

# Single-Image Vignetting Correction

Yuanjie Zheng, *Member, IEEE*, Stephen Lin, *Member, IEEE*, Chandra Kambhamettu, *Member, IEEE*, Jingyi Yu, *Member, IEEE*, Sing Bing Kang, *Senior Member, IEEE*

**Abstract**—In this paper, we propose a method for robustly determining the vignetting function given only a single image. Our method is designed to handle both textured and untextured regions in order to maximize the use of available information. To extract vignetting information from an image, we present adaptations of segmentation techniques that locate image regions with reliable data for vignetting estimation. Within each image region, our method capitalizes on the frequency characteristics and physical properties of vignetting to distinguish it from other sources of intensity variation. Rejection of outlier pixels is applied to improve the robustness of vignetting estimation. Comprehensive experiments demonstrate the effectiveness of this technique on a broad range of images with both simulated and natural vignetting effects. Causes of failures using the proposed algorithm are also analyzed.

**Index Terms**—Vignetting correction, camera calibration, low-level vision.

## I. INTRODUCTION

Vignetting refers to the phenomenon of brightness attenuation away from the image center, and is an artifact that is prevalent in photography (see Fig. 1 (a) for an example). Although not objectionable to the average viewer, it can significantly impair computer vision algorithms that rely on precise intensity data to analyze a scene. Applications in which vignetting distortions can be particularly damaging include photometric methods such as shape from shading, appearance-based techniques such as object recognition, and image mosaicing. As an example, see the banding effects in Fig. 1 (b) for the segmentation of the image in Fig. 1 (a).

There are several factors that contribute to vignetting. Some arise from the optical properties of camera lenses, the most prominent of which is off-axis illumination falloff or the  $\cos^4$  law. Here, vignetting is a result of foreshortening of the lens when viewed at increasing angles from the optical axis [13]. Other sources of vignetting are geometric in nature—light arriving at oblique angles to the optical axis may be partially obstructed by the field stop or lens rim.

To determine the vignetting effects in an image, the most straightforward approach involves capturing an image completely spanned by a uniform scene region, such that brightness variations can solely be attributed to vignetting [1], [11], [20], [28], [29]. In such a calibration image, ratios of intensity with respect to the pixel on the optical axis describe the vignetting

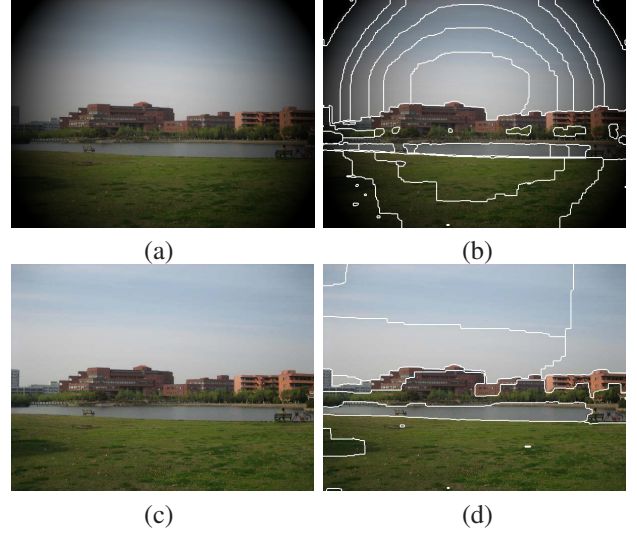


Fig. 1. (a) Image with added vignetting to draw interest to the building. (b) Banding effects in segmentation caused by vignetting. (c) Image after vignetting is estimated and removed from (a). (d) Segmentation of the image in (c).

function. Suitable imaging conditions for this approach, however, can be challenging to produce due to uneven illumination and camera tilt, and the vignetting measurements are valid only for images captured by the camera under the same camera settings. Moreover, a calibration image can be recorded only if the camera is at hand; consequently, this approach cannot be used to correct images captured by unknown cameras, such as images downloaded from the web.

A vignetting function can alternatively be computed from image sequences with overlapping views of an arbitrary static scene [7], [8], [12], [15]. In this approach, point correspondences are first determined in the overlapping image regions. Since a given scene point has a different position in each image, its brightness may be differently attenuated by vignetting. From the aggregate attenuation information from all correspondences, the vignetting function can be accurately recovered without any assumptions on the scene. The image sequence can alternatively be obtained by capturing projector images at different exposures and different aperture settings [10].

These previous approaches require either a collection of overlapping images or an image of a calibration scene. However, often in practice only a single image of an arbitrary scene is available, e.g., when applying vignetting correction to images downloaded from the web.

Previous techniques derive information for vignetting correction from pixels with equal scene radiance but differing

Y. Zheng, C. Kambhamettu and J. Yu are with Department of Computer and Information Sciences, University of Delaware, Newark, DE 19716-2712. E-mail: zhengyuanjie@gmail.com and {chandra,yu}@cis.udel.edu.

S. Lin is with Microsoft Research Asia, 5/F, Beijing Sigma Center No.49, Zhichun Road Hai Dian District Beijing, China 100080. E-mail: stevelin@microsoft.com.

S.B. Kang is with Microsoft Corporation, One Microsoft Way, Redmond, WA 98052. E-mail: sbkang@microsoft.com.

attenuation of brightness. For a single arbitrary input image, this information becomes challenging to obtain, since it is difficult to identify pixels having the same scene radiance while differing appreciably in vignetting attenuation.

In this paper, we present a method for single-image vignetting correction that acquires vignetting data by finding image regions in which vignetting can be distinguished from other sources of intensity variation. These regions are located using adaptive segmentation techniques designed to reveal image areas with uniform scene radiance characteristics and gradual brightness variations that can be attributed to vignetting. To maximize the use of available information in the image, our technique extracts vignetting data from *both textured and untextured regions*. In extracting vignetting information from a given region, we take advantage of physical vignetting characteristics to diminish the influence of textures and other intensity variations. With the vignetting information discerned from various disparate regions, the vignetting function is estimated.

In principle, if segmentation is perfect, the vignetting parameters can be estimated directly. However, vignetting typically causes inaccuracies in segmentation that can result in suboptimal estimation. We handle this problem by iteratively segmenting, estimating the vignetting parameters, and correcting the image, with the reasonable assumption that the accuracy of estimated parameters improves after every iteration.

The remainder of the paper is structured as follows. Section II briefly discusses our imaging model and vignetting model. Section III presents the problem definition and an overview of our method. Section IV describes vignetting estimation from a segmented image. Section V details the image segmentation process with a given estimate of the vignetting function. Section VI explains how important parameters related to vignetting estimation and segmentation are determined. Section VII demonstrates the effectiveness of our approach with experiments on a wide variety of images. Concluding remarks are given in Section VIII.

## II. IMAGING MODEL WITH VIGNETTING

For a perfect lens, the imaging model can be approximated as  $\mathcal{I} = f(\mathcal{R} + \epsilon_1) + \epsilon_2$  [25], with  $\mathcal{I}$  being the observed image,  $\mathcal{R}$  being the scene radiance,  $f$  being the radiometric response function of the imaging system,  $\epsilon_1$  being mostly shot-noise and thermal noise, and  $\epsilon_2$  being other noise effects (quantization error, amplifier noise, D/A and A/D noise).

Since vignetting occurs at the lens, the scene radiance is modulated by a vignetting function  $\mathcal{V}$ , yielding the modified imaging model  $\mathcal{I} = f(\mathcal{R} \cdot \mathcal{V} + \epsilon_1) + \epsilon_2$ , with “ $\cdot$ ” representing pixelwise multiplication. Assuming that the image is captured under well-lit conditions,  $\epsilon_1 \gg \epsilon_2$ , and we can ignore  $\epsilon_2$  to obtain

$$\mathcal{I} = f(\mathcal{R} \cdot \mathcal{V} + \epsilon_1). \quad (1)$$

In this paper, our goal is to estimate  $\mathcal{V}$  given  $\mathcal{I}$ . To accomplish this, we can either solve for  $\mathcal{V}$  at discrete radii (where  $\mathcal{V}$  is a rotationally symmetric function), or solve for parameters of a physical vignetting model. We chose to use

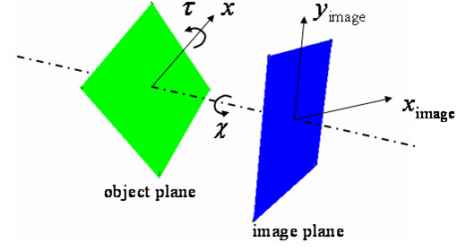


Fig. 2. Tilt angles  $\tau$  and  $\chi$  in the Kang-Weiss vignetting model.

a physical model for two reasons: (1) the parameters tell us about the nature of vignetting (thus providing us hints on the lens type), and (2) there is robustness to noise or sparse data.

To model  $\mathcal{V}$ , most methods for vignetting correction use a parametric vignetting model to simplify estimation and minimize the influence of image noise. Typically used are empirical models such as polynomial functions [7], [20] and hyperbolic cosine functions [28]. Models based on physical considerations include that of Asada *et al.* [1], which accounts for off-axis illumination and light path obstruction, and that of Kang and Weiss [11] which additionally incorporates scene-based tilt effects. Tilt describes intensity variations within a scene region that are caused by differences in distance from the camera, i.e., closer points appear brighter due to the inverse square law of illumination. This is particularly true for indoor scenes. The tilt results in an elliptical iso-intensity distribution, and is handled by the Kang-Weiss model. Although not intrinsic to the imaging system, the intensity attenuation effects caused by tilt must be accounted for in single-image vignetting estimation.

In this work, we use an extension of the Kang-Weiss model, originally designed for a single planar surface of constant albedo, to multiple surfaces of possibly different color. Additionally, we generalize its linear model of geometric vignetting to a polynomial form.

### A. Kang-Weiss model

The Kang-Weiss model [11] assumes that the scene radiance is modulated in the imaging process by a vignetting-tilt function  $\varphi$  which consists of not only the vignetting  $\mathcal{V}$  but also a tilt effect  $\mathcal{T}$ . With this model, Eq. (1) can be rewritten as

$$\mathcal{I} = f(\mathcal{R} \cdot \varphi + \epsilon_1). \quad (2)$$

In the model, the vignetting  $\mathcal{V}$  is represented by an off-axis illumination factor  $\mathcal{A}$  and a geometric factor  $\mathcal{G}$ . We consider an image with zero skew, an aspect ratio of 1, and principal point at the image center with image coordinates  $(u, v) = (0, 0)$ . For a pixel  $i$  at  $(u_i, v_i)$  with distance  $r_i$  from the image center,  $\varphi_i$  is expressed with  $\mathcal{A}$ 's value  $A_i$ ,  $\mathcal{G}$ 's value  $G_i$ ,  $\mathcal{T}$ 's value  $T_i$  and  $\mathcal{V}$ 's value  $V_{r_i}$  as

$$\varphi_i = A_i G_i T_i = V_{r_i} T_i \quad \text{for } i = 1 \cdots N, \quad (3)$$

where

$$\begin{aligned} A_i &= \frac{1}{(1 + (r_i/f)^2)^2}, \\ G_i &= (1 - \alpha_1 r_i), \\ V_{r_i} &= A_i G_i, \\ T_i &= \cos \tau \left( 1 + \frac{\tan \tau}{f} (u_i \sin \chi - v_i \cos \chi) \right)^3. \end{aligned} \quad (4)$$

$N$  is the number of pixels in the image,  $f$  is the effective focal length of the camera, and  $\alpha_1$  represents a coefficient in the geometric vignetting factor. The tilt parameters  $\chi, \tau$  respectively describe the rotation angle of a planar scene surface around an axis parallel to the optical axis, and the rotation angle around the  $x$ -axis of this rotated plane, as illustrated in Fig. 2. Note that  $\mathcal{V}$  is rotationally symmetric; thus, it can be specified as a 1D function of the radial distance  $r_i$  from the image center.

### B. Extended vignetting model

In an arbitrary input image, numerous regions with different local tilt factors may exist. To account for multiple surfaces in an image, we present an extension of the Kang-Weiss model in which different image regions can have different tilt angles. The tilt factor of Eq. (4) is modified to

$$T_i = \cos \tau_{s_i} \left( 1 + \frac{\tan \tau_{s_i}}{f} (u_i \sin \chi_{s_i} - v_i \cos \chi_{s_i}) \right)^3, \quad (5)$$

where  $s_i$  indexes the region containing pixel  $i$ .

We also extend the linear geometric factor to a more general polynomial form:

$$G_i = (1 - \alpha_1 r_i - \dots - \alpha_p r_i^p), \quad (6)$$

where  $p$  represents a polynomial order that can be arbitrarily set according to a desired precision. This generalized representation provides a closer fit to the geometric vignetting effects that we have observed in practice. The amount of rays being blocked by the lens' rim as a function of field height (or radial distance from optical axis) can be highly nonlinear for wide-angled views. In contrast to using a polynomial as the overall vignetting model, representing only the geometric component by a polynomial allows the overall model to explicitly account for local tilt effects and global off-axis illumination.

In the next section, we provide the analysis leading to the estimation of vignetting parameters.

## III. PROBLEM DEFINITION AND ALGORITHM OVERVIEW

Given an observed image  $\mathcal{I}$  (specified by Eq. (1)), we wish to recover the vignetting function  $\mathcal{V}$ . Undoing the effects of  $\mathcal{V}$  on  $\mathcal{I}$  leads to a vignetting-free image  $\mathcal{Y}$ :

$$\mathcal{Y} = f(\mathcal{R} \cdot \mathcal{T} + \epsilon_1). \quad (7)$$

The corresponding linearized vignetting-free image is  $\mathcal{Y}' = g(\mathcal{Y}) = \mathcal{R} \cdot \mathcal{T} + \epsilon_1$ , where  $g$  is the inverse radiometric response function.

### A. Dealing with radiometric response function

To solve for  $\mathcal{Y}$ , the response function  $f$  should be computed in advance (e.g., using the single image-based radiometric calibration method of Lin *et al.* [14]). However, vignetting  $\mathcal{V}$  is not considered in the estimation of the response function in [14], which may adversely affect the result. Although simultaneous estimation of  $f$  and  $\mathcal{V}$  can resolve this problem, it is considerably more complex and may generate a solution space consisting of many local minima. Instead, we approximate  $f$  based on  $\mathcal{I}$  first, and then solve for  $\mathcal{V}$  after accounting for  $f$ .

By applying the inverse of the estimated radiometric response function  $\hat{f}$  to Eq. (2), we get the linearized version of  $\mathcal{I}$ , which we denote as  $\mathcal{I}'$ :

$$\mathcal{I}' = \hat{g}(\mathcal{I}) = \mathcal{R} \cdot \varphi + \epsilon_1 = \mathcal{R} \cdot \mathcal{V} \cdot \mathcal{T} + \epsilon_1. \quad (8)$$

A recovered vignetting function  $\mathcal{V}$  can be factored out to yield a linearized vignetting-free image  $\mathcal{Y}'$ :

$$\mathcal{Y}' = \mathcal{R} \cdot \mathcal{T} + \epsilon_1 \approx \mathcal{I}' / \mathcal{V} = \mathcal{R} \cdot \mathcal{T} + \epsilon_1 / \mathcal{V} = \mathcal{R} \cdot \mathcal{T} + \epsilon_{\mathcal{V}}, \quad (9)$$

where  $/$  is the de-vignetting process (pixelwise division). Assuming that  $\mathcal{V}$  spatially varies slowly and  $\epsilon_1 \sim (0, \sigma_1^2)$ , it is reasonable that  $\epsilon_{\mathcal{V}} \sim (0, \sigma_1^2)$  as well. Therefore,  $\mathcal{I}' / \mathcal{V}$  is a good approximation of  $\mathcal{Y}'$ .

### B. Estimating vignetting

Given a linearized image  $\mathcal{I}'$ , we estimate the vignetting function  $\mathcal{V}$  using Eq. (8) and an assumption that  $\mathcal{R}$  is piecewise constant. Natural images formed by a piecewise constant latent image plus zero-mean Gaussian noise are popularly modeled by a Markov Random Field (MRF), as in image de-noising and image segmentation. There are efficient algorithms to solve the MRF, such as graph cuts [30].

Given  $\mathcal{I}'$ , we solve for  $\mathcal{V}$  by estimating  $\mathcal{R}$ ,  $\mathcal{V}$  and  $\mathcal{T}$  simultaneously with a *maximum a posteriori* (MAP) criterion:

$$\mathcal{V}, \mathcal{T}, \mathcal{R} = \arg \max_{\mathcal{V}, \mathcal{T}, \mathcal{R}} \mathcal{P}(\mathcal{V}, \mathcal{T}, \mathcal{R} | \mathcal{I}'). \quad (10)$$

In Eq. (10), the conditional probability of  $(\mathcal{V}, \mathcal{T}, \mathcal{R})$  may be stated as follows based on Bayes' theorem:

$$\mathcal{P}(\mathcal{V}, \mathcal{T}, \mathcal{R} | \mathcal{I}') = \frac{\mathcal{P}(\mathcal{I}' | \mathcal{V}, \mathcal{T}, \mathcal{R}) \mathcal{P}(\mathcal{V}, \mathcal{T}, \mathcal{R})}{\mathcal{P}(\mathcal{I}').} \quad (11)$$

For Eq. (11), we assume the denominator  $\mathcal{P}(\mathcal{I}')$  to be a constant and drop it. Moreover, we assume  $\mathcal{V}$ ,  $\mathcal{T}$  and  $\mathcal{R}$  are statistically independent. These assumptions yield

$$\mathcal{P}(\mathcal{V}, \mathcal{T}, \mathcal{R} | \mathcal{I}') = \mathcal{P}(\mathcal{I}' | \mathcal{V}, \mathcal{T}, \mathcal{R}) \mathcal{P}(\mathcal{V}) \mathcal{P}(\mathcal{T}) \mathcal{P}(\mathcal{R}). \quad (12)$$

To maximize the above probability, we can either simultaneously estimate  $\mathcal{V}$ ,  $\mathcal{T}$  and  $\mathcal{R}$  in a single step, or alternate between updating  $\mathcal{V}$  plus  $\mathcal{T}$  and  $\mathcal{R}$  (while keeping the other constant). The former is usually intractable in practice because of the large number of unknowns. In contrast, the latter is considerably simpler, and we employ it in this paper.

To illustrate the alternating process that maximizes the probability in Eq. (12) for estimating  $\mathcal{V}$ ,  $\mathcal{T}$  and  $\mathcal{R}$ , the two step updating scheme would be

$$\{\mathcal{V}, \mathcal{T}\}^{(k+1)} = \arg \max_{\mathcal{V}} \mathcal{P}(\mathcal{I}' | \mathcal{V}, \mathcal{T}, \mathcal{R}^{(k)}) \mathcal{P}(\mathcal{V}) \mathcal{P}(\mathcal{T}) \quad (13)$$

and

$$\mathcal{R}^{(k+1)} = \arg \max_{\mathcal{R}} \mathcal{P} \left( \mathcal{I}' | \{\mathcal{V}, \mathcal{T}\}^{(k+1)}, \mathcal{R} \right) \mathcal{P}(\mathcal{R}). \quad (14)$$

The two steps in Eqs. (13) and (14) are repeated until convergence (starting with  $k = 0$ ).

Equation (13) can be seen as estimating the parameters in the chosen vignetting-tilt function of the extended Kang-Weiss vignetting model. We assume the noise in  $\mathcal{I}'$  (Eq. (8)) to have a Gaussian distribution, so the conditional probability in Eq. (13) can be represented by a Gaussian function. We set the prior probabilities of  $\mathcal{V}$  and  $\mathcal{T}$  (given by  $\mathcal{P}(\mathcal{V})$  and  $\mathcal{P}(\mathcal{T})$ ) both to 1, since vignetting is represented with a physical model. By taking the negative logarithm on the right hand side of Eq. (13) and dropping the constant, we get the following energy function:

$$E_{vgn}^{(k+1)} = \sum_{i \in \Omega} \left( I'_i - R_i^{(k)} V_i T_i \right)^2 / \sigma_{R_i}^2, \quad (15)$$

where  $I'_i$ ,  $R_i^{(k)}$ ,  $V_i$  and  $T_i$  are the values of pixel  $i$  in the image domain  $\Omega$  of  $\mathcal{I}'$ ,  $\mathcal{R}^{(k)}$ ,  $\mathcal{V}$  and  $\mathcal{T}$ , respectively.  $\sigma_{R_i}^2$  is the estimated variance of the Gaussian model for  $R_i$ . Note that the vignetting parameters are components of  $\mathcal{V}_i$  in Eq. (15).

We solve Eq. (14) using an MRF-based segmentation process. We first account for the current vignetting and tilt estimation  $\{\mathcal{V}, \mathcal{T}\}^{(k+1)}$  in  $\mathcal{I}'$ , i.e.,  $\mathcal{Y}''^{(k+1)} = \frac{\mathcal{I}'}{\mathcal{V}^{(k+1)} \cdot \mathcal{T}^{(k+1)}}$ . We then treat  $\mathcal{Y}''^{(k+1)}$  as the observed image in the segmentation process in which  $\mathcal{R}$  is cast as an MRF. More specifically, segmentation is accomplished by minimizing the two-term energy function as

$$E_{seg}^{(k+1)} = \sum_{i \in \Omega} D \left( R_i, Y_i''^{(k+1)} \right) + \sum_{(i,j) \in \mathcal{N}_{\Omega}} S(R_i, R_j), \quad (16)$$

where the two terms are obtained by taking the negative logarithm of the right hand side of Eq. (14). These two terms correspond to the conditional probability and priori probability, respectively.

$D \left( R_i, Y_i''^{(k+1)} \right)$  measures the fidelity between the radiance estimation  $R_i$  and the observed value  $Y_i''^{(k+1)}$  for pixel  $i$  in image domain  $\Omega$ . We use  $Y_i''^{(k+1)}$  to denote the value of a pixel in  $\mathcal{Y}''^{(k+1)}$ . We use a Gaussian model to represent the conditional probability in Eq. (14) (see Eq. (9)):

$$D \left( R_i, Y_i''^{(k+1)} \right) = \left( R_i - Y_i''^{(k+1)} \right)^2 / \sigma_{R_i}^2. \quad (17)$$

$S(R_i, R_j)$  imposes a penalty for a pair of neighboring pixels  $i$  and  $j$  in the neighboring-pixel-pair set  $\mathcal{N}_{\Omega}$  to have different radiance estimations, i.e., enforcing smoothness on radiance estimations in image space. Graph cut [30] can efficiently minimize this energy function. The initialization of  $\mathcal{R}^{(0)}$  in Eq. (13) is obtained with graph cuts performed directly on  $\mathcal{I}'$ . In the segmentation, in addition to radiance, we also added other features such as texture features for texture segmentation. We simply replaced  $R_i$  and  $Y''$  in Eq. (16) and (17) with the corresponding feature values. More details on the selected features for our technique can be found in Section V.

In Eqs. (15) and (17), we can assume  $\sigma_{R_i}^2$  to be a constant, or compute estimates for different values of  $R_i$ . In this paper,

we treat  $\sigma_{R_i}^2$  as a constant over the given image. To obtain this constant value, we first subtract the given image from a bilateral filtered version of itself [24] (with a photometric spread  $\sigma_r = 20$  and a geometric spread  $\sigma_d = 3$  in our implementation) to get an approximate noise image. The noise is converted from color to grayscale, and the standard deviation of the noise is computed with this grayscale noise image. Other techniques may alternatively be employed for estimating this constant [16], [21].

The processes for vignetting model estimation and image segmentation are represented by Eqs. (13) and (14), and are detailed in Sections IV and V, respectively. To more robustly estimate the vignetting parameters, we augment our technique with features such as recursively segmenting regions at finer scales to get more reliable information for vignetting estimation, using a robust statistic function in Eq. (15), and handling texture and outliers in the segmented image regions.

### C. Procedural overview

The high-level flow of our algorithm for joint estimation of  $\mathcal{V}$ ,  $\mathcal{T}$  and  $\mathcal{R}$  is illustrated in Fig. 3. In each iteration, the image is first segmented at a coarse scale, and for each region a reliability measure of the region data for vignetting estimation is computed. For regions that exhibit greater consistency with physical vignetting characteristics than other regions, a higher reliability weight is assigned. Low weights may indicate regions with multiple distinct surfaces, so these regions are recursively segmented at incrementally finer scales until weights of the smaller regions exceed a threshold or regions become negligible in size. With this segmentation approach, the segmentation scale varies spatially in a manner that facilitates collection of vignetting data.

After spatially adaptive segmentation, pixels with high reliability weights are used to estimate the vignetting model parameters. Since the preceding segmentations may be corrupted by the presence of vignetting, the subsequent iteration of the procedure re-computes segmentation boundaries from an image corrected using the currently estimated vignetting model. Better segmentation results lead to improved vignetting estimates, and these iterations are repeated until the estimates converge.

### D. Image pyramid in vignetting estimation

We use an image pyramid in a coarse-to-fine strategy to improve initialization and solution quality in the vignetting estimation. Once the image pyramid is obtained, the iterative algorithm for the MAP estimation of  $\mathcal{V}$ ,  $\mathcal{T}$  and  $\mathcal{R}$  is performed at the coarsest level. Estimation at each successively finer level uses the result from the previous level as the initial condition.

The Gaussian pyramid is used, which produces a sequence of copies of the original image by low-pass-filtering and subsampling by a factor of two to obtain the next pyramid level. Pyramid construction is equivalent to convolving the original image with a set of Gaussian filters.

In order to eliminate changes in model parameters due to different image sizes at the different resolution levels, these parameters are expressed in terms of the original image

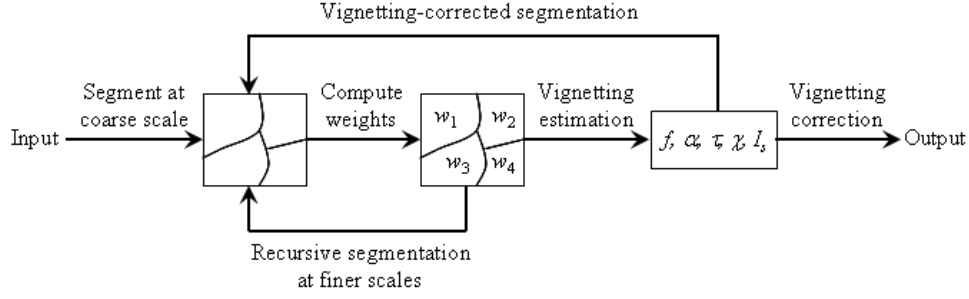


Fig. 3. Overview of vignetting function estimation.

resolution, and are computed using subsampled image values at coarser pyramid levels. In this way, the same position relative to the world coordinate system will have consistent values for  $r$ ,  $u$  and  $v$  in Eqs. (3), (4), (5) and (6).

#### IV. SEGMENTATION BASED VIGNETTING ESTIMATION

Consider a vignetted image that has been segmented by our algorithm in Section V. For a scene composed of piecewise constant regions (based either on color or texture)<sup>1</sup>, this segmentation will result in regions each with constant scene radiance but with variations in vignetting attenuation<sup>2</sup>. We take advantage of these variations of color or texture within each segmented region to infer the vignetting parameters of the image.

In this estimation, we solve for parameters of the vignetting model that transform a piecewise constant scene as closely as possible to the measured image. We construct an energy function measuring the errors between the original image and an image simulated using estimated vignetting parameters and the estimated scene radiances of the segmented, piecewise flat regions. A smaller error indicates a more accurate estimation of vignetting and scene radiance.

An important challenge is to effectively deal with noise, outliers, and texture in this estimation. To handle noise and outliers, we apply a robust cost function to the error term. Fluctuations in region color due to texture can degrade estimation as well, so we measure the texture of a segmented region and compensate for it in the error term. Since this texture compensation may mask outliers with respect to texture in the robust cost function, a texture-based outlier measure is used as well.

##### A. Vignetting energy function

The energy function specified in Eq. (15), while correct in principle, lacks robustness in handling noise, outliers, and texture. In this section, we show how the function can be modified for more accurate estimation. For simplicity, we

remove the superscripts in Eq. (15) with the understanding that the energy function is specified based on the latest estimated value of  $\mathcal{R}$ .

Let the scene radiance  $R_s$  of a piecewise flat region  $s$  be expressed by its ratio  $\lambda_s$  to the scene radiance  $R_0$  of the center pixel, i.e.,  $R_s = \lambda_s R_0$ . Given an image with  $M$  regions of different scene radiance, we formulate the recovery of vignetting parameters as minimization of the following energy function:

$$E = \sum_{s=1}^M \sum_{i=1}^{N_s} \left[ \rho \left( \frac{\lambda_s R_0 T_i V_{r_i} - I'_i}{1 + c_k K_s} \right) \delta(H_i > H_{th}) \right], \quad (18)$$

where  $i$  is the pixel index in region  $s$  (with  $N_s$  pixels),  $I'_i$  is the pixel value in the linearized image  $\mathcal{I}'$ ,  $\rho(\cdot)$  is the error norm function,  $K_s$  is a measure of texture for region  $s$ , and  $c_k$  is a constant (the determination of its value will be explained in Section VI).  $H_i$  represents a measure of likelihood that pixel  $i$  is not an outlier,  $H_{th}$  is a threshold value (set to 0.93 in our implementation), and  $\delta(\cdot)$  is the Kronecker delta. The vignetting model fit term  $\lambda_s R_0 T_i V_{r_i} - I'_i$  in the error norm function  $\rho(\cdot)$  is modulated by texture measure  $K_s$  to downplay errors caused by the texture effect. The  $\delta(\cdot)$  function serves to reject outliers with respect to the region's texture. For color images,  $I'_i$  is an RGB vector. For ease of explanation, we express  $I'_i$  in this paper as a single color channel, and overall energies are averaged over the separate color components.

We use three mechanisms to improve the robustness of vignetting estimation in Eq. (18): (1) using a robust statistic function  $\rho(\cdot)$  on the error term to remove outliers recognized by large intensity errors between a simulated image and the real one, (2) applying the texture measure  $K_s$  to the error term so as to avoid emphasis on errors due to texture fluctuations within regions, and (3) using an additional outlier measure  $H_i$  to specify the outliers which might not be detected by  $\rho(\cdot)$  due to the compensation factor  $K_s$  in texture regions. We detail each of these robustness mechanisms in the following.

1) *Robust Statistic Function*: With an appropriate function  $\rho(\cdot)$  in the energy expressed by Eq. (18), outliers which do not conform to statistical assumptions can be discarded, thus making the algorithm more robust [3]. We use the *Lorentzian* estimator:

$$\rho(x) = \log \left( 1 + (x/\sigma(t))^2/2 \right), \quad (19)$$

where  $\sigma$  is a scale parameter to control the shape of function  $\rho(x)$  and  $t$  is the iteration number of the vignetting estimation.

<sup>1</sup>Piecewise constant regions come from the assumption that the scene radiance are from uniform planar surfaces.

<sup>2</sup>A piecewise constant scene region will not be piecewise constant in the observed image due to vignetting. However, most segmentation algorithms tolerate color/texture variations in a segmented region to get more practical segmentations. The segmentation algorithm can group together pixels with the same underlying color/texture but different degrees of vignetting attenuation.

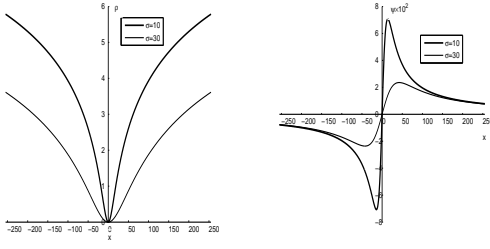


Fig. 4. The Lorentzian error norm function (left) and its influence function (right).

With the *Lorentzian* function, pixels with small error values that better fit the current estimated vignetting parameters in Eq. (18) will have large “influence” on the solution whereas outlier pixels with large error values will have significantly less “influence”. Moreover, as  $\sigma$  decreases in Eq. (19), the “influence” of pixels with smaller error values increases, as shown in Fig. 4. The influence function characterizes the effect of an error value on the energy function and is proportional to the derivative of the error norm function [3]. A larger value of an influence function corresponds to a larger degree of “influence” by the pixel.

Since vignetting parameters are more accurate with more iterations, pixels with smaller errors are given increasingly higher emphasis while pixels with larger errors are increasingly penalized in Eq. (18). At each image pyramid level,  $\sigma$  in Eq. (19) is gradually decreased in time (iteration  $t$ ) according to  $\sigma(t) = \frac{\sigma'}{t}$ , where  $\sigma'$  is computed from the first iteration. Robust statistics [2], [19] is used to automatically estimate  $\sigma'$ . By denoting all pixel errors in Eq. (18) with  $\mathcal{X}$ ,  $\sigma'$  is computed using

$$\sigma' = c_x \text{MED}(|\mathcal{X} - \text{MED}(\mathcal{X})|) \quad (20)$$

where MED refers to the median value, and  $c_x$  is a constant that depends on the statistical distribution of  $\mathcal{X}$ . We simply set  $c_x = 1.4826$  by assuming  $\mathcal{X}$  is normally distributed.

2) *Textureness Measurement*: We measure the textureness in a segmented region by computing the average value of intensity variance within local windows of a fixed size ( $7 \times 7$  in our experiments):

$$K_s = \frac{1}{N_s \cdot N_W} \sum_{i \in \Omega_s} \sum_{j \in \mathcal{W}_i} (I_j' - \mu_i')^2 \quad (21)$$

where  $\Omega_s$  denotes the pixels in the segmented region  $s$ ,  $N_s = |\Omega_s|$ ,  $N_W = 49$  represents the number of pixels in a local window,  $\mathcal{W}_i$  signifies the local window of pixel  $i$ , and  $\mu_i'$  is the mean linearized value of pixels in  $\mathcal{W}_i$ .

3) *Texture Outlier Measurement*: We suppose there is a prevailing texture pattern in a segmented region, and treat pixels as outliers whose texture pattern deviates too far from the estimated prevailing texture pattern. More precisely, a histogram of intensities in a local window is used to represent the texture pattern of the central pixel. We use the same set of histogram bins for all pixels in a region, and organize each histogram into a vector. Then, Principal Components Analysis (PCA) [9] is performed on the collection of vectors

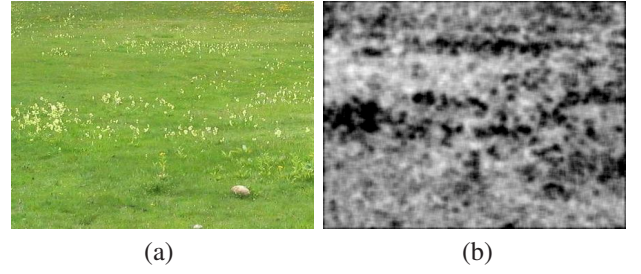


Fig. 5. Texture outlier detection. (a) A natural picture in which grass is the prevailing texture pattern, and flowers are considered outliers in the vignetting estimation. (b) An image indicating texture outliers according to  $H_i$ , where darker pixels are more likely to be outliers while brighter pixels are likely to be part of the prevailing texture pattern.

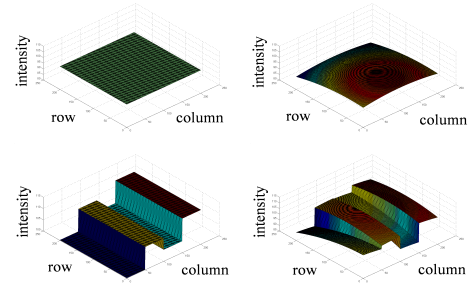


Fig. 6. Vignetting over multiple regions. Left to right: Without vignetting for a single uniform region, With vignetting for a single uniform region, Without vignetting for multiple regions, With vignetting for multiple regions.

for the given region. We suppose that the prevailing texture pattern can be represented by a small number of low-order principal components. For simplicity, we selected 4 lowest order principal components, although we note that there exist various automatic selection methods [26], [27]. Finally, we compute for each pixel  $i$  the angle  $\beta_i$  in the high dimensional space between the original histogram vector and its projection onto the lower-dimensional subspace spanned by the selected principal components, and measure the likelihood that  $i$  is not an outlier by

$$H_i = \frac{-\beta_i}{\pi} + 1. \quad (22)$$

Fig. 5 shows the result of using  $H_i$  to segment a natural image consisting of grass and flower regions. The areas associated with the flowers have low  $H_i$  values and are tagged as outliers.

With these three robust estimation components, the vignetting in an image can be more reliably recovered. In the energy function (18), the parameters to be estimated are the focal length  $f$  in the off-axis component, the  $\alpha$  coefficients of the geometric factor, the tilt angles  $\tau_s$  and  $\chi_s$ , the scene radiance of the center pixel  $R_0$ , and the radiance ratio  $\lambda_s$  of each region. In processing multiple image regions as illustrated in Fig. 6, minimization of this energy function can intuitively be viewed as simultaneously solving for local region parameters  $R_s$ ,  $\tau_s$  and  $\chi_s$  that give a smooth alignment of vignetting attenuation between regions, while optimizing the underlying global vignetting parameters  $f$ ,  $\alpha_1, \dots, \alpha_p$ . With the estimated parameters, the vignetting corrected image is then given by  $I_i'/V_{r_i}$  (see Eq. (9)). Note that we retain the local tilt factor in

the corrected image so as not to produce an unnatural-looking result.

### B. Vignetting estimation

For a collection of segmented regions, the many unknown parameters create a complicated solution space. To simplify optimization, we use a stepwise method for parameter initialization prior to estimating the vignetting function.

In the first step, initial values of relative scene radiance  $\lambda_s$  are determined for each region without consideration of vignetting and tilt parameters. For pixels  $i$  and  $j$  at the same radius  $r$  but from different regions, their vignetting attenuation should be equal, so their image values  $I'_i$  and  $I'_j$  should differ only in scene radiance. Based on this property, relative scene radiance values are initialized by minimizing the function

$$E_1 = \sum_r \sum_{r_i, r_j=r; s_i \neq s_j} w_i w_j \left( \frac{I'_i}{\lambda_{s_i}} - \frac{I'_j}{\lambda_{s_j}} \right)^2 \quad (23)$$

where  $w_i$  and  $w_j$  are the weight values (their computation will be explained in next section).

The  $\lambda_s$  values in Eq. 23 are solved in the least squares sense by singular value decomposition (SVD) on a system of equations  $\sqrt{w_i w_j} \left( \frac{I'_i}{\lambda_{s_i}} - \frac{I'_j}{\lambda_{s_j}} \right) = 0$  where  $\frac{1}{\lambda_{s_i}}$  and  $\frac{1}{\lambda_{s_j}}$  are unknowns. To expedite minimization of this function, a set of pixels at a given radius and within the same region may be represented by a single pixel with the average color of the set.

With the initial values of  $\lambda_s$ , the second step initializes the parameters  $f$ ,  $R_0$ , and  $\alpha_1, \dots, \alpha_p$ , where  $p$  is the polynomial order used in the geometric factor of Eq. 6. Ignoring local tilt factors, this is computed with the energy function

$$E_2 = \sum_{s=1}^M \sum_{i=1}^{N_s} w_i (\lambda_s R_0 V_{r_i} - I'_i)^2. \quad (24)$$

This function is iteratively solved by incrementally increasing the polynomial order from  $k = 1$  to  $k = p$ , and using the previously computed polynomial coefficients  $\alpha_1, \dots, \alpha_{k-1}$  as initializations. In our implementation, we use a polynomial order of  $p = 4$ .

In the third step, the local tilt parameters  $\tau_s, \chi_s$  are estimated by optimizing the energy function in Eq. 18 with the other parameters fixed to their initialization values. After this initialization stage, all the parameters are jointly optimized in Eq. 18 to finally estimate the vignetting function. The optimizations of Eq. 18 and Eq. 24 are computed using the Levenberg-Marquardt algorithm [17].

The initialization techniques explained above are performed only in the first iteration between segmentation and vignetting estimation. At the end of each iteration, an estimate of the vignetting function is determined and used as the initialization in the following iteration. The vignetting parameters are progressively refined during the iterating process by estimating jointly all the parameters with the Levenberg-Marquardt algorithm [17]. Meanwhile, more refined estimates of vignetting lead to more accurate segmentations of the image.

## V. VIGNETTING-BASED IMAGE SEGMENTATION

To obtain information for vignetting estimation, pixels having the same scene radiance need to be identified in the input image by minimizing the segmentation energy function shown in Eqs. (16) and (17). As mentioned in Section III-B, we treat this as an MRF problem. In our implementation, we employ graph cut segmentation [30] with per-pixel feature vectors composed of six color/texture attributes. The color components are the RGB values, and the local texture descriptors are the polarity, anisotropy and normalized texture contrast described in [5]. As described in Section III-B, we assume a Gaussian model for each segmented region of  $\mathcal{Y}''$ . For the MRF problem, we use an 8-neighborhood system and the Potts smoothness term [30]. Details of the graph cut algorithm can be found in [4], [30]; its associated parameter setting will be explained in next section.

In this section, we present the proposed adaptations to graph cut for obtaining more reliable information in vignetting estimation. To facilitate the location of reliable vignetting data, segmentation scales are spatially varied over the image, and the adverse effects of vignetting on segmentation are progressively reduced as the vignetting function estimate is refined.

### A. Spatial variations in scale

Sets of pixels with the same scene radiance provide more valuable information if they span a broader range of vignetting attenuation. In the context of segmentation, larger regions are therefore preferable. While relatively large regions can be obtained with a coarse segmentation scale, many of these regions may be unreliable for vignetting estimation since they may contain multiple surfaces or include areas with non-uniform illumination. In an effort to gain useful data from an unreliable region, our method recursively segments it into smaller regions that potentially consist of better data for vignetting estimation. This recursive segmentation proceeds until regions have a higher reliability weight than a pre-set weight threshold  $w_{th}$  or become of negligible size according to a threshold of 225 pixels used in our implementation. Regions of very small size generally contain insignificant changes in vignetting attenuation, and the inclusion of such regions would bias the optimization process.

The weight for a region is computed based on its consistency with physical vignetting characteristics (i.e., our physically-based vignetting model) and how well its vignetting characteristics conform to that observed elsewhere in the image. This weight is first computed per pixel as

$$w_i = \frac{1}{c_w \cdot \rho \left( \frac{\lambda_s R_0 T_i V_{r_i} - I'_i}{1 + c_k K_s} \right) + 1} \quad (25)$$

where  $c_w$  is determined by

$$c_w = \frac{1 - p}{\sqrt{2}\sigma p}, p \in [0 \ 1]. \quad (26)$$

In Eq. (25),  $w_i$  is a value within the range  $[0 \ 1]$ , and the input to function  $\rho$  is the error term between the original pixel intensity and its estimation with our vignetting model. Eq. (25) is a decreasing function relative to the error term, and the

weight value is equal to  $p$  when the error equals to  $\sqrt{2}\sigma$  (which corresponds to the saddle point in the Lorentzian function's influence function as shown in Fig. 4).

For our iterative algorithm on segmentation and vignetting estimation, initially, no vignetting estimates are known, so reliability is measured in the first iteration of the algorithm according to how closely the region data can be represented by our physically-based vignetting model. For a given region, an estimate  $\hat{\mathcal{V}}$  of the vignetting function is computed similarly to the technique described in Section IV-B.

The weight of an image region is specified as the mean weight value of the region, and is defined as

$$w_s = \frac{1}{N_s} \sum_{i=1}^{N_s} w_i, \quad (27)$$

where  $i$  indexes the pixels in this region.

The weight threshold for the recursive segmentation is specified as follows:

$$w_{th} = \frac{1}{c_w \cdot \rho(\sqrt{2}\sigma) + 1} \quad (28)$$

where  $c_w$  is defined in Eq. (26). It is determined that a region needs to be subdivided when its weight is less than that of regions whose error term lies outside the two saddle points of the Lorentzian influence function.

In the recursive segmentation procedure, incrementally finer scales of segmentation are used. For methods such as mean shift [6] and region competition [22], segmentation scale is essentially controlled by a parameter on variation within each feature class, where a feature may simply be pixel intensity or color. With such approaches, a finer partitioning of a low-weight region can be obtained by segmenting the region with a decreased parameter value. In other techniques such as graph cuts [30] and Blobworld [5], the degree of segmentation is set according to a given number of feature classes in an image. There exist several ways to set the number of classes, including user specification, data clustering, and minimum description length criteria [18]. For recursive segmentation, since each region belongs to a certain class, a finer partitioning of the region can be obtained by segmenting it with the number of feature classes specified as two.

With this general adaptation, segmentation scale varies over an image in a manner designed to maximize the quality of vignetting data.

### B. Accounting for vignetting

Two pixels with the same scene radiance may exhibit significantly different image intensities due to variations in vignetting attenuation. In segmentation, a consequence of this vignetting is that a homogeneous scene area may be divided into separate image regions. Vignetting may also result in heterogeneous image areas being segmented together due to lower contrasts at greater radial distances. For better stability in vignetting estimation, the effects of vignetting on segmentation should be minimized.

To address vignetting effects in segmentation, after each iteration through the procedure in Fig. 3, the estimated vignetting function is accounted for in the segmentation step of

the subsequent iteration. Specifically, the vignetting corrected image computed with the currently estimated parameters is used in place of the original input image in determining segmentation boundaries. The corrected image is used only for segmentation purposes, and the colors in the original image are still used for vignetting estimation.

As the segmentations improve from reduced vignetting effects, the estimated vignetting function also is progressively refined. This process is repeated until the difference between vignetting functions in consecutive iterations falls below a prescribed threshold, where the difference is measured as

$$\Delta\mathcal{V} = \frac{1}{k} \sum_r ||V_r(t) - V_r(t-1)||. \quad (29)$$

$\mathcal{V}(t)$  represents the global vignetting function at iteration  $t$ , and radial distances  $r$  are sampled at  $k$  uniform intervals, where  $k = 100$  in our implementation.

## VI. PARAMETER ESTIMATION

In our segment-based vignetting estimation technique, there are two key parameters:  $c_k$  in Eq. (18) used in vignetting estimation with a given segmentation, and a parameter for the graph cut algorithm [4], [30] used in segmentation with a given vignetting estimate. We determine their values empirically as explained below.

The value of  $c_k \geq 0$  in Eq. (18) controls the emphasis on texture regions in comparison to non-texture regions in vignetting estimation. When  $c_k = 0$ , the texture regions and non-texture regions are treated equally in the error measure. As  $c_k$  is increased, the error norm decreases for pixels in texture regions in relation to pixels in non-texture regions. This is because pixels in texture regions have a larger value of textureiness  $K_s$ , and results in de-emphasis of pixels within texture regions. This feature is desirable because errors in texture regions are mostly caused by texture fluctuations. However, texture regions do contain useful information, and can particularly help in cases of relatively homogeneous textures. Based on extensive experimentation, we found  $c_k = 13$  to be an effective setting.

As Eq. (16) shows, the energy function used in the graph cut algorithm [4], [30] consists of two terms: the fidelity term and the smoothness term. Only one parameter is involved as a penalty term for violating the smoothness constraint [23]. It determines the value of the smoothness term in Eq. (16). If this parameter is too small, there will be little coherence between neighboring pixels. This leads to over-segmentation, and the small segments will not contribute in an optimal manner to vignetting estimation. Conversely, an overly large parameter results in under-segmentation. Such segments are likely to straddle regions with different radiances, which in turn, produces incorrect vignetting parameters. Based on experiments, satisfactory segmentation results are obtained by setting the parameter to 1.3.

## VII. RESULTS

To evaluate our algorithm, we design multiple experiments on a wide variety of images with both simulated and real

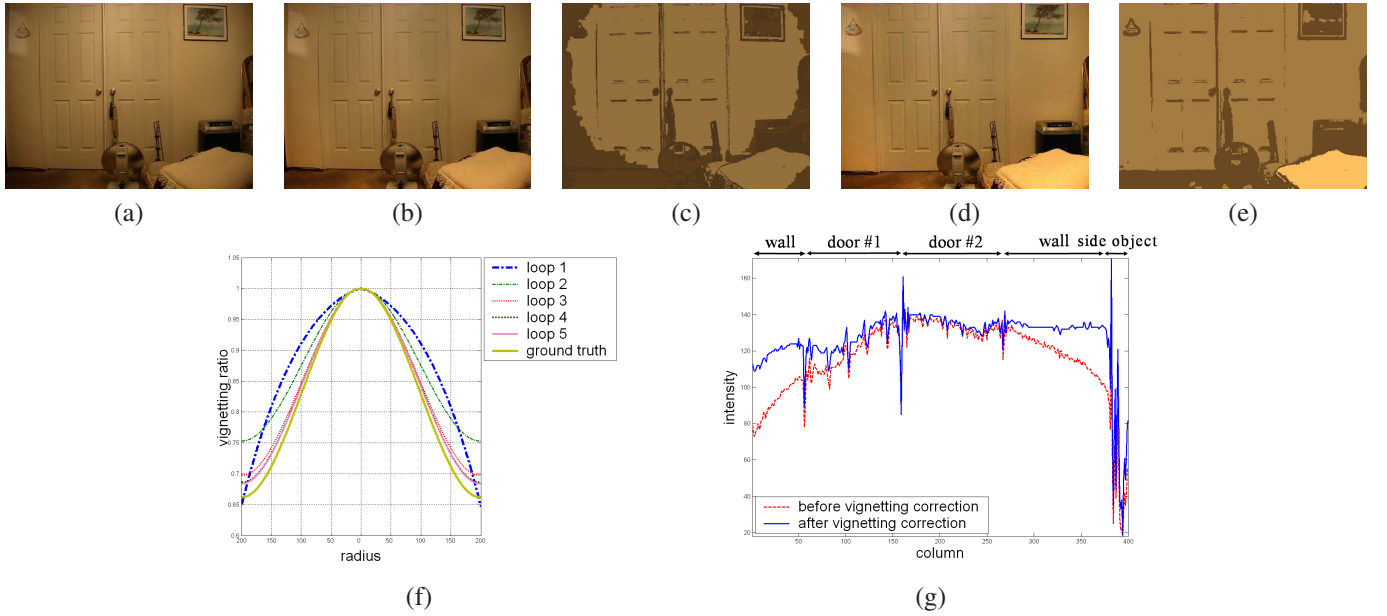


Fig. 7. Effects of vignetting compensation in segmentation. (a) Original image; (b) Vignetting correction without vignetting compensation; (c) Segmentation without vignetting compensation; (d) Vignetting correction with vignetting compensation; (e) Segmentation with vignetting compensation; (f) Estimated vignetting functions after each iteration in comparison to the ground truth; (g) Intensity profile before (red) and after (blue) correction, shown for the image row that passes through the image center.

vignetting. First, we present experiments to examine the effects of the proposed segmentation adaptation. Second, we provide a quantitative analysis on vignetting parameter estimation based on images with simulated vignetting, and explain how to obtain the ground truth vignetting in practice. Third, we evaluate our algorithm on images with real vignetting, and provide a quantitative error analysis. Fourth, we compare the speed between our method and our previous single-image vignetting correction method [31]. Fifth, we show our algorithm's performance on estimating tilt effects, and present some results on image mosaicing. Finally, we present some failure mode examples of our algorithm and analyze the causes.

#### A. Segmentation Adaptation

Accounting for vignetting in segmentation leads to progressive improvements in the estimated vignetting function. To exemplify this, we run our algorithm on an image captured indoors with heavy vignetting (see Fig. 7 (a)). The original image is directly processed without an image pyramid in these experiments. The correction and segmentation results in (b) and (c) without vignetting compensation are equivalent to those after a single pass through the overall procedure shown in Fig. 3. With additional iterations, the enhanced segmentations lead to vignetting estimates that trend towards the ground truth.

The effect of recursive segmentation on a given region is illustrated in Fig. 8. Further segmentation of a low-weight region can produce sub-regions of higher weight. With this improvement in data quality, a more accurate vignetting function can be estimated.

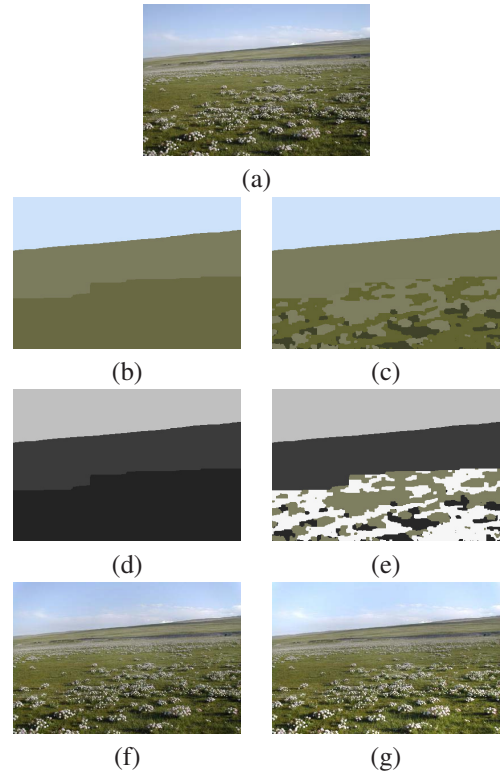


Fig. 8. Recursive segmentation on low-weight regions. (a) Original image; (b) Regions prior to recursive segmentation; (c) Regions after recursive segmentation of one region; (d) Region weights of (b), where higher intensity indicates a higher weight; (e) Region weights of (c); (f) Vignetting correction result using region information of (b); (g) Vignetting correction result using region information of (c).

$f, \alpha_1, \dots, \alpha_5$	mean of error $f, (\alpha_1, \dots, \alpha_5) \times 10^{-3}$	std of error $f, (\alpha_1, \dots, \alpha_5) \times 10^{-3}$
800, 0.02, 0, 0, 0, 0	10.2, 1.0, 4.3, 5.0, 2.6, 4.2	2.5, 0.1, 0.9, 2.4, 4.9, 24.0
800, 0.0, 0.3, 0, 0, 0	25.1, 1.3, 6.2, 3.7, 3.0, 14.3	2.8, 0.1, 0.3, 1.5, 2.8, 19.8
800, 0.0, 0.0, 0.4, 0, 0	13.7, 0.9, 5.7, 3.1, 3.3, 10.1	1.7, 0.3, 1.1, 0.9, 3.9, 22.4
800, 0, 0, 0, 0, 0.7	5.1, 0.3, 4.4, 5.1, 6.0, 8.1	3.7, 0.2, 1.4, 1.8, 12.1, 13.9
800, 0, 0, 0, 0, 0	4.3, 0.4, 5.7, 6.1, 1.3, 7.4	1.8, 0.3, 3.7, 4.7, 6.1, 7.9
1900, 0.001, 0, 0, 0, 0.5	50.5, 1.8, 6.6, 6.4, 2.7, 8.1	14.2, 2.6, 3.1, 1.7, 2.4, 9.3

TABLE I

ERROR IN ESTIMATED OPTICAL AND GEOMETRIC FACTORS OF THE DIFFERENT SIMULATED VIGNETTING FUNCTIONS. THE MEAN AND STANDARD VARIANCE OF THE ERRORS FOR OUR 50 TEST IMAGES ARE LISTED.

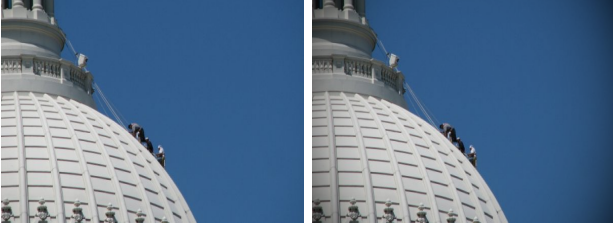


Fig. 9. On the left is the vignetting free image. Right image is produced by adding vignetting with model parameters  $f = 1342$ ,  $\alpha_1 = \alpha_4 = \alpha_5 = 0$ ,  $\alpha_2 = 0.02$  and  $\alpha_3 = 0.03$ .

### B. Evaluations on Image with Simulated Vignetting

We obtained images free of vignetting, simulated various vignetting effects by setting different parameter values of the vignetting model explained in Section II, and added the simulated vignetting to the vignetting free images. The vignetting model parameters estimated by our algorithm are then compared with the simulated vignetting values.

It is not an easy task to acquire images totally free of vignetting. Although variable density filters or postprocessing techniques have been used to compensate for vignetting, the effect usually cannot be completely accounted for.

We design a variational experimental method to obtain the ground truth vignetting with a given camera setting. Then, we capture images with the same settings, and account for the ground truth vignetting to produce ground truth vignetting free images.

Ground truth vignetting functions of the cameras at different focal lengths were computed from multiple images of a distant white surface under approximately uniform illumination. A distant surface was used to minimize tilt effects, but generally a distant surface does not fully cover the image plane. We captured multiple images with camera translation such that each image pixel views the surface in at least one view. The image fragments of the white surface were joined and blended to obtain an accurate calibration image. We note that the illumination variation during the capturing process needs to be reduced as much as possible, in order to bring higher accuracy.

We used a Nikon E775 camera to capture more than 50 images. After the ground truth vignetting was removed, the simulated vignetting was imposed with multiple kinds of camera settings which are shown in Table I. An example image with simulated vignetting is shown in Fig. 9. The statistics of the differences between the simulated and recovered vignetting

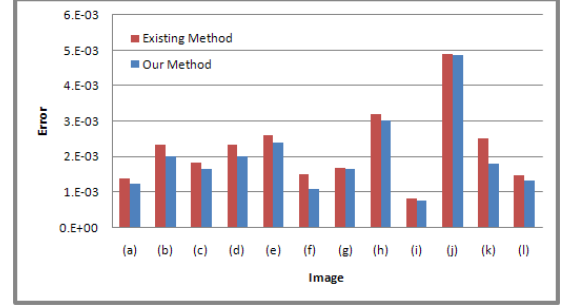


Fig. 11. Errors by our new method and previous method in [31] in removing vignetting in the images of Fig. 10.

	Our previous method [31]	Our new method
Outdoor	1.9/0.5	1.7/0.3
Indoor	2.9/1.8	2.6/1.3
Texture	5.7/2.1	5.1/0.8

TABLE II

COMPARISON OF MEAN/STANDARD-DEVIATION OF THE MEAN SQUARED ERRORS ( $\times 10^{-3}$ ) FOR 90 IMAGES.

parameters, given by mean and variance, are shown in Table I. Note that we did not include tilt effects in this error analysis because tilt is always region specific and it is difficult to determine ground truth tilt values in natural images.

In Table I, errors in  $\alpha_1$  are seen to be lower than those of the other coefficients in the geometric factor. This can be explained by the more pronounced effect of  $\alpha_1$  on the vignetting function, particularly towards the center of the image.

### C. Evaluations on Images with Real Vignetting

Our algorithm was evaluated on a total of 90 outdoor, indoor, and texture images, captured with a Canon G3, a Canon EOS 20D, and a Nikon E775. The ground truth vignetting was obtained for all cameras and settings with the method described in the preceding subsection.

Some vignetting correction results of our technique are presented in Fig. 10, along with the vignetting-based segmentation regions and their weights.

Errors computed (similarly to Eq. (29)) between the estimated vignetting functions and the ground truth functions for the images in Fig. 10 are listed in the chart of Fig. 11. In the chart, we also list the errors produced by our preliminary

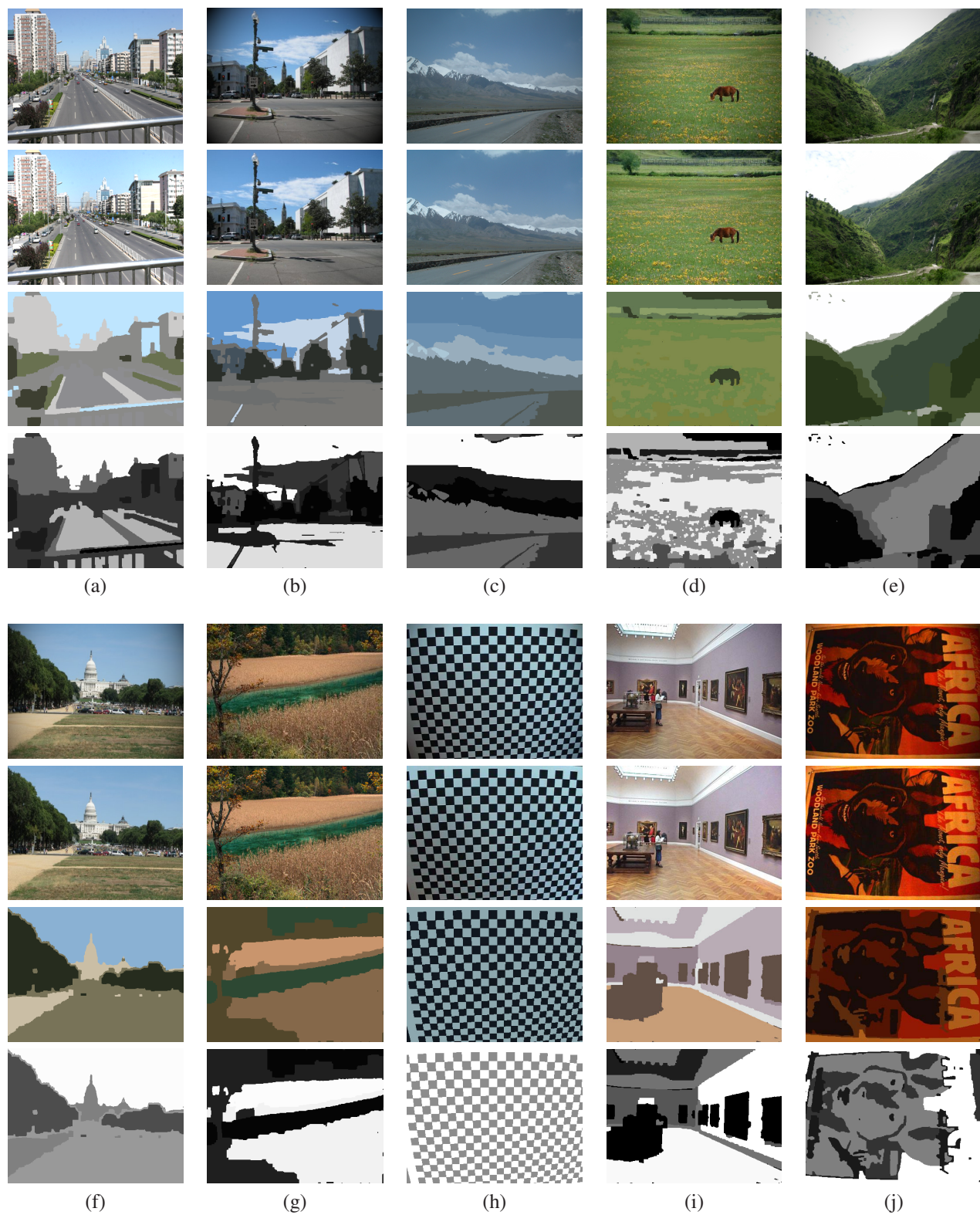


Fig. 10. Vignetting correction results. Each set from top to bottom: original image, vignetting corrected image, vignetting-based segmentation, region weights (brighter pixels indicate higher weights).

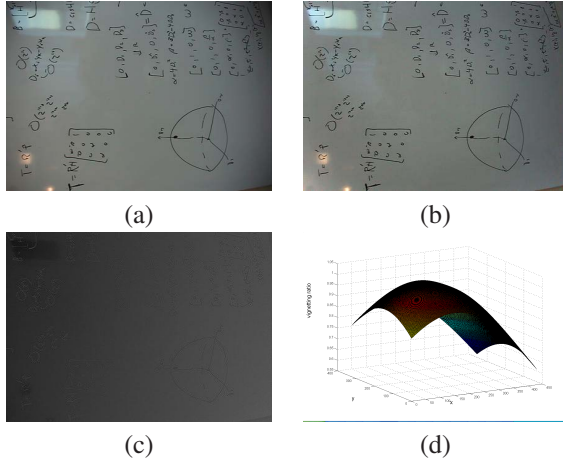


Fig. 12. Tilt effects in vignetting estimation. (a) Original image with vignetting and tilt; (b) Image corrected for only vignetting using the proposed method; (c) Tilt image, where brighter areas indicate more distant points on a surface; (d) Estimated attenuation function with both vignetting and tilt.

work [31]. Moreover, we list in Table II the residual errors for our new method as well as our previous method [31] on the 90 images. It is obvious that the introduction of the robust cost function and texture outlier detection improves the performance of our previous single image based vignetting estimation method [31].

While some slight vignetting artifacts may be visible under close examination, the correction quality is reasonable especially considering that only a single arbitrary input image is processed. For some indoor images, the amount of reliable data can possibly be low due to greater illumination non-uniformity. Images with poor data quality could potentially be identified within our method by examination of region weights, and indicated to the user.

#### D. Speed

We have compared the speed between the new method in this paper and our previous single-image vignetting correction method [31] on the 90 outdoor, indoor, and textured images. All images have a resolution of  $450 \times 600$  and all algorithms were implemented in Matlab (except for the segmentation component of [31] in C++) and run on a Dell PC with 2.39 GHz Intel Core 2 CPU. Our previous method [31] on average spends 285 seconds on one image while the method in this paper takes 347 seconds with the coarse-to-fine scheme based on an image pyramid as explained in Section III.

#### E. Other usages

While the goal of this work is to estimate and correct for the global vignetting function of the camera, tilt effects computed in vignetting estimation as shown in Fig. 12 could potentially provide some geometric information of the scene. It should be noted though that estimated tilt values are only accurate for reliable regions with high weights.

In Fig. 13, we show the application of our method to image mosaicing. Even though vignetting correction was performed independently on each image of the sequence, a reasonable

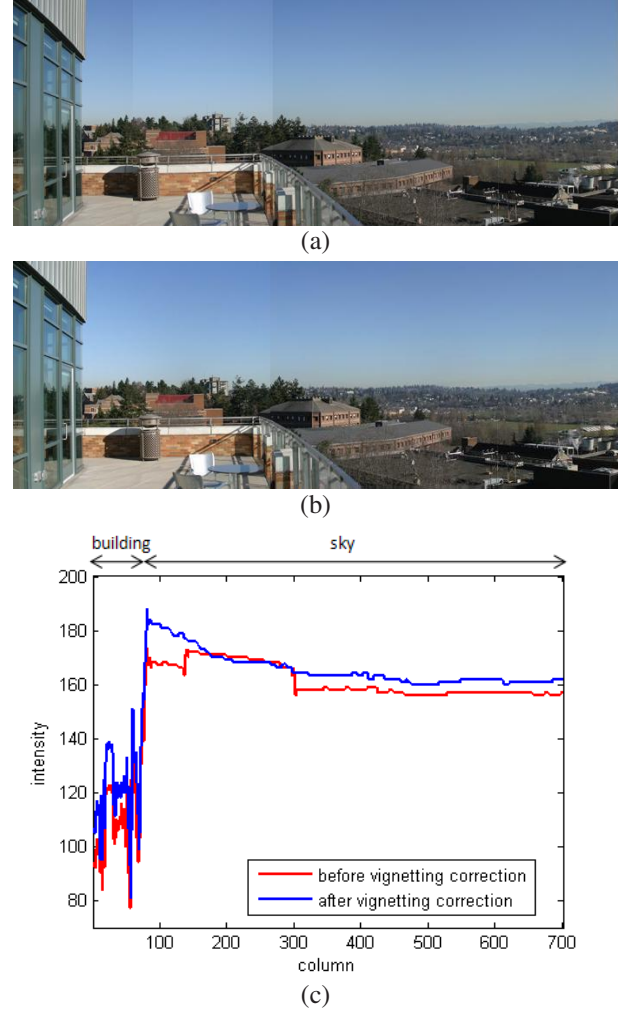


Fig. 13. The image mosaic on top exhibits obvious vignetting effects. (a) No image blending has been applied. (b) After vignetting is corrected separately in each image using our method. (c) Intensity profiles before (red) and after (blue) correction shown for one image row passing through the “sky” for the images in (a) and (b).



Fig. 14. Upper and lower rows show two failed examples of the proposed algorithm. On left side are original images with vignetting, and the right side displays the weight values computed in our algorithm.

mosaic was still obtained. In cases where overlapping images are available, joint consideration of the vignetting data among all images in the sequence would likely lead to better results. In contrast to previous works on image mosaicing [7], [8], [15], our proposed method can also jointly process data from images containing completely different content if they are captured by the same camera under the same camera setting.

#### F. Failure mode examples

The proposed algorithm can fail when only few useful segmented regions are available in the test image. On the one hand, a region is not useful to vignetting estimation when it contains illumination variations which cannot be captured by the tilt effect in our vignetting model. Ideally, tilt encompasses only smooth changes for which a plot of pixel intensities with respect to 2-D spatial coordinates forms a plane in the 3-D space. In practice, the illumination might be more complex. Both of the example images shown in Fig. 14 have this kind of non-useful regions, e.g., the wall and leaves regions in the upper and lower rows, respectively. On the other hand, a region is unsuitable for vignetting estimation when it is composed of highly non-homogeneous textures. The texture components in a region are possibly very complicated and full of variation, such as exhibited in the segmented flowers and shrub regions in Fig. 14. Regions with highly non-homogeneous texture patterns bring larger errors to vignetting estimation.

Our algorithm will produce poor results if no useful regions are found in an image. We note that all of the segmented regions in Fig. 14 received low weights in our estimation procedure because they were deemed to be not useful for vignetting estimation.

### VIII. CONCLUSION

In this paper, we introduced a method for vignetting correction using only the information available in a single arbitrary image. Adaptations to general segmentation techniques were presented for locating regions with reliable vignetting data. Within an image region, the proposed method takes advantage of the frequency characteristics and physical properties of vignetting to distinguish it from other sources of intensity variation. Experimental results demonstrate effective vignetting correction on a broad range of images.

Two mechanisms were introduced to improve robustness in estimating the proposed vignetting model's parameters, i.e., the robust statistic function and the texture outlier measure. They were shown by our experiments to improve accuracy in vignetting estimation.

Accurate correction results are generally obtained despite many regions having non-planar geometry and non-uniform illumination. The detrimental effects of non-planar geometry are reduced when distance variations of surface points from the camera are small in comparison to the distance of the surface itself, since variations in scene radiance become negligible. In many instances, the effects of non-uniform illumination appear similar to tilt, such that its effects on image intensity are incorporated into the estimated tilt factor. Low frequency

vignetting effects also remain distinct when geometry and illumination exhibit texture-like high-frequency variations, such as among leaves on a tree. As a result, reliable vignetting data often exists even in image areas with significant geometry and illumination variation.

We also showed that our algorithm fails for scenes with substantial non-uniform illumination that cannot be represented by tilt effect. Our algorithm also has difficulties with images composed primarily of highly non-uniform texture regions.

Directions for future work include joint estimation of camera parameters such as the principal point, aspect ratio, and skew, in addition to the vignetting function. In our current method, these camera parameters are assumed to be known from prior geometric calibration, but could potentially be recovered from vignetting information. Another interesting topic for future investigation is the examination of data in the RGB channels for region weighting, since vignetting should attenuate RGB values in a similar way, while other causes of region variation may not affect the channels equally.

### ACKNOWLEDGMENT

We would like to thank the anonymous reviewers for their detailed and constructive feedback, which helped improve the quality of this article.

### REFERENCES

- [1] N. Asada, A. Amano, and M. Baba. Photometric calibration of zoom lens systems. In *ICPR*, pages 186–190, 1996.
- [2] M. J. Black and G. Sapiro. Edges as outliers: Anisotropic smoothing using local image statistics. *Proceedings of the Second International Conference on Scale-Space Theories in Computer Vision*, 1682:259–270, 1999.
- [3] M. J. Black, G. Sapiro, D. H. Marimont, and D. Heeger. Robust anisotropic diffusion. *IEEE Trans. Image Processing*, 7:421–432, 1998.
- [4] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE TPAMI*, 23:1222–1239, 2001.
- [5] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: Image segmentation using expectation-maximization and its application to image querying. *IEEE TPAMI*, 24(8):1026–1038, 2002.
- [6] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE TPAMI*, 24(5):603–619, 2002.
- [7] D. Goldman and J. Chen. Vignette and exposure calibration and compensation. In *ICCV*, pages 899–906, 2005.
- [8] J. Jia and C. Tang. Tensor voting for image correction by global and local intensity alignment. *IEEE TPAMI*, 27(1):36–50, 2005.
- [9] I. T. Jolliffe. Principal component analysis. *2nd Edition, Springer*, 2002.
- [10] R. Juang and A. Majumder. Photometric self-calibration of a projector-camera system. In *CVPR*, 2007.
- [11] S. Kang and R. Weiss. Can we calibrate a camera using an image of a flat textureless lambertian surface? In *ECCV*, volume II, pages 640–653, 2000.
- [12] S. Kim and M. Pollefeys. Robust radiometric calibration and vignetting correction. *IEEE TPAMI*, 40(4):562–576, 2008.
- [13] M. Klein and T. Furtak. *Optics*. John Wiley and Sons, 1986.
- [14] S. Lin, J. Gu, S. Yamazaki, and H.-Y. Shum. Radiometric calibration using a single image. In *CVPR*, pages II-938–II-945, 2004.
- [15] A. Litvinov and Y. Schechner. Addressing radiometric nonidealities: A unified framework. In *CVPR*, pages 52–59, 2005.
- [16] C. Liu, W. T. Freeman, R. Szeliski, and S. B. Kang. Noise estimation from a single image. In *CVPR*, volume 1, pages 901–908, 2006.
- [17] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 1992.
- [18] J. Rissanen. Modelling by shortest data description. *Automatica*, 14:465–471, 1978.
- [19] P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection*. John Wiley & Sons, New York, 1987.
- [20] A. A. Sawchuk. Real-time correction of intensity nonlinearities in imaging systems. *IEEE Trans. on Computers*, 26(1):34–39, 1977.

- [21] C. Staelin and H. Nachlieli. Image noise estimation using color information. Technical Report HPL-2005-2R1, Hewlett-Packard Development Company, L.P., September 2007.
- [22] M. Tang and S. Ma. General scheme of region competition based on scale space. *IEEE TPAMI*, 23(12):1366–1378, 2001.
- [23] M. F. Tappen and W. T. Freeman. Comparison of graph cuts with belief propagation for stereo, using identical MRF parameters. In *ICCV*, volume 2, pages 900–906, 2003.
- [24] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *CVPR*, pages 839–846, 1998.
- [25] Y. Tsin, V. Ramesh, and T. Kanade. Statistical calibration of the CCD imaging process. In *CVPR*, pages 480–487, 2001.
- [26] S. Valle, W. Li, and S. J. Qin. Selection of the number of principal components: the variance of the reconstruction error criterion with a comparison to other methods. *Industrial and Engineering Chemistry Research*, 38:4389–4401, 1999.
- [27] W. S. Yambor, B. A. Draper, and J. R. Beveridge. *Analyzing PCA-based Face Recognition Algorithms: Eigenvector Selection and Distance Measures*. Empirical evaluation methods in computer vision. World Scientific, Singapore, 2002.
- [28] W. Yu. Practical anti-vignetting methods for digital cameras. *IEEE Trans. on Cons. Elect.*, 50:975–983, 2004.
- [29] W. Yu, Y. Chung, and J. Soh. Vignetting distortion correction method for high quality digital imaging. In *ICPR*, volume 3, pages 666–669, 2004.
- [30] R. Zabih and V. Kolmogorov. Spatially coherent clustering using graph cuts. In *CVPR*, volume 2, pages II–437–II–444, 2004.
- [31] Y. Zheng, S. Lin, and S. B. Kang. Single-image vignetting correction. In *CVPR*, pages 461–468, 2006.



**Yuanjie Zheng** received his B.S. degree in Agricultural Mechanical Design and Manufacture from Laiyang Agricultural College and M.E. degree in Industrial Mechanical Design and Theory from Jilin University of China. He received his Ph.D. degree in Pattern Recognition and Intelligent Systems from Shanghai Jiao Tong University of China in 2006. His current research interests are in computer vision and medical image analysis, including segmentation, matting, registration, camera calibration, feature extraction, and

classification.



**Stephen Lin** received the B.S.E degree in electrical engineering from Princeton University, Princeton, NJ, and the Ph.D. degree in computer science and engineering from the University of Michigan, Ann Arbor. He is a Lead Researcher in the Internet Graphics Group, Microsoft Research Asia. His current research interests include photometric analysis in computer vision and reflectance modeling in computer graphics.



**Chandra Kambhamettu** received his PhD in Computer Science and Engineering from the University of South Florida in 1994. From 1994–96, he was a research scientist at NASA-Goddard, where he received the “1995 Excellence in Research Award” from the Universities Space Research Association (USRA). He is currently an Associate Professor at the Department of Computer and Information Sciences, University of Delaware, where he leads the Video/Image Modeling and Synthesis (VIMS) group. His research interests include video modeling and image analysis for biomedical, remote sensing and multimedia applications. Dr. Kambhamettu is best known for his work in motion analysis of deformable bodies, for which he received NSF CAREER award in 2000. He published over 130 peer-reviewed papers, supervised nine PhD students and several Masters students in his areas of interest. He served as Area Chair for ACCV and MICCAI, has been technical committee member in leading computer vision conferences. He also served as associate editor for Pattern Recognition journal and IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI).



**Jingyi Yu** is an assistant professor at Computer and Information Science Department at the University of Delaware. He received his B.S. from Caltech in 2000 and M.S. and Ph.D. degree in EECS from MIT in 2005. His research interests span a range of topics in computer graphics, computer vision, and image processing, including computational photography, medical imaging, non-conventional optics and camera design, tracking and surveillance, and graphics hardware.



**Sing Bing Kang** received his Ph.D. in robotics from Carnegie Mellon University, Pittsburgh in 1994. He is currently Principal Researcher at Microsoft Corporation working on image-based modeling as well as image and video enhancement. Sing Bing has co-edited two books in computer vision (“Panoramic Vision” and “Emerging Topics in Computer Vision”), and coauthored a book on image-based rendering. He has served as area chair and member of technical committee for the three major computer vision conferences (ICCV, CVPR, and ECCV). He has also served as papers committee member for SIGGRAPH’07 and SIGGRAPH Asia’08. Sing Bing is currently CVPR’09 program co-chair and IEEE TPAMI Associate Editor.