Submitted to *IEEE Transactions on Pattern Analysis and Machine Intelligence* October, 2006

Automatic Estimation and Removal of Noise from a Single Image

Ce Liu* Richard Szeliski† Sing Bing Kang†
C. Lawrence Zitnick† William T. Freeman*

*Computer Science and Artificial Intelligence Lab, Massachusetts Institute of Technology
†Interactive Visual Media Group, Microsoft Research

Abstract

Image denoising algorithms often assume an additive white Gaussian noise (AWGN) process that is independent of the actual RGB values. Such approaches are not fully automatic and cannot effectively remove color noise produced by today's CCD digital camera. In this paper, we propose a unified framework for two tasks: automatic estimation and removal of color noise from a single image using piecewise smooth image models. We introduce the noise level function (NLF), which is a continuous function describing the noise level as a function of image brightness. We then estimate an upper bound of the real noise level function by fitting a lower envelope to the standard deviations of per-segment image variances. For denoising, the chrominance of color noise is significantly removed by projecting pixel values onto a line fit to the RGB values in each segment. Then, a Gaussian conditional random field (GCRF) is constructed to obtain the underlying clean image from the noisy input. Extensive experiments are conducted to test the proposed algorithm, which is shown to outperform state-of-the-art denoising algorithms.

Keywords: image denoising, piecewise smooth image model, segmentation-based computer vision algorithms, noise estimation, Gaussian conditional random field, automatic vision system

1 Introduction

Image denoising has been studied for decades in computer vision, image processing and statistical signal processing. This problem not only provides a good platform to examine natural image models and signal separation algorithms, but also becomes an important part to digital image acquiring systems to enhance image qualities. These two directions are both important and will be explored in this paper.

Most of the existing image denoising work assumes additive white Gaussian noise (AWGN) and removes the noise independent of RGB channels. However, the type and level of the noise generated by digital cameras are unknown if the series and brand of the camera, as well as the camera settings (ISO, shutter speed, aperture, and flash on/off), are unknown. For instance, the exchangeable image file format (EXIF) metadata attached with each picture can be lost in image format conversion and image file transferring. Meanwhile, the statistics of the color noise is not independent of the RGB channels because of the demosaic process embedded in cameras. Therefore, the current denoising approaches are *not* truly automatic and *cannot* effectively remove color noise. This fact prevents the noise removal techniques from being practically applied to digital image denoising and enhancing applications.

In some image denoising software, the user is required to specify a number of smooth image regions to estimate the noise level. This motivated us to adopt a segmentation-based approach to automatically estimate the noise level from a single image. Since the noise level is dependent on the image brightness, we propose to estimate an upper bound of the *noise level function* (NLF) from the image. The image is partitioned into piecewise smooth regions in which the mean is the estimate of brightness and the standard deviation is an overestimate of noise level. The prior of the noise level functions are learnt by simulating the digital camera imaging process and are used to help estimate the curve correctly where there is missing data.

Since separating signal and noise from a single input is under-constrained, it is in theory impossible to completely recover the original image from the noise contaminated observation. The goal of image denoising is to preserve image features as much as possible while eliminating noise. There are a number of principles we want to match in designing image denoising algorithms.

(a) The perceptually *flat regions* should be as smooth as possible. Noise should be completely removed from these regions.

(b) *Image boundaries* should be well preserved. This means that the boundary should not be blurred or sharpened.

(c) *Texture detail* should not be lost. This is one of the hardest criteria to match. Since image denoising algorithms tend to smooth the image, it is easy to lose texture detail during smoothing.

(d) The *global contrast* should be preserved (i.e. the *low-frequencies* of the denoised and input images should be identical).

(e) No *artifacts* should appear in the denoised image.

The global contrast is probably the easiest to match, whereas some of the rest principles are almost incompatible. For instance, (a) and (c) are difficult to adjust together since most denoising algorithms cannot distinguish flat and textured regions from a single input image. Also, satisfying (e) is important. For example, wavelet-based denoising algorithms tend to generate ringing artifacts.

Ideally, the same image model should be used for both noise estimation and denoising. We found that a segmentation-based approach is equally suited to both tasks. After a natural image is over-segmented into piecewise smooth regions, the pixel values within each segment approximately lie on a 1D line in RGB space due to the physical law of image formation [25, 23, 19]. This important fact helps to significantly reduce color noise. We improve the results by constructing a Gaussian conditional random field (GCRF) to estimate the clean image (signal) from the noisy image.

Experiments are conducted, with both quantitatively convincing and visually pleasing results to demonstrate that our segmentation-based denoising algorithm outperforms the state of the art. Our approach is distinctively automatic since the noise level is automatically estimated. Automatically estimating the noise level can benefit other computer vision algorithms as well. For example, the parameters of stereo, motion estimation, edge detection and super resolution algorithms can be set as a function of the noise level so that we can avoid tweaking the parameters for different noise levels.

The paper is organized as follows. After reviewing relevant work in Section 2, we introduce our piecewise smooth image model in Section 3. In Section 4, we propose the method for noise estimation from a single image. Our segmentation-based image denoising algorithm is presented in detail in Section 5, with results shown in Section 6. We discuss issues of color noise, modeling, and automation in Section 7, and provide concluding remarks in Section 8.

2 Related Work

In this section, we briefly review previous work on image denoising and noise estimation. Image denoising techniques differ in the choice of image prior models while existing noise estimation techniques assume additive white Gaussian noise (AWGN).

2.1 Image Denoising

In the past three decades, a variety of denoising methods have been developed in the image processing and computer vision communities. Although seemingly very different, they all share the same property: to keep the meaningful edges and remove less meaningful ones. We categorize the existing image denoising work by different natural image prior models and the corresponding representation of natural image statistics.

Wavelets. When a natural image is decomposed into multiscale oriented subbands [30], we observe highly kurtotic marginal distributions [15]. To enforce the marginal distribution to have high kurtosis, we can simply suppress low-amplitude values while retaining high-amplitude values, a technique known as coring [39, 43].

In [42], the joint distribution of wavelets were found to be dependent. A joint coring technique is developed to infer the wavelet coefficients in a small neighborhood across different orientation and scale subbands simultaneously. The typical joint distribution for denoising is a Gaussian scale mixture (GSM) model [37]. In addition, wavelet-domain hidden Markov models have been applied to image denoising with promising results [8, 13].

Although the wavelet-based method is popular and dominant in denoising, it is hard to remove the ringing artifacts of wavelet reconstruction. In other words, wavelet-based methods tend to introduce additional edges or structures in the denoised image.

Anisotropic Diffusion. The simplest method for noise removal is Gaussian filtering, which is equivalent to solving an isotropic heat diffusion equation [46], a second order linear PDE. To keep sharp edges, anisotropic diffusion can be performed using $I_t = div(c(x, y, t)\nabla I)$ [34], where $c(x, y, t) = g(\|\nabla I(x, y, t)\|)$, and g is a monotonically decreasing function. As a result, for high gradient pixels, $c(x, y, t)$ is small and therefore gets less diffused. For low gradient pixels, $c(x, y, t)$ has a higher value and these pixels get blurred with neighboring pixels. A more sophisticated way of choosing $g(\cdot)$ is discussed in [3]. Compared to simple Gaussian filtering, anisotropic diffusion smooths out noise while keeping edges. However, it tends to over-blur the image and sharpen the boundary with many texture details lost.

More advanced partial differential equations (PDEs) have been developed so that a specific regularization process is designed for a given (user-defined) underlying local smoothing geometry [52], preserving more texture details than the classical anisotropic diffusion methods.

FRAME & FOE. As an alternative to measuring marginal or joint distributions on wavelet coefficients, a complete prior model over the whole image can be learnt from marginal distributions [18, 55]. Thus, it is natural to use a Bayesian inference for denoising or restoration [54, 40], which has the form $I_t = \sum_{i=1}^{n} F_i^{-1} * \lambda_i'(F_i * I) + \frac{1}{\sigma^2}(I^{obs} - I)^2$, where $\{F_i\}$ are linear filters (F_i^{-1} is the filter obtained by mirroring F_i around its center pixel), $\{\lambda_i\}$ are the corresponding Gibbs potential functions, σ^2 is the variance of noise, and t is the index of iteration. Because the derivative λ_i' typically has high values close to zero and low values at high amplitude, the above PDE is very similar to anisotropic diffusion if the F_is are regarded as derivative filters at different directions [54].

Learning a Gibbs distribution using MCMC can be inefficient. Meanwhile, these methods have the same drawbacks as anisotropic diffusion: over smoothing and edge sharpening.

Bilateral Filtering. An alternative way of adapting Gaussian filtering to preserve edges is bilateral filtering [51], where both space and range distances are taken into account. The essential relationship between bilateral filtering and anisotropic diffusion is derived in [2]. Fast bilateral filtering algorithm is also proposed in [10, 33].

Bilateral filtering has been widely accepted as a simple and effective algorithm for denoising, particularly for color images in recovering HDR images [10]. However, it cannot handle speckle noise and it also has the tendency of over smoothing and edge sharpening.

Nonlocal Methods. If both the scene and camera are static, we can simply take multiple pictures and use the mean to remove the noise. This method is impractical for a single image, but a temporal mean can be computed from a spatial mean–as long as there are enough similar patterns in the single image. We can find the similar patterns to a query patch and take the mean or other statistics to estimate the true pixel value, e.g., in [1, 5]. A more rigorous formulation of this approach is through sparse coding of the noisy input [11].

Nonlocal methods work well for texture-like images containing many repeated patterns. Compared to other denoising algorithms that have $O(n^2)$ complexity where n is the image width, these algorithms have $O(n^4)$ time complexity, which is prohibitive for real-world applications.

Conditional Random Fields. Recently, conditional random fields (CRFs) [26] have been a promising model for statistical inference. Without an explicit prior model on the signal, CRFs are flexible at modeling short and long range constraints and statistics. Since the noisy input and the clean image are well aligned at image features, CRFs, in particular Gaussian conditional random fields (GCRFs) can be well applied to image denoising. Preliminary success has been shown in the denoising work of [47]. Learning GCRFs is also addressed in [49].

2.2 Noise Estimation

Image-dependent noise can be estimated from multiple or a single image. Estimation from multiple images is an over-constrained problem, and was addressed in [24]. Estimation from a single image, however, is an under-constrained problem and further assumptions have to be made for the noise. In the image denoising literature, noise is often assumed to be additive white Gaussian noise (AWGN). A widely used estimation method is based on mean absolute deviation [9]. In [16], the noise level is estimated from the gradient of smooth or small textured regions, and the signal-dependent noise level is estimated for each intensity interval. In [45], the authors proposed three methods to estimate noise levels based on training samples and the statistics (Laplacian) of natural images. In [36], a generalized expectation maximization algorithm is proposed to estimate the spectral features of a noise source corrupting an observed image.

Techniques for noise estimation followed by noise reduction have been proposed, but they tend to be heuristic. For example, in [21], a set of statistics of film grain noise are used to estimate and remove the noise produced from scanning the photographic element under uniform exposures. In [44], signal-dependent noise is estimated from the smooth regions of the image by segmenting the image gradient with an adaptive threshold. The estimated signal-dependent noise is applied to the whole image for noise reduction. This work was further extended in [20] by associating a default film-related noise model to the image based on its source identification tag. The noise model is then adjusted using the image statistics. In certain commercially available image enhancement software, such as Neat Image™[1], the noise level can be semi-automatically estimated by specifying featureless areas to profile noise. Neat Image™also provides calibration tools to estimate the amount of noise for a specific camera and camera

[1] `http://www.neatimage.com`

setting; pre-calibrated noise profiles for various cameras are also available to directly denoise images.

By comparison, our technique avoids the tedious noise measurement process for each camera used. Furthermore, our technique provides a principled way for estimating a continuous noise level function from a single image under the Bayesian inference framework.

3 Piecewise Smooth Image Model

The piecewise smooth image model was first introduced to computer vision literature by Terzopoulos [50] to account for the regularization of the reconstructed image. The concept of piecewise smooth (or continuous) was elaborated by Blake and Zisserman [4]. In this section we discuss the reconstruction of piecewise smooth image model from an image and some important properties of this model.

3.1 Image Segmentation

Image segmentation algorithms are designed based on piecewise smooth image prior to partition pixels into regions with both similar spatial coordinates and RGB pixel values. There are a variety of segmentation algorithms, such as mean shift [7] and graph-based methods [14]. Since the focus of this paper is not on segmentation, we choose a K-means clustering method for grouping pixels into regions as described in [56]. Each segment is represented by a mean color and spatial extent. The spatial extent is computed so that the shape of the segment is biased towards convex shapes and that all segments have similar size.

3.2 Segment Statistics and Affine Reconstruction

Let the image lattice be $\mathcal{L}$. It is completely partitioned to a number of regions $\{\Omega_i\}$ where $\mathcal{L} = \bigcup_i \Omega_i$ and $\Omega_i \cap \Omega_j = \emptyset$ for $i \neq j$. Let $v \in \mathbb{R}^2$ be the coordinate variable, and $I(v) \in \mathbb{R}^3$ be the RGB value of the pixel. Since in this section we focus on the statistics within each segment, we shall use Ω to represent a segment and $v \in \Omega$ to index pixels in segment Ω.

We can fit an affine model in segment Ω to minimize the square error

$$\mathbf{A}^* = \arg\min_{\mathbf{A}} \sum_{v \in \Omega} \left\| I(v) - \mathbf{A}[v^T \ 1]^T \right\|^2, \tag{1}$$

where $\mathbf{A} \in \mathbb{R}^{3\times3}$ is the affine matrix. We call the reconstruction $f(v) = \mathbf{A}^*[v^T\ 1]^T$ the *affine reconstruction* of segment Ω. The residual is $r(v) = I(v) - f(v)$.

We assume that the residual consists of two parts, subtle texture variation $h(v)$, which is also part of signal, and noise $n(v)$, i.e., $r(v) = h(v) + n(v)$. In other words, the observed image can be decomposed into $I(v) = f(v) + h(v) + n(v)$. The underlying clean image or signal is thus $s(v) = f(v) + h(v)$, which is to be estimated from the noisy input. $s(v)$, $h(v)$, $n(v)$ are all 3D vectors in RGB space.

Let the covariance matrices of $I(v)$, $s(v)$, $h(v)$ and $n(v)$ be $\mathbf{\Sigma}_I$, $\mathbf{\Sigma}_s$, $\mathbf{\Sigma}_h$ and $\mathbf{\Sigma}_n$, respectively. We assume that $f(v)$ is a non-random process and $r(v)$ and $n(v)$ are random variables. Therefore, $\mathbf{\Sigma}_s = \mathbf{\Sigma}_h$. Suppose signal $s(v)$ and noise $n(v)$ are independent, we have

$$\mathbf{\Sigma}_r = \mathbf{\Sigma}_s + \mathbf{\Sigma}_n, \tag{2}$$

which leads to

$$\mathbf{\Sigma}_r \geqslant \mathbf{\Sigma}_n. \tag{3}$$

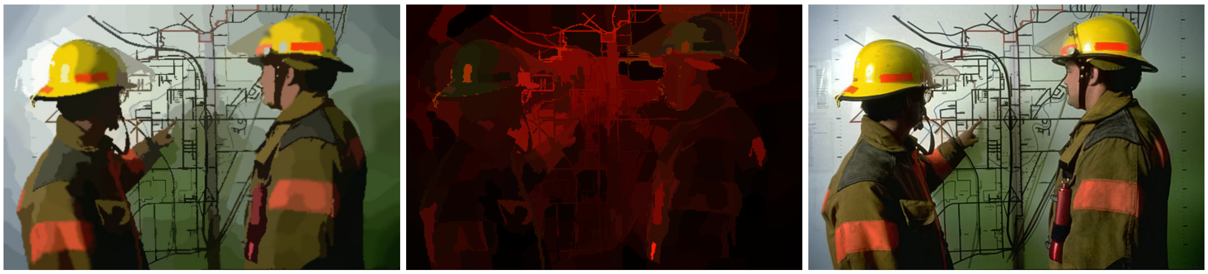
3.3 Boundary Blur Estimation

If we merely use per-segment affine reconstruction, the reconstructed image has artificial boundaries, and the original boundaries get sharpened. The amount of blur is thus estimated from the original image. For each hypothesized blur b from $b_{\min}(= 0)$ to $b_{\max}(= 2.5)$ in steps of $\Delta b(= 0.25)$, we compute the blurred image $f_{\text{blur}}(v; b) = f(v) * G(u; b)$, where $G(u; b)$ is a Gaussian kernel with sigma b. We then compute the error image I_{err} such that $I_{\text{err}}(v; b) = \|I(v) - f_{\text{blur}}(v; b)\|^2$. We dilate each boundary curve $\mathcal{C}_{ij}$ five times into regions Ω_i and Ω_j to obtain a mask Γ_{ij}. The best blur b^*_{ij} for $\mathcal{C}_{ij}$ corresponds to the minimum aggregate error $I_{err}(v; b)$ over Γ_{ij}, or $b^*_{ij} = \arg\min_b \sum_{v\in\Gamma_{ij}} I_{err}(v, b)$.

To reinstate the blur in the transition region Γ_{ij}, we simply replace $f(v)$ with $f_{\text{blur}}(v; b^*_i j)$. Note that this assumes that the amount of blur in Ω_i and Ω_j is the same, which is strictly not true in general. However, we found that this approximation generates good enough results. After this process is done for every pair of regions we obtain boundary blurred piecewise affine reconstruction $f_{\text{blur}}(v)$.

The piecewise smooth image model is illustrated in Figure 1. The example image (a) taken from Berkeley image segmentation database [31] is partitioned to piecewise smooth regions (b) by the segmentation algorithm. The per-segment affine reconstruction is shown in (c) where we can see artificial boundaries between regions and the true boundaries are sharpened. After

(a) Original Image (b) Segmentation (c) Per-segment affine reconstruction



(d) Affine reconstruction plus boundary blur (e) The sorted eigenvalues in each segment (f) RGB values projected onto the largest eigenvector

Figure 1. Illustration of piecewise smooth image model

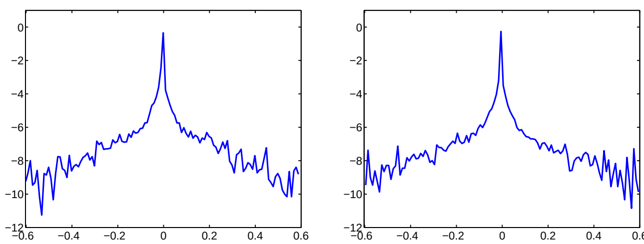


Figure 2. The log histograms of the horizontal (left) and vertical (right) derivative filter responses of the reconstruction in Figure 1 (d).

blur estimation and reinstatement, the boundaries become much smoother.

3.4 Important Properties of the Piecewise Smooth Image Model

There are three important properties of our piecewise smooth image model that made us choose it as the model for both noise estimation and removal. They are:

I. The piecewise smooth image model is consistent with a sparse image prior.

II. The color distribution per each segment can be well approximated by a line segment, due to the physical law of image formation [25, 23, 19].

III. The standard deviation of residual per each segment is the upper bound of the noise level in that segment.

The last property is straightforward from Equation (3). For the first two properties, we again use the example image in Figure 1 (a) to examine them. For the reconstructed image (d) we compute the log histograms of the horizontal and vertical derivatives and plotted them in Figure 2. The long tails clearly show that the piecewise smooth reconstruction match the high-kurtosis statistics of natural images [32]. This image model also shares some similarity with the dead leaves model [28].

For the second property, we compute the eigenvalues and eigenvectors of the RGB values $\{I(v)\}$ in each region. The eigenvalues are sorted decreasingly and displayed in Figure 1 (e). Obviously, the red channel accounts for the majority of the RGB channels, a fact that proves the first eigenvalue of each segment is significantly larger than the second eigenvalue. Therefore, when we project the pixel values onto the first eigenvalue while ignoring the other two, we get an almost perfect reconstruction in (f). The mean square error (MSE) between the original image (a) and projected (f) is 5.31×10^{-18} or a PSNR of 35.12dB. These numbers demonstrate that the RGB values in each segment lie in a line.

Having demonstrated these properties of our piecewise smooth image model, we are ready to develop models for both noise estimation and removal.

4 Noise Estimation from a Single Image

Although the standard deviation of each segment of the image is the upper bound of noise as shown in Equation (3), it is not guaranteed that the means of the segments cover the full spectrum of image intensity. Besides, the estimate of standard deviation itself is also a random variable which has variance as well. Therefore, a rigorous statistical framework is needed for the inference. In this section, we introduce the noise level functions (NLFs) and a simulation approach to learn the priors. A Bayesian approach is proposed to infer the upper bound of the noise level function from a single input.

4.1 Learning the Prior of Noise Level Functions

According to the definition, the noise standard deviation as function of brightness, the noise level function for a particular brand of camera and a fixed parameter setting can be estimated by fixing the camera on a tripod, taking multiple shots towards a static scene, and then compute the mean as the estimate of the brightness, and standard deviation as the noise level for each pixel of every RGB channel. The function of the standard deviation with respect to the mean is the desired NLF. The ground truth of NLF can be obtained by this approach and we shall use it as the reference method to test our algorithm, but it is expensive and time consuming.

As an alternative, we propose a simulation-based approach to obtain NLFs. We build a model for the noise level functions of CCD cameras. We introduce the terms of our camera noise model, showing the dependence of the noise level function on the camera response function (a.k.a. CRF, the image brightness as function of scene irradiance). Given a camera response function, we can synthesize realistic camera noise. Thus, from a parameterized set of CRFs, we derive the set of possible noise level functions. This restricted class of NLFs allows us to accurately estimate the NLF from a single image.

4.1.1 Noise Model of CCD Camera

The CCD digital camera converts the irradiance, the photons coming into the imaging sensor, to electrons and finally to bits. See Figure 3 for the imaging pipeline of CCD camera. There are mainly five noise sources as stated in [24], namely *fixed pattern noise*, *dark current noise*, *shot noise*, *amplifier noise* and *quantization noise*. These noise terms are simplified in [53]. Following the imaging equation in [53], we propose the following noise model of a CCD camera

$$I = f(L + n_s + n_c) + n_q, \tag{4}$$

where I is the observed image brightness, $f(\cdot)$ is camera response function (CRF), n_s accounts for all the noise components that are dependent on irradiance L, n_c accounts for the independent noise before gamma correction, and n_q is additional quantization and amplification noise. Since n_q is the minimum noise attached to the camera and most cameras can achieve very low noise, n_q will be ignored in our model. We assume noise statistics $\mathrm{E}(n_s) = 0, \mathrm{Var}(n_s) = L\sigma_s^2$ and $\mathrm{E}(n_c) = 0, \mathrm{Var}(n_c) = \sigma_c^2$. Note the linear dependence of the variance of n_s on the irradiance L [53].

4.1.2 Camera Response Function (CRF)

The camera response function models the nonlinear processes in a CCD camera that perform tonescale (gamma) and white balance correction [41]. There are many ways to estimate camera response functions given a set of images taken under different exposures. To explore the common properties of many different CRFs, we downloaded 201 real-world response functions from `http://www.cs.columbia.edu/CAVE` [22]. Note that we only chose 190 satu-

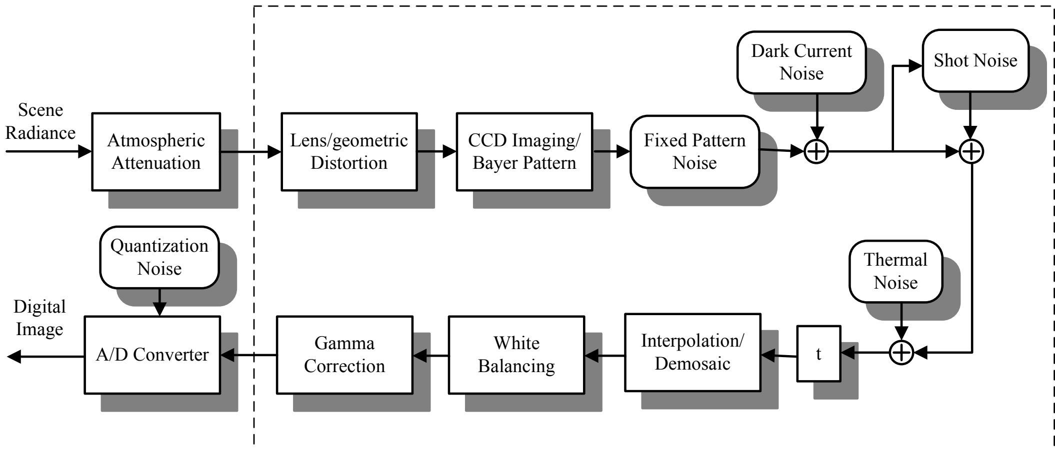
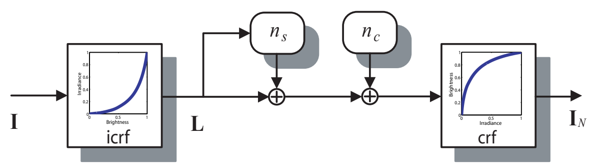
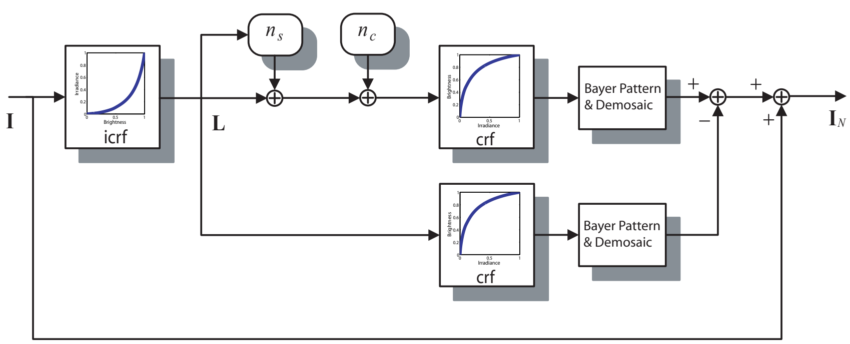


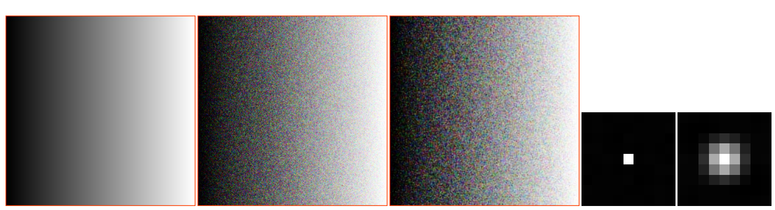
Figure 3. CCD camera imaging pipeline, redrawn from [53].



(a) White (independent) CCD noise synthesis



(b) Correlated CCD noise synthesis by going through Bayer pattern



(c) (d) (e) (f) (g)

Figure 4. Block diagrams showing noise simulations for color camera images. (a) shows independent white noise synthesis; (b) adds CCD color filter pattern sensing and demosaicing to model spatial correlations in the camera noise [27]. (c): test pattern. (d) and (e): the synthesized images of (a) and (b). (f) and (g): the corresponding autocorrelation.

rated CRFs since the unsaturated curves are mostly synthetic. Each CRF is a 1024-dimensional vector that represents the discretized $[0,1] \rightarrow [0,1]$ function, where both irradiance L and brightness I are normalized to be in the range $[0,1]$. We use the notation crf(i) to represent the i^{th} function in the database.

4.1.3 Synthetic CCD Noise

In principle, we could set up optical experiments to measure precisely for each camera how the noise level changes with image brightness. However, this would be time consuming and might still not adequately sample the space of camera response functions. Instead, we use numerical simulation to estimate the noise function. The basic idea is to transform the image $\mathbf{I}$ by the inverse camera response function f^{-1} to obtain an irradiance plane $\mathbf{L}$. We then take $\mathbf{L}$ through the processing blocks in Figure 3 to obtain the noisy image $\mathbf{I}_N$.

A direct way from Eqn. (4) is to reverse transform $\mathbf{I}$ to irradiance $\mathbf{L}$, add noise independently to each pixel, and transform to brightness to obtain $\mathbf{I}_N$. This process is shown in Figure 4 (a). The synthesized noise image, for the test pattern (c), is shown in Figure (d).

Real CCD noise is not white, however; there are spatial correlations introduced by "demosaicing" [38], i.e., the reconstruction of three colors at every pixel from the single-color samples measured under the color filter array of the CCD. We simulate this effect for a common color filter pattern (Bayer) and demosaicing algorithm (gradient-based interpolation [27]); we expect that other filter patterns and demosaicing algorithms will give comparable noise spatial correlations. We synthesized CCD camera noise in accordance with 4 (b) and took the difference between the demosaiced images with and without noise, adding that to the original image to synthesize CCD noise. The synthesized noise is shown in Figure 4 (e). The autocorrelation functions for noise images (d) and (e) are shown in (f) and (g), respectively, showing that the simulated CCD noise shows spatial correlations after taking into account the effects of demosaicing.

4.1.4 The Space of Noise Level Functions

We define the *noise level function* (NLF) as the variation of the standard deviation of noise with respect to image intensity. This function can be computed as

$$\tau(I) = \sqrt{\mathrm{E}[(I_N - I)^2]}, \tag{5}$$

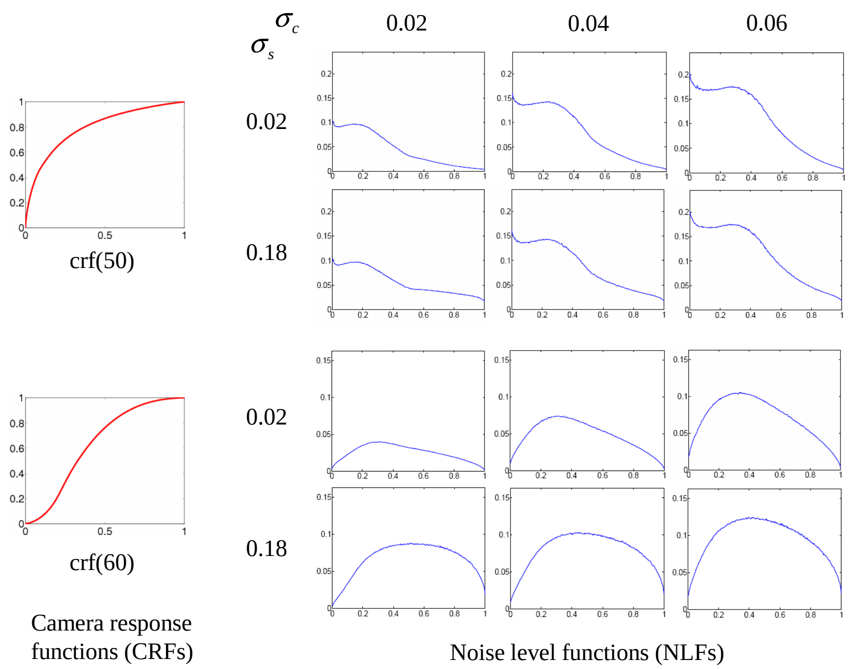
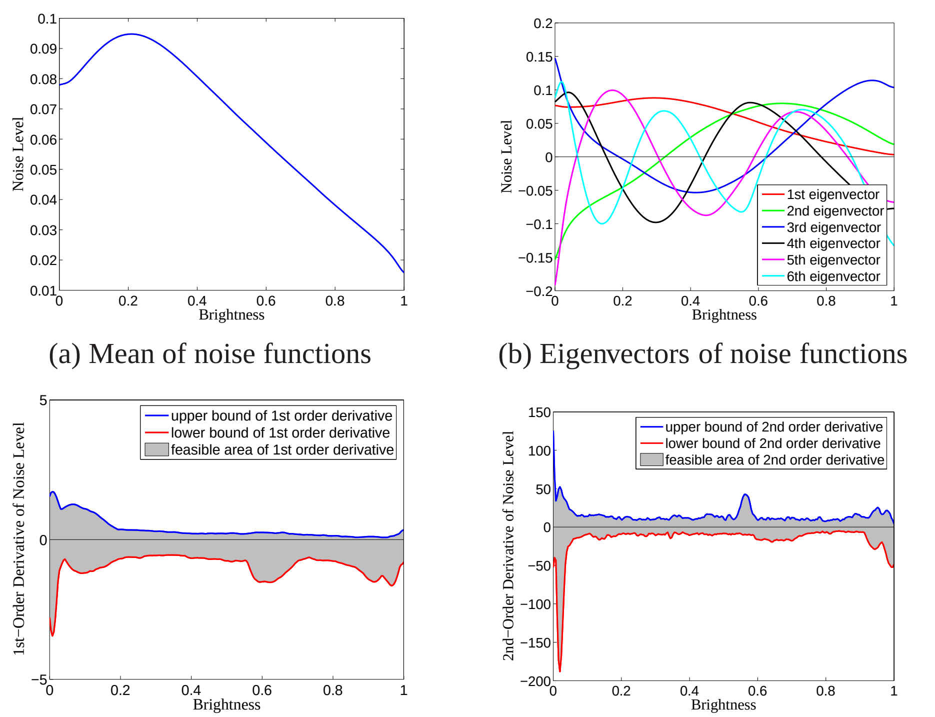


Figure 5. The procedure of sampling noise level functions.



(a) Mean of noise functions

(b) Eigenvectors of noise functions

(c) The bounds of 1st-order derivatives

(d) The bounds of 2nd-order derivatives

Figure 6. The prior of noise level functions.

where I_N is the observation and $I = \mathrm{E}(I_N)$. Essentially this is a function of how standard deviation changes with respect to the mean value.

Based on the CCD camera noise model Eqn. (4) and noise synthesis process, I_N is a random variable dependent on the camera response function f and noise parameters σ_s and σ_c. Because $L = f^{-1}(I)$, the noise level function can also be written as

$$\tau(I; f, \sigma_s, \sigma_c) = \sqrt{\mathrm{E}[(I_N(f^{-1}(I), f, \sigma_s, \sigma_c) - I)^2]}, \tag{6}$$

where $I_N(\cdot)$ is the noise synthesis process.

We use numerical simulation to estimate the noise function given f, σ_s and σ_c, for each of red, green and blue channels. This procedure is shown in Figure 5. The smoothly changing pattern in Figure 4 (c) is used to estimate Eqn. (6). To reduce statistical fluctuations, we use an image of dimension 1024×1024 and take the mean of 20 samples for each estimate.

To represent the whole space of noise level functions, we draw samples of $\tau(\cdot; f, \sigma_s, \sigma_c)$ from the space of f, σ_s and σ_c. The downloaded 190 CRFs are used to represent the space of f. We found that $\sigma_s = 0.16$ and $\sigma_c = 0.06$ result in very high noise, so these two values are set as the maximum of the two parameters. We sample σ_s from 0.00 to 0.16 with step size 0.02, and sample σ_c from 0.01 to 0.06 with step size 0.01. We get a dense set of samples $\{\tau_i\}_{i=1}^K$ of NLFs, where $K = 190 \times 9 \times 6 = 10,260$. Using principal component analysis (PCA), we get mean noise level function $\overline{\tau}$, eigenvectors $\{w_i\}_{i=1}^m$ and the corresponding eigenvalues $\{\upsilon_i\}_{i=1}^m$. Thus, a noise function can be represented as

$$\tau = \overline{\tau} + \sum_{i=1}^{m} \beta_i w_i, \tag{7}$$

where coefficient β_i is Gaussian distributed $\beta_i \sim \mathcal{N}(0, \upsilon_i)$, and the function must be positive everywhere, i.e.,

$$\overline{\tau} + \sum_{i=1}^{m} \beta_i w_i \geqslant 0, \tag{8}$$

where $\overline{\tau}, w_i \in \mathbb{R}^d$ and $d = 256$. This inequality constraint implies that noise functions lie inside a cone in β space. The mean noise function and eigenvectors are displayed in Figure 6 (a) and (b), respectively.

Eigenvectors serve as basis functions to impose smoothness to the function. We also impose upper and lower bounds on 1st and 2nd order derivatives to further constrain noise functions. Let $\mathbf{T} \in \mathbb{R}^{(d-1)\times d}$ and $\mathbf{K} \in \mathbb{R}^{(d-2)\times d}$ be the matrix of 1st- and 2nd-order derivatives [46]. The constraints can be represented as

$$b_{min} \leqslant \mathbf{T}\tau \leqslant b_{max},\ h_{min} \leqslant \mathbf{K}\tau \leqslant h_{max}, \tag{9}$$

where $b_{min}, b_{max} \in \mathbb{R}^{d-1}, h_{min}, h_{max} \in \mathbb{R}^{d-2}$ are estimated from the training data set $\{\tau_i\}_{i=1}^K$.

4.2 Likelihood Model

Since the estimated standard deviation of each segment is an over-estimate of the noise level, we obtain an upper bound estimate of the noise level function by fitting a lower envelope to the samples of standard deviation versus mean of each RGB channel. The examples of these sample points are shown in Figure 8. We could simply fit the noise function in the learnt space to lie below all the sample points yet close to them. However, because the estimates of variance in each segment are noisy, extracting these estimates with hard constraints could result in bias due to a bad outlier. Instead, we follow a probabilistic inference framework to let every data point contribute to the estimation.

Let the estimated standard deviation of noise from k pixels be $\hat{\sigma}$, with σ being the true standard deviation. When k is large, the square root of chi-square distribution is approximately $\mathcal{N}(0, \sigma^2/k)$ [12]. In addition, we assume a noninformative prior for large k, and obtain the posterior of the true standard deviation σ given $\hat{\sigma}$:

$$p(\sigma|\hat{\sigma}) \propto \frac{1}{\sqrt{2\pi\sigma^2/k}} \exp\{-\frac{(\hat{\sigma}-\sigma)^2}{2\sigma^2/k}\} \approx \frac{1}{\sqrt{2\pi\hat{\sigma}^2/k}} \exp\{-\frac{(\sigma-\hat{\sigma})^2}{2\hat{\sigma}^2/k}\}. \tag{10}$$

Let the cumulative distribution function of a standard normal distribution be $\Phi(z)$. Then, given the estimate $(I, \hat{\sigma})$, the probability that the underlying standard deviation σ is larger than u is

$$\Pr[\,\sigma \geqslant u|\hat{\sigma}] = \int_u^\infty p(\sigma|\hat{\sigma})d\sigma = \Phi(\frac{\sqrt{k}(\hat{\sigma}-u)}{\hat{\sigma}}). \tag{11}$$

To fit the noise level function to the lower envelope of the samples, we discretize the range of brightness $[0, 1]$ into uniform intervals $\{nh, (n+1)h\}_{n=0}^{\frac{1}{h}-1}$. We denote the set $\Omega_n = \{(I_i, \hat{\sigma}_i)|nh \leqslant$

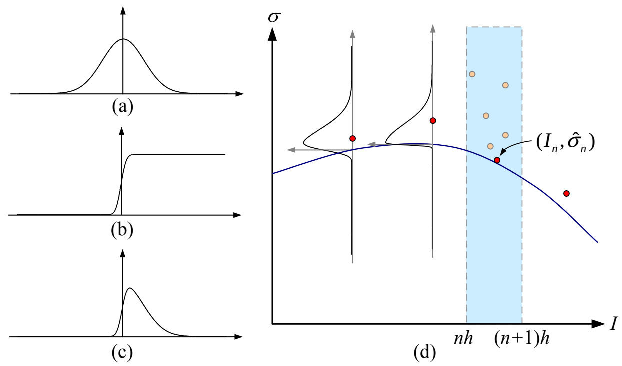


Figure 7. The likelihood function of Eq. 12. Each single likelihood function (c) is a product of a Gaussian pdf (a) and Gaussian cdf (b).

$I_i \leqslant (n+1)h\}$, and find the pair $(I_n, \hat{\sigma}_n)$ with the minimum variance $\hat{\sigma}_n = \min_{\Omega_n} \hat{\sigma}_i$. Lower envelope means that the fitted function should most probably be lower than all the estimates while being as close as possible to the samples. Mathematically, the likelihood function is the probability of seeing the observed image intensity and noise variance measurements given a particular noise level function. It is formulated as

$$\begin{aligned} \mathcal{L}(\tau(I)) &= P(\{I_n, \hat{\sigma}_n\}|\tau(I)) \\ &\propto \prod_n \Pr[\sigma_n \geqslant \tau(I_n)|\hat{\sigma}_n] \exp\{-\frac{(\tau(I_n)-\hat{\sigma}_n)^2}{2s^2}\} \\ &= \prod_n \Phi(\frac{\sqrt{k_n}(\hat{\sigma}_n - \tau(I_n))}{\hat{\sigma}_n}) \exp\{-\frac{(\tau(I_n)-\hat{\sigma}_n)^2}{2s^2}\}, \end{aligned} \tag{12}$$

where s is the parameter to control how close the function should approach the samples. This likelihood function is illustrated in Figure 7, where each term (c) is a product of a Gaussian pdf with variance s^2 (a) and a Gaussian cdf with variance $\hat{\sigma}_n^2$ (b). The red dots are the samples of minimum in each interval. Given the function (blue curve), each red dot is probabilistically beyond but close to the curve with the pdf in (c).

4.3 Bayesian MAP Inference

The parameters we want to infer are actually the coefficients on the eigenvectors $x_l = [\beta_1 \ \cdots \ \beta_m]^T \in \mathbb{R}^m$, $l = 1, 2, 3$ of the noise level function for RGB channels. We denote the sample set to fit $\{(I_{ln}, \hat{\sigma}_{ln}, k_{ln})\}$. Bayesian inference turns out to be an optimization problem

$$\{x_l^*\} = \arg\min_{\{x_l\}} \sum_{l=1}^{3} \Bigg\{ \sum_n \Bigg[-\log\Phi\Big(\frac{\sqrt{k_{ln}}}{\hat{\sigma}_n}(\hat{\sigma}_{ln} - e_n^T x_l - \overline{\tau}_n)\Big) + \frac{(e_n^T x_l + \overline{\tau}_n - \hat{\sigma}_{ln})^2}{2s^2} \Bigg] + x_l^T \Lambda^{-1} x_l + \sum_{j=1, j>l}^{3} (x_l - x_j)^T \mathbf{E}^T (\gamma_1 \mathbf{T}^T \mathbf{T} + \gamma_2 \mathbf{K}^T \mathbf{K}) \mathbf{E} (x_l - x_j) \Bigg\} \tag{13}$$

subject to

$$\overline{\tau} + \mathbf{E} x_l \geqslant 0, \tag{14}$$

$$b_{min} \leqslant \mathbf{T}(\overline{\tau} + \mathbf{E} x_l) \leqslant b_{max}, \tag{15}$$

$$h_{min} \leqslant \mathbf{K}(\overline{\tau} + \mathbf{E} x_l) \leqslant h_{max}. \tag{16}$$

In the above formula, the matrix $\mathbf{E} = [w_1 \ \cdots \ w_m] \in \mathbb{R}^{d \times m}$ contains the principal components, e_n is the n^{th} row of $\mathbf{E}$, and $\Lambda = \mathrm{diag}(v_1, \cdots, v_m)$ is the diagonal eigenvalue matrix. The last term in the objective function accounts for the similarity of the NLF for RGB channels. Their similarity is defined as a distance on 1st and 2nd order derivative. Since the dimensionality of the optimization is low, we use the MATLAB standard nonlinear constrained optimization function `fmincon` for optimization. The function was able to find an optimal solution for all the examples we tested.

4.4 Experimental Results on Noise Estimation

We have conducted experiments on both synthetic and real noisy images to test the proposed noise estimation algorithm. First, we applied our CCD noise synthesis algorithm in Sect 3.3 to 17 randomly selected pictures from the Berkeley image segmentation database [31] to generate synthetic test images. To generate the synthetic CCD noise, we specify a CRF and two parameters σ_s and σ_c. From this information, we also produce the ground truth noise level function using the training database in Sect 4.1.4. For this experiment, we selected crf(60), $\sigma_s = 0.10$ and $\sigma_c = 0.04$. Then, we applied our method to estimate the NLF from the synthesized noisy images. Both L^2 and L^∞ norms are used to measure the distance between the estimated NLF and the ground truth. The error statistics under the two norms are listed in Table 1, where the mean and maximum value of the ground truth are 0.0645 and 0.0932, respectively.

Some estimated NLFs are shown in Figure 8. In (a) we observe many texture regions especially at high intensity values, which implies high signal variance. The estimated curves (in red, green and blue) do not tightly follow the lower envelope of the samples at high intensi-

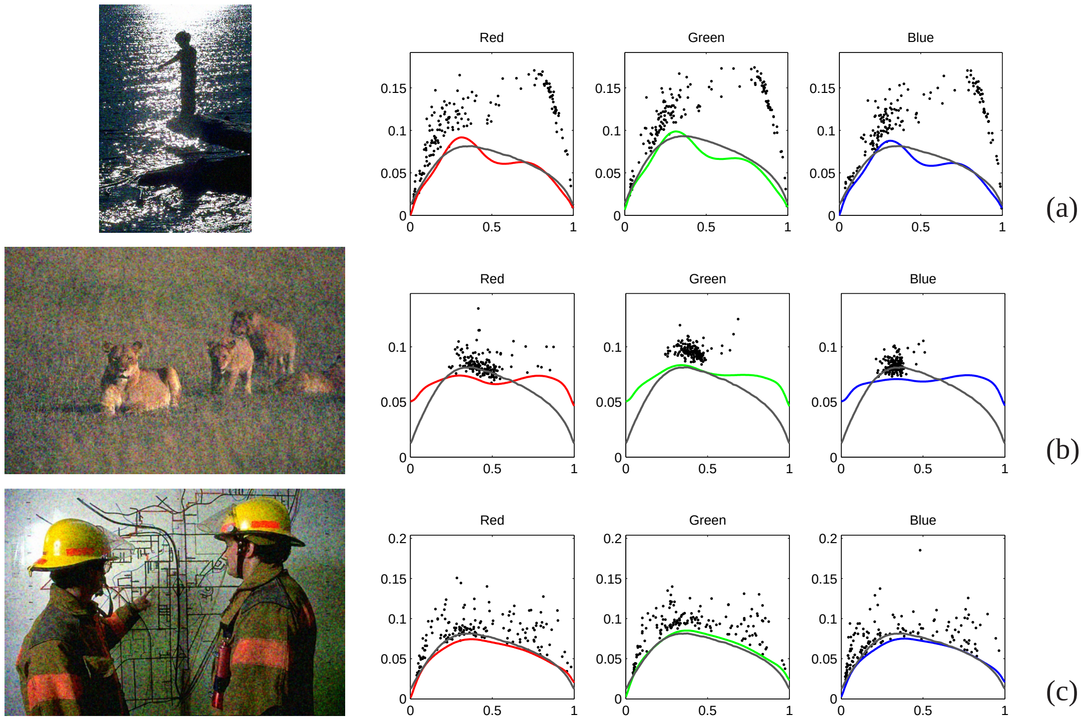


Figure 8. Synthesized noisy images and their corresponding noise level functions (noise standard deviation as a function of image brightness). The red, green and blue curves are estimated using the proposed algorithm, whereas the gray curves are the true values for the synthetically generated noise.

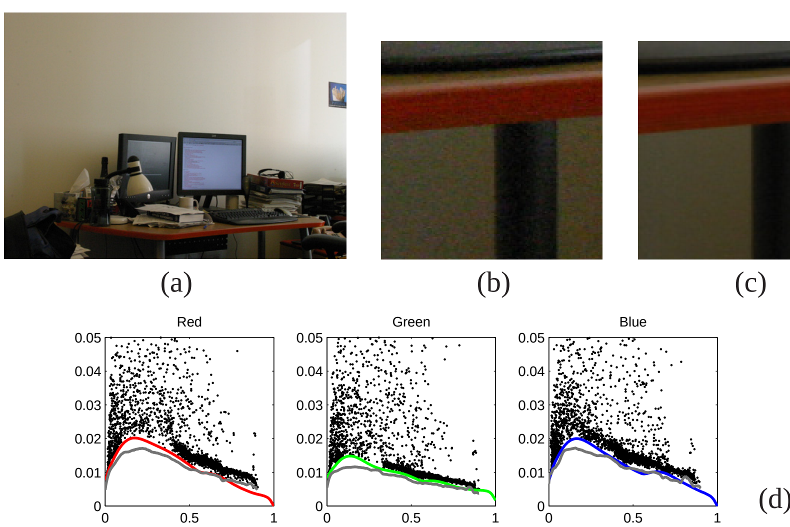


Figure 9. Comparison of estimated camera noise with experimental measurement. (a) shows one of the 29 images taken with a Canon$^{\mathrm{TM}}$EOS 10D. An enlarged patch is shown for (b) a single image, and (c) the mean image. (d) is the estimated NLF from a single image (color), showing good agreement with the ground truth (gray), measured from the noise variations over all 29 images.

Norm	mean	std. deviation
L^2	0.0048	0.0033
L^∞	0.0110	0.0120

Table 1. The statistics of the L^2 and L^∞ norms between the estimated NLF and the ground truth.

ties, although they deviate from the true noise function (in gray) slightly. In (b) the samples do not span the full intensity range, so our estimate is only reliable where the samples appear. This shows a limit of the prior model: the samples are assumed to be well-distributed. The estimation is reliable if the color distribution span the full range of the spectrum and there are textureless regions, as in (c).

We conducted a further experiment as a sanity check. We took 29 images of a static scene by Canon$^{\mathrm{TM}}$EOS 10D (ISO 1600, exposure time 1/30 s and aperture f/19) and computed the mean image. One sample is shown in Figure 9 (a). A close-up view of sample (a) and the mean image is shown in (b) and (c), respectively. Clearly the noise is significantly reduced in the mean image. Using the variance over the 29 images as a function of mean intensity, we calculated the "ground truth" NLF and compared that to the NLF estimated by our method from only one image. The agreement between the NLFs in each color band is very good, see Figure 9 (d).

We also applied the algorithm to estimating noise level functions from the other images taken by a CCD camera. We evaluated our results based on repeatability: pictures taken by the same camera with the same setting on the same date should have the same noise level function, independent of the image content. We collected two pictures taken by a Canon$^{\mathrm{TM}}$ EOS DIGITAL REBEL and estimated the corresponding noise level functions, as shown in Figure 10 (a) and (b). Even though image (a) is missing high intensity values, the estimated NLFs are similar.

5 Segmentation-Based Denoising

Recall from Section 3.2 that the observation $I(v)$ is decomposed to signal $s(v)$ and noise $n(v)$. Given the characteristics of the noise that have been estimated from the previous section, we are now ready to separate the signal and noise from the observation.

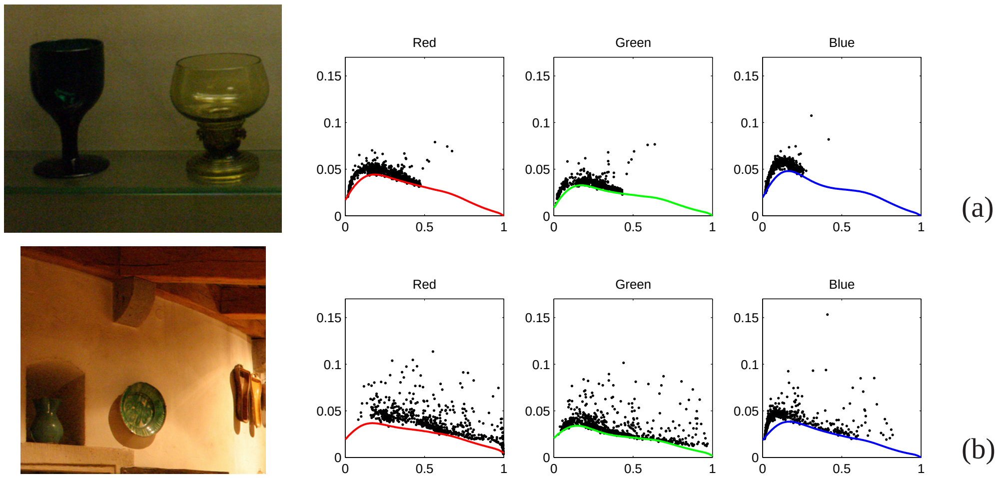


Figure 10. The two images are taken by a Canon$^{\mathrm{TM}}$EOS DIGITAL REBEL and the estimated noise level functions. Very similar noise level functions are derived, even though the two images have very different tonescales.

5.1 0th-Order Model

Let $\mu = [\mu_1\ \mu_2\ \mu_3]^T \in \mathbb{R}^3$ be the mean color for segment Ω after the piecewise smooth image reconstruction to the input image I. Suppose the noise is independent for RGB channels, and we obtain the covariance matrix of noise in this segment

$$\hat{\mathbf{\Sigma}}_n = \mathrm{diag}\big(\tau^2(\mu_1), \tau^2(\mu_2), \tau^2(\mu_3)\big). \tag{17}$$

From the independence assumption of the noise and signal, we obtain (from Equation (2))

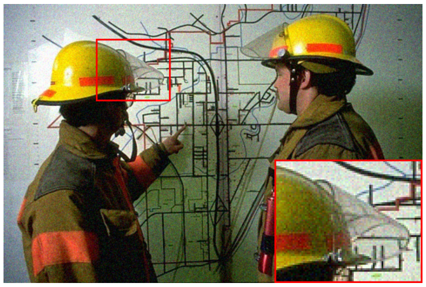
$$\hat{\mathbf{\Sigma}}_s = \mathbf{\Sigma}_r - \hat{\mathbf{\Sigma}}_n. \tag{18}$$

It is possible that the estimated $\hat{\mathbf{\Sigma}}_s$ is not positive definite. For this case we simply enforce the minimum eigenvalue of $\hat{\mathbf{\Sigma}}_s$ to be a small value (0.0001).

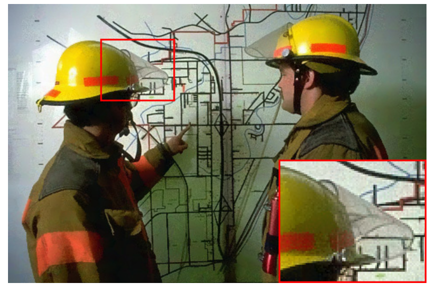
We simply run Bayesian MAP estimation for each pixel to estimate the noise based on the obtained 2nd order statistics. Since

$$\begin{aligned} & p(s(v)|I(v)) \propto p(I(v)|s(v))p(s(v)) \\ & \propto \exp\Big\{-\frac{1}{2}[I(v)-s(v)]^T\hat{\mathbf{\Sigma}}_n^{-1}[I(v)-s(v)]\Big\}\exp\Big\{-\frac{1}{2}[s(v)-f(v)]^T\hat{\mathbf{\Sigma}}_s^{-1}[s(v)-f(v)]\Big\}, \end{aligned} \tag{19}$$

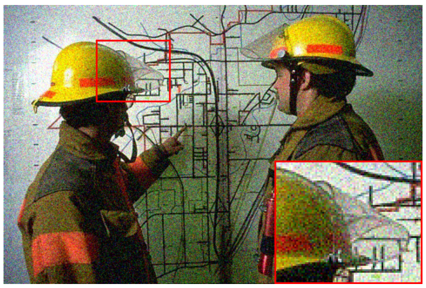
where $f(v)$ is the piecewise smooth reconstruction, the optimal estimation has a simple closed-



(a) 5% AWGN



(b) Denoised by the 0th-order model (PSNR=32.04)



(c) 10% AWGN

(d) Denoised by the 0th-order model (PSNR=27.05)

Figure 11. Noise contaminated images and the denoised results by the 0th-order model. A patch at a fixed position marked by a red rectangle is zoomed-in and inset at the bottom-right of each image. Clearly, the 0th-order model significantly removes the chrominance component of color noise. For high noise level, the discontinuities between the neighboring segments are further removed by the 1st-order model (see Figure 13, 14 and Table 2).

form solution

$$s^*(v) = \arg\max p(s(v)|I(v)) = (\hat{\mathbf{\Sigma}}_n^{-1} + \hat{\mathbf{\Sigma}}_s^{-1})^{-1}(\hat{\mathbf{\Sigma}}_n^{-1} I(v) + \hat{\mathbf{\Sigma}}_s^{-1} f(v)), \tag{20}$$

which simply down-weighs the pixel values from $I(v)$ to $f(v)$ using the covariance matrices as weights. For a scaled identity $\hat{\mathbf{\Sigma}}_n$, it is easy to show that the attenuation along each principal direction in the color covariance matrix is $\lambda_i/(\lambda_i + \sigma_n)$, where λ_i is the variance in the ith direction. Qualitatively, as this variance tends towards zero (either because the non-dominant direction has low variance, or the region is untextured), the cleaned up residual is progressively more attenuated.

Equation (20) is applied to every pixel, where $\hat{\mathbf{\Sigma}}_n$ and $\hat{\mathbf{\Sigma}}_s$ vary from segment to segment. Since there is no spatial relationship of pixels in this model, we call it *0th-order model*. An example of denoising using 0th-order model is shown in Figure 11, where the algorithm is tested by synthetic AWGN with noise levels of 5% and 10%. Clearly the 0th-order model significantly removes the chrominance component of color noise. The results are acceptable for 5% noise level, and we can see discontinuities between the neighboring segments for 10% noise level because the spatial correlation has been accounted for.

5.2 1st-Order Gaussian Conditional Random Field

The values of the neighboring pixels are correlated in natural images. We chose to regularize with a conditional random field (CRF) [26, 47] where the spatial correlation is a function of the local patch of the input image, over the Markov random field (MRF) [18] to avoid having a complete prior models on images as in [40], Moreover, we model it as a Gaussian CRF since all the energy functions are quadratic. We call it *1st-order model* because the spatial correlation is captured by the 1st-order derivative filters. Likewise, we can have 2nd-order model or even higher order. But we found that 1st-order model is sufficient for the denoising task.

Let the estimated covariance matrices of signal and noise be $\hat{\mathbf{\Sigma}}_s(i)$ and $\hat{\mathbf{\Sigma}}_n(i)$ for segment Ω_i. The CRF is formulated as

$$p(s|I) = \frac{1}{Z}\exp\left\{-\frac{1}{2}\sum_i\sum_{v\in\Omega_i}\left[\Big(s(v)-I(v)\Big)^T\hat{\mathbf{\Sigma}}_n^{-1}(i)\Big(s(v)-I(v)\Big)+\right.\right.$$
$$\left.\left.\Big(s(v)-f(v)\Big)^T\hat{\mathbf{\Sigma}}_s^{-1}(i)\Big(s(v)-f(v)\Big)+\xi_i w(v)\sum_{j=1}^{m}F_j^2(v)\right]\right\}. \qquad (21)$$

In the above equation, $F_j = \phi_j * s$ is the filter response of s being convolved with filter ϕ_j. For this 1st-order GCRF, we choose horizontal and vertical filters (i.e., m=2). $w(v)$ and ξ_i are both weights to balance the importance of spatial correlation. ξ_i is the weight for each segment. We find that ξ_i can be a linear function of the mean noise level in segment Ω_i. $w(v)$ is derived from the filter responses of the original image. Intuitively, $w(v)$ should be small when there is clear boundary at v to weaken spatial correlation, and be large when there is no boundary to strengthen spatial correlation. Boundaries can be detected by Canny edge detection [6], but we found that the algorithm is more stable when $w(v)$ is set to be a function of local filter responses. We use orientationally elongated Gabor sine and cosine filters [17] to capture the boundary energy of the underlying noise-free image. The boundary energy is the sum over all the orientations and sin/cos phases. We then use a nonlinear function to map the energy to the local value of the weight matrix, i.e., $y = (1-\tanh(t_1 x))^{t_2}$ where $t_1 = 0.6$ and $t_2 = 12$ in our implementation. Solving Equation (21) is equivalent to solving a linear system, which can be effectively computed by conjugate gradient method that only needs iterations of linear filtering.

6 Experimental Results on Image Denoising

Our automatic image denoising system consists of two parts, noise estimation and denoising. To have a fair comparison with other denoising algorithms, we first test our denoising algorithms using synthetic AWGN with constant and known noise level (σ). Then the whole system is tested with the images contaminated with real CCD camera noise.

6.1 Synthetic AWGN

We selected 17 images covering different types of objects and scenes from the Berkeley segmentation dataset [31] and added AWGN with 5% and 10% noise level to test our denoising algorithm. The noise contaminated images with 10% noise level are shown in Figure 12. We also ran standard bilateral filtering [51] (our implementation), curvature preserving PDE [52] (publicly available implementation[2]) and wavelet joint coring, GSM [37] (publicly available implementation[3]). Default parameter settings are used for the downloaded code. For curvature preserving PDE, we tweaked the parameters and found that the best results can be obtained by setting $alpha = 1,\ iter = 4$ for $\sigma = 10\%$ and $alpha = 0.5\ iter = 7$ for $\sigma = 5\%$. We compare their results to our own using both visual inspection in Figure 13 and 14, and peak signal to noise ratio (PSNR) statistics in Table 2.

It is clear that our technique consistently outperforms bilateral filtering, curvature preserving PDE and wavelet joint coring. In terms of PSNR, our technique outperforms these algorithms by a significant margin. When $\sigma = 0.05$, i.e., the noise level is low, even the 0th-order model outperforms the state-of-the-art wavelet GSM. When $\sigma = 0.10$, i.e., the noise level is high, the 1st-order model outperforms wavelet GSM by 1.3 PSNR on average.

The results are also visually inspected in Figure 13 from (a) to (e), corresponding to image 35008, 23084, 108073, 65010 and 66075 in Figure 12, respectively. Some close-up views of the denoising results are shown in Figure 14. The curvature preserving PDE method generates color fringing artifacts around the strong edges. Wavelet coring tends to produce color and blurring artifacts, especially in (a) and (d). Our algorithm, however, is able to smooth out flat regions, preserve sharp edges, as well as keep subtle texture details. In Figure 13 and 14 (a), our algorithm achieved sharper boundaries of the bug and preserved the texture of the flower. In (b), many curves with a variety of width are well reconstructed, whereas the wavelet coring

[2] `http://www.greyc.ensicaen.fr/~dtschump/greycstoration/download.html`
[3] `http://decsai.ugr.es/~javier/denoise/`

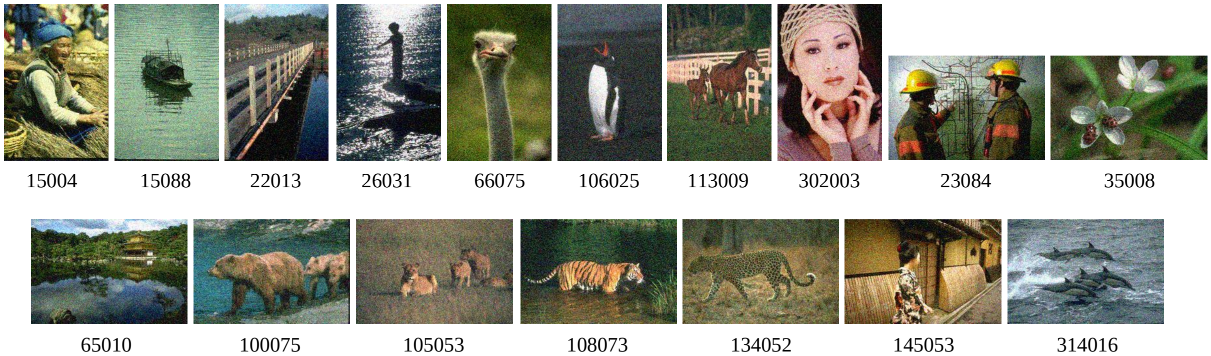
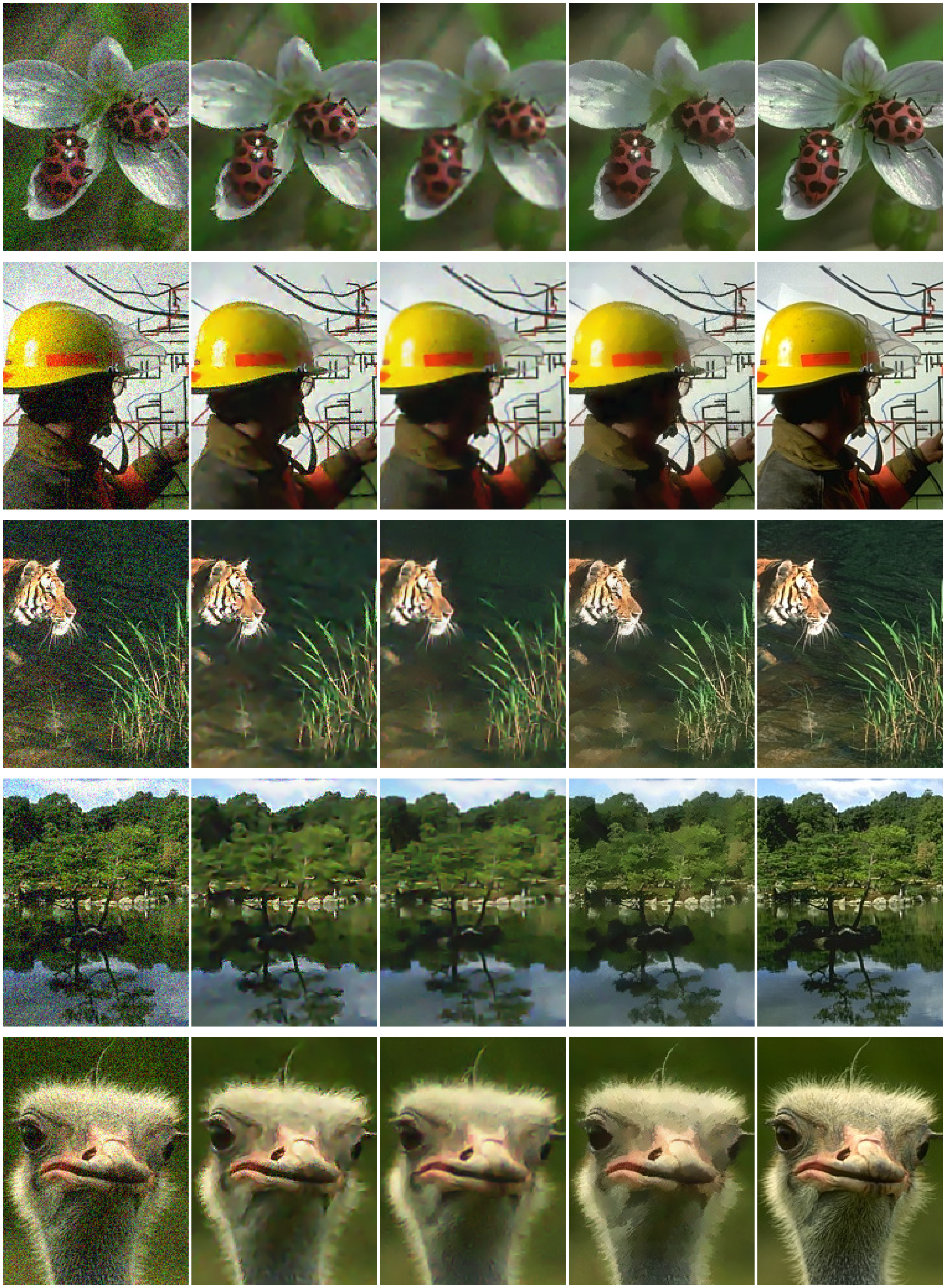


Figure 12. Seventeen images are selected from Berkeley image segmentation database [31] to evaluate the proposed algorithm. The file names (numbers) are shown beneath each picture

PSNR	$\sigma = 5\%$					$\sigma = 10\%$				
File name	bilat	PDE	wavelet	0th	1st	bilat	PDE	wavelet	0th	1st
100075	29.32	29.76	31.27	31.69	31.68	26.47	27.72	28.31	28.14	28.96
105053	32.33	32.54	34.01	33.77	34.02	30.05	30.56	31.41	30.63	31.95
106025	34.47	34.29	36.13	35.75	36.44	30.94	31.58	32.57	32.03	34.22
108073	29.98	29.84	31.48	31.94	31.98	25.61	26.98	27.94	28.33	29.21
113009	30.73	30.72	32.89	32.31	32.61	27.38	27.80	29.91	28.89	30.19
134052	30.02	30.03	32.09	32.58	32.88	25.71	27.38	28.20	28.61	29.55
145053	29.51	29.12	31.72	31.88	32.26	23.84	25.83	27.23	27.46	28.71
15004	28.61	28.24	30.74	30.98	31.51	23.38	24.77	26.35	25.58	27.50
15088	29.55	29.19	33.36	32.41	32.74	25.02	26.64	28.83	27.71	28.76
22013	29.92	29.50	31.31	32.17	32.33	25.09	26.42	27.12	27.14	28.84
23084	30.31	29.76	32.14	32.04	32.64	24.63	26.34	27.24	27.05	29.23
26031	28.76	27.93	28.87	31.20	31.24	21.58	22.97	23.95	25.55	26.65
302003	31.29	30.93	33.70	32.94	33.96	26.85	27.43	29.47	27.91	30.84
314016	28.43	29.26	31.28	31.57	31.44	25.00	26.93	27.64	26.81	27.83
35008	33.28	33.40	35.74	34.84	35.97	29.25	30.85	31.23	30.24	33.27
65010	29.62	29.46	30.95	31.99	32.18	25.20	26.45	26.73	27.18	28.41
66075	32.57	32.46	33.36	35.02	35.03	28.33	29.78	29.69	29.79	31.80
mean	30.51	30.78	32.41	32.65	32.99	26.14	27.43	28.46	28.18	29.76

Table 2. PSNR for the images in Berkeley image segmentation database. “bilat”, “PDE”, “wavelet”, “0th” and “1st” stand for bilateral filtering [51], curvature preserving PDE [52], wavelet (GSM) [37], 0th-order and 1st-order model, respectively. The images with green are cropped, zoomed-in, and displayed in Figure 13.

introduced color fringing artifacts around boundaries. In (c), the whiskers of the tiger are sharper and clearer by our algorithm, and so are the stems and leaves of the grasses. In (d), the texture details of the leaves are preserved, while the clouds are well smoothed. The ostrich head in (e) is a failure example, where the upper neck part is over-smoothed and some artificial



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

10% AWGN | PDE | Wavelet GSM | Ours (1st-order model) | Original

Figure 13. Close-up view of the denoising results. See text for the explanation.

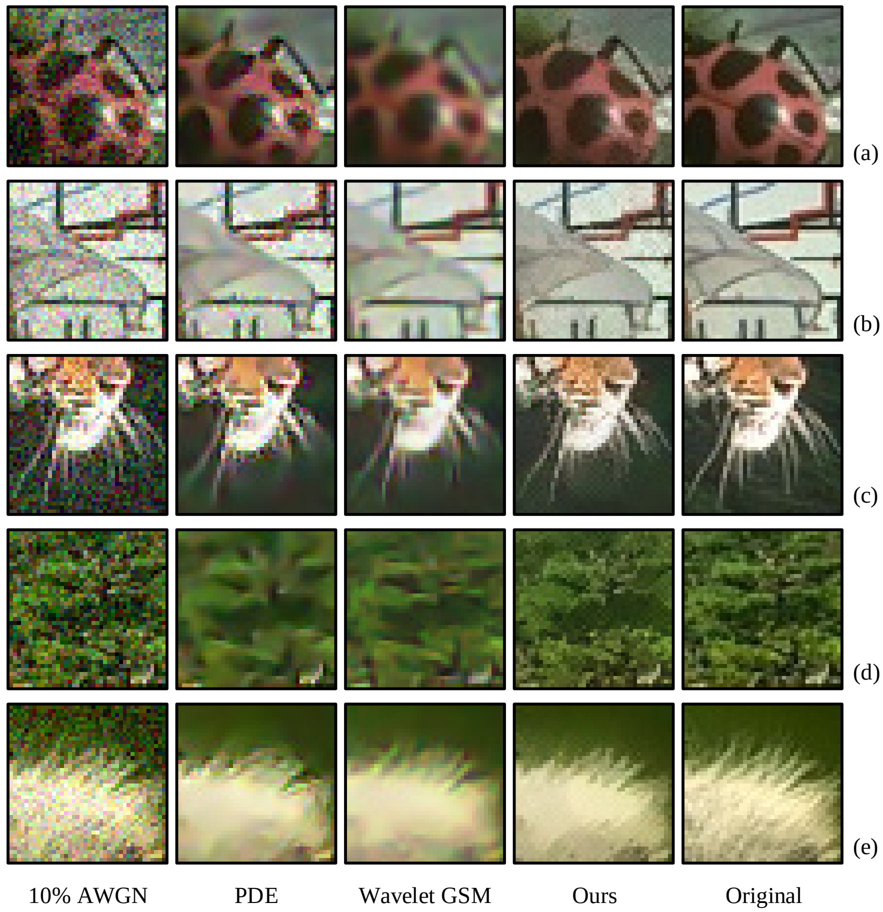


Figure 14. Some close-up views of the denoising results in Figure 13 are shown here. Our algorithm generated crispier images without color fringing artifacts as produced by PDE [52] and wavelet GSM [37] approaches.

boundaries are generated for the furry edges. Note that our system does not always completely remove the noise for the texture regions, but it looks visually pleasing since the chrominance component of the noise is removed. In addition, the remaining noise in the texture regions as in (d) is not noticeable.

Overall, our algorithm outperforms the state-of-the-art denoising algorithms on the synthetic noise case. It takes our un-optimized MATLAB$^{\mathrm{TM}}$implementation less than one minute on average to denoise one picture (with a typical resolution of 481×321) in the Berkeley database. Our experiments were run on a 2.8 GHz Pentium D PC.

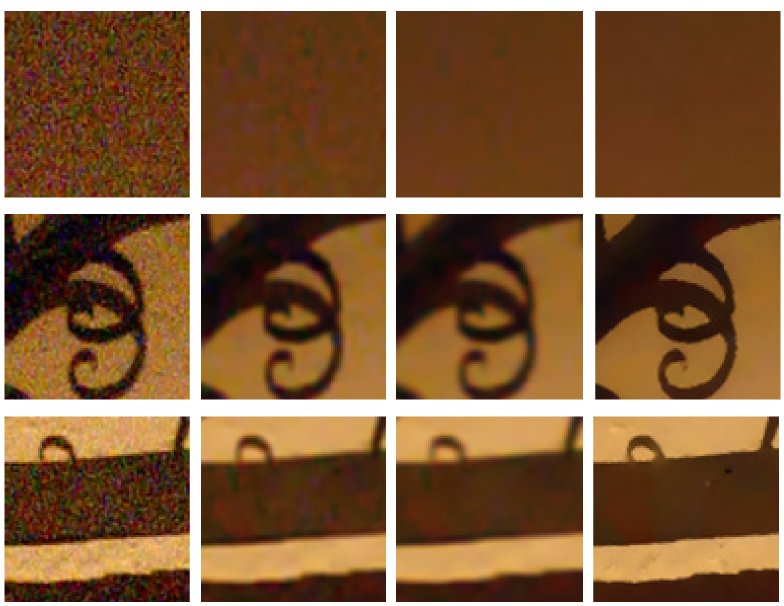
(a) Noisy input

(b) Wavelet GSM $\sigma = 10\%$

(c) Wavelet GSM $\sigma = 15\%$

(d) Ours

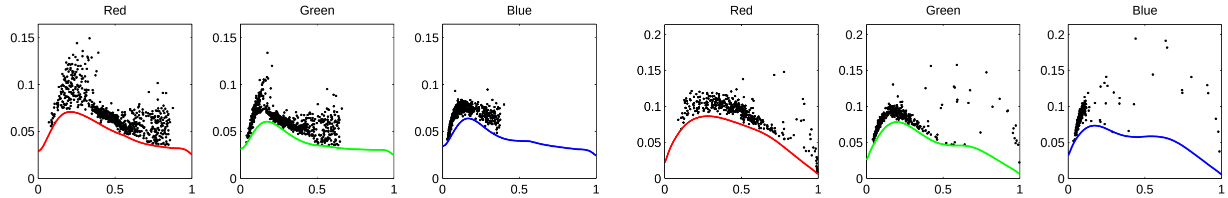
Figure 15. Comparison of denoising algorithms on a real CCD camera noise input.



(1) (2) (3)

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

Figure 16. Close-up view of the denoising results in Figure 15.



(a) (b)

Figure 17. Estimated noise level functions. (a): NLFs for the noisy sample in Figure 15. (b): NLFs for the example in Figure 18.

(a) Noisy input (b) Denoised by wavelet GSM [37] (c) Denoised by our algorithm

Figure 18. The denoising results of a very challenging example.

6.2 Real CCD Noise

We further tested our automatic denoising system using the pictures taken by CCD cameras with remarkable noise [35]. The picture in Figure 15 (a) was taken by Canon$^{\mathrm{TM}}$EOS DIGITAL REBEL, with intense noise for the dim pixels, but less for the bright ones. The noise level functions are estimated and displayed in Figure 17, which agree with the observation. To compare, we also run wavelet coring (GSM), with $\sigma = 10\%$ and $\sigma = 15\%$, and the results are shown in Figure 15 (b) and (c), respectively. The denoising result automatically generated by our system is shown in (d). The close-up inspections of these results are shown in Figure 16. Clearly with the constant noise level assumption, the wavelet coring algorithm cannot balance the high and low noise areas. When $\sigma = 10\%$ it does a better job for the bright pixels with sharper edges, and when $\sigma = 15\%$ it does a better job for the dark pixels with smoother regions. But overall we can still see blocky color artifacts, overly smoothed boundaries and loss of texture details. The result produced by our system successfully overcomes these problems. In Figure 16 row (1) our method produces almost flat patch. In row (2) the boundary is much sharper, whereas in row (3) many subtle texture details are preserved. Overall our algorithm generates visually more appealing result (we cannot compute PSNR since there is no ground truth clean image).

We tested our algorithm on another challenging example shown in Figure 18 (a). As shown in (b) the wavelet coring cannot effectively remove the color noise because of the spatial correlation of the color noise. Even though the result generated by our automatic denoising system in (c) overly sharpens the edges to have cartoon style, the noise gets completely removed and the image looks visually more pleasing.

Since our segmentation algorithm generates bigger segments for flat regions and smaller segments for texture regions, the parameter setting of segmentation does not significantly influence the overall performance of the system if it is within a reasonable range. We used the same parameter settings for the benchmark test, and found that visually more pleasing results are achieved for the real CCD noise examples in Figure 18 if bigger segments are allowed (unfortunately we could not measure the PSNR). Certainly better segmentation will further improve the denoising system, but our current segmentation algorithm is sufficient.

7 Discussion

Having shown the success of our model using both synthetic and real noise, we want to provide some insights to the denoising problem and our modeling.

7.1 Color Noise

As shown in Section 3, the color of the pixels in a segment is approximately distributed along a 1D subspace of the three-dimensional RGB space. This agrees with the fact that the strong sharp boundaries are mainly produced by the change of materials (or reflectance), whereas the weak smooth boundaries are mainly produced by the change of lighting [48]. Since the human vision system is accustomed to these patterns, color noise, which breaks the 1D subspace rule, appears annoying to our eyes. Our denoising system was designed based on this 1D subspace rule to effectively remove the chrominance component of color noise. The results of our 0th-order model in Figure 11 demonstrate that the images look significantly more pleasing when the chrominance component is removed.

7.2 Conditional Model vs. Generative Model

In our system we do segmentation only once to obtain a piecewise smooth model of the input image. If we treat the region partitions as a hidden variable that generates the noise image, then the conditional model becomes a generative model. Inference in the generative model will require integration over the region partitions. Intuitively, the segmentation of the noisy input image could be noisy and unreliable and could result in many possible segmentations.

One way of this full Bayesian approach is to sample partitions from the input image, obtain the denoised image for each segmentation, and compute the mean as the output. This approach would possibly improve the results by removing some of the boundary artifacts, but it is intractable in practice because of the huge space of partitions. Another way is to treat the partition as missing data and use expectation-maximization (EM) algorithm to iterates between segmenting the image based on the denoised image (E-step), and estimating the denoised image based on the segmentation (M-step). This approach is also intractable in practice because many iterations are required. Nevertheless, these full Bayesian approaches might be promising directions for future segmentation-based image processing systems with more powerful computation.

7.3 Automation of Computer Vision System

The performance of a computer vision system is sensitive to peripheral parameters, e.g., noise level, blur level, resolution/image quality, lighting and view point. For example, for image denoising the noise level is an important parameter to the system. Poor results may be produced with a wrong estimate of the noise level. Most existing computer vision algorithms focus on addressing the problems with known peripheral parameters, but the algorithms have to be tweaked to fit different imaging conditions. Therefore, it is an important direction to make computer vision systems account for the important peripheral parameters to be fully automatic. Our automatic image denoising system is one of the first attempts to make the denoising algorithm robust to noise level.

8 Conclusion

Based on a simple piecewise-smooth image prior, we proposed a segmentation-based approach to automatically estimate and remove noise from color images. The NLF is obtained by estimating the lower envelope of the standard deviations of image variance per segment. The chrominance of the color noise is significantly removed by projecting the RGB pixel values to a line in color space fitted to each segment. The noise is removed by formulating and solving a Gaussian conditional random field. Experiments were conducted to test both the noise estimation and removal algorithms.

We verified that the estimated noise level is a tight upper bound of the true noise level in three ways: (1) by showing good agreement with experimentally measured noise from repeated exposures of the same image, (2) by repeatedly measuring the same NLF with the same camera for different image content, and (3) by accurately estimating known synthetic noise functions. Our noise estimation algorithm can be applied to not only denoising algorithms, but other computer vision applications to make them independent of noise level [29].

Our denoising algorithm outperforms the state-of-the-art wavelet denoising algorithms on both synthetic and real noise-contaminated images by generating shaper edges, producing smoother flat regions and preserving subtle texture details. These features match our original criteria we proposed for a good denoising algorithm.

9 Acknowledgement

Ce Liu is supported by a Microsoft Fellowship. Funding for this research was provided by NGA NEGI-1582-04-0004 and Shell Research.

References

[1] S. P. Awate and R. T. Whitaker. Higher-order image statistics for unsupervised, information-theoretic, adaptive, image filtering. In *Proc. IEEE Conf. Computer Vision and Patter Recognition*, 2005.

[2] D. Barash. A fundamental relationship between bilateral filtering, adaptive smoothing, and the non-linear diffusion equation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(6):844–847, 2002.

[3] M. J. Black, G. Sapiro, D. H. Marimont, and D. Heeger. Robust anisotropic diffusion. *IEEE Trans. on Image Processing*, 7(3):421–432, 1998.

[4] A. Blake and A. Zisserman. *Visual Reconstruction*. MIT Press, Cambridge, Massachusetts, 1987.

[5] A. Buades, B. Coll, and J.-M. Morel. A non-local algorithm for image denoising. In *Proc. IEEE Conf. Computer Vision and Patter Recognition*, 2005.

[6] J. Canny. A computational approach to edge detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

[7] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24:603–619, 2002.

[8] M. S. Crouse, R. D. Nowak, and R. G. Baraniuk. Wavelet-based statistical signal processing using hidden Markov models. *IEEE Trans. on Signal Processing*, 46(4):886–902, 1998.

[9] D. Donoho. De-noising by soft-thresholding. *IEEE Trans. on Information Theory*, 41(3):613–627, 1995.

[10] F. Durand and J. Dorsey. Fast bilateral filtering for the display of high-dynamic-range images. In *SIGGRAPH*, pages 257–266, 2002.

[11] M. Elad and M. Aharon. Image denoising via learned dictionaries and sparse representation. In *Proc. IEEE Conf. Computer Vision and Patter Recognition*, 2006.

[12] M. Evans, N. Hastings, and B. Peacock. *Statistical Distributions*. A Wiley-Interscience Publication, 2000.

[13] G. Fan and X.-G. Xia. Image denoising using a local contextual hidden Markov model in the wavelet domain. *IEEE Signal Processing Letters*, 8(5):125–128, 2001.

[14] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *Int'l Journal of Compter Vision*, 59:167–181, 2004.

[15] D. Field. Relations between the statistics of natural images and the response properties of cortial cells. *J Opt Soc Am A*, 4(12):2379–2394, Dec 1987.

[16] W. Förstner. Image preprocessing for feature extraction in digital intensity, color and range images. In *Springer Lecture Notes on Earth Sciences*, 1998.

[17] D. Gabor. Theory of communication. *Journal of IEE*, 93(26):429–457, 1946.

[18] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the bayesian restoration of images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.

[19] T. Gevers and A. W. M. Smeulders. Color-based object recognition. *Pattern Recognition*, 32(3):453–464, 1999.

[20] E. B. Gindele and N. Serrano. Us patent #7,054,501: Estimating noise for a digital image utilizing updated statistics, 2006.

[21] R. T. Gray and D. R. Cok. Us patent #5,641,596: Adjustint film grain properties in digital images, 1997.

[22] M. D. Grossberg and S. K. Nayar. Modeling the space of camera response functions. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(10):1272–1282, October 2004.

[23] G. Healey. Segmenting images using normalized color. *IEEE Trans. on Systems, Man and Cybernetics*, 22(1):64–73, 1992.

[24] G. E. Healey and R. Kondepudy. Radiometric CCD camera calibration and noise estimation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(3):267–276, March 1994.

[25] G. Klinker, S. Shafer, and T. Kanade. A physical approach to color image understanding. *Int'l Journal of Compter Vision*, 4(1):7–38, January 1990.

[26] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of Int'l Conference on Machine Learning*, 2001.

[27] C. Laroche and M. Prescott. *Apparatus and Methods for Adaptively Interpolating a Full Color Image Utilizaing Chrominance Gradients*. U.S. Patent No. 5,373,322 (1994).

[28] A. B. Lee, D. Mumford, and J. Huang. Occlusion models for natural images: A statistical study of a scale-invariant dead leaves model. *Int'l Journal of Compter Vision*, 41(1/2):35–59, 2001.

[29] C. Liu, W. T. Freeman, R. Szeliski, and S. B. Kang. Noise estimation from a single image. In *Proc. IEEE Conf. Computer Vision and Patter Recognition*, pages 901–908, Jun 2006.

[30] S. G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 11(7):674–693, 1989.

[31] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. IEEE Int'l Conf. Computer Vision*, volume 2, pages 416–423, July 2001.

[32] D. Mumford and B. Gidas. Stochastic models for generic images. *Quaterly Journal of Applied Mathematics*, LIX(1):85–111, 2001.

[33] S. Paris and F. Durand. A fast approximation of the bilateral filter using a signal processing approach. In *Proc. European Conference on Computer Vision*, 2006.

[34] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(7):629–639, 1990.

[35] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama. Digital photography with flash and no-flash image pairs. *ACM Trans. Graph.*, 23(3):664–672, 2004.

[36] J. Portilla. Full blind denoising through noise covariance estimation using gaussian scale mixtures in the wavelet domain. In *Proc. IEEE Int'l Conf. Image Proc.*, pages 1217–1220, 2004.

[37] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli. Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Trans. Image Processing*, 12(11):1338–1351, Nov 2003.

[38] R. Ramanath, W. E. Snyder, G. L. Bilbro, and W. A. Sander. Demosaicking methods for Bayer color arrays. *Journal of Electronic Imaging*, 11(3):306–315, July 2002.

[39] J. Rossi. Digital techniques for reducing television noise. *JSMPTE*, 87:134–140, 1978.

[40] S. Roth and M. J. Black. Fields of experts: A framework for learning image priors. In *Proc. IEEE Conf. Computer Vision and Patter Recognition*, 2005.

[41] W. F. Schreiber. *Fundamentals of electronic imaging systems*. Springer-Verlag, 1986.

[42] E. P. Simoncelli. Statistical models for images: Compression, restoration and synthesis. In *31st Asilomar Conf on Signals, Systems and Computers*, pages 673–678, 1997.

[43] E. P. Simoncelli and E. H. Adelson. Noise removal via Bayesian wavelet coring. In *Proc. IEEE Int'l Conf. Image Proc.*, volume I, pages 379–382, 1996.

[44] P. D. Snyder, T. F. Pawlicki, and R. S. Gaborski. Us patent #5,923,775: Apparatus and method for signal dependent noise estimation and reduction in digital images, 1999.

[45] A. Stefano, P. White, and W. Collis. Training methods for image noise level estimation on wavelet components. *EURASIP Journal on Applied Signal Processing*, 16:2400–2407, 2004.

[46] G. Strang. *Introduction to Applied Mathematics*. Wellesley-Cambridge Press, 1986.

[47] M. F. Tappen, E. H. Adelson, and W. T. Freeman. Estimating intrinsic component images using non-linear regression. In *Proc. IEEE Conf. Computer Vision and Patter Recognition*, 2006.

[48] M. F. Tappen, W. T. Freeman, and E. H. Adelson. Recovering intrinsic images from a single image. In *Advances in Neural Information Processing System*, pages 1343–1350, 2003.

[49] M. F. Tappen, C. Liu, E. H. Adelson, and W. T. Freeman. Learning Gaussian conditional random fields for low-level vision. In *Proc. IEEE Conf. Computer Vision and Patter Recognition*, 2007.

[50] D. Terzopoulos. Regularization of inverse visual problems involving discontinuities. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8(4):413–424, July 1986.

[51] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *ICCV '98: Proceedings of the Sixth International Conference on Computer Vision*, pages 839–846, 1998.

[52] D. Tschumperlé. Fast anisotropic smoothing of multi-valued images using curvature-preserving PDE's. *Int'l Journal of Compter Vision*, 68(1):65–82, 2006.

[53] Y. Tsin, V. Ramesh, and T. Kanade. Statistical calibration of CCD imaging process. In *Proc. IEEE Int'l Conf. Computer Vision*, pages 480–487, 2001.

[54] S. C. Zhu and D. Mumford. Prior learning and Gibbs reaction-diffusion. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(11):1236–1250, 1997.

[55] S. C. Zhu, Y. Wu, and D. Mumford. Filters, random fields and maximum entropy (frame): Towards a unified theory for texture modeling. *Int'l Journal of Compter Vision*, 27(2):107–126, 1998.

[56] C. L. Zitnick, N. Jojic, and S. B. Kang. Consistent segmentation for optical flow estimation. In *Proc. IEEE Int'l Conf. Computer Vision*, 2005.