Blind Deconvolution Using Alternating Maximum a Posteriori Estimation with Heavy-Tailed Priors*

Jan Kotera[1,2], Filip Šroubek[1], and Peyman Milanfar[3]

[1] Institue of Information Theory and Automation,
Academy of Sciences of the Czech Republic, Prague, Czech Republic
{kotera,sroubekf}@utia.cas.cz
[2] Charles University in Prague, Faculty of Mathematics and Physics, Czech Republic
[3] University of California Santa Cruz, Electrical Engineering Department,
Santa Cruz CA 95064 USA
milanfar@ee.ucsc.edu

Abstract. Single image blind deconvolution aims to estimate the unknown blur from a single observed blurred image and recover the original sharp image. Such task is severely ill-posed and typical approaches involve some heuristic or other steps without clear mathematical explanation to arrive at an acceptable solution. We show that a straightforward maximum a posteriory estimation combined with very sparse priors and an efficient numerical method can produce results, which compete with much more complicated state-of-the-art methods.

Keywords: image blind deconvolution, blur estimation, heavy-tailed priors, augmented Lagrangian method.

1 Introduction

Single channel blind deconvolution amounts to estimating an image u from a single observed image g satisfying a convolutional degradation model

$$g = u * h + n, \tag{1}$$

where h, called point spread function (PSF), is unknown and n is random additive noise. Since we have only one observation (Single-Channel) and no knowledge of the PSF, the problem is extremely ill-posed. One way to tackle this problem is to assume a parametric model of the PSF and search in the space of parameters and not in the full space of PSFs. Chang *et al.* in [2] investigated zero patterns in the Fourier transform or in the cepstrum, and assumed only parametric motion or out-of-focus blurs. More low-level parametric methods for estimating general motion blurs were proposed in [14,16]. Unfortunately

* This work was supported by GA UK under grant 938213, by GACR under grant 13-29225S, and by AVCR under grant M100751201.

R. Wilson et al. (Eds.): CAIP 2013, Part II, LNCS 8048, pp. 59–66, 2013.

real PSFs seldom follow parametric models and this prevents the parametric methods from finding the exact solution.

There has been a considerable effort in the image processing community in the last three decades to find a reliable algorithm for SC blind deconvolution with general (non-parametric) PSFs. First algorithms appeared in telecommunication and signal processing in early 80's [6]. For a long time, the problem seemed too difficult to be solvable for arbitrary blur kernels. Proposed algorithms usually worked only for special cases, such as symmetric PSFs or astronomical images with uniform black background, see [1].

Over the last few years, SC blind deconvolution based on the Bayesian paradigm experiences a renaissance. In probabilistic point of view, simultaneous recovery of u and h amounts to solving standard MAP (Maximum A Posteriori) estimation

$$P(u,h|g) \propto P(g|u,h)P(u,h) = P(g|u,h)P(u)P(h)$$

where $P(g|u,h) \propto \exp(-\frac{\gamma}{2}\|u * h - g\|^2)$ is the noise distribution (in this case assumed Gaussian) and $P(u)$, $P(h)$ are the prior distributions on the latent image and blur kernel, respectively. The key idea of new algorithms is to address the ill-posedness of blind deconvolution by characterizing the prior $P(u)$ using natural image statistics and by a better choice of estimators.

Levin *et al.* in [10,9] claim that a proper estimator matters more than the shape of priors. They showed that marginalizing the posterior with respect to the latent image u leads to the correct solution of the PSF h. The marginalized probability $P(h|g)$ can be expressed in a closed form only for simple priors that are, e.g., Gaussian. Otherwise approximation methods such as variational Bayes [11] or the Laplace approximation [5] must be used. A frantic activity in this area started with the work of Fergus *et al.* [4], who applied variational Bayes to approximate the posterior $P(u,h|g)$ by a simpler distribution $q(u,h) = q(u)q(h)$. Other authors [7,8,13,15] stick to the "good old" alternating MAP approach, but by using ad hoc steps, which often lack rigorous explanation, they converge to the correct solution.

The main contribution of our paper is to show that a simple alternating MAP approach without any ad hoc steps results in an efficient blind deconvolution algorithm that outperforms sophisticated state-of-the-art methods. The novelty is to use image priors $P(u)$ that are more heavy-tailed than Laplace distribution and apply a method of augmented Lagrangian to tackle this non-convex optimization problem.

In the next section we define the energy function of u and h that we want to minimize. Sec. 3 provides a detailed description of the optimization algorithm and the final experimental section illustrates algorithm's performance.

2 Mathematical Model

Let us assume that the variables in (1) are discrete quantities (vectors) with indexing denoted as u_i or $[u]_i$. Maximization of the posterior $P(u,h|g)$ is equivalent to minimization of its negative logarithm, i.e.,

$$L(u,h) = -\log(P(u,h|g)) + const = \frac{\gamma}{2}\|u * h - g\|_2^2 + Q(u) + R(h) + const, \quad (2)$$

where $Q(u) = -\log P(u)$ and $R(h) = -\log P(h)$ can be regarded as regularizers that steer the optimization to the right solution and away from infinite number of trivial or other unwanted solutions. A typical prior on u allows only few nonzero resulting coefficients of some linear or nonlinear image transform. The most popular choice is probably the l_1 norm of the image derivatives, either directionally separable $Q(u) = \sum_i |[D_x u]_i| + |[D_y u]_i|$ (this corresponds to the Laplace distribution of image derivatives) or isotropic (in terms of image gradient) $Q(u) = \sum_i \sqrt{[D_x u]_i^2 + [D_y u]_i^2}$, where D_x and D_y are partial derivative operators. The prior on h depends on the task at hand, for motion blurs it again favors sparsity and, in addition, disallows negative values.

It has been reported (e.g. [10]) that the distribution of gradients of natural images is even more heavy-tailed than Laplace distribution, we therefore use a generalized version of $Q(u)$ defined as

$$Q(u) = \Phi(D_x u, D_y u) = \sum_i \left([D_x u]_i^2 + [D_y u]_i^2\right)^{\frac{p}{2}}, \quad 0 \le p \le 1.$$

For the blur kernel we use Laplace distribution on the positive kernel values to force sparsity and zero on the negative values. This results in the following regularizer R:

$$R(h) = \sum_i \Psi(h_i), \quad \Psi(h_i) = \begin{cases} h_i & h_i \ge 0 \\ +\infty & h_i < 0. \end{cases}$$

3 Optimization Algorithm

In order to numerically find the solution u, h, we alternately minimize the functional L in (2) with respect to either u or h while keeping the other constant, this allows for easy minimization of the joint data fitting term. In each minimization subproblem we use the augmented Lagrangian method (ALM) (see e.g. [12, Chap. 17]), let us describe the procedure in detail.

3.1 Minimization with Respect to u

We wish to solve

$$\min_u \frac{\gamma}{2}\|Hu - g\|^2 + \Phi(D_x u, D_y u),$$

where H denotes a (fixed) convolution operator constructed from the h estimate from the previous iteration. This problem is equivalent to introducing new variables $v_x = D_x u$, $v_y = D_y u$ and solving

$$\min_{u,v} \frac{\gamma}{2}\|Hu - g\|^2 + \Phi(v_x, v_y) \quad \text{s.t. } v_x = D_x u,\ v_y = D_y u.$$

ALM adds quadratic penalty term for each constraint to the traditional Lagrangian, which (after some reshuffling) results in the functional

$$L_u(u, v_x, v_y) = \frac{\gamma}{2}\|Hu-g\|^2+\Phi(v_x, v_y)+\frac{\alpha}{2}\|D_x u-v_x-a_x\|^2+\frac{\alpha}{2}\|D_y u-v_y-a_y\|^2,$$

where the new variables a_x, a_y are proportional to the estimates of the Lagrange multipliers of the corresponding constraints. After such reformulation, the data term $\|Hu-g\|^2$ and the regularizer $\Phi(v_x, v_y)$ can be minimized separately since they depend on different variables. By introducing penalty terms, ALM allows us to treat the constrained variables $D_x u$ and v_x (similarly $D_y u$ and v_y) as though they were unrelated and by keeping the penalty weight α sufficiently large, we will obtain the solution to the original problem [3, Thm. 8].

We solve the minimization of L_u via coordinate descent in the u, v_x, v_y "direction" alternately. That is, we compute derivative with respect to one variable while keeping others fixed, solve it for minimum and update that variable accordingly, then move on to the next variable and so on for sufficiently many iterations. Let us state the whole process at once and explain the individual steps afterwards.

1: Set $v_x^0 := 0, v_y^0 := 0, a_x^0 := 0, a_y^0 := 0$, and $j := 0$
2: **repeat**
3: Solve $(H^T H+\frac{\alpha}{\gamma}(D_x^T D_x+D_y^T D_y))u^{j+1} = H^T g+\frac{\alpha}{\gamma}(D_x^T(v_x^j+a_x^j)+D_y^T(v_y^j+a_y^j))$ for u^{j+1}
4: $\{[v_x^{j+1}]_i, [v_y^{j+1}]_i\} := \mathrm{LUT}_p([D_x u^{j+1} - a_x^j]_i, [D_y u^{j+1} - a_y^j]_i), \quad \forall i$
5: $a_x^{j+1} := a_x^j - D_x u^{j+1} + v_x^{j+1}$
6: $a_y^{j+1} := a_y^j - D_y u^{j+1} + v_y^{j+1}$
7: $j := j + 1$
8: **until** stopping criterion is satisfied
9: **return** u^j

After differentiating L_u w.r.t. u and setting the derivative to zero, we must solve the linear system on line 3 for u. If we treat the $u*h$ convolution as circular, then H and consequently the whole matrix on the left-hand side is block-circulant and can be digonalized by 2D Fourier transform. Thus, the solution u can be computed directly and only at the cost of Fourier transform.

Minimization of L_u w.r.t. v_x, v_y on line 4 is trickier. If we disregard terms not depending on v_x, v_y, we get $\Phi(v_x, v_y) + \frac{\alpha}{2}\|D_x u - v_x - a_x\|^2 + \frac{\alpha}{2}\|D_y u - v_y - a_y\|^2$, where all three terms are summations of simpler terms over all image pixels. Derivatives and minimzation can be therefore carried out pixel by pixel independently. Let i be fixed pixel index. Let $t = ([v_x]_i, [v_y]_i)$ and $r = (D_x u_i - [a_x]_i, D_y u_i - [a_y]_i)$, then the problem of minimizing L_u w.r.t. $[v_x]_i, [v_y]_i$ can be rewritten as

$$\min_t \|t\|^p + \frac{\alpha}{2}\|t - r\|^2. \tag{3}$$

For some p a closed form solution can be computed. After simple calculation it can be seen that for the common choice of $p = 1$, minimization of (3) results

in vector soft thresholding $t = \frac{r}{\|r\|} \max\left(\|r\| - \frac{1}{\alpha}, 0\right)$. Similarly, for the binary penalty $p = 0$ we get hard thresholding with threshold $\sqrt{2/\alpha}$. For the general case $0 < p < 1$, no closed form solution exists, but becasue p is known beforehand and (3) is basically 1D minimization, it can be precomputed numerically and used in the minimization of L_u w.r.t. v_x, v_y in the form of lookup table (LUT), which is then used independently for each ith component of v_x, v_y.

Update equations for a_x, a_y on lines 5 and 6 are reminiscent of simple gradient descent but actually originate from the ALM theory, [12].

3.2 Minimization with Respect to h

Minimizing with respect to h can be done in similar fashion. To separate the minimization of data term and regularizer, we again make the substitution $v_h = h$, which yields the following optimization problem

$$\min_{h,v_h} \frac{\gamma}{2}\|Uh - g\|^2 + R(v_h) \quad \text{s.t. } h = v_h,$$

where U is (fixed) convolutional operator constructed from u. Applying ALM again results in the functional

$$L_h(h, v_h) = \frac{\gamma}{2}\|Uh - g\|^2 + R(v_h) + \frac{\beta}{2}\|h - v_h - a_h\|^2,$$

where a_h is again related to ALM method and is proportional to the Lagrange multiplier of the prescribed constraint. This functional can be minimized by the following coordinate descent algorithm:

1: Set $v_h^0 := 0, a_h^0 := 0$, and $j := 0$
2: **repeat**
3: Solve $(U^T U + \frac{\beta}{\gamma} I)h^{j+1} = U^T g + \frac{\beta}{\gamma}(v_h^j + a_h^j)$ for h^{j+1}
4: $[v_h^{j+1}]_i := \max([h^{j+1} - a^j]_i - \frac{1}{\beta}, 0), \quad \forall i$
5: $a_h^{j+1} := a_h^j - h^{j+1} + v_h^{j+1}$
6: $j := j + 1$
7: **until** stopping criterion is satisfied
8: **return** h^j

As in the previous case, the linear system on line 3, originating from differentiating L_h w.r.t. h, can be diagonalized by 2D Fourier transform and therefore solved directly. I denotes identity matrix.

Minimization w.r.t. v_h can be again done component-wise. Let i be a pixel index, $t = [v_h]_i$, $r = [h - a_h]_i$, then the problem on line 4 can be rewritten as $\min_t \frac{\beta}{2}(r - t)^2 + \psi(t)$, which is basically scalar version of (3) for $p = 1$ with the additional constraint that only positive values of t are allowed. The solution is thus component-wise soft thresholding as specified on line 4. Line 5 originates again from the ALM theory.

3.3 Implementation Details

To avoid getting trapped in a local minimum, we estimate the PSF in the multiscale fashion. The input image g is downsampled such that the estimated PSF at this scale is small (3×3 pixels or similar), then we upsample such estimated PSF (with factor 2) and use this as the initial point of the next level estimation. This procedure is repeated until the target PSF size is reached.

The no-blur solution is favored by blind deconvolution algorithms based on MAP. It is thus advantageous to exaggerate some parameters to push the optimization away from this trivial solution. We have discovered that setting the parametr γ lower than its correct value (as it corresponds to the observed image noise level) and slowly increasing it during the optimization helps the PSF estimation. Also, we set the sparsity parameter p to much lower value than would be expected for natural images and only after estimating the PSF we run the u estimation one last time with p and γ set to realistic values.

For our experiments we use for the PSF estimation $\gamma = 1$, $\alpha = 1$, $\beta = 10^4$, $p = 0.3$ and we multiply the γ by 1.5 after each pass of the u-estimation and h-estimation pair. For the final nonblind deconvolution, we use $\gamma = 10, p = 1$.

4 Experimental Results

We tested our algorithm on the dataset provided by [10] consisting of four grayscale images and eight PSFs of true motion blur, resulting in 32 test images. We compare our method to the method of [15], which is arguably currently the best performing single-channel blind deconvolution method, and the method of [4], which frequently appears in comparisons of blind deconvolution methods. In our comparison, we focus on the accuracy assesment of the estimated PSF, which we measure by the MSE of the (registered) estimated PSF to the ground truth.

Fig. 2 shows the result of kernel estimation measured as MSE from the ground truth kernel. We see that in most cases our method is superior. Fig. 3 shows the estimeted PSFs for the first input image. The remaining 24 estimates look

Fig. 1. The dataset of [10]. First row contains sharp images, second row measured motion blur PSFs.

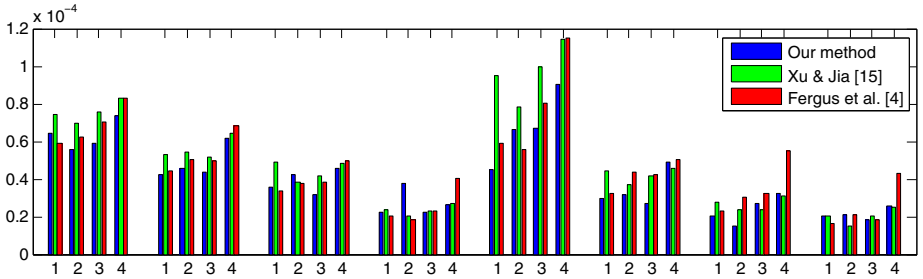

Fig. 2. MSE of estimated kernels (low values mean better performance) in the 32 test examples, grouped by PSFs. Numbers on x-axis indicate image index.

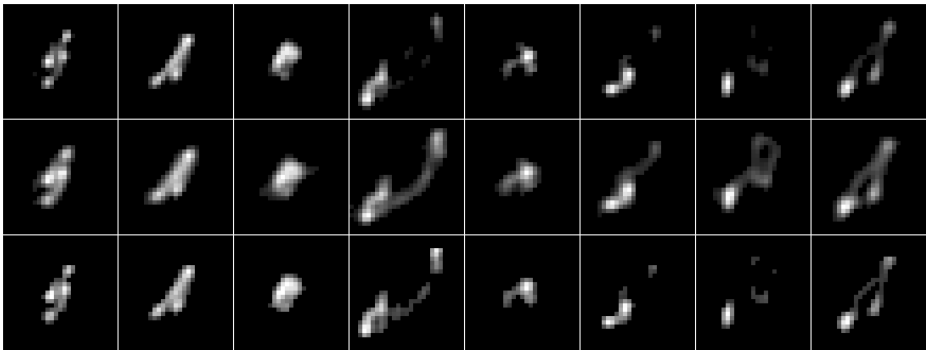

Fig. 3. Estimated PSFs, image 1. Rows from top to bottom: our method, method of [15], method of [4]. Compare with ground truth in Fig. 1.

Fig. 4. Deconvolution of true motion blur. Left: captured image, right: deconvolved result (estimated PSF superimposed).

similar. All methods perform very well but it can be seen that our method produces slightly more accurate results. The last experiment in Fig. 4 shows the deconvolution result of severly motion-blurred photo captured by handheld camera, the improvement in quality is evident.

References

1. Chan, T.F., Wong, C.K.: Total variation blind deconvolution. IEEE Trans. Image Processing 7(3), 370–375 (1998)
2. Chang, M.M., Tekalp, A.M., Erdem, A.T.: Blur identification using the bispectrum. IEEE Trans. Signal Processing 39(10), 2323–2325 (1991)
3. Eckstein, J., Bertsekas, D.P.: On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. Math. Program. 55(3), 293–318 (1992)
4. Fergus, R., Singh, B., Hertzmann, A., Roweis, S.T., Freeman, W.T.: Removing camera shake from a single photograph. In: SIGGRAPH 2006: ACM SIGGRAPH 2006 Papers, pp. 787–794. ACM, New York (2006)
5. Galatsanos, N.P., Mesarovic, V.Z., Molina, R., Katsaggelos, A.K.: Hierarchical Bayesian image restoration from partially known blurs. IEEE Transactions on Image Processing 9(10), 1784–1797 (2000)
6. Godard, D.: Self-recovering equalization and carrier tracking in two-dimensional data communication systems. IEEE Transactions on Communications 28(11), 1867–1875 (1980)
7. Jia, J.: Single image motion deblurring using transparency. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition CVPR 2007, June 17–22, pp. 1–8 (2007)
8. Joshi, N., Szeliski, R., Kriegman, D.J.: PSF estimation using sharp edge prediction. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition CVPR 2008, June 23–28, pp. 1–8 (2008)
9. Levin, A., Weiss, Y., Durand, F., Freeman, W.T.: Understanding blind deconvolution algorithms. IEEE Transactions on Pattern Analysis and Machine Intelligence 33(12), 2354–2367 (2011)
10. Levin, A., Weiss, Y., Durand, F., Freeman, W.T.: Understanding and evaluating blind deconvolution algorithms. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009, pp. 1964–1971 (2009)
11. Miskin, J., MacKay, D.J.C.: Ensemble learning for blind image separation and deconvolution. In: Girolani, M. (ed.) Advances in Independent Component Analysis, pp. 123–142. Springer (2000)
12. Nocedal, J., Wright, S.: Numerical Optimization. Springer Series in Operations Research. Springer (2006)
13. Shan, Q., Jia, J., Agarwala, A.: High-quality motion deblurring from a single image. In: SIGGRAPH 2008: ACM SIGGRAPH 2008 Papers, pp. 1–10. ACM, New York (2008)
14. Shan, Q., Xiong, W., Jia, J.: Rotational motion deblurring of a rigid object from a single image. In: Proc. IEEE 11th International Conference on Computer Vision ICCV 2007, October 14–21, pp. 1–8 (2007)
15. Xu, L., Jia, J.: Two-phase kernel estimation for robust motion deblurring. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 157–170. Springer, Heidelberg (2010)
16. Yitzhaky, Y., Kopeika, N.S.: Identification of blur parameters from motion blurred images. Graphical Models and Image Processing 59(5), 310–320 (1997)