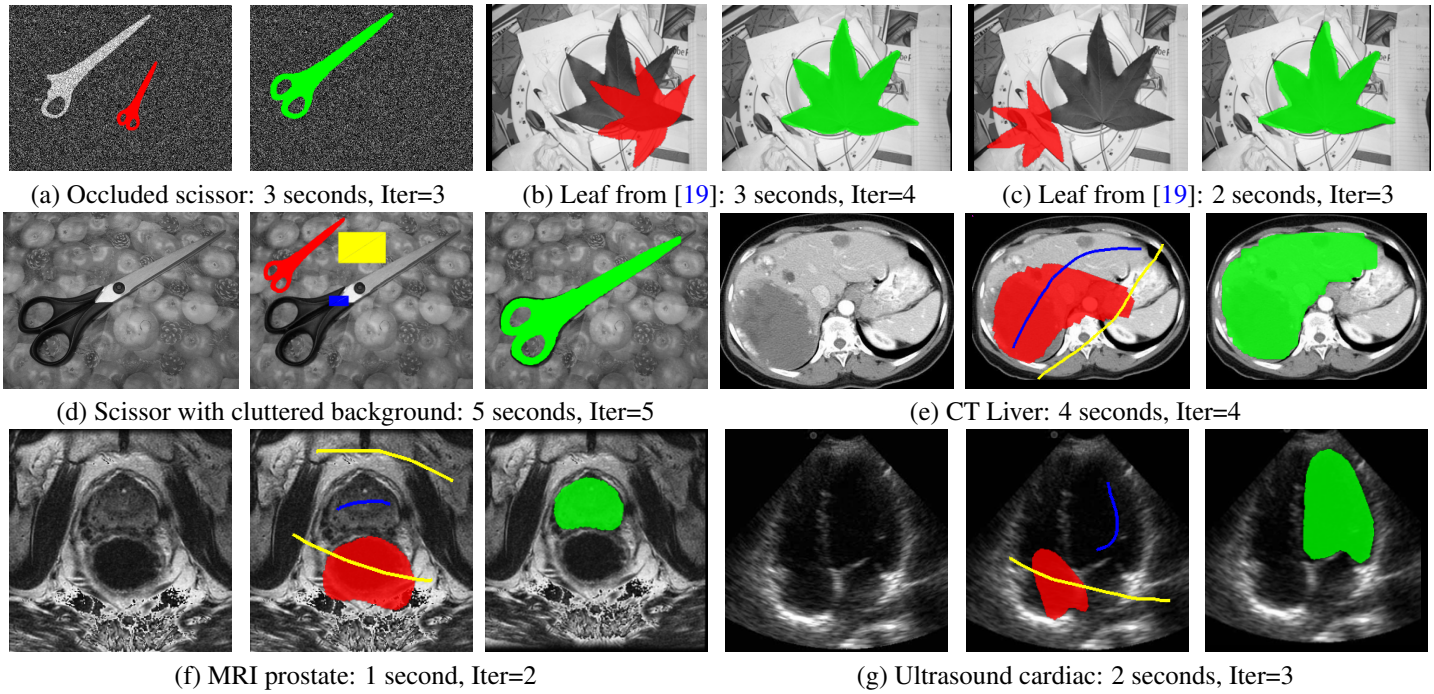
Input:Shape prior template u_0 and the image I.
Initialization: Compute initial GMM parameters and Q.
while Not converged **do**
1.Compute translation $\mathbf{t}$ from u_i and Q. (Section 4.1)
2.Update u_i with $\mathbf{t}$ to $u_{i\mathbf{t}}$
3.Compute s and r from $u_{i\mathbf{t}}$ and Q. (Section 4.2).
4.Update $u_{i\mathbf{t}}$ with s and r to u_{i+1}
5.Update the GMM paramters and Q.
end while
Deformable transformation estimation. (Section 4.3).

(a) (b) (c)

Figure 11: Shape-prior segmentation with GMM (a) Image with the reference shape in red. (b) The segmentation result after similarity transformation. (c) The segmentation result after deformable transformation. This figure is best viewed in color.

Given the initial shape template, we create K components of the GMM for both the foreground and background. A standard approach is to use Expectation-Maximization (EM) algorithm, which softly assigns probabilities for each component to a given intensity observation. However, as observed by other researchers also [22], this involves significant computation complexity with negligible practical benefit. Instead, we divide both foreground and background regions into K pixel clusters, where one Gaussian component is generated from each pixel cluster. The key problem is to find well separated and low-variance clusters. Inspired by Ruzon and Tomasi [25], we use the color quantization technique proposed by Orchard and Bouman [18] which generates tight and well-separated clusters. This technique uses the eigenvector of the color variance to determine how to split the clusters. For a detailed discussion, we refer the readers to the original article [25, 18].

As we iteratively update of the similarity transform, the intermediate segmentation result will change. To update the GMM, a naive way is to rebuild the GMM from scratch with the same technique we described above. However, this is very computationally expensive. Instead, we follow the EM type update, where each pixel in the new segmentation is assigned to one of K component, by evaluating its likelihood of belonging to each component with old GMM. Then for a given component k, say, the foreground label, the subset of foreground pixels $F(k) = \{\mathbf{x_n} : k_n = k\}$ is collected and the new GMM parameters are recomputed from it. Note that the weight component $\pi(k)$ is defined as $\pi(k) = \frac{|F(k)|}{\sum_k |F(k)|}$, where $|\cdot|$ denotes the size of a set. If desired, user input such as strokes and bounding box can also be incorporated by setting $Q(\mathbf{x})$ to large positive or negative values depending on whether $\mathbf{x}$ belongs to the *a priori* known foreground or background region.



(a) Occluded scissor: 3 seconds, Iter=3 (b) Leaf from [19]: 3 seconds, Iter=4 (c) Leaf from [19]: 2 seconds, Iter=3

(d) Scissor with cluttered background: 5 seconds, Iter=5 (e) CT Liver: 4 seconds, Iter=4

(f) MRI prostate: 1 second, Iter=2 (g) Ultrasound cardiac: 2 seconds, Iter=3

Figure 12: Experimental results for similarity estimation only. This figure is best viewed in color.

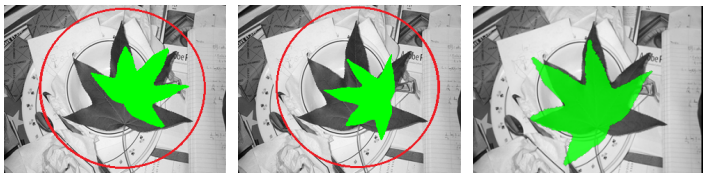
Fig. 11 illustrates one segmentation example of using GMM as the intensity model. Fig. 11a shows the input image along with the reference shape template. Fig. 11b shows the segmentation result after similarity transformation estimation. Fig. 11c illustrates the final segmentation result after deformation estimation. Notice that the stem is correctly recovered due to the deformable registration step.

6. Experiments

In this section, we apply our segmentation algorithm to various applications. For all the experiments, the red shape template indicates the prior shape template (also initial shape template) and the green shape template indicates the segmentation result. User provided foreground and background strokes (if any) are illustrated in blue and yellow respectively. User provided bounding box is shown in magenta.

Fig. 12 shows the segmentation result with Gaussian and simple histogram intensity models. If user strokes are provided, the log-likelihood map Q is estimated from the strokes, otherwise, it is estimated from the user-provided (M_{in}, M_{out}) for a two-phase model as in (3). Deformable registration is not applied to these experiments. We also show the segmentation time and the number of similarity estimation iterations in the subcaptions.

As we can see, the performance of our algorithm on these challenging image segmentation problems is very satisfactory, obtaining the near-global optimum result in a matter of a few seconds. In contrast, regularization-based methods are typically much slower and less optimal as illustrated in Fig. 1. To further compare with other approaches, we implemented the standard level-set based shape-prior image segmentation algorithms [4, 10] along with the recently proposed continuous-cuts method [19]. We compared the three methods on the same leaf image (taken from [19]) in Fig. 13, with carefully tuned parameters. The reference shape template is the same as Fig. 12b, and the initialization for the level-set evolution is illustrated as the red circle in Fig. 13ab. The number of iterations for each experiment is set to 100, where each iteration has an alternating step of updating shape and pose. For level-set algorithm [4, 10], narrow-band approach is used to speed-up the computation. Fast marching is also used to periodically reinitialize the signed distance function. All three algorithms are trapped into the local optimum, while our algorithm successfully converges to the near global optimal solution in a few seconds (Fig. 12b).



(a) (b) (c)

Figure 13: Comparison segmentation result with three techniques. (a) [4]. (b) [10]. (c) [19].

Fig. 14 shows the real-world challenging segmentation results [**?**] using full transformation (both similarity and deformable) with GMM as the intensity mode. User input such as strokes and bounding box can be incorporated in the same manner as described in Section 5.2. Fig. 14a shows the original image with user input if any. Fig. 14b shows the reference shape template. Fig. 14c shows the segmentation result after similar-

Figure 14: Segmentation results of full transformation estimation with GMM as intensity model. (a) Image with user input. (b) Reference shape template. (c) Segmentation result after similarity transformation. (d) Final segmentation result. This figure is best viewed in color.

Figure 15: Walking data experimental results. (a)–(c) 3 representative shape templates, (d)-(f) The optimum segmentation results.

ity transformation estimation. This intermediate segmentation result is near-optimal with respect to similarity transformation. This optimality guarantee greatly reduces the chance that deformable estimation being trapped into local optimum. Fig. 14d shows the final segmentation result.

We also tested our algorithm on publicly available data provided by the authors of [9]. In this experiment, we take 3 representative shape silhouettes from the training database, and 3 representative images from the test database. For each test image, we run our algorithm (two-phase model) with all 3 shapes and pick the result with the highest objective function value. The result is illustrated in Fig. 15. This is a straightforward preliniminary application of our model to the problem of image segmentation with multiple shape priors. Loosely speaking, it is an exhaustive search in a coarsely discretized shape space.

7. Conclusion

By reformulating the problem of image segmentation with one shape prior into a template-based framework, we designed a highly efficient segmentation framework. The preliminary experimental results indicate the satisfactory performance of our algorithm. In the future, we plan to investigate a more complicated segmentation algorithm for multiple shape templates. We are also extending our current framework to affine and projective transformation. In addition, we are also planning to apply our algorithm to challenging 3D medical segmentation applications, where many current methods fail to provide a fast, good solution.

References

[1] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In *ECCV*, 2006. 3

[2] Y. Boykov and G. Funka-Lea. Graph cuts and efficient Nd image segmentation. *IJCV*, 70(2):109–131, 2006. 7

[3] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *IJCV*, 22(1):61–79, 1997. 2

[4] T. Chan and W. Zhu. Level set based shape prior segmentation. In *CVPR*, 2005. 1, 2, 9

[5] S. Chen, G. Charpiat, and R. Radke. Converting level set gradients to shape gradients. In *ECCV*, 2010. 2

[6] Y. Chen, H.D. Tagare, S. Thiruvenkadam, et al. Using prior shapes in geometric active contours in a variational framework. *IJCV*, 50(3):315–328, 2002. 2, 3

[7] J.W. Cooley and J.W. Tukey. An algorithm for the machine calculation of complex Fourier series. *Mathematics of Computation*, 19(90):297–301, 1965. 4

[8] D. Cremers, S.J. Osher, and S. Soatto. Kernel density estimation and intrinsic alignment for shape priors in level set segmentation. *IJCV*, 69(3):335–351, 2006. 2

[9] D. Cremers, F.R. Schmidt, and F. Barthel. Shape priors in variational image segmentation: Convexity, Lipschitz continuity and globally optimal solutions. In *CVPR*, 2008. 10

[10] D. Cremers and S. Soatto. A pseudo-distance for shape priors in level set segmentation. In *VLSM*, 2003. 2, 9

[11] D. Freedman and T. Zhang. Interactive graph cut based segmentation with shape priors. In *CVPR*, 2005. 1, 2

[12] J. Gorski, F. Pfeuffer, and K. Klamroth. Biconvex sets and optimization with biconvex functions: a survey and extensions. *Mathematical Methods of Operations Research*, 66(3):373–407, 2007. 7

[13] MP Kumar, PHS Ton, and A. Zisserman. OBJ CUT. In *CVPR*, 2005. 1, 2

[14] J.P. Lewis. Fast template matching. In *Vision Interface*, volume 95, pages 120–123, 1995. 4

[15] D.G. Lowe. Object recognition from local scale-invariant features. In *ICCV*, 1999. 3

[16] S. Manay, D. Cremers, A. Yezzi, and S. Soatto. One-shot integral invariant shape priors for variational segmentation. In *EMMCVPR*, 2005. 2

[17] D. Mattes, D.R. Haynor, H. Vesselle, et al. Pet-ct image registration in the chest using free-form deformations. *Medical Imaging, IEEE Transactions on*, 22(1):120–128, 2003. 5

[18] M.T. Orchard and C.A. Bouman. Color quantization of images. *IEEE Trans. Signal Processing*, 39(12):2677–2690, 1991. 8

[19] N. Overgaard, K. Fundana, and A. Heyden. Pose Invariant Shape Prior Segmentation Using Continuous Cuts and Gradient Descent on Lie Groups. *SSVM*, 2009. 1, 2, 9

[20] BS Reddy and BN Chatterji. An FFT-based technique for translation, rotation, and scale-invariant image registration. *IEEE Trans. Image Processing*, 5(8):1266–1271, 1996. 4, 7

[21] T. Riklin-Raviv, N. Kiryati, and N. Sochen. Prior-based segmentation and shape registration in the presence of perspective distortion. *IJCV*, 72(3):309–328, 2007. 2

[22] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 23(3):309–314, 2004. 7, 8

[23] M. Rousson and N. Paragios. Shape priors for level set representations. In *ECCV*, 2002. 3

[24] D. Rueckert, L.I. Sonoda, C. Hayes, et al. Nonrigid registration using free-form deformations: application to breast mr images. *Medical Imaging, IEEE Transactions on*, 18(8):712–721, 1999. 5

[25] M.A. Ruzon and C. Tomasi. Alpha estimation in natural images. In *CVPR*, 2000. 8

[26] T. Schoenemann and D. Cremers. Globally optimal image segmentation with an elastic shape prior. In *ICCV*, 2007. 1, 2

[27] A. Tsai, A. Yezzi Jr, W. Wells, et al. A shape-based approach to the segmentation of medical imagery using level sets. *IEEE Trans. Medical Imaging*, 22(2):137–154, 2003. 2

[28] R.E. Wendell and A.P. Hurter Jr. Minimization of a non-separable objective function subject to disjoint constraints. *Operations Research*, 24(4):643–657, 1976. 7

[29] M. Werlberger, T. Pock, M. Unger, and H. Bischof. A variational model for interactive shape prior segmentation and real-time tracking. *SSVM*, 2009. 1, 2

[30] G. Wolberg and S. Zokai. Robust image registration using log-polar transform. In *ICIP*, 2000. 4, 5

[31] S.C. Zhu and A. Yuille. Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. *IEEE TPAMI*, 18(9):884–900, 2002. 2

[32] B. Zitova and J. Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21(11):977–1000, 2003. 3, 4