User Assisted Separation of Reflections from a Single Image Using a Sparsity Prior

Anat Levin and Yair Weiss

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. This sparsity prior is optimized using the iterative reweighted least squares (IRLS) approach. 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 modest number of labeled gradients.

Index Terms—Transparency, image statistics, low-level vision, interactive image editing.

1 Introduction

THE room in which Leonardo's Mona Lisa is displayed at the Louvre is shown in Fig. 1a. 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 Fig. 1b). Fig. 1c 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 the polarizing lens appropriately, one can reduce (but not eliminate) the reflection. As suggested in [2], [15] 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. Agrawal et al. [1] use a similar approach to handle reflections given a flash and no-flash image pair. An alternative solution is to use *multiple* input images [18], [4], [13], [14] in which the reflection and the nonreflected images have different motions. By analyzing the movie sequence, the two layers can be recovered. In [20], 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 Fig. 1c, 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?

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

• *The authors are with the School of Computer Science and Engineering, The Hebrew University of Jerusalem, 91904 Jerusalem, Israel. E-mail: {alevin, yweiss}@cs.huji.ac.il.*

Manuscript received 24 Apr. 2006; revised 27 Sept. 2006; accepted 5 Dec. 2006; published online 1 Feb. 2007.
Recommended for acceptance by G. Finlayson.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0316-0406.
Digital Object Identifier no. 10.1109/TPAMI.2007.1106.

Mathematically, the problem is massively ill-posed. The input image $I(x,y)$ is a linear combination of two unknown images the image behind the glass, I_1 and the image reflected by the glass, I_2. These two images sum linearly [2], [15] as

$$I(x,y) = I_1(x,y) + I_2(x,y). \quad (1)$$

Obviously, there are an infinite number of solutions to (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 [19], [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 [6], 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. Fig. 1d shows the Mona Lisa image with manually marked gradients: Blue gradients are marked as belonging to the Mona Lisa layer and yellow 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 postprocess 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 million pixels and assume the user marks a hundred edges. Each marked edge gives an additional constraint for the problem in (1). However, with these additional equations, the total number of equations is only a 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 [10], [16], we use a prior on images that is based on the sparsity of derivative filters. This sparsity prior is optimized using the iterative reweighted least squares (IRLS) approach, which poses the problem as a sequence of standard least squares problems, each least squares problem reweighted by the previous step solution. We show that by using a prior derived from the statistics of natural scenes, one can obtain excellent separations using a relatively 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 [10], [16], [23]. Figs. 2a 2b, 2c, and 2d 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 [8] and Simoncelli [17] in characterizing these distributions in terms of the shape of their logarithm. As shown in Figs. 2b and 2d, 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

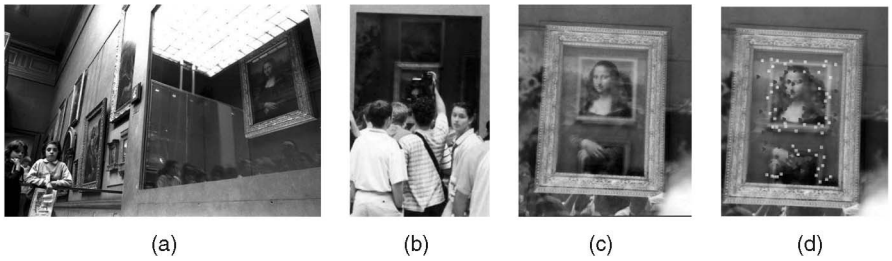

Fig. 1. (a) and (b) The scene near the Mona Lisa in the Louvre. The painting is housed behind glass to protect it from the many tourists. (c) A photograph taken by a tourist at the Louvre. The photograph captures the painting as well as the reflection of the wall across the room. (d) The user assisted 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 log domain) (Fig. 2e). In [6], 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 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 et al. [21] have suggested describing the histograms of natural images with an infinite Gaussian mixture model.

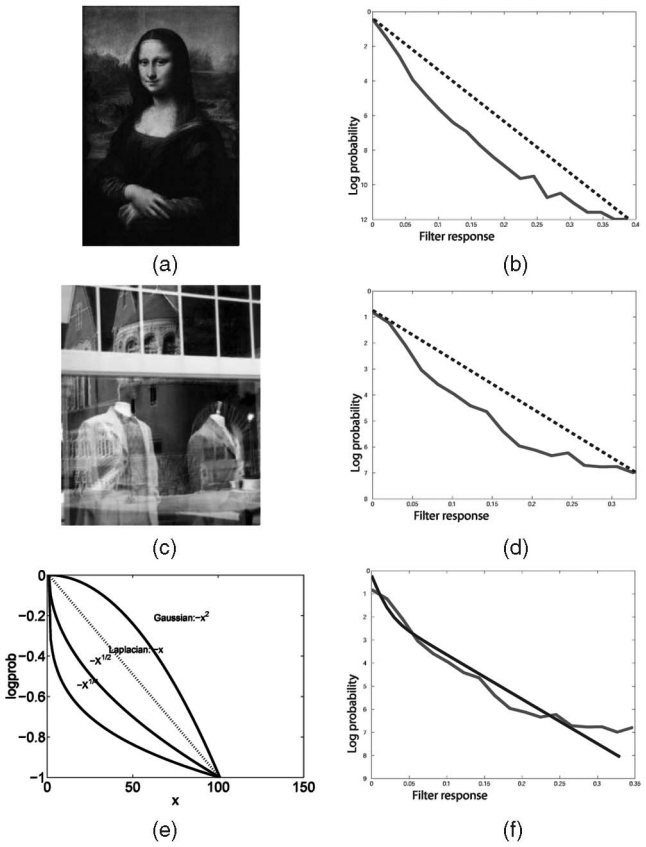

Fig. 2. (a) and (c) Input images. (b) and (d) Log-histogram of d_y derivative. A robust property of natural images is that the log-histograms of derivative filters lie below the straight line connecting the minimal and maximal values. We refer to such distributions as sparse (e) Log probabilities for distributions of the form e^{-x^α}. 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.

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. Fig. 2f 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 Fig. 2f. The learned values we found are $s_1 = 0.01$, $s_2 = 0.05$, $\pi_1 = 0.4$, and $\pi_2 = 0.6$.

Given the histograms over derivative filters, we follow [22] 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

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

where $f \cdot I$ denotes the inner product between a linear filter f and an image I and $f_{i,k}$ is the kth 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). Note that the independence assumption used here is definitely wrong—there are more filter outputs then pixels, so they certainly cannot be independent. Nevertheless, we follow previous research in adapting this simplifying assumption.

We approximate the filters likelihood using the Laplacian mixture model (2), thus

$$\begin{aligned} \log \Pr(f_{i,k} \cdot I) &\approx -\rho(f_{i,k} \cdot I) \\ \rho(x) &= \log\left(\frac{\pi_1}{2s_1} e^{-|x|/s_1} + \frac{\pi_2}{2s_2} e^{-|x|/s_2}\right). \end{aligned} \tag{4}$$

Equation (3) gives the probability of a single layer. We follow [6] 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 $I = I_1 + I_2$ and
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.

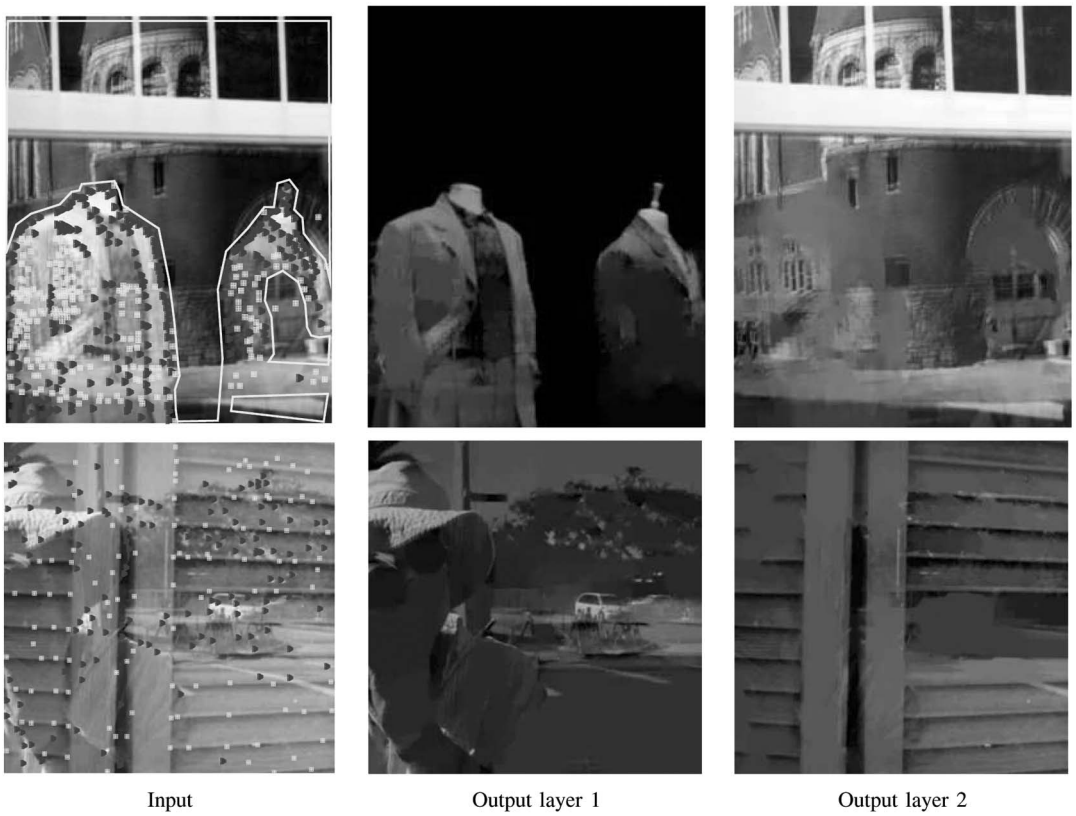

Fig. 3. Decomposition results.

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 (3). This is equivalent to minimizing

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

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 a 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

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

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

We can rewrite the cost J_2 as:

$$J_3(v) = \sum_j \rho_j(A_{j\rightarrow} v - b_j), \tag{7}$$

where v is a vectorized version of the image I_1, the matrix A has rows that correspond to the derivative filters, and the vector b either has input image derivatives or zero.

3.1 Iterative Reweighted Least Squares Optimization

In [5], we have optimized the cost of (7) using the expectation-maximization algorithm, where each maximization step involved solving a linear programming problem. Here, we take a simpler approach, which involves solving least square problems only. A simple and useful approach for optimizing the costs discussed in this paper is the iterative reweighted least squares technique (see, for example, [9]). The IRLS approach minimizes costs of the form

$$\sum_j \rho(A_{j\rightarrow} x - b_j) \tag{8}$$

by posing the problem as a sequence of standard least squares problems, each least squares problem reweighted by the previous step solution. The minimization of each least squares problem is equivalent to solving a sparse set of linear equations.

The IRLS algorithm proceeds as follows:

- Initialization: set $\psi_j^0 = 1$
- repeat till convergence:
  - Let $\bar{A} = \sum_j A_{j\rightarrow}^T \psi_j^{t-1} A_{j\rightarrow}$ and $\bar{b} = \sum_j A_{j\rightarrow}^T \psi_j^{t-1} b_j$. x^t is the solution for $\bar{A}x = \bar{b}$.
  - Set $u_j = A_{j\rightarrow} x^t - b_j$ and

$$\psi_j^t(u_j) = \frac{1}{u_j} \frac{d\rho(u_j)}{du}.$$

In this paper, we are concerned with costs of the form $\rho(u_j) = \log(\sum_l \frac{\pi_l}{2s_l} e^{-|u_j|/s_l})$. The reweighting term for this cost reduces to

$$\psi(u_j) = \frac{1}{\max(|u_j|, \epsilon)} \frac{\sum_l \frac{\pi_l}{2s_l^2} e^{-|u_j|/s_l}}{\sum_l \frac{\pi_l}{2s_l} e^{-|u_j|/s_l}},$$

where $1/|u_j|$ was replaced with $1/\max(|u_j|, \epsilon)$ to avoid division by zero.

In our implementation, we used a fixed number of 10 IRLS iterations (rather than tasting for convergence). When iterative

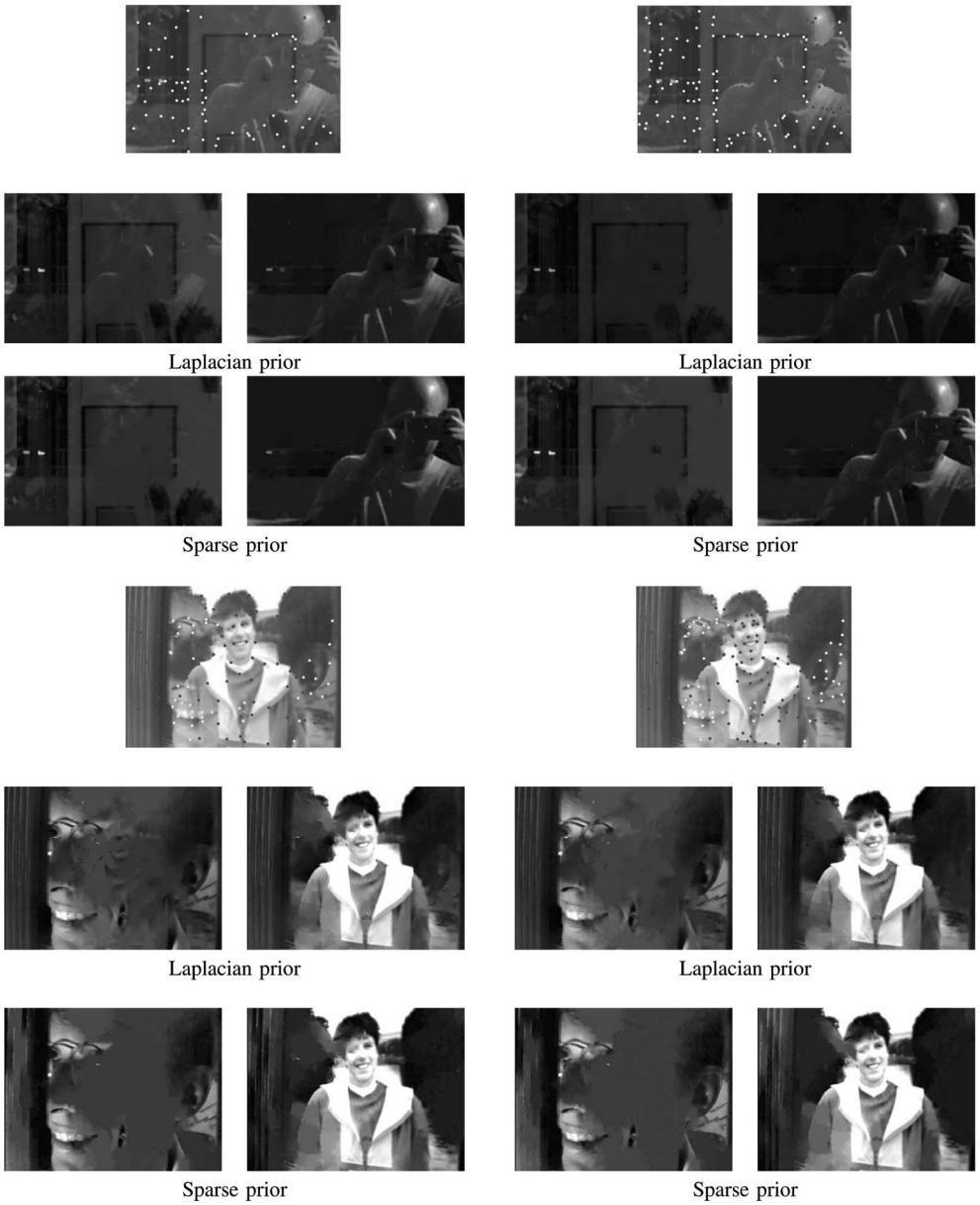

Fig. 4. Comparing Laplacian prior with a sparse prior. (left) When a few gradients are labeled, the sparse prior gives noticeably better results. (right) When more gradients are labeled, the Laplacian prior results are similar to the sparse prior.

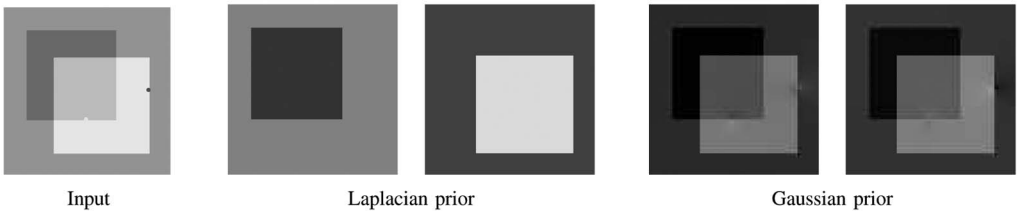

Fig. 5. A very simple image with two labeled points. The Laplacian prior gives the correct decomposition for this image while the Gaussian prior prefers to split edges into two low contrast edges.

reweighted least squares is applied on a convex cost such as the L1 cost, it converges only to the global optimum. When it is applied to the sparse prior of (4), one cannot guarantee that the global optimum will be achieved. All results in this paper use the initialization $\psi_j = 1$ which means the layers are initialized with the solution of the Gaussian prior as in Fig. 6. We found that other initialization procedures gave markedly worse results. Section 4.1.1 compares the IRLS approach to the optimization of [5].

4 Results

4.1 Qualitative Results

The implementation of the decomposition algorithm described in this paper is available at the authors' Web page, www.cs.huji.ac.il/~alevin/reflections.zip.

We show qualitative results of our algorithm on five images of scenes with reflections. While our algorithm is based on the assumption of linear camera response, four of the images were

Fig. 6. Gaussian prior results using the labels in the second column of Fig. 4.

Fig. 7. Removing shading artifacts.

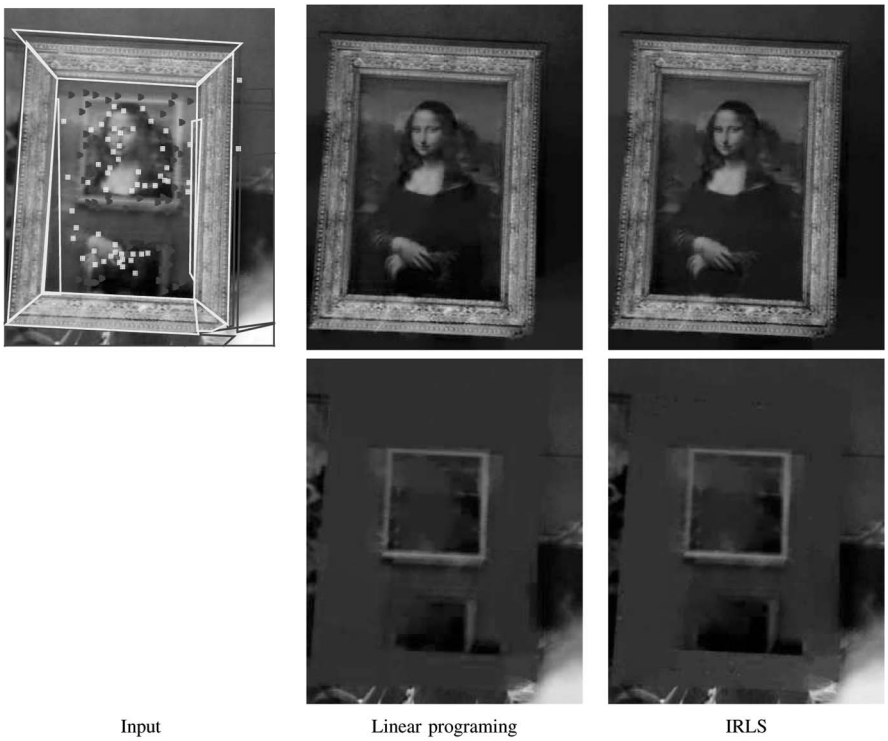

Fig. 8. Decomposition results with iterated linear programing [5] and with the iterative reweighted least squares approach described in this paper.

downloaded from the Web and we had no control over the camera parameters or the compression methods used. Yet, the algorithm was applied on the images directly without any gamma correction. (A standard 2.2 gamma correction did not have a significant effect on the result.) For color images, we ran the algorithm separately on each of the R, G, and B channels.

Figs. 3, 8, and 4 show the input images with labeled gradients and our results. In Fig. 4, we compare the Laplacian prior and the sparse prior, 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. Figs. 5 and 6 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 nonsparse 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.

As mentioned above, our technique is based on the assumption of linear camera responses, and we are not modeling correctly the nonlinear aspects of images with limited dynamic range. This problem can be observed in the second example of Fig. 4. The images in this figure were separated automatically in [18] 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

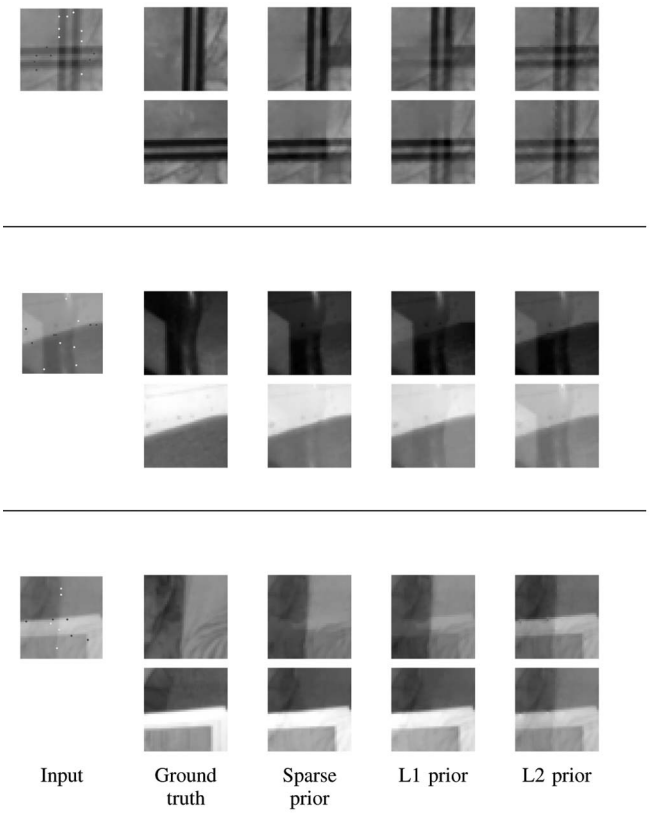

Fig. 9. Visual comparison of prior models using first and second order derivative filters.

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. 7, the technique was applied for removing shading artifacts. For this problem, the same algorithm was applied in the log-domain (since the color observed in an image can be modeled as the reflectance times the light, the problem is log-linear in the log-domain).

4.1.1 Comparison of Optimization Methods

When iterative reweighted least squares is applied on a convex cost such as the L_1 cost, it converges only to the global optimum. When it is applied to the sparse prior of (4) one cannot guarantee that the global optimum will be achieved. However, we found that in practice, for our problem the iterative reweighted least squares can find solutions whose quality is visually similar to the those of [5]. For example, Fig. 8 presents the results of the two algorithms on the Mona Lisa image. The results are visually similar. Since the formulation of the transparency problem is invariant to additive constant, we constrained both solutions to have the same mean in each color channel. However, the IRLS algorithm is significantly faster. The algorithm of [5] runs on the 260×320 Mona Lisa image in about 2.5 hours on a 2.4GH CPU (using the LOQO linear programing solver which is the fastest solver we have been able to find). On the other hand, the IRLS algorithm processes the same image within only 12 minutes, when each of the least square problems is solved exactly using the Matlab's "backslash" operator. Moreover, the subject of least squares optimization has attracted much more research than linear programing and is understood much better. Thus, for the least square problem there exists a variety of fast numerical solvers (e.g., multigrid solvers [11]) which could replace the exact Matlab solver and farther speed the performances.

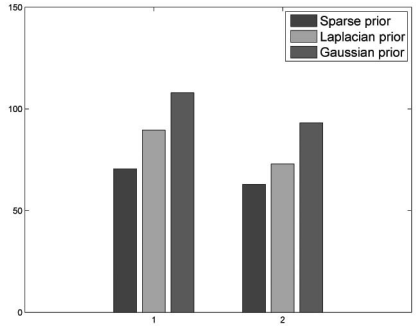

Fig. 10. Quantitative evaluation of different prior models. Bars are plotted in two groups, representing the number of labeled gradients in each experiment (15 or 50).

4.2 Quantitative Evaluation of Likelihood Models

In this section, we investigate the selection of the filters and likelihood models used in our decomposition cost function (6).

To perform a quantitative evaluation of the different models, we selected at random 250 pairs of 40×40 patches from natural images. The superpositions of those pairs served us as test images, for which the ground truth decompositions are known. For each patch in the pair, Canny edge detector was applied and sets of 15 or 50 points over the edges were selected at random as "marked gradients." Figs. 9, 12, and 15 illustrate some of the test images. Given the marked gradients, we were trying to decompose each test image using several likelihood models. We measured the sum of absolute differences between the recovered layers and the ground truth layers. In Figs. 10, 13, and 16, we present bar charts. The bar chart for each experiment are plotted in two groups, corresponding the number of labeled gradients used in each experiment (15 or 50).

We have performed three experiments, the first one was designed to test the prior choice and the other two experiments test the filters choice.

We start by investigating the importance of the sparse likelihood model. We were using first and second order derivative filters, and compared the sparse likelihood that was fitted to the distribution of edges in natural images with the simpler Laplacian distribution prior and Gaussian priors. The bar charts of the three models are plotted in Fig. 10. As can be seen, the highly nonsparse nature of the Gaussian prior result in a very bad decomposition. The Laplacian prior behaves much better then the Gaussian prior, but the actual sparse prior that was fitted to the distribution of filters in real images outperforms the Laplacian prior. Fig. 9 presents visual results for one of the test images. A qualitative comparison of the different priors was also presented in Figs. 4, 5, and 6.

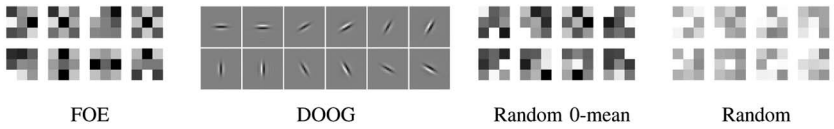

Fig. 11. Evaluated filters sets.

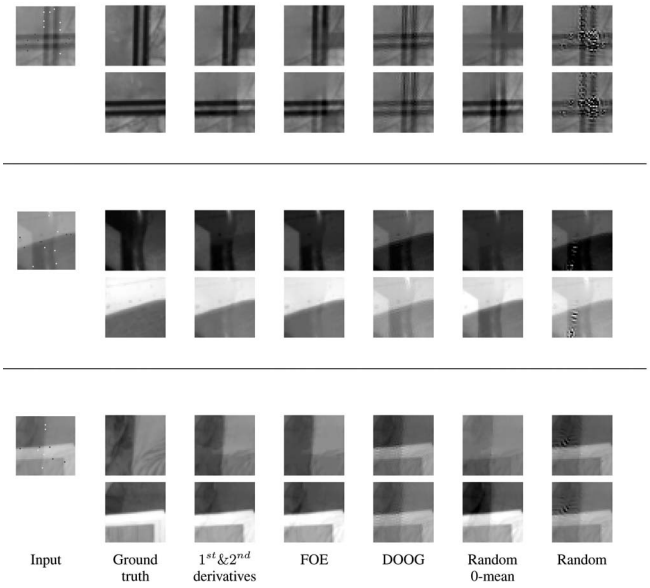

Fig. 12. Visual comparison of different filters using a sparse prior.

In addition to fitting the prior to the real distribution in natural images, there is also a question which filters to use.

In our second comparison, we experimented with several of the popular sets of low-level filters using a sparse prior (the sparsity prior was obtained by fitting a mixture of Laplacians to the first and second order derivatives histograms and the same prior was used for all filters). In particular, we chose to test the filters sets listed below. The filters sets are also presented visually in Fig. 11.

1. High frequency first order derivatives evaluated by the $[1\ -1]$ filter, in the horizontal and vertical directions, plus high frequency second order derivatives. Second order derivatives were evaluated by convolving each pair of first order filters. The obtained filers are:
$$[1\quad -2\quad 1], \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}.$$
2. The set of 3×3 filters learned by Roth and Black using the field of experts (FOE) model [12].
3. The set of difference of oriented Gaussians (DOOG) filters used in [7]. We used six orientations, two phases, and one scale. (Using filters in coarser scales significantly increases complexity, since as the filters have wider support, the matrix that we need to invert is less sparse.)
4. A set of zero mean white noise 3×3 filters. Those were selected by randomizing nine numbers from a uniform distribution on the $[-1, 1]$ interval and subtracting the mean.
5. A set of white noise 3×3 filters without zero mean.

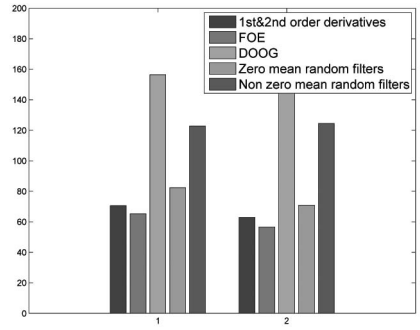

Fig. 13. Quantitative evaluation of different filter sets. Bars are plotted in two groups, representing the number of labeled gradients in each experiment (15 or 50).

The bar charts resulting from the usage of the above filter sets are presented in Fig. 13. Fig. 12 visualizes the results for three of the test patches. It seems that the best decomposition results were obtained by the FOE filters of [12], and the set of second and first order derivatives performed almost the same. Random zero-mean filters also provided relatively good results. The DOOG filters of [7] didn't perform that well, despite the fact that they were designed to be particularly sparse filters. It seems that the problem with those filters is that the filters support is too wide and the filter response in each location averages responses of different edges (in many cases, it averages responses of edges from different layers). Also, those filters may suffer most from the independence assumption. The worst results were obtained with nonzero mean filters, suggesting that the sparse prior is only suitable when the filter output is sparse. To see this, we have evaluated the responses of the different filter sets on real images. Fig. 14 plots the log histogram of filter responses. While the first four filter sets follow a sparse distribution, filters with arbitrary mean don't tend to have sparse responses on images, and the evaluated histogram is almost uniform.

In our third experiment, we test the influence of the filters support size and the amount of high order details it captures. In this experiment, we have compared results of three filters groups: 1) Both first and second order derivatives, 2) first order derivatives alone, and 3) second order derivatives alone. Numerical results are presented in the bars of Fig. 16 and visual results at Fig. 15. It can be observed that the results of using both first and second order derivatives are better then each group alone. This is because the width of the support of a filter performs an important task. First order derivatives alone are not strong enough since they cannot capture wide edges. For example, Fig. 17 presents a 1D profile of an edge and two possible decompositions of this edge. First, a desirable decomposition, places the entire edge in one layer. The second decomposition places the transition between pixels 3 and 4 in one layer and the second half of the transition (occurring between pixels 4 and 5) on the second layer. A cost which penalize the first derivative alone cannot distinguish the two decompositions, since it will pay for both decompositions $\rho(I_2 - I_3) + \rho(I_3 - I_4)$. However, if

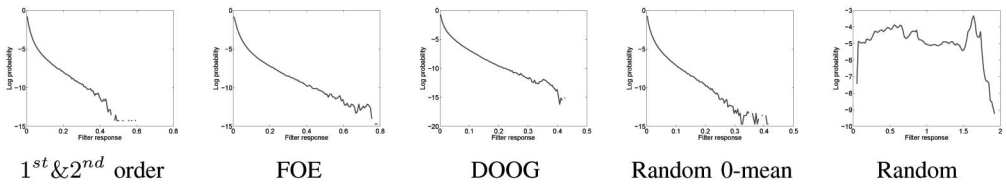

Fig. 14. Log-histogram of filter sets responses in natural images. While classical filter sets follow a sparse distribution, the distribution of nonzero mean filters is almost uniform.

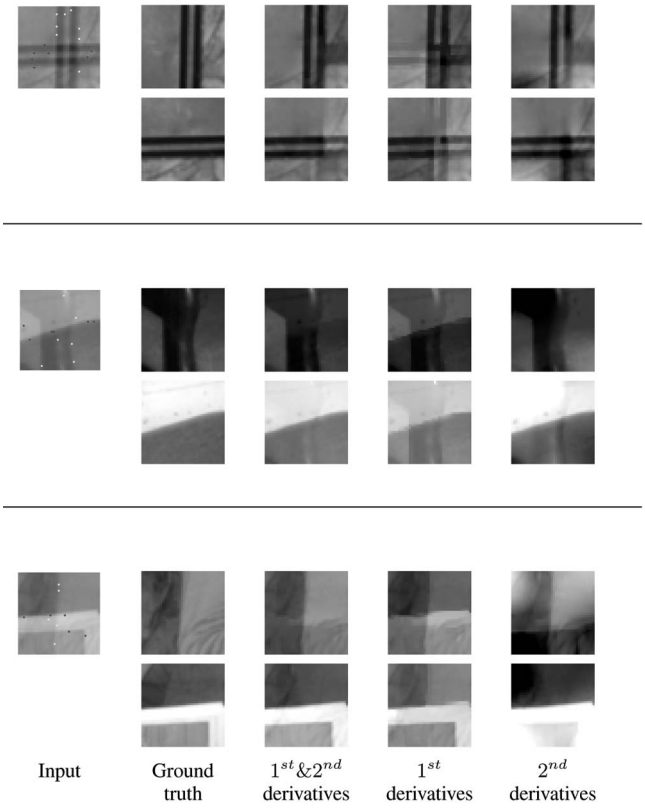

Fig. 15. Visual test of filters support.

the second order derivative is calculated, the second order derivative for the first decomposition fires only on one layer, and in the second decomposition, it fires on the two layers. Therefore, if the cost favor sparse second order derivatives, it will favor the first decomposition, as desired. However, the fact that a wide support is important, does not mean that the high order details are neglectable, as demonstrated by the fact that using second order derivatives alone provides bad results.

A second aspect that should be taken into consideration when selecting the filters set is computational complexity. The computation time reduces when the matrix A of (7) contains more zero entries.

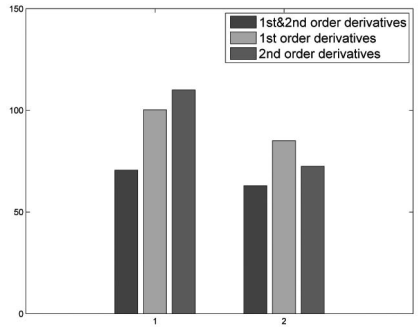

Fig. 16. Quantitative evaluation of the influence of the filter support. Bars are plotted in two groups, representing the number of labeled gradients in each experiment (15 or 50).

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 by solving a set of linear systems. Our results show the clear advantage of a technique that is based on natural scene statistics rather than simply assuming a Gaussian distribution.

We have performed a quantitative comparison of different likelihood models and different filters sets. It will be interesting to develop algorithms that will learn the optimal set of filters and the optimal likelihood model from real images data. One approach for that is the unsupervised approach that was explored by [12]. The unsupervised approach tries to construct a likelihood function that will model the distribution of natural images. Then, assuming that the two layers are independent, an optimal decomposition is a decomposition that will maximize the sum of log-likelihoods of the two layers. A second approach is the supervised approach. Which will translate to finding a likelihood function that when combined with user marks, will minimize the error of decomposed images.

In the current approach, the amount of user interaction required to achieve good results is still quite large. We hope that the usage of better statistical models will enable us to reduce the amount of user efforts.

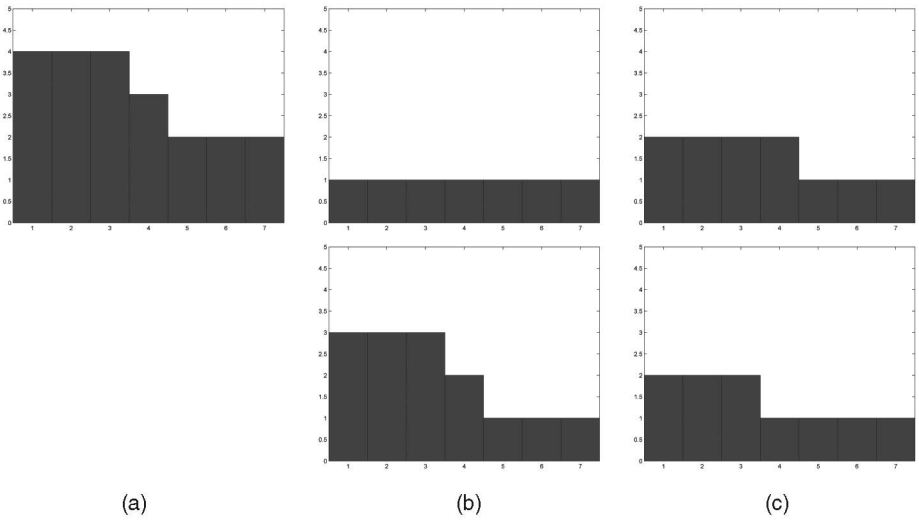

Fig. 17. Decomposing 1D edge. (a) Input edge. (b) and (c) Possible decompositions.

REFERENCES

[1] A. Agrawal, R. Raskar, S.K. Nayar, and Y. Li, "Removing Photography Artifacts Using Gradient Projection and Flash-Exposure Sampling," *ACM Trans. Graphics,* vol. 24, no. 3, pp. 828-835, July 2005.

[2] H. Farid and E.H. Adelson, "Separating Reflections from Images by Use of Independent Components Analysis," *J. Optical Soc. Am.,* vol. 16, no. 9, pp. 2136-2145, 1999.

[3] G.D. Finlayson, S.D. Hordley, and M.S. Drew, "Removing Shadows from Images," *Proc. European Conf. Computer Vision,* 2002.

[4] M. Irani and S. Peleg, "Image Sequence Enhancement Using Multiple Motions Analysis," *Proc. Conf. Computer Vision and Pattern Recognition,* pp. 216-221, June 1992.

[5] A. Levin and Y. Weiss, "User Assisted Separation of Reflections from a Single Image Using a Sparsity Prior," *Proc. European Conf. Computer Vision,* May 2004.

[6] A. Levin, A. Zomet, and Y. Weiss, "Learning to Perceive Transparency from the Statistics of Natural Scenes," *Advances in Neural Information Processing Systems 15,* S. Becker, S. Thrun, and K. Obermayer, eds., 2002.

[7] J. Malik, S. Belongie, T. Leung, and J. Shi, "Contour and Texture Analysis for Image Segmentation," *Perceptual Organization for Artificial Vision Systems,* K.L. Boyer and S. Sarkar, eds., Kluwer Academic, 2000.

[8] S. Mallat, "A Theory for Multiresolution Signal Decomposition : The Wavelet Representation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 11, pp. 674-693, 1989.

[9] P. Meer, "Robust Techniques for Computer Vision," *Emerging Topics in Computer Vision,* 2004.

[10] B.A. Olshausen and D.J. Field, "Emergence of Simple-Cell Receptive Field Properties by Learning a Sparse Code for Natural Images," *Nature,* vol. 381, pp. 607-608, 1996.

[11] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing.* Cambridge Univ. Press, 1992.

[12] S. Roth and M.J. Black, "Fields of Experts: A Framework for Learning Image Priors," *Proc. Conf. Computer Vision and Pattern Recognition,* 2005.

[13] B. Sarel and M. Irani, "Separating Transparent Layers through Layer Information Exchange," *Proc. European Conf. Computer Vision,* May 2004.

[14] B. Sarel and M. Irani, "Separating Transparent Layers of Repetitive Dynamic Behaviors," *Proc. Int'l Conf. Computer Vision,* 2005.

[15] Y. Shechner, J. Shamir, and N. Kiryati, "Polarization-Based Decorrelation of Transparent Layers: The Inclination Angle of an Invisible Surface," *Proc. Int'l Conf. Computer Vision,* pp. 814-819, 1999.

[16] E.P. Simoncelli, "Statistical Models for Images: Compression Restoration and Synthesis," *Proc Asilomar Conf. Signals, Systems, and Computers,* pp. 673-678, 1997.

[17] E.P. Simoncelli, "Bayesian Denoising of Visual Images in the Wavelet Domain," *Wavelet Based Models,* P Müller and B Vidakovic, eds., 1999.

[18] R. Szeliksi, S. Avidan, and P. Anandan, "Layer Extraction from Multiple Images Containing Reflections and Transparency," *Proc. Conf. Computer Vision and Pattern Recognition,* 2000.

[19] M. Tappen, W.T. Freeman, and E.H. Adelson, "Recovering Intrinsic Images from a Single Image," *Advances in Neural Information Processing Systems 15,* S. Becker, S. Thrun, and K. Obermayer, eds., 2002.

[20] Y. Tsin, S.B. Kang, and R. Szeliski, "Stereo Matching with Reflections and Translucency," *Proc. Conf. Computer Vision and Pattern Recognition,* pp. 702-709, 2003.

[21] M.J. Wainwright, E.P. Simoncelli, and A.S. Willsky, "Random Cascades of Gaussian Scale Mixtures for Natural Images," *Proc. Int'l Conf. Image Processing,* pp. I:260-263, 2000.

[22] Y. Weiss, "Deriving Intrinsic Images from Image Sequences," *Proc. Int'l Conf. Computer Vision,* pp. 68-75, 2001.

[23] M. Zibulevsky, P. Kisilev, Y. Zeevi, and B. Pearlmutter, "Blind Source Separation via Multinode Sparse Representation," *Advances in Neural Information Processing Systems 14,* T. Dietterich, S. Becker, and Z. Ghahramani, eds., 2001.