User assisted separation of reflections from a single image using a sparsity prior

Anat Levin and Yair Weiss

School of Computer Science and Engineering
The Hebrew University of Jerusalem
91904 Jerusalem, Israel
{*alevin,yweiss*}*@cs.huji.ac.il*

Abstract. When we take a picture through transparent glass the image we obtain is often a linear superposition of two images: the image of the scene beyond the glass plus the image of the scene reflected by the glass. Decomposing the single input image into two images is a massively ill-posed problem: in the absence of additional knowledge about the scene being viewed there are an infinite number of valid decompositions. In this paper we focus on an easier problem: user assisted separation in which the user interactively labels a small number of gradients as belonging to one of the layers.

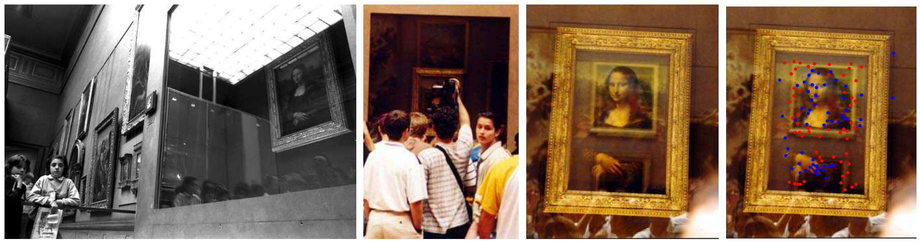
Even given labels on part of the gradients, the problem is still ill-posed and additional prior knowledge is needed. Following recent results on the statistics of natural images we use a sparsity prior over derivative filters. We first approximate this sparse prior with a Laplacian prior and obtain a simple, convex optimization problem. We then use the solution with the Laplacian prior as an initialization for a simple, iterative optimization for the sparsity prior. Our results show that using a prior derived from the statistics of natural images gives a far superior performance compared to a Gaussian prior and it enables good separations from a small number of labeled gradients.

1 Introduction

Figure 1(a) shows the room in which Leonardo's Mona Lisa is displayed at the Louvre. In order to protect the painting, the museum displays it behind a transparent glass. While this enables viewing of the painting, it poses a problem for the many tourists who want to photograph the painting (see figure 1(b)). Figure 1(c) shows a typical picture taken by a tourist[1] : the wall across from the painting is reflected by the glass and the picture captures this reflection superimposed on the Mona-Lisa image.

A similar problem occurs in various similar settings: photographing window dressings, jewels and archaeological items protected by glass. Professional photographers attempt to solve this problem by using a polarizing lens. By rotating

[1] All three images are taken from www.studiolo.org/Mona/MONA09.htm



(a) (b) (c) (d)

Fig. 1. (a),(b) The scene near the Mona Lisa in the Louvre. The painting is housed behind glass to protect it from the many tourists. (c) A photograph taken by a tourist at the Louvre. The photograph captures the painting as well as the reflection of the wall across the room. (d) The user assisted reflection problem. We assume the user has manually marked gradients as belonging to the painting layer or the reflection layer and wish to recover the two layers.

the polarizing lens appropriately, one can reduce (but not eliminate) the reflection. As suggested in [2, 8] the separation can be improved by capturing two images with two different rotations of the polarizing lens and taking an optimal linear combination of the two images. An alternative solution is to use *multiple* input images [11, 4] in which the reflection and the non-reflected images have different motions. By analyzing the movie sequence, the two layers can be recovered. In [13], a similar approach is applied to stereo pairs.

While the approaches based on polarizing lenses or stereo images may be useful for professional photographers, they seem less appealing for a consumer-level application. Viewing the image in figure 1(c), it seems that the information for the separation is present in a single image. Can we use computer vision to separate the reflections from a single image ?

Mathematically, the problem is massively ill-posed. The input image $I(x, y)$ is a linear combination of two unknown images $I_1(x, y), I_2(x, y)$:

$$I(x, y) = I_1(x, y) + I_2(x, y) \tag{1}$$

Obviously, there are an infinite number of solutions to equation 1: the number of unknowns is twice the number of equations. Additional assumptions are needed. On the related problem of separating shading and reflectance, impressive results have been obtained using a single image [12, 3]. These approaches make use of the fact that edges due to shading and edges due to reflectance have different statistics (e.g. shading edges tend to be monochromatic). Unfortunately, in the case of reflections, the two layers have the same statistics, so the approaches used for shading and reflectance are not directly applicable. In [5], a method was presented that used a prior on images to separate reflections with no user intervention. While impressive results were shown on simple images, the technique used a complicated optimization that often failed to converge on complex images.

In this paper, we present a technique that works on arbitrarily complex images but we simplify the problem by allowing user assistance. We allow the user to *manually* mark certain edges (or areas) in the image as belonging to one of the two layers. Figure 1(d) shows the Mona Lisa image with manually marked gradients: blue gradients are marked as belonging to the Mona Lisa layer and red are marked as belonging to the reflection layer. The user can either label individual gradients or draw a polygon to indicate that all gradients inside the polygon belong to one of the layers. This kind of user assistance seems quite natural in the application we are considering: imagine a Photoshop plugin that a tourist can use to post-process the images taken with reflections. As long as the user needs only to mark a small number of edges, this seems a small price to pay.

Even when the user marks a small number of edges, the problem is still ill-posed. Consider an image with a million pixels and assume the user marks a hundred edges. Each marked edge gives an additional constraint for the problem in equation 1. However, with these additional equations, the total number of equations is a only million and a hundred, far less than the two million unknowns. Unless the user marks every single edge in the image, additional prior knowledge is needed.

Following recent studies on the statistics of natural scenes [7, 9], we use a prior on images that is based on the sparsity of derivative filters. We first approximate this prior with a Laplacian prior and this approximation enables us to find the most likely decomposition using *convex* optimization. We then use the Laplacian prior solution as an initial guess for a simple, iterative optimization of the sparsity prior. We show that by using a prior derived from the statistics of natural scenes, one can obtain excellent separations using a small number of labeled gradients.

2 Statistics of natural images

A remarkably robust property of natural images that has received much attention lately is the fact that when derivative filters are applied to natural images, the filter outputs tend to be sparse [7, 9, 17]. Figure 2(a-d) illustrates this fact: the histogram of the vertical derivative filter is peaked at zero and fall off much faster than a Gaussian. These distributions are often called "sparse" and there are a number of ways to formulate this property mathematically , (e.g. in terms of their tails or their kurtosis).

We will follow Mallat [6] and Simoncelli [10] in characterizing these distributions in terms of the shape of their logarithm. As shown in figure 2(b,d), when we look at the logarithm of the histogram the curve is always below the straight line connecting the maximum and minimum values. This should be contrasted with the Gaussian distribution (that is always above the straight line) or the Laplacian distribution (that is simply a straight line in the log domain) (figure 2(e)). In [5] it was shown that the fact that the log distribution is always below the straight line, is crucial for obtaining transparency decompositions from a single

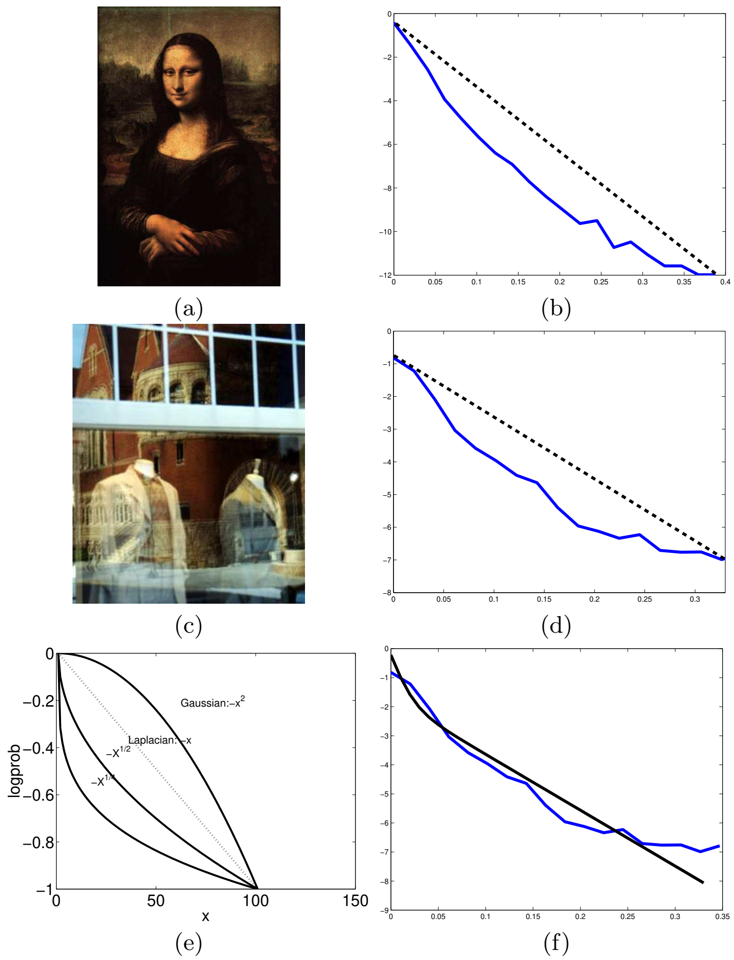


Fig. 2. (a),(c) input images. (b),(d) log-histogram of d_y derivative. A robust property of natural images is that the log-histograms of derivative filters lie below the straight line connecting the minimal and maximal values. We refer to such distributions as sparse (e) Log probabilities for distributions of the form $e^{-x^{\alpha}}$. The Gaussian distribution is not sparse (it is always above the straight line) and distributions for which $\alpha < 1$ are sparse. The Laplacian distribution is exactly at the border between sparse and non sparse distributions. (f) Matching a mixture model to a filter output histogram. The mixture parameters were selected to maximize the likelihood of the histogram. A mixture of Laplacians is sparse even though the individual components are not.

image. Distributions that are above the straight line will prefer to split an edge of unit contrast into two edges (one in each layer) with half the contrast, while distributions below the line will prefer decompositions in which the edge only appears in one of the layers but not in the other. We will refer to distributions that have this property in the log domain as being sparse.

Wainwright and Simoncelli [15] have suggested describing the histograms of natural images with an infinite Gaussian mixture model. By adding many Gaussians, each with a mean at zero but with different variances one can obtain sparse distributions. This can also be achieved by mixing only two distributions: a narrow distribution centered on zero and a broad distribution centered on zero will give a sparse distribution. Figure 2(f) shows a mixture of two Laplacian distributions:

$$\Pr(x) = \frac{\pi_1}{2s_1} e^{-|x|/s_1} + \frac{\pi_2}{2s_2} e^{-|x|/s_2} \tag{2}$$

Although the Laplacian distributions are not sparse based on our definition, the mixture is. For the experiments in this paper, the mixture parameters were learned from real images. That is, the parameters were selected to maximize the likelihood of the histogram of derivative filters, as in Figure 2(f). The learned values we found are $s_1 = 0.01, s_2 = 0.05, \pi_1 = 0.4, \pi_2 = 0.6$.

Given the histograms over derivative filters, we follow [16] in using it to define a distribution over images by assuming that derivative filters are independent over space and orientation so that our prior over images is given by:

$$\Pr(I) = \prod_{i,k} \Pr(f_{i,k} \cdot I) \tag{3}$$

where $f \cdot I$ denotes the inner product between a linear filter f and an image I, and $f_{i,k}$ is the k'th derivative filter centered on pixel i. The derivative filters set we use includes two orientations (horizontal and vertical) and two degrees (i.e. first derivative filters as well as second derivative). The probability of a single derivative is given by equation 2.

Equation 3 gives the probability of a single layer. We follow [5] in defining the probability of a decomposition I^1, I^2 as the product of the probabilities of each layer (i.e. assuming the two layers are independent).

3 Optimization

We are now ready to state the problem formally. We are given an input image I and two sets of image locations S_1, S_2 so that gradients in location S_1 belong to layer 1 and gradients in location S_2 belong to layer 2. We wish to find two layers I_1, I_2 such that:

1. the two layers sum to form the input image $I = I_1 + I_2$
2. the gradients of I_1 at all locations in S_1 agree with the gradients of the input image I and similarly the gradients of I_2 at all locations in S_2 agree with the gradients of I.

Subject to these two constraints we wish to maximize the probability of the layers $\Pr(I^1, I^2) = \Pr(I^1)\Pr(I^2)$ given by equation 3.

Our approximation proceeds in two steps. We first approximate the sparse distribution with a Laplacian prior. This leads to a *convex* optimization problem for which the global maximum can be found using linear programming. We then use the solution with a Laplacian prior as an initial condition for a simple, iterative maximization of the sparse prior.

3.1 Exactly maximizing a Laplacian prior using linear programming

Under the Laplacian approximation, we approximate $\Pr(I)$ with an approximate $\tilde{\Pr}(I)$ defined as:

$$P\tilde{r}(I) = \prod_{i,k} e^{-|f_{i,k} \cdot I|} \tag{4}$$

To find the best decomposition under the Laplacian approximation we need to minimize:

$$J(I_1, I_2) = \sum_{i,k} |f_{i,k} \cdot I_1| + |f_{i,k} \cdot I_2| \tag{5}$$

subject to the two constraints given above: that $I_1 + I_2 = I$ and that the two layers agree with the labeled gradients. This is an L_1 minimization with linear constraints. We can turn this into an unconstrained minimization by substituting in $I_2 = I - I_1$ so that we wish to find a single layer I^1 that minimizes:

$$\begin{aligned} J_2(I_1) = & \sum_{i,k} |f_{i,k} \cdot I_1| + |f_{i,k} \cdot (I - I_1)| \\ & + \lambda \sum_{i \in S_1, k} |f_{i,k} \cdot I_1 - f_{i,k} \cdot I| \\ & + \lambda \sum_{i \in S_2, k} |f_{i,k} \cdot I_1| \end{aligned} \tag{6}$$

where the last two terms enforce the agreement with the labeled gradients.

This minimization can be performed exactly using linear programming. This is due to the fact that the derivatives are linear functions of the unknown image. To see this, define v to be a vectorized version of the image I_1 then we can rewrite J_2 as:

$$J_2(v) = \|Av - b\|_1 \tag{7}$$

where $\| \, \|_1$ is the L_1 norm, the matrix A has rows that correspond to the derivative filters and the vector b either has input image derivatives or zero so that equation 7 is equivalent to equation 6.

Minimization of equation 7 can be done by introducing slack variables and solving:

$$\begin{gathered} Min : \sum_i (z_i^+ + z_i^-) \\ Subject \;\; to : \\ Av + (z^+ - z^-) = b \\ z^+ \geq 0, z^- \geq 0 \end{gathered}$$

The idea is that at the optimal solution one of the variables z_i^+, z_i^- is zero, and the over is equal to $|A_{i \rightarrow} v - b_i|$. The above problem is a standard linear programming one and we use the LOQO [14] linear programming package to solve it.

3.2 Optimization of the sparse prior using iterated linear programming

To find the most likely decomposition under the sparse prior we need to maximize the probability of the two layers as given by equation 3. Using the same algebra as in the previous section this is equivalent to finding a vector v that minimizes:

$$J_3(v) = \sum_i \rho\left(A_{i\rightarrow} v - b_i\right) \tag{8}$$

where $\rho(x)$ is the log probability shown in figure 2. $\rho(x)$ is similar to a robust error measure and hence minimizing J_3 is *not* a convex optimization problem. Nevertheless, using EM we can iteratively solve convex problems.

Since we use a mixture model to describe the sparse prior, we can use expectation-maximization (EM) [1] to iteratively improve the probability of a decomposition. We introduce a binary hidden variable h_i for every row of the matrix A that denotes which Laplacian generated the measurement in b_i. In the E step we calculate the expectation of h_i and in the M step we use this expected value and optimize an expected complete data log likelihood. A standard derivation shows that the EM algorithm reduces to:

- E step. calculate two weights w_1, w_2 for every row of the matrix A:

  $$w_j(i) \propto \frac{\pi_j}{s_j} e^{-|A_{i\rightarrow} v - b_i|/s_j} \tag{9}$$

  the proportion constant is set so that $w_1(i) + w_2(i) = 1$ for all i.
- M step: perform an L_1 minimization given by:

  $$v^* \leftarrow \arg\min_v \|DAv - Db\|_1 \tag{10}$$

  with D a diagonal matrix whose elements are given by:

  $$D(i,i) = w_1(i)/s_1 + w_2(i)/s_2 \tag{11}$$

At every iteration, we are provably decreasing the cost function J_3 in equation 8. The optimization in the M step was performed using the same linear programming software as in the Laplacian approximation. 3 EM iterations are usually sufficient.

4 Results

We show results of our algorithm on five images of scenes with reflections. Four of the images were downloaded from the internet and we had no control over the camera parameters or the compression methods used. For color images we ran the algorithm separately on the R,G and B channels.

Figures 3, 4 and 5 show the input images with labeled gradients, and our results. In Figures 4,5 we compare the Laplacian prior and the sparse prior,

Fig. 3. Results: (a) input image. (b-c) decomposition.

versus the number of labeled points. The Laplacian prior gives good results although some ghosting effects can still be seen (i.e. there are remainders of layer 2 in the reconstructed layer 1). These ghosting effects are fixed by the sparse prior. Good results can be obtained with a Laplacian prior when more labeled gradients are provided. Figures 6, 7 compares the Laplacian prior with a Gaussian prior (i.e. minimizing $\|Av - b\|$ under the L_2 norm) using both simple and real images. The non sparse nature of the Gaussian distribution is highly noticeable, causing the decomposition to split edges into two low contrast edges, rather then putting the entire contrast in one of the layers.

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

Fig. 4. Comparing Laplacian prior(first iteration results) with a sparse prior. When a few gradients are labeled (left) the sparse prior gives noticeably better results. When more gradients are labeled (right), the Laplacian prior results are similar to the sparse prior. (a-b) labeled input images. (c-d) decomposition with Laplacian prior. (e-f) decomposition using a sparse prior.

The images in figure 5 were separated automatically in [11] using multiple images. An advantage of using multiple images is that they can deal better with saturated regions (e.g. the cheekbone of the man in the image that is superimposed on the white shirt of the woman) since the saturated region location varies along the sequence. However, working with a single image, we cannot recover structure in saturated regions.

In Fig 8 the technique was applied for removing shading artifacts. For this problem, the same algorithm was applied in the log-domain.

5 Discussion

Separating reflections from a single image is a massively ill-posed problem. In this paper we have focused on slightly easier problem in which the user marks a small number of gradients as belonging to one of the layers. This is still an ill-posed problem and we have used a prior derived from the statistics of natural scenes: that derivative filters have sparse distributions. We showed how to efficiently find the most probable decompositions under this prior using linear programming. Our results show the clear advantage of a technique that is based on natural scene statistics rather than simply assuming a Gaussian distribution.

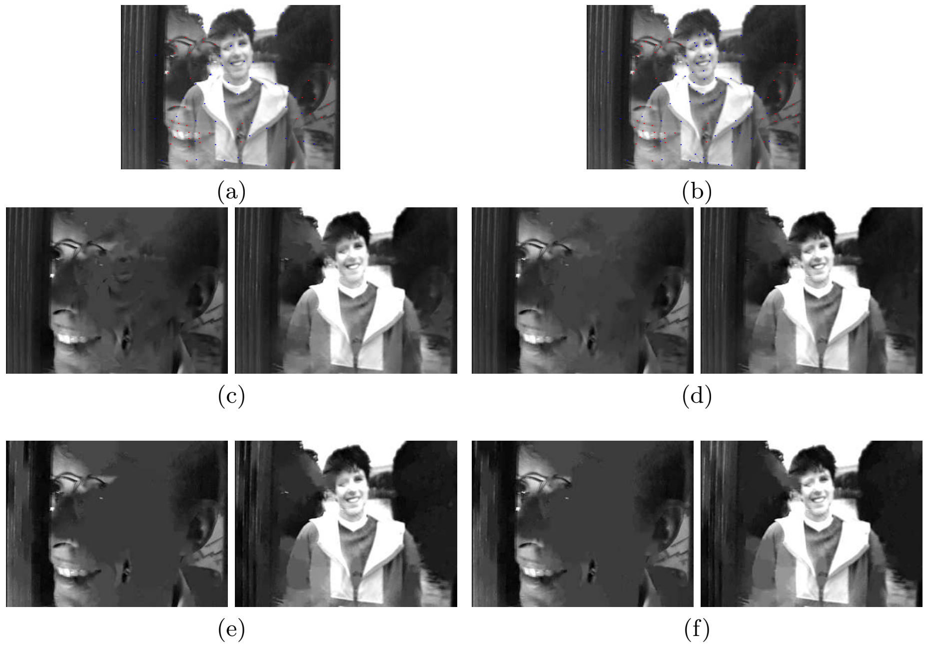


Fig. 5. Comparing Laplacian prior(first iteration results) with a sparse prior. When a few gradients are labeled (left) the sparse prior gives noticeably better results. When more gradients are labeled (right), the Laplacian prior results are similar to the sparse prior. (a-b) labeled input images. (c-d) decomposition with Laplacian prior. (e-f) decomposition using a sparse prior.

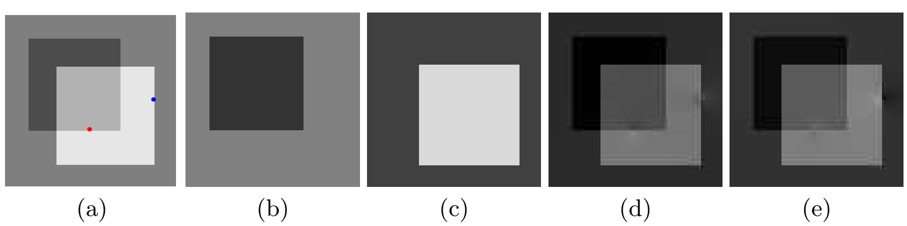


Fig. 6. (a) A very simple image with two labeled points. (b-c) The Laplacian prior gives the correct results for this image while the Gaussian prior (c-d) does not. The Gaussian prior prefers to split edges into two low contrast edges.

Since we are using an off-the-shelf linear programming package, we are not taking advantage of the spatial properties of the optimization problem. The current run time of the linear programming for images of size $240x320$ is a few minutes on a standard PC. We have not performed an extensive comparison of linear programming packages so that with other packages the run times may be significantly faster. We are currently working on deriving specific algorithms for minimizing L_1 cost functions on image derivatives. Since this is a convex

(a) (b)

Fig. 7. Gaussian prior results: (a) results on the second column of fig4. (b) results on the second column of fig5.

(a) (b) (c) (d)

Fig. 8. Removing shading artifacts (a) original image. (b) labeled image. (c-d) decomposition

problem, local minima are not an issue and so a wide range of iterative algorithms may be used. In preliminary experiments, we have found that a multigrid algorithm can minimize such cost functions significently faster. We are also investigating using a mixture of Gaussians rather than a mixture of Laplacians to describe sparse distributions. This leads to M steps in which L_2 minimizations need to be performed, and there are a wide range of efficient solvers for such minimizations.

We are also investigating the use of other features other than derivatives to describe the statistics of natural images. Our experience shows that when stronger statistical models are used, we need less labeled points to achieve a good separation. We hope that using more complex statistical models will still enable us to perform optimization efficiently. This may lead to algorithms that separate reflections from a single image, without any user intervention.

References

1. A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Statist. Soc. B*, 39:1–38, 1977.
2. H. Farid and E.H. Adelson. Separating reflections from images by use of independent components analysis. *Journal of the optical society of america*, 16(9):2136–2145, 1999.
3. G. D. Finlayson, S. D. Hordley, and M. S. Drew. removing shadows from images. In *European Conf. on Computer Vision*, 2002.

4. M. Irani and S. Peleg. Image sequence enhancement using multiple motions analysis. In *Conf. on Computer Vision and Pattern Recognition*, pages 216–221, Champaign, Illinois, June 1992.
5. A. Levin, A. Zomet, and Y. Weiss. Learning to perceive transparency from the statistics of natural scenes. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems 15*, 2002.
6. S. Mallat. A theory for multiresolution signal decomposition : the wavelet representation. *IEEE Trans. PAMI*, 11:674–693, 1989.
7. B.A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–608, 1996.
8. Y. Shechner, J. Shamir, and N. Kiryati. Polarization-based decorrelation of transparent layers: The inclination angle of an invisible surface. In *Int. Conf. on Computer Vision*, pages 814–819, 1999.
9. E.P. Simoncelli. Statistical models for images:compression restoration and synthesis. In *Proc Asilomar Conference on Signals, Systems and Computers*, pages 673–678, 1997.
10. E.P. Simoncelli. Bayesian denoising of visual images in the wavelet domain. In P Mller and B Vidakovic, editors, *Wavelet based models*, 1999.
11. R. Szeliksi, S. Avidan, and P. Anandan. Layer extraction from multiple images containing reflections and transparency. In *Conf. on Computer Vision and Pattern Recognition*, 2000.
12. M. Tappen, W.T. Freeman, and E.H. Adelson. Recovering intrinsic images from a single image. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems 15*, 2002.
13. Y. Tsin, S.B. Kang, and R. Szeliski. Stereo matching with reflections and translucency. In *Conf. on Computer Vision and Pattern Recognition*, pages 702–709, 2003.
14. R. Vanderbei. Loqo, http://www.princeton.edu/ rvdb/, 2000.
15. M.J. Wainwright, E.P. Simoncelli, and A.S. Willsky. Random cascades of gaussian scale mixtures for natural images. In *Int. Conf. on Image Processing*, pages I:260–263, 2000.
16. Y. Weiss. Deriving intrinsic images from image sequences. In *Proc. Intl. Conf. Computer Vision*, pages 68–75. 2001.
17. M. Zibulevsky, P. Kisilev, Y. Zeevi, and B. Pearlmutter. Blind source separation via multinode sparse representation. In T. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, 2001.