

Template Matching Based Object Recognition With Unknown Geometric Parameters

Roger M. Dufour, Eric L. Miller, *Member, IEEE*, and Nikolas P. Galatsanos, *Senior Member, IEEE*

Abstract—In this paper, we examine the problem of locating an object in an image when size and rotation are unknown. Previous work has shown that with known geometric parameters, an image restoration method can be useful by estimating a delta function at the object location. When the geometric parameters are unknown, this method becomes impractical because the likelihood surface to be minimized across size and rotation has numerous local minima and areas of zero gradient. In this paper, we propose a new approach where a smooth approximation of the template is used to minimize a well-behaved likelihood surface. A coarse-to-fine approximation of the original template using a diffusion-like equation is used to create a library of templates. Using this library, we can successively perform minimizations which are locally well-behaved. As detail is added to the template, the likelihood surface gains local minima, but previous estimates place us within a well-behaved “bowl” around the global minimum, leading to an accurate estimate. Numerical experiments are shown which verify the value of this approach for a wide range of values of the geometric parameters.

I. INTRODUCTION

A COMMON image processing problem is to determine the location of an object using a template when the size and rotation of the true target are unknowns [1]–[3]. An algorithm for finding the solution should be robust to noise, accurate across a wide range of object configurations, and computationally efficient. With known geometric parameters (i.e., size and rotation) and additive Gaussian noise, the classic solution for object localization is a whitening filter followed by a matched filter. Other estimators have been proposed, such as the phase only matched filter (POMF) and the symmetric phase only matched filter (SPOMF) which give better location discrimination than the standard matched filter [3]. However these filters do not use additional information regarding the background noise which may be available or estimable. Alternatively, one may formulate the localization problem in the framework of image reconstruction, where the image to be recovered is a delta function at the location of the object and the blurring kernel is the template to be matched [1]. This method follows the expansion matching (EXM) method, which seeks to localize by expanding the image

on a basis composed as translated versions of the object [4]–[6]. This approach makes available image reconstruction methods such as the linear least squares estimator (LLSE) and maximum likelihood estimator which can incorporate background statistics. The final step in either the delta restoration or the EXM method is simply to select the point of highest response in the recovered image.

Since they rely upon an accurate template the methods mentioned above are not sufficient when geometric parameters such as size and rotation are unknown. Here, one must compose an estimator which is invariant to these parameters or simultaneously estimate these parameters. With the invariant approach [7]–[10], a discriminant function is composed in the spatial, frequency or another domain which is invariant to an affine transformation of the template. The discriminant function is usually computed from local or global features of the image, boundaries within the image or regions of the image. The invariant function score can then be used for detection or classification. The second approach and the one followed in this paper is to estimate the parameters. By estimating the parameters, an accurate template match can be achieved as in the methods mentioned earlier. This approach is also taken with the Fourier Mellin Matched Filter (FMMF) [2]. In the FMMF, the amplitude of the Fourier transform is mapped to a log-polar coordinate system. The translational dependence is eliminated because it appears only in the phase of the Fourier transform. The transformation to the log-polar coordinate system converts the rotation and scaling parameters into translational parameters. Scale and rotation are then solved with a matched filter or phase only matched filter. While computationally fast and a simple method, it is not very robust to noise or clutter. Additionally if the template does not accurately match the object in the image, this method again will often fail. Another approach is to minimize a likelihood surface defined from the difference between the matched template and the image. However, this surface is not amenable to minimization due to numerous local minima and areas of small or zero gradient. Numerous randomized techniques for minimizing a surface with multiple minima have been developed, such as simulated annealing [11], [12] or by using a jump diffusion technique [13], [14]. With these techniques, the parameter set is updated by a step increment. As the algorithm progresses, the increment becomes smaller until the parameters settle into a minima. With proper selection of the increment size, the final solution is, with high probability, the global minimum.

We propose a deterministic method for searching this surface using a progression of templates. The early templates are smooth approximations of the exact template. This results in a search of a well-behaved approximation of the true likelihood

Manuscript received February 27, 2001; revised July 11, 2002. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Josiane B. Zerubia.

R. M. Dufour is with the MIT Lincoln Laboratory, Lexington, MA 02420-9185 USA (e-mail: dufour@ll.mit.edu).

E. L. Miller is with the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA 02115 USA.

N. P. Galatsanos is with the Department of Electrical and Computer Science, University of Ioannina, Ioannina, Greece, and also with the Department of Computer Science, University of Ioannina, Ioannina, Greece.

Digital Object Identifier 10.1109/TIP.2002.806245

surface and therefore we can get close to the global minimum of the cost surface. A standard optimization routine such as the Newton algorithm can be used to find the best fit solution. Using this estimate as a starting point, we add more detail to our template and search again. As we add detail, the surface becomes more ill-behaved, but the previous estimates have placed us within a “basin of attraction” of the global minimum. This method is similar to the Graduated Non-Convexity (GNC) approach [15]–[17] in that it allows us to locally search a series of approximations to the likelihood surface and in a small number of iterations will carry us to an accurate solution. To generate the templates for the search, we use a diffusion like equation which allows fast Fourier based computations of the templates, coupled with the Fourier based image restoration method [1]. This leads to a Fourier domain algorithm which is not computationally burdensome.

We demonstrate the performance of the algorithm using synthetic images, infrared images and optical images. The experimental evidence shows that this approach is able to accurately estimate the size, rotation and location of an object across a wide range of signal to noise ratios and clutter. In addition, it was demonstrated that it also works well for a wide range of values of the rotation and scaling parameters.

In Section II we will introduce the problem setup and the solution method via an impulse estimation routine. In Section III, we present a method for developing the template library. Section IV shows the Newton algorithm used for parameter estimation. Section V discusses the information loss and performance of the parameter estimates. In Section VI, we present our numerical experiments. Finally, in Section VII we present some conclusions and directions for future work.

II. BACKGROUND

In [18] the problem of template matching was formulated using a novel nonorthogonal image expansion approach. In [1] it was shown that this approach in essence was an image restoration approach. According to this approach the signal to be restored is a delta function at the template location. This leads to the convolution equation

$$g(\mathbf{r}) = f(\mathbf{r}; \boldsymbol{\theta}^0) * \delta(\mathbf{r} - \bar{\mathbf{r}}^0) + n(\mathbf{r}) \quad (1)$$

where the position vector $\mathbf{r}^T = [r_x \ r_y]$, and $*$ represents two-dimensional (2-D) convolution. Here the object is represented by the template, f , at some location $\bar{\mathbf{r}}^0$. We have described the template using a parameter set, $\boldsymbol{\theta}$ (where $\boldsymbol{\theta}^0$ is the true parameter set), which specifies the geometric parameters of the template (i.e., the size s and rotation ϕ), as

$$\boldsymbol{\theta}^T = [s \ \phi]. \quad (2)$$

The template is a rotated and resized version of the standard template, $f(\mathbf{r})$, given as

$$f(\mathbf{r}; \boldsymbol{\theta}) = f\left(\frac{1}{s}\mathbf{M}(\phi)\mathbf{r}\right), \text{ where } \mathbf{M}(\phi) = \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix}. \quad (3)$$

As in [1] for estimation purposes we assume that the noise and the δ are zero-mean stochastic functions with spectra given as

$$S_\delta(\mathbf{k}) = \sigma_\delta^2 \text{ and } S_n(\mathbf{k}) = \sigma_n^2 \quad (4)$$

where $\mathbf{k}^T = [k_x \ k_y]$ are the spatial frequencies used in the 2-D Fourier Transform

$$\tilde{f}(\mathbf{k}) = \mathcal{F}\{f(\mathbf{r})\} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\mathbf{r}) \exp(-j2\pi\mathbf{k} \cdot \mathbf{r}) d\mathbf{r}. \quad (5)$$

With this framework, we now use a maximum likelihood estimate of the parameter set $\hat{\boldsymbol{\theta}}$, while using the delta restoration method [1] for the location estimation $\hat{\mathbf{r}}$. Since the additive noise is assumed Gaussian, the negative log-likelihood for our parameters is

$$l(\boldsymbol{\theta}, \bar{\mathbf{r}}|g) = \frac{1}{\sigma_n^2} \|f(\mathbf{r}; \boldsymbol{\theta}) * \delta(\mathbf{r} - \bar{\mathbf{r}}) - g(\mathbf{r})\|_2^2. \quad (6)$$

The maximum likelihood solution is then achieved by minimizing (6) as

$$\{\hat{\mathbf{r}}, \hat{\boldsymbol{\theta}}\} = \arg \min_{\{\bar{\mathbf{r}}, \boldsymbol{\theta}\}} l(\boldsymbol{\theta}, \bar{\mathbf{r}}|g). \quad (7)$$

In general, determining the exact solution to this optimization problem (i.e., the ML estimate of location, size, and rotation) is not a trivial task. This is seen by the wide range of localization techniques discussed in the Introduction both when the geometry is known and when it is not known. Hence, in this paper, we propose a suboptimal estimation scheme which is both computationally tractable and is demonstrated to work effectively on real data. Our approach has two steps. First, the delta restoration technique of Abu-Naser *et al.* [1] is used to determine an almost closed-form estimate of the location of the target in terms of the geometric parameters. Using this estimate to reduce the space of unknowns only to the size and rotation parameters, a nonlinear least squares approach is employed to determine these remaining quantities. Because even this problem is plagued by local minima of the cost function, the template progression approach is introduced to “ease” the estimate of the geometric parameters into the global basin of attraction.

With a given set of geometric parameters $\boldsymbol{\theta}$, we can construct an estimate of position following Abu-Naser [1] by first making a Linear Least Squares Estimate (LLSE) of the delta function,

$$\hat{\delta}(\mathbf{r}) = \mathcal{F}^{-1} \left\{ \frac{\tilde{f}_0^*(\mathbf{k}; \boldsymbol{\theta})}{|\tilde{f}_0(\mathbf{k}; \boldsymbol{\theta})|^2 + \left(\frac{\sigma_n}{\sigma_\delta}\right)^2} \tilde{g}(\mathbf{k}) \right\} \quad (8)$$

and selecting the point of maximum response as our position estimate, as¹

$$\hat{\mathbf{r}}(\boldsymbol{\theta}) = \arg \max_{\mathbf{r}} \hat{\delta}(\mathbf{r}). \quad (9)$$

¹While $\hat{\delta}$ is the statistically optimal estimate of δ , taking the maximum in (9) does not guarantee that the final estimate of position is in fact the maximum likelihood estimate

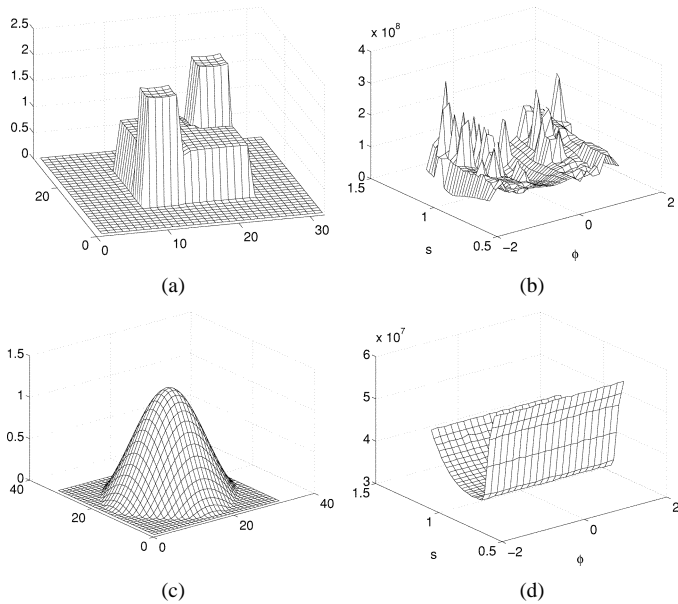


Fig. 1. Likelihood surfaces and associated templates. (a) Exact template, (b) exact surface, (c) smooth approximate template, and (d) smooth approximate surface.

Substituting (9) into (7), we have a nonlinear least squares estimator, $\hat{\theta}$ as

$$\begin{aligned} \hat{\theta} &= \arg \min_{\theta} \frac{1}{\sigma_n^2} \|f(\mathbf{r}, \theta) * \delta(\mathbf{r} - \hat{\mathbf{r}}(\theta)) - g(\mathbf{r})\|_2^2 \\ &\equiv \arg \min_{\theta} J(\theta; f). \end{aligned} \quad (10)$$

The minimization of $J(\theta; f)$ could be accomplished by calculating $J(\theta; f)$ on a dense grid in the $\phi - s$ space to find the minimum point. However, the calculations required to perform this are generally prohibitive. The computation of the surface shown in Fig. 1 needed 67 billion flops. Instead, we seek to use standard optimization tools to find the minimum, the gradient descent algorithm we propose used only 260 million flops to find the minimum of the surface in Fig. 1. The problem which we encounter when we attempt to do this is that the likelihood surface is not amenable to a gradient descent methods because of local minima and other areas of zero gradient. We find that these problem areas are a direct result of the shape of the template, f ; specifically, multiple maxima in the template lead to multiple maxima in the likelihood surface. This can be seen in Fig. 1 where we have in (a) a simple target template with multiple maxima and in (b) the $J(\theta; f)$ surface beside it. The example given here is for estimating the same two peaked object with size $s = 1$ and rotation $\phi = 0$ with noiseless data. The minimum of this surface is a sharp point at size 1.0 and rotation 0, but the irregularity of the surface makes descent-type minimizations impractical. Conversely, if we use a smooth rotationally invariant template, shown in (c), to perform the match we obtain the $J(\theta; f)$ surface in (d) which is far better behaved. The minimum of this surface while close to 1.0 is no longer a sharp point, but has a broad minima with much less curvature. This broader curve implies that the estimate will be more sensitive to noise in the data relative to that produced by the exact template. Thus the primary objective of this work is to capture the

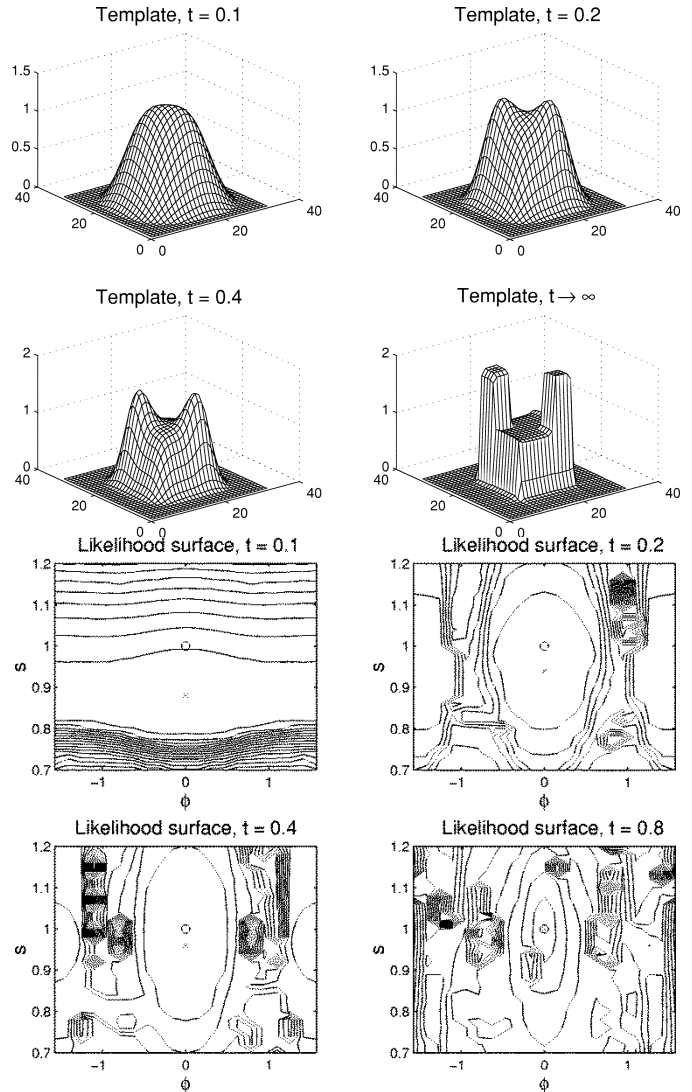


Fig. 2. Four templates and contour plots of the associated likelihood surfaces. The exact parameters are marked by the “o”, the local minimum found by gradient descent is marked by the “x.”

accuracy of the exact template estimation with the more easily minimizable surface of the approximate template.

We formulate a method which uses smooth approximate templates which allow us to minimize on a well-behaved surface. Since the results of the minimizations which we obtain from the approximations are not as accurate as those which could be obtained from the exact template, we will add detail back to our templates and minimize again with the previous estimate starting our current estimate within a well-behaved region around the global minimum.

III. TEMPLATE PROGRESSION

In this section we will detail a method of generating target templates which are approximations of the true template. This will allow us to make successive estimates beginning at smooth approximation which will generally locate us near the global optimum and refine that estimate with more detailed approximations of the template until a sufficiently accurate estimate of the parameters is reached.

TABLE I
VALUES OF ELEMENTS OF $\nabla_{\theta} \hat{\mathbf{r}}(\theta)$ FOR 1095 ITERATIONS

$ \nabla_{\theta} f(\mathbf{r}; \theta) $	$\frac{\partial \hat{r}_x}{\partial s}$		$\frac{\partial \hat{r}_y}{\partial s}$		$\frac{\partial \hat{r}_x}{\partial \phi}$		$\frac{\partial \hat{r}_y}{\partial \phi}$	
	mean	std	mean	std	mean	std	mean	std
1.3534	0.0090	0.0334	0.0090	0.0334	0.0087	0.0406	0.0087	0.0406

We will generate a spectrum of templates, indexed by t as f_t , ranging from a smooth template f_0 to the exact template $f_E = \lim_{t \rightarrow \infty} f_t$. We desire that the most smooth template be monomodal to induce the behavior in the likelihood surface which we discussed in Section II. A relatively simple choice of this is a Gaussian blob matched (in the two-norm sense) in size and amplitude to the original template given as

$$f_0(\mathbf{r}) = A \exp\left(-\frac{\|\mathbf{r}\|_2^2}{2w^2}\right). \quad (11)$$

While successive estimation will locate us inside a ‘‘well of attraction’’ of the global minimum where we will avoid local minima, discontinuities in any template can yield discontinuities on our surface. We therefore choose a method for generating templates which in the continuous domain produces continuous templates. A diffusion like process performs this adequately. We specify the Fourier transform of a template f_t as

$$\tilde{f}_t(\mathbf{k}) = \left(\tilde{f}_E(\mathbf{k}) - \tilde{f}_0(\mathbf{k})\right) \exp\left(\frac{-\|\mathbf{k}\|^2}{t}\right) + \tilde{f}_0(\mathbf{k}) \quad (12)$$

where \tilde{f}_E and \tilde{f}_0 are the Fourier transforms of the exact and most smooth templates, respectively.

A progression for the two peak template is shown in Fig. 2. We see that at small values of t , the template is a smooth approximation of the true template, as t increases, the true template emerges. Associated with this are the likelihood surfaces related to each template. Here we see that at small values of t , the surface is very smooth has no rotational localization and a very broad scale localization. As t increases, the ill-behavior returns, but previous estimates place us within the area of the global minimum, and our estimate becomes more accurate. The final solution for the parameters will of course be a local minimum, but may not necessarily be the global minimum. As will be shown later, the rate at which the templates evolve influences the final outcome of the optimization. Slower t -schedules as expected lead to more accurate estimates of the global minimum.

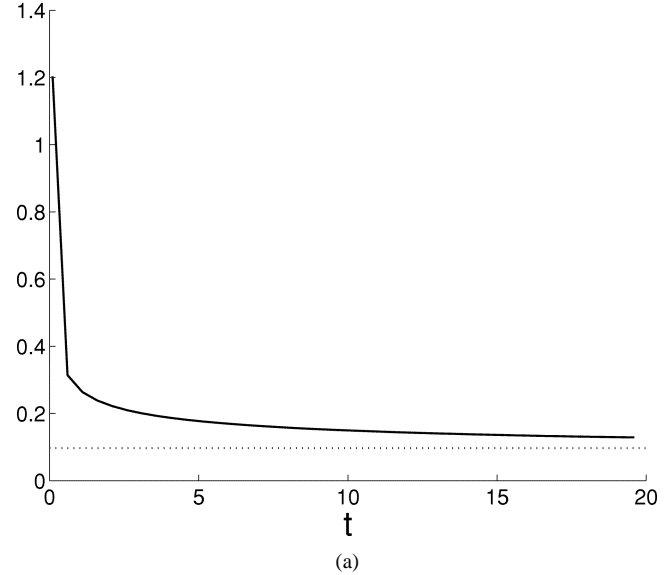
We can now describe a complete algorithm using the template progression as

1. Begin at $t = 0$.
2. Construct f_t with (12).
3. Minimize the likelihood surface constructed via the equations

$$\hat{\mathbf{r}}_t(\theta) = \arg \max_{\mathbf{r}} \mathcal{F}^{-1} \left\{ \frac{\tilde{f}_t^*(\mathbf{k}; \theta)}{|\tilde{f}_t(\mathbf{k}; \theta)|^2 + \left(\frac{\sigma_n}{\sigma_s}\right)^2} \tilde{g}(\mathbf{k}) \right\} \quad (13)$$

$$\hat{\theta}_t = \arg \min_{\theta} \frac{1}{\sigma_n^2} \|f_t(\mathbf{r}; \theta) * \delta(\mathbf{r} - \hat{\mathbf{r}}_t(\theta)) - g(\mathbf{r})\|_2^2. \quad (14)$$

Std. Dev. of Size Estimate versus t



Curvature with respect to ϕ

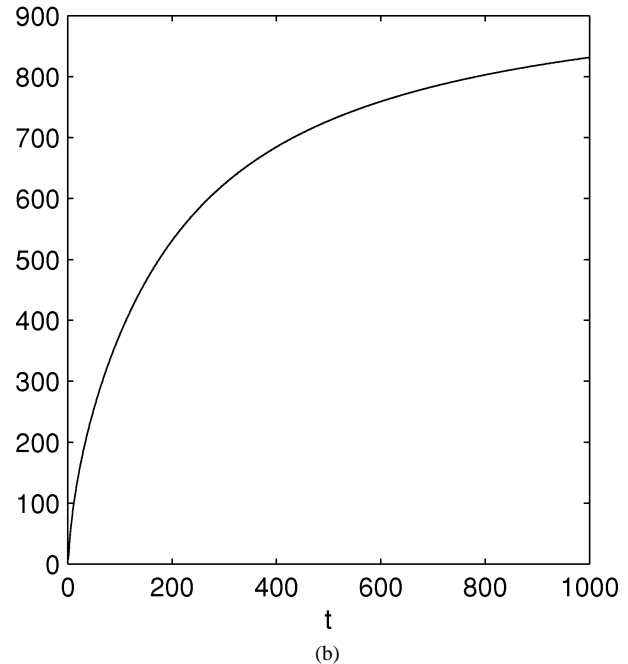


Fig. 3. The curvature of the likelihood surface with respect to t for (a) size estimates and (b) rotation estimates.

The minimization in (14) is performed via a Newton algorithm given in Section IV. 4. Increase t and proceed to step 2. The t -schedule should be chosen to take small steps at low values of t where the algorithm is more sensitive to changes in the likelihood surface. This agrees with

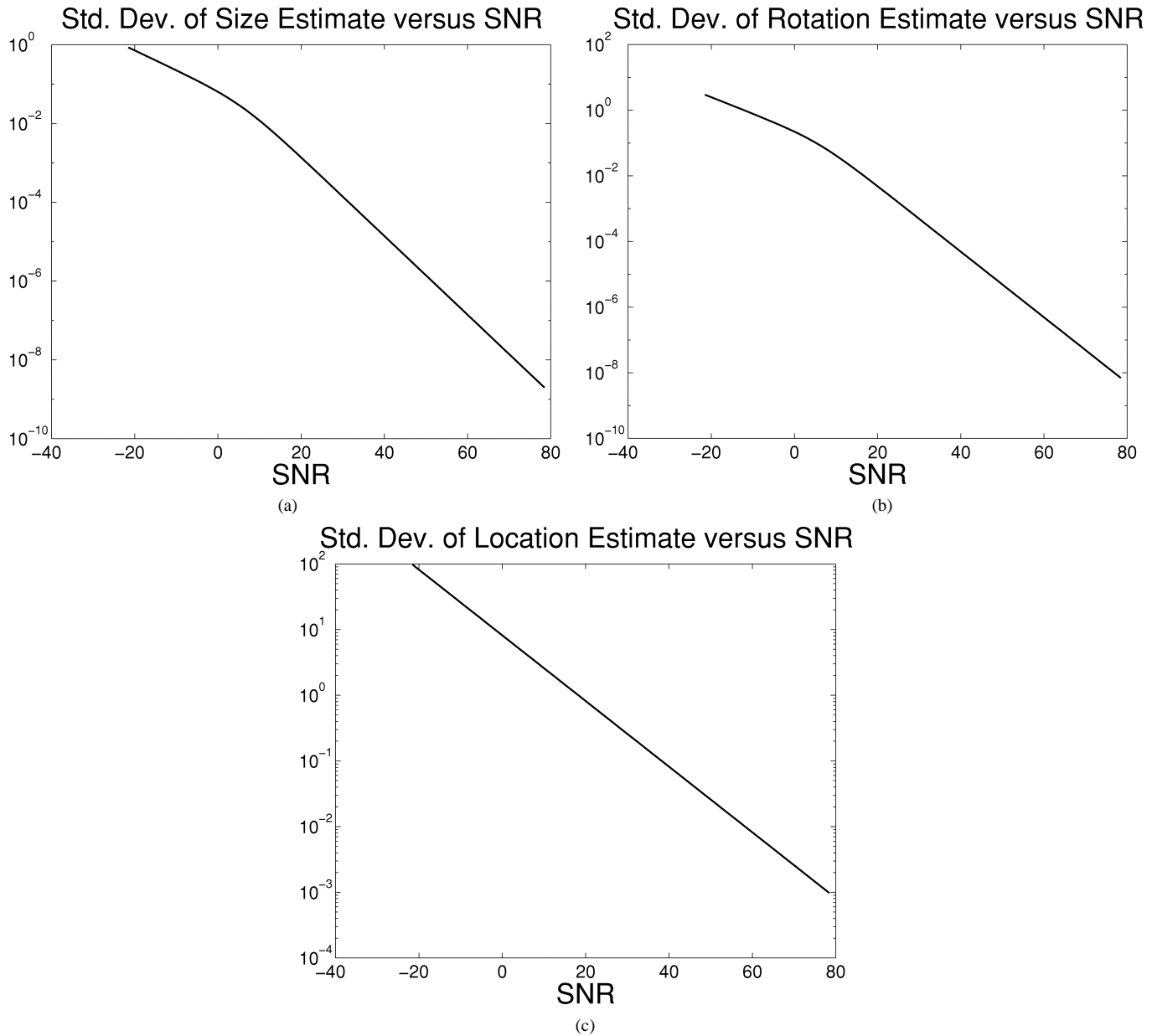


Fig. 4. The bounds upon the parameter estimates with respect to SNR for (a) size estimates, (b) rotation estimates, and (c) location estimates.

the observations in [16] for the method of graduated nonconvexity and also demonstrated in the analysis of the bounds on estimation accuracy in Section V. In our implementation, a small value is chosen for the initial value of t and it is doubled for each subsequent t . If a longer schedule is desired, a smaller multiplier is used.

IV. NEWTON ALGORITHM

The Newton algorithm [19] is used for the minimizations in our paper. The Newton algorithm implemented here seeks to minimize the squared error in the estimated image against the data according to

$$\hat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}; f_t) = \arg \min_{\boldsymbol{\theta}} \int e_t^2(\mathbf{r}; \boldsymbol{\theta}) d\mathbf{r} \quad (15)$$

where e_t is the error image given as

$$e_t(\mathbf{r}; \boldsymbol{\theta}) = f_t(\mathbf{r}; \boldsymbol{\theta}) * \delta(\mathbf{r} - \hat{\mathbf{r}}_t(\boldsymbol{\theta})) - g(\mathbf{r}). \quad (16)$$

The Newton iterative procedure produces updates of the parameter vector as [19]

$$\begin{aligned} \boldsymbol{\theta}^{(k+1)} &= \boldsymbol{\theta}^{(k)} + \mathbf{p}^{(k)} \quad (17) \\ \text{with } \left(\mathbf{U}_t(\boldsymbol{\theta}^{(k)}) + \mathbf{S}_t(\boldsymbol{\theta}^{(k)}) \right) \mathbf{p}^{(k)} &= - \int \mathbf{J}_t(\mathbf{r}; \boldsymbol{\theta}^{(k)}) \\ &\quad \times e_t(\mathbf{r}; \boldsymbol{\theta}^{(k)}) d\mathbf{r} \quad (18) \end{aligned}$$

where $\mathbf{p}^{(k)}$ is the update vector for the parameter set, \mathbf{J}_t is the Jacobian vector of the error function, and \mathbf{U}_t and \mathbf{S}_t are functions of the Jacobian and Hessian that are described below.

The Jacobian vector is the vector of first derivative functions of the of the error as

$$\begin{aligned} \mathbf{J}_t(\mathbf{r}; \boldsymbol{\theta}) &= \nabla_{\boldsymbol{\theta}} e_t(\mathbf{r}; \boldsymbol{\theta}) = \left[\frac{\partial e_t(\mathbf{r}; \boldsymbol{\theta})}{\partial s} \quad \frac{\partial e_t(\mathbf{r}; \boldsymbol{\theta})}{\partial \phi} \right]^T \\ &= \nabla_{\boldsymbol{\theta}} f_t(\mathbf{r}; \boldsymbol{\theta}) * \delta(\mathbf{r} - \hat{\mathbf{r}}_t(\boldsymbol{\theta})) + f_t(\mathbf{r}; \boldsymbol{\theta}) \\ &\quad * \nabla_{\boldsymbol{\theta}} \delta(\mathbf{r} - \hat{\mathbf{r}}_t(\boldsymbol{\theta})) \end{aligned} \quad (19)$$

$$\begin{aligned} &= \nabla_{\boldsymbol{\theta}} f_t(\mathbf{r} - \hat{\mathbf{r}}_t(\boldsymbol{\theta}); \boldsymbol{\theta}) \\ &\quad - \nabla_{\boldsymbol{\theta}} \hat{\mathbf{r}}_t(\boldsymbol{\theta}) \nabla_{\mathbf{r}} f_t(\mathbf{r} - \hat{\mathbf{r}}_t(\boldsymbol{\theta}); \boldsymbol{\theta}). \end{aligned} \quad (20)$$

The first term of (20) is directly computable from the template function, however the second term is troublesome. The gradient $\nabla_{\boldsymbol{\theta}} \hat{\mathbf{r}}_t(\boldsymbol{\theta})$ is not guaranteed to exist everywhere, and even if it does exist, it cannot be calculated in closed form since $\hat{\mathbf{r}}_t(\boldsymbol{\theta})$ involves a maximization. Numerical computation of $\nabla_{\boldsymbol{\theta}} \hat{\mathbf{r}}_t(\boldsymbol{\theta})$ is also complicated by the large granularity of image pixels with respect to the usual size of $\nabla_{\boldsymbol{\theta}} \hat{\mathbf{r}}_t(\boldsymbol{\theta})$. By this, we mean that if we attempt to approximate an element of $\nabla_{\boldsymbol{\theta}} \hat{\mathbf{r}}_t(\boldsymbol{\theta})$ (for example $\partial \hat{r}_x(\boldsymbol{\theta}) / \partial s$) by the relation

$$\frac{\partial \hat{r}_x(s, \phi)}{\partial s} \approx \frac{\hat{r}_x(s + \Delta s, \phi) - \hat{r}_x(s, \phi)}{\Delta s} \quad (21)$$

we find that the numerator is either zero (if the two estimates of position are usually the same pixel), or arbitrarily large (if they are different pixels) because Δs is made small. Therefore we used an alternative method to obtain a more accurate approximation of the elements of $\nabla_{\boldsymbol{\theta}} \hat{\mathbf{r}}_t(\boldsymbol{\theta})$. First we increase s until $\hat{r}_x(s, \phi)$ moved at least by a pixel and denoted this point by s_1 . Similarly, we then find another point by decreasing s and this point is denoted by s_2 . The derivative is then approximated by

$$\frac{\partial \hat{r}_x(s, \phi)}{\partial s} \approx \frac{\hat{r}_x(s_2, \phi) - \hat{r}_x(s_1, \phi)}{s_2 - s_1}. \quad (22)$$

A similar approach was used to compute the remaining three elements of $\nabla_{\boldsymbol{\theta}} \hat{\mathbf{r}}_t(\boldsymbol{\theta})$, that is $\partial \hat{r}_y(\boldsymbol{\theta}) / \partial s$, $\partial \hat{r}_x(\boldsymbol{\theta}) / \partial \phi$, and $\partial \hat{r}_y(\boldsymbol{\theta}) / \partial \phi$.

The disadvantage of this method is that it is computationally intensive since we must calculate $\hat{\mathbf{r}}_t(\boldsymbol{\theta})$ at many points. Using this method, we find that the elements of the matrix $\nabla_{\boldsymbol{\theta}} \hat{\mathbf{r}}_t(\boldsymbol{\theta})$ are typically two to three orders of magnitude smaller than $\nabla_{\boldsymbol{\theta}} f_t(\mathbf{r} - \hat{\mathbf{r}}_t(\boldsymbol{\theta}); \boldsymbol{\theta})$ in (20). This is demonstrated by the figures in Table I, which shows a comparison between the average magnitude of $\nabla_{\boldsymbol{\theta}} f_t(\mathbf{r}; \boldsymbol{\theta})$ and the elements of $\nabla_{\boldsymbol{\theta}} \hat{\mathbf{r}}_t(\boldsymbol{\theta})$ for 20 runs (1095 iterations) of the algorithm for the two-peak example shown earlier. It is reasonable therefore to simplify the calculation by eliminating this term, and using the approximation for the Jacobian given by

$$\mathbf{J}_t(\mathbf{r}; \boldsymbol{\theta}) \approx \nabla_{\boldsymbol{\theta}} f_t(\mathbf{r} - \hat{\mathbf{r}}_t(\boldsymbol{\theta}); \boldsymbol{\theta}). \quad (23)$$

The Hessian of the error, or the matrix of second derivatives, is

$$\begin{aligned} \mathbf{H}_t(\mathbf{r}; \boldsymbol{\theta}) &= \nabla_{\boldsymbol{\theta}}^2 e_t(\mathbf{r}; \boldsymbol{\theta}) \\ &= \left[\nabla_{\boldsymbol{\theta}}^2 f_t(\mathbf{r}; \boldsymbol{\theta}) - \nabla_{\boldsymbol{\theta}}^2 \hat{\mathbf{r}}_t(\boldsymbol{\theta}) \nabla_{\mathbf{r}} f_t(\mathbf{r}; \boldsymbol{\theta}) \right. \\ &\quad \left. - 2 \nabla_{\boldsymbol{\theta}} \hat{\mathbf{r}}_t(\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} \nabla_{\mathbf{r}} f_t(\mathbf{r}; \boldsymbol{\theta}) + (\nabla_{\boldsymbol{\theta}} \hat{\mathbf{r}}_t(\boldsymbol{\theta}))^2 \nabla_{\mathbf{r}}^2 f_t(\mathbf{r}; \boldsymbol{\theta}) \right] \\ &\quad * \delta(\mathbf{r} - \hat{\mathbf{r}}_t(\boldsymbol{\theta})). \end{aligned} \quad (24)$$

Similarly to the Jacobian, the terms of the Hessian which contain the gradients of $\hat{\mathbf{r}}_t(\boldsymbol{\theta})$ are usually not significant and can cause computational problems, so we disregard those terms. The ap-

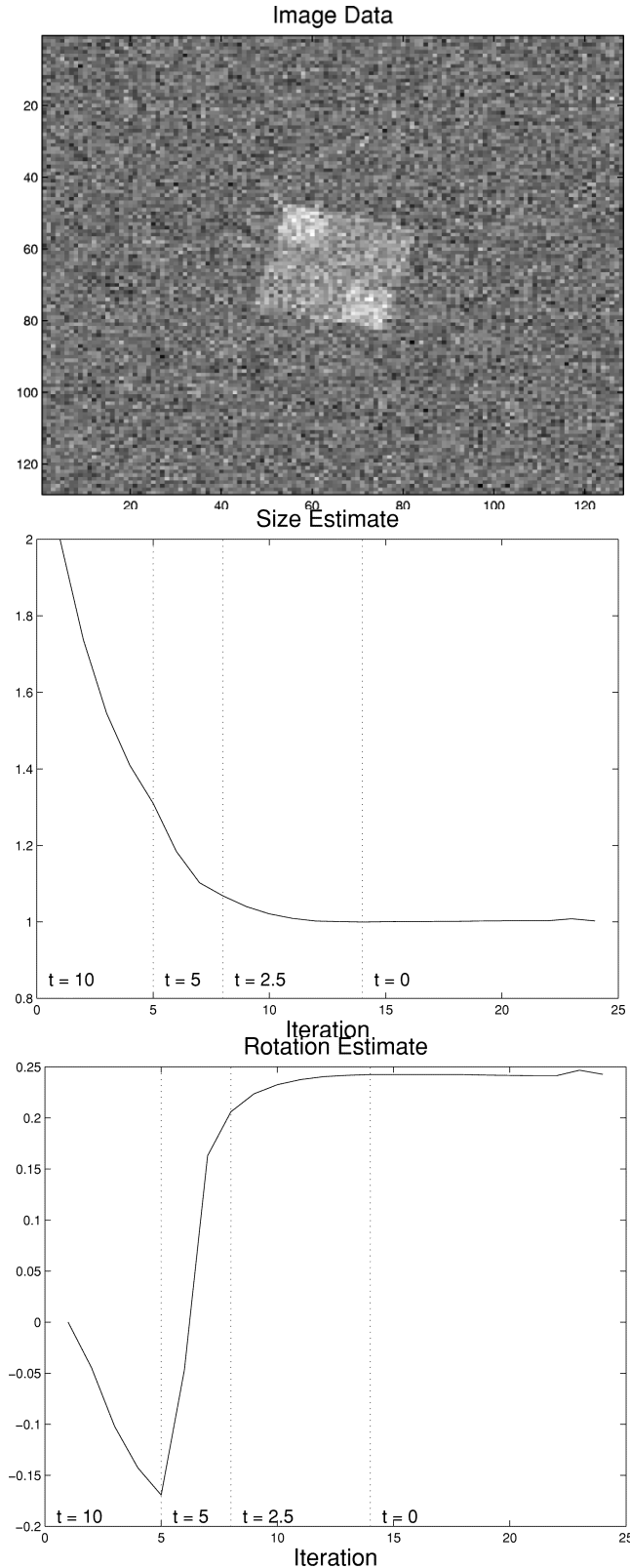


Fig. 5. Synthetic example. The two-peak target in 0 dB SNR. Accurate parameter estimation was achieved in 24 iterations with four values of t .

proximate Hessian which we used is given by

$$\mathbf{H}_t(\mathbf{r}; \boldsymbol{\theta}) \approx \nabla_{\boldsymbol{\theta}}^2 f_t(\mathbf{r} - \hat{\mathbf{r}}_t(\boldsymbol{\theta}); \boldsymbol{\theta}). \quad (25)$$

The two matrices on the left hand side of (18) are computed from the Jacobian and Hessian. The first is the inner product of

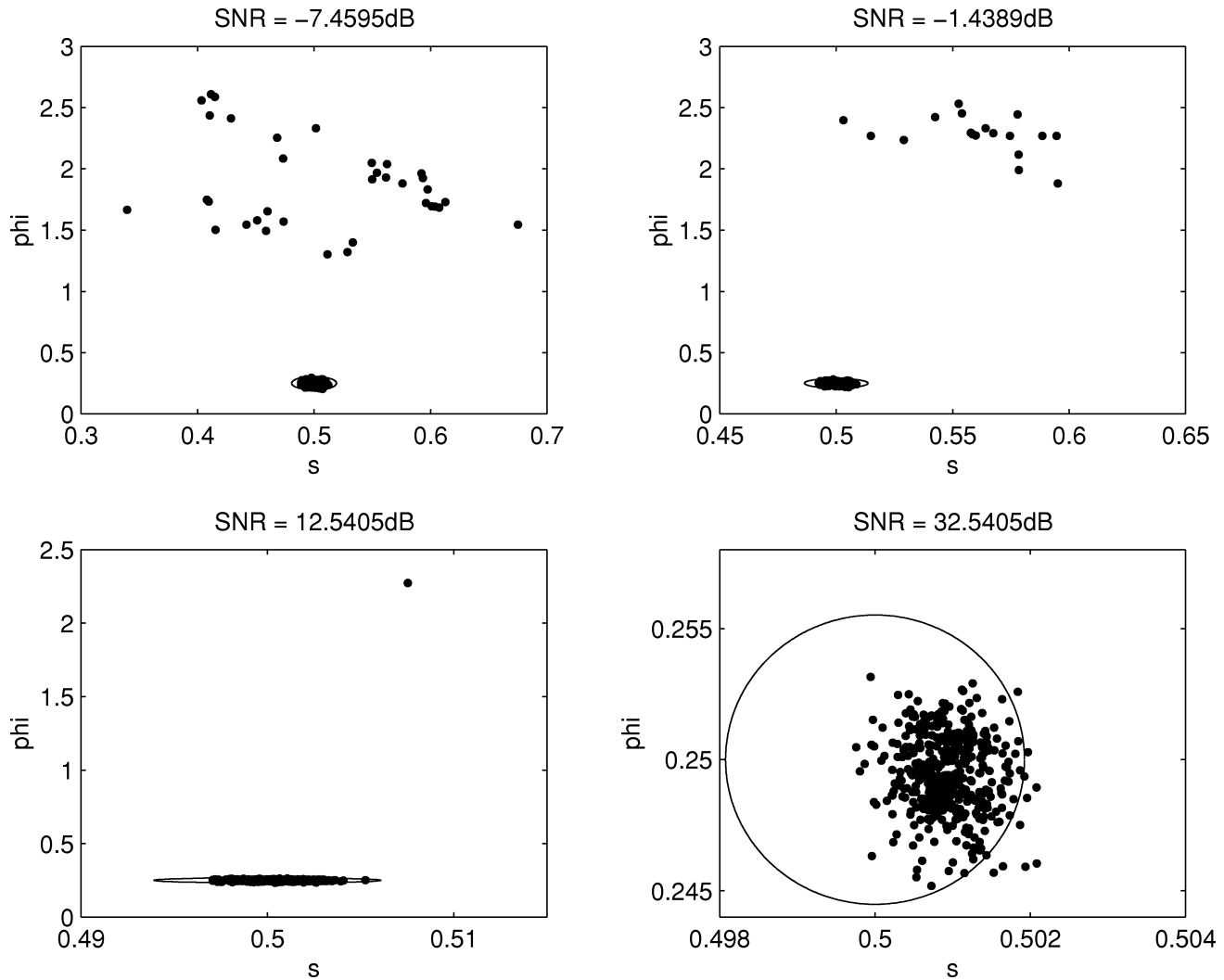


Fig. 6. Scatter plots of estimations from 400 simulations of the two-peak example. The ellipse is drawn at 3 times the bounds around the exact parameters, and is used to define “good” estimations.

the Jacobian with itself, and the second is given as the integral of the Hessian components with the error. Therefore,

$$\mathbf{U}_t(\boldsymbol{\theta}) = \int \mathbf{J}_t(\mathbf{r}; \boldsymbol{\theta}) \mathbf{J}_t^T(\mathbf{r}; \boldsymbol{\theta}) d\mathbf{r}. \quad (26)$$

Specifying the elements of $\mathbf{S}_t(\boldsymbol{\theta})$, by $s_{i,j}(\boldsymbol{\theta})$ and the elements of $\mathbf{H}_t(\mathbf{r}; \boldsymbol{\theta})$ by $h_{i,j}(\mathbf{r}; \boldsymbol{\theta})$ we have

$$s_{i,j}(\boldsymbol{\theta}) = \int h_{i,j}(\mathbf{r}; \boldsymbol{\theta}) e_t(\mathbf{r}; \boldsymbol{\theta}) d\mathbf{r}. \quad (27)$$

The Newton algorithm is then an iteration of (17) and (18) until the likelihood as evaluated by (10) ceases to change significantly. That is, while

$$J(\boldsymbol{\theta}^{(k)}; f_t) - J(\boldsymbol{\theta}^{(k-1)}; f_t) > \tau \quad (28)$$

for some small value of τ .

V. COMPUTING THE t -SCHEDULE

In the last section we presented the Newton algorithm which finds the minimum of the cost surface with respect to the parameters. The minimization is performed successively for many values of t . As explained earlier, the values of t must be chosen

to induce better behavior from the minimization. The better behavior is induced by smoothing, or flattening, the surface about the global minimum. The selection of the appropriate values of t has a direct affect upon the amount of computation and whether the algorithm will converge to a local minimum or the global minimum. It seems appropriate then to base the t -schedule on the expected value of the local curvature around the global minimum.

The expected local curvature at the minimum for any value of t can be approximated by the expected values of the elements of the Hessian matrix [20]. The elements of the Hessian matrix, with respect to the parameters, are given as

$$E \frac{\partial^2 l_t}{\partial \theta_i \partial \theta_j} = \frac{2}{\sigma_n^2} \int (f_t(\mathbf{r}; \boldsymbol{\theta}) - f_E(\mathbf{r})) \frac{\partial^2 f_t(\mathbf{r}; \boldsymbol{\theta})}{\partial \theta_i \partial \theta_j} \mathbf{r} + \frac{2}{\sigma_n^2} \int \frac{\partial f_t(\mathbf{r}; \boldsymbol{\theta})}{\partial \theta_i} \frac{\partial f_t(\mathbf{r}; \boldsymbol{\theta})}{\partial \theta_j} \mathbf{r}. \quad (29)$$

In Fig. 3, we plot the value of the local curvature with respect to t for the two-peaked target examined earlier. We see that the value of the curvature is small for small value of t when the template is a smooth approximate template. The curvature quickly increases as t is raised until it asymptotically approaches the

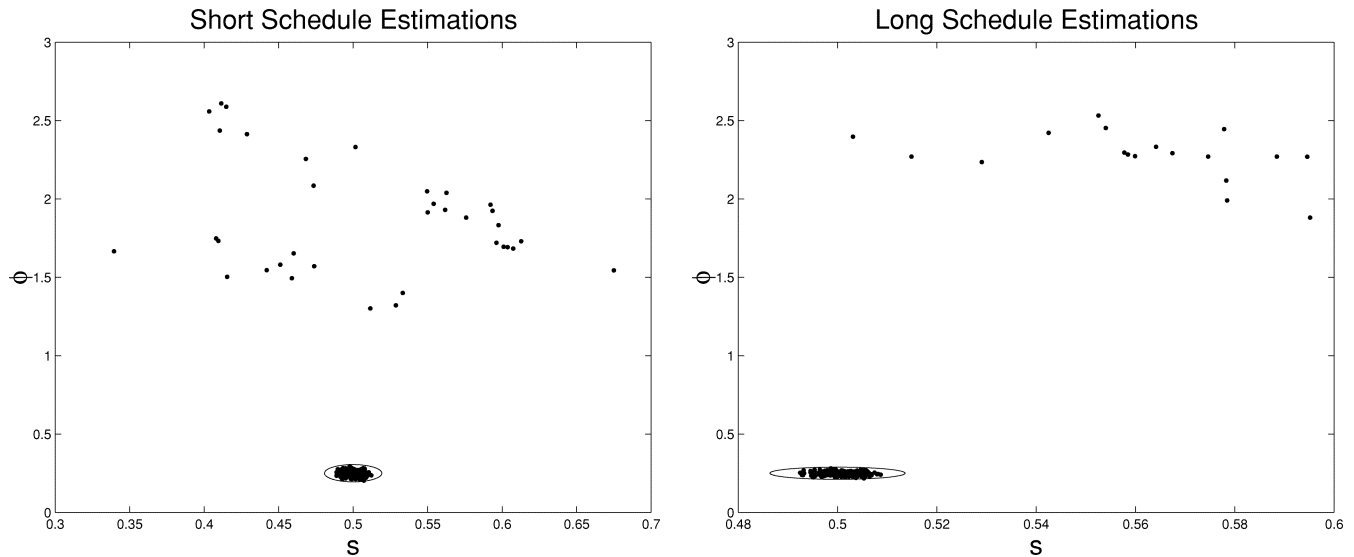


Fig. 7. Scatter plots of estimations from 400 simulations of the two-peak example for two schedules, longer schedules result in fewer misses.

TABLE II
ESTIMATION ERRORS FOR SIZE, ROTATION AND LOCATION

SNR (dB)	Schedule	Size		Rotation		\hat{r}_x		\hat{r}_y	
		Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
-7.4595	short	0.50538	0.020603	0.42739	0.57641	64.8575	3.4557	65.0825	2.5947
-1.4389	short	0.50357	0.013774	0.33951	0.42302	64.9575	2.2921	65.0325	1.5756
12.5405	short	0.50105	0.0025995	0.25451	0.10889	65.025	0.5	64.9825	0.35
18.5611	short	0.50097	0.00095061	0.24918	0.0034543	65	0	65	0
32.5405	short	0.50104	0.00013374	0.24942	0.00046999	65	0	65	0
38.5611	short	0.50105	0.0002985	0.24943	0.0010505	65	0	65	0
-7.4595	long	0.5013	0.024512	0.38909	0.4705	64.785	2.5238	64.905	1.6379
-1.4389	long	0.50221	0.015672	0.32243	0.34788	64.895	1.9425	65.0325	1.1245
12.5405	long	0.50084	0.0014079	0.25412	0.10135	65.03	0.6	64.985	0.3
18.5611	long	0.50085	0.00096719	0.24918	0.0035128	65	0	65	0
32.5405	long	0.50093	0.00043025	0.24944	0.0015144	65	0	65	0
38.5611	long	0.50092	0.00030425	0.24944	0.0010707	65	0	65	0

curvature of the exact template. For efficient and accurate estimation, it is important that we step through t quickly, however, if t increases too rapidly then we are likely to end up in a local minimum. At $t = 0$, the template is smoothest and the surface is the most well behaved and has the broadest well around the global minimum. The template must evolve in such a way that the estimate remains within the well. Using the curvature as a gauge of this well, we see that initially small changes in t are necessary so as not to too drastically change the surface. However as t increases larger steps can be taken since the surface evolves slower with respect to t . Using this knowledge, we have computed a geometric series for our t -schedules in the examples presented in this paper. This type of schedule is similar to that used in the Graduated Non-Convexity approach in [15]–[17].

VI. PERFORMANCE AND BOUNDS

Closely related to the curvature of the surface is the Cramer-Rao bound on the variance of the parameter estimation. The CRB for the estimates is arrived at by inverting the Fisher information matrix, which is computed by evaluating the Hessian as given above at $t = \infty$. The CRB establishes the lower

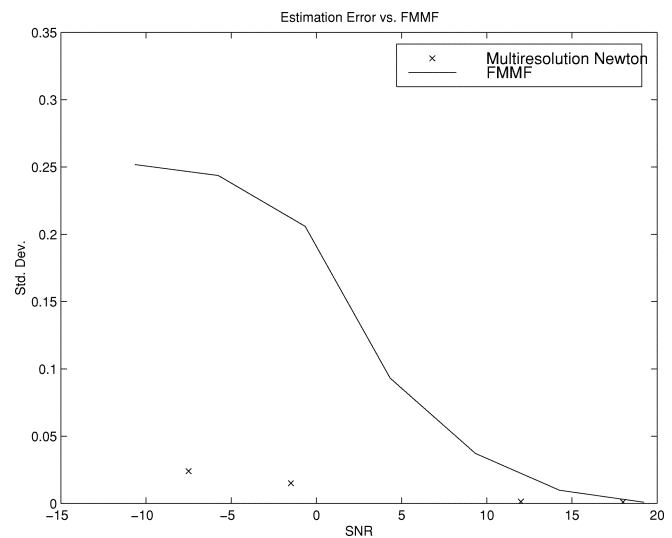


Fig. 8. Estimation error for location for 400 simulations using the Fourier Mellin Matched Filter and the current algorithm. The current algorithm performs significantly better at low SNR values.

limit on the variance of the estimates for an unbiased estimator. The CRB is only achievable for an unbiased estimator which

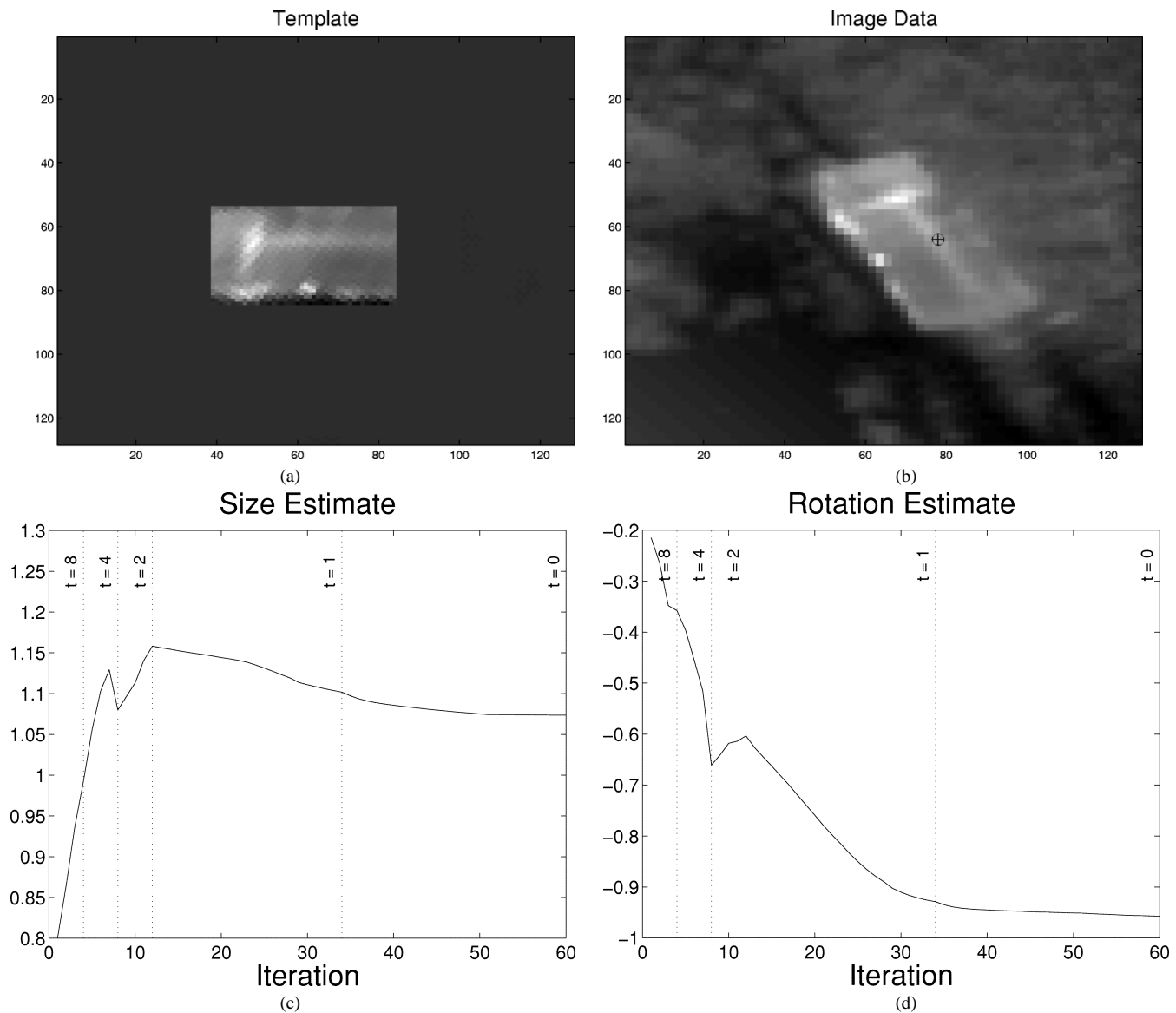


Fig. 9. Infrared target example. (a) The template, (b) the data with estimated center location, (c) the estimated size at each iteration, and (d) the estimated rotation at each iteration.

uses the sufficient statistic for the parameter to be estimated. If such a statistic is not available, then the CRB cannot be achieved. Nonetheless, it is useful to examine the lower limit to the variance of the estimates

$$\mathbf{I} = \begin{bmatrix} E \left(\frac{\partial^2 l}{\partial s^2} \right) & E \frac{\partial^2 l}{\partial s \partial \phi} & E \frac{\partial^2 l}{\partial s \partial r_x} & E \frac{\partial^2 l}{\partial s \partial r_y} \\ E \frac{\partial^2 l}{\partial s \partial \phi} & E \left(\frac{\partial^2 l}{\partial \phi^2} \right) & E \frac{\partial^2 l}{\partial \phi \partial r_x} & E \frac{\partial^2 l}{\partial \phi \partial r_y} \\ E \frac{\partial^2 l}{\partial s \partial r_x} & E \frac{\partial^2 l}{\partial \phi \partial r_x} & E \left(\frac{\partial^2 l}{\partial r_x^2} \right) & E \frac{\partial^2 l}{\partial r_x \partial r_y} \\ E \frac{\partial^2 l}{\partial s \partial r_y} & E \frac{\partial^2 l}{\partial \phi \partial r_y} & E \frac{\partial^2 l}{\partial r_x \partial r_y} & E \left(\frac{\partial^2 l}{\partial r_y^2} \right) \end{bmatrix}. \quad (30)$$

Then from the elements of the inverse of the information matrix we have the lower bounds upon the variance of the parameter estimations, as

$$\text{var } s \geq [\mathbf{I}^{-1}]_{1,1} \quad (31)$$

$$\text{var } \phi \geq [\mathbf{I}^{-1}]_{2,2} \quad (32)$$

$$\text{var } \mathbf{r} \geq [\mathbf{I}^{-1}]_{3,3} + [\mathbf{I}^{-1}]_{4,4}. \quad (33)$$

In Fig. 4 we see the CRB versus noise. These behave as expected with exact estimation possible in the no noise case and estimation performance degrading as noise increases.

VII. NUMERICAL EXPERIMENTS

In this section we present numerical experiments from the proposed algorithm and performance comparisons with the Fourier Mellin Matched Filter estimate [2]. We first present a synthetic target estimation example using a Monte Carlo simulation analysis. Then we show results of the proposed algorithm when applied to infrared (IR) and optical images.

Fig. 5(a) shows the two-peak target shown earlier buried in noise with 0 dB SNR. Fig. 5(b) and (c) show the estimates of s and ϕ at each iteration of the Newton algorithm. Here we used a t -schedule with four values, $\{0, 2.5, 5, 10\}$. We see here that the algorithm converged to close to the true parameters in 25 iterations. The true parameters were a size of 1.0 and a rotation angle of 0.25 the estimations converged to values of 0.9914 for size and 0.2354 for rotation.

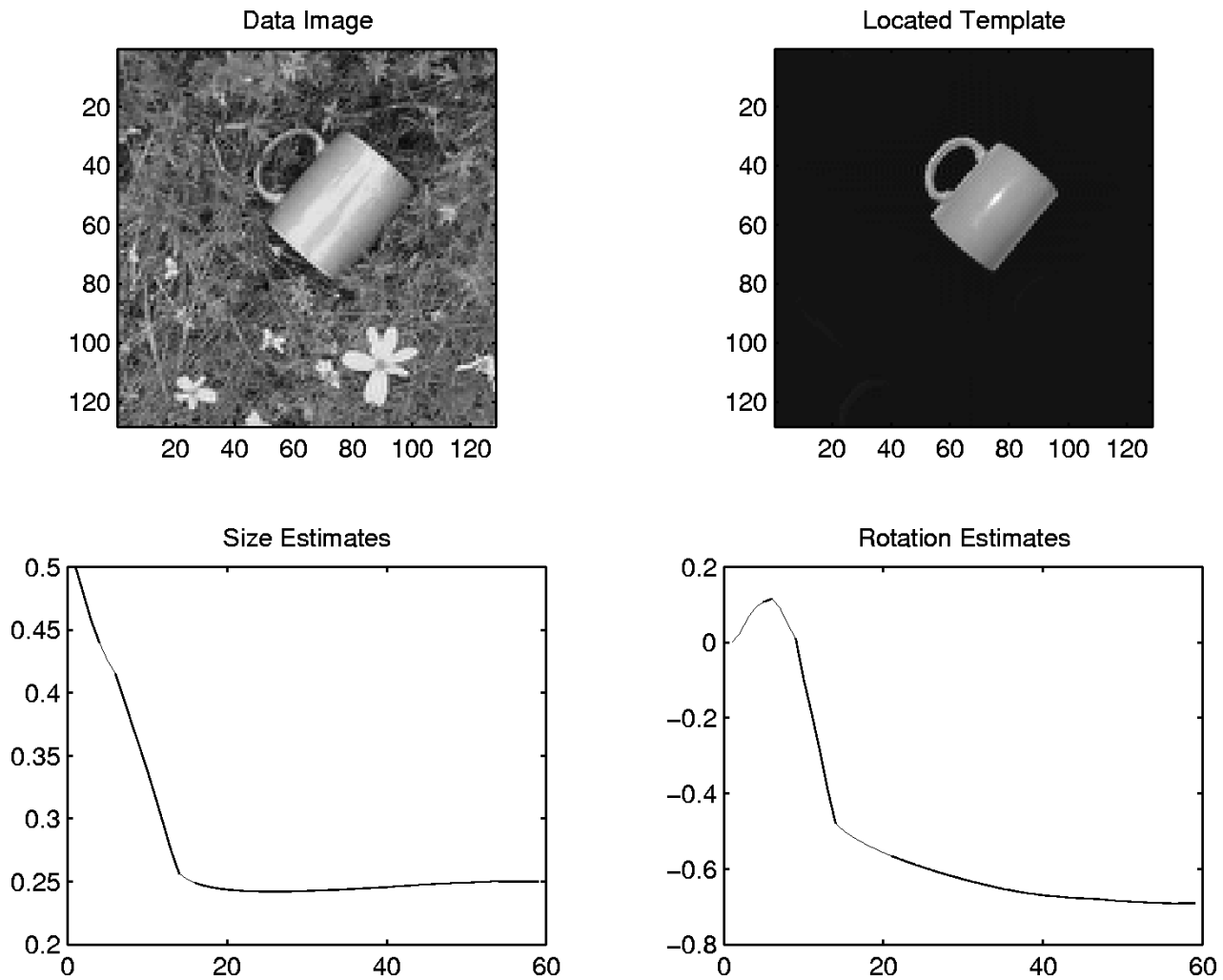


Fig. 10. Optical example of a cup in a cluttered scene. (a) The data, (b) the template at estimated size, rotation and location, (c) the estimated size at each iteration, and (d) the estimated rotation at each iteration.

In Fig. 6, we show scatter plots of estimates from 400 simulations of the two-peak example at varying signal to noise ratios. The true parameters are a size of 0.5 and a rotation of 0.25. The ellipses are used to show estimates which are “accurate”, that is are close to the true parameters. The center of the ellipse is at the true parameters and the ellipses are drawn at 3 times the CRB computed from the equations in Section VI. The interiors of these ellipses can be viewed as regions of sufficient confidence in which “accurate” estimates of the parameters fall. We observe that in (a)–(c), the number of estimates which are outside the ellipse, and are therefore considered misses, decreases. The number of misses increases for the highest SNR because of a bias in the size estimates which when combined with the tightness of the CRB causes us to register more misses at high SNR levels than may be warranted by the fact that these estimates are close to the true parameters.

Table II summarizes the Monte Carlo runs of the example just presented. Each line shows the result of 400 simulations of the algorithm for a specific level of noise and t -schedule. The error in the estimates is composed of two components, one caused by the variance of the local minimum around the true parameters, and the second caused by the the algorithm becoming trapped in a local minimum which is far from the

true parameters. The first type of error is bounded below by the CR bounds, and in practice is usually so small as to effect the target location estimation by less than one pixel. The second type of error constitutes a catastrophic miss by the algorithm. This error is demonstrated in Fig. 7 with a pair of Monte Carlo runs at two different t -schedules. From the first scatter plot, we see that the majority of estimates are grouped around the true parameters, but 35 of the estimates missed dramatically. These are instances where the algorithm became caught in a local minimum which was not the global minimum, and thus resulted in an enormous error. The second scatter plot shows the same Monte Carlo runs with a t -schedule which has twice as many steps and thus has twice the computational burden. Here, the number of misses was reduced from 35 to 18. By progressing through the t -schedule at a slower rate, we can reduce the chances of become trapped in a local minimum at the expense of more computations.

In Fig. 8 the performance for these Monte Carlo runs is compared with that of the Fourier Mellin Matched Filter [2]. The plot shows the standard deviation of the error in the location estimate versus SNR. We see that at high SNR, both methods perform well, but as the SNR decreases then error in the FMMF increases at a faster rate.

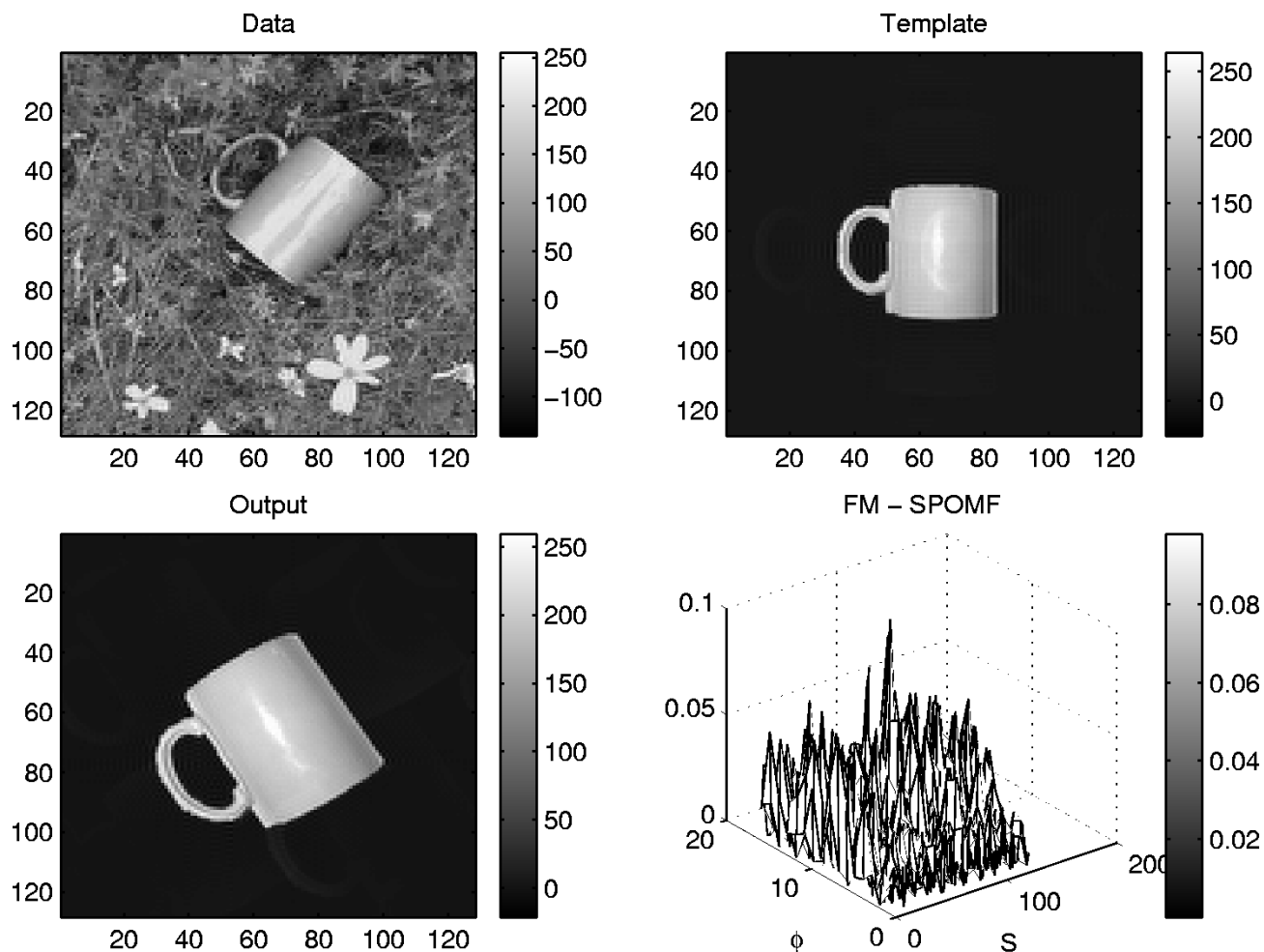


Fig. 11. FMMF of the cup in a cluttered scene. The FMMF fails to accurately estimate the parameters.

Fig. 9 shows an example of estimating the size, rotation and location of a vehicle in an IR image. Shown first is the target template which we are using followed by the data image. Below this is are the estimation values for size and rotation for each iteration. The iterations are divided into several sections by vertical dotted lines denoting the respective value of t for each stage of the algorithm. We can see that the algorithm converges in 60 iterations to values which upon matching the template to the data appear to be appropriate. Also shown is the position estimate of the target which also appears to be correct.

Fig. 10 shows an example with a real optical image. The first image shows a cup in a simple background which is close to white noise. The second image shows the estimated position of the template with proper size and rotation. We see here that the algorithm settled into this estimate after 42 iterations and that it is an accurate estimate. In Fig. 11, we show the output of the Fourier-Mellin matched filter which fails to isolate the true size and rotation for this image. The performance of the FMMF is degraded by the clutter in the scene. The FMMF operates across the entire image while the template matching is isolated to the support of the template.

The last topic of consideration is the ranges of scale and rotation over which this algorithm converges to the proper values. This would be highly dependent upon the shape of the template, the resolution to which template data is available

TABLE III
TOTAL MISSES OUT OF 20 MONTE CARLO SIMULATIONS OF ESTIMATING SEVERAL SCALES AND ROTATIONS. PROPER ESTIMATION WAS USUALLY ACHIEVED FOR SCALES DOWN TO 0.6, AND FOR ROTATIONS LESS THAN 30°

		ϕ^0			
		0°	30°	60°	90°
s^0	1.0	0	0	15	13
	0.8	0	0	15	14
	0.6	2	2	4	2
	0.4	3	3	2	2
	0.2	8	8	7	6

and the t -schedule used. For the two-peak example used in this paper, and the t -schedule discussed earlier, we ran Monte Carlo simulations of the convergence for values of scale of $\{0.2, 0.4, 0.6, 0.8, 1.0\}$ and values of rotation of $\{0, \pi/6, \pi/3, \pi/2\}$. The noise variance was set to produce a SNR of 10 dB across the support of the target. Table III summarizes the results of this experiment as the number of misses out of 20 simulations. The results show that the algorithm converges to the proper parameters across a wide range of values of the rotation and scale parameters. The algorithm converges to the correct parameter for scale values down to 0.6 and the correct rotation. For values of rotation higher than $\pi/3$, the algorithm often converged to a local minimum around $\pi/2$.

Interestingly, the performance at higher rotations improves for smaller scale objects. We believe that this is a result of the smoothing operation. Since the smoothing is constant, the relative amount of smoothing is greater for smaller objects. This results in an effectively finer t -schedule than the t -schedule for larger objects. We believe that if the t -schedule were made sufficiently fine, performance would improve for all areas of Table III. However, this is part of the larger issue of optimal t -scheduling. This topic is beyond the scope of the paper, but one which we intend to pursue in the future.

VIII. CONCLUSIONS AND FUTURE WORK

In this work, we have examined the problem of finding a target in a noisy image. Following in the work of Abu-Naser [1], we represent the problem as an image restoration problem where the object to be reconstructed is a delta function encoding the target location and the blur is a target template. Previous work showed that the restoration can be performed using a LLSE and choosing the location of maximum response [1]. However, this did not consider the problem of unknown geometric parameters, i.e., the size and rotation of the target within the image. To estimate size and rotation of the object, we choose to follow the previous approach for location estimation and minimize the negative of the likelihood function across size and rotation to find the true parameters.

This approach was not amenable to minimization, however, due to local minima and regions of zero derivative on the likelihood surface. To overcome these difficulties, we present a new method of generating a library of target templates which range from smooth monomodal approximations to the exact target. Using this library of templates, it is possible to sequentially estimate the parameter set using standard optimization tools.

Furthermore, since this method relies upon a functional description of the target template, we believe it to be extensible to more complicated target detection problems. Three-dimensional (perspective mappings of targets could be functionally described, allowing us to locate a target under a three-dimensional rotation. Also, since all targets degenerate to the same template at the extremely smooth scale and the algorithm returns a likelihood score as it converges toward the true parameters, we could perform target classification by allowing our library to branch toward several targets or target classes and pruning those which score poorly. Overall, we believe this algorithm could be extended for multiple target, multiple template recognition with reasonable computational costs.

REFERENCES

- [1] A. Abu-Naser, N. P. Galatsanos, M. N. Wernick, and D. Schonfeld, "Object recognition based on impulse restoration using the expectation maximization algorithm," *J. Opt. Soc. Amer. A: Opt. Image Sci.*, vol. 15, p. 2327, 1998.
- [2] Q. Chen, M. DeFrise, and F. Deconinck, "Symmetric phase-only matched filtering of fourier-mellin transforms for image registration and image recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 16, Dec. 1994.
- [3] H. Lester and S. R. Arridge, "Survey of hierarchical nonlinear medical image registration," *Pattern Recognit.*, vol. 32, no. 1, pp. 129–149, 1999.
- [4] J. Ben-Arie and K. R. Rao, "Non-orthogonal image expansion by restoration with applications to template matching," *Proc. SPIE*, vol. 1778, pp. 208–220, Mar. 1992.
- [5] K. R. Rao and J. Ben-Arie, "Restoration with equivalence to nonorthogonal image expansion for feature extraction and edge detection," in *Proc. SPIE Conf. Visual Communications and Image Processing*, Nov. 1992, pp. 187–197.
- [6] J. Ben-Arie and R. K. Rao, "Optimal template matching by nonorthogonal image expansion using restoration," *Int. J. Mach. Vis. Applicat.*, vol. 7, no. 2, pp. 69–81, Mar. 1994.
- [7] Z. Wang and J. Ben-Arie, "Model based segmentation and detection of affine transformed shapes in cluttered images," in *IEEE Int. Conf. Image Processing*, Oct. 1998.
- [8] I. Weiss, "Geometric invariants and object recognition," *Int. J. Comput. Vis.*, vol. 10, no. 3, pp. 207–231, 1993.
- [9] D. Cyganski and R. Vaz, "A linear signal decomposition approach to affine invariant contour identification," *Pattern Recognit.*, vol. 28, no. 12, pp. 1845–1853, 1995.
- [10] J. Ben-Arie and Z. Wang, "Pictorial recognition of objects employing affine invariance in the frequency domain," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 20, pp. 604–618, June 1998.
- [11] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, pp. 671–680, 1983.
- [12] E. Aarts and J. Korst, *Simulated Annealing and Boltzman Machines*. New York: Wiley, 1990.
- [13] M. Cooper, U. Grenander, M. Miller, and A. Srivastava, "Accommodating geometric and thermodynamic variability for forward-looking infrared sensors," in *Proc. Aerosense*, Apr. 1997.
- [14] A. Srivastava, "A bayesian approach to geometric subspace estimation," in *Proc. IEEE SSAP Workshop*, Sept. 1998.
- [15] B. Andrew and Z. Andrew, *Visual Reconstruction*. Cambridge, MA: MIT Press, 1987.
- [16] M. Nikolova, J. Idier, and A. Mohammad-Djafari, "Inversion of large-support ill-posed linear operators using a piecewise gaussian MRF," *IEEE Trans. Image Processing*, vol. 7, pp. 571–585, Apr. 1998.
- [17] M. Nikolova, "Markovian reconstruction using a GNC approach," *IEEE Trans. Image Processing*, vol. 8, pp. 1204–1220, Sept. 1999.
- [18] J. Ben-Arie and R. K. Rao, "A novel approach to template matching by nonorthogonal image expansion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, no. 1, pp. 71–84, 1993.
- [19] L. E. Scales, *Introduction to Non-Linear Optimization*. New York: Springer-Verlag, 1985.
- [20] J. J. Koenderink, *Solid Shape*. Cambridge, MA: MIT Press, 1990.

Roger M. Dufour received the B.S.E.E. in 1994 from Worcester Polytechnic Institute and the M.S.E.E. in 1997 from Northeastern University. He is pursuing the Ph.D. degree at Northeastern University.

He is currently employed at MIT Lincoln Laboratory, Lexington, MA. His research interests include statistical methods for target detection and parameter estimation.

Mr. Dufour is a member of Tau Beta Pi and Eta Kappa Nu.

Eric L. Miller (S'90–M'95) received the S.B. degree in 1990, the S.M. in 1992, and the Ph.D. degree in 1994 all in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge.

He is currently an Associate Professor in the Department of Electrical and Computer Engineering at Northeastern University. His research interests include the use of multiscale and statistical methods for the solution of inverse problems in general and inverse scattering problems in particular and the development of computationally efficient, physically-based models for use in applications such as mine detection, target recognition, medical imaging, and environmental monitoring and remediation.

Dr. Miller is a member of Tau Beta Pi, Eta Kappa Nu, and Phi Beta Kappa and received the CAREER Award from the National Science Foundation in 1996. He is currently serving as an Associate Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING.

Nikolas P. Galatsanos (SM'94) received the Diploma of electrical engineering from the National Technical University of Athens, Greece, in 1982. He received the M.S.E.E. and Ph.D. degrees from the Electrical and Computer Engineering Department of the University of Wisconsin-Madison in 1984 and 1989, respectively.

He has been on the faculty of the Electrical and Computer Engineering Department at the Illinois Institute of Technology since 1989. Presently, he is also on the faculty of the Computer Science Department at the University of Ioannina, Ioannina, Greece. His research interests center around image processing and statistical learning problems for medical imaging and visual communications applications. He coedited the book *Image Recovery Techniques for Image and Video Compression and Transmission* (Norwell, MA: Kluwer, 1998).

Dr. Galatsanos has served as an Associate Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING and the *IEEE Signal Processing Magazine*.