

# Natural Shadow Matting

TAI-PANG WU and CHI-KEUNG TANG

The Hong Kong University of Science and Technology  
and

MICHAEL S. BROWN

Nanyang Technological University  
and

HEUNG-YEUNG SHUM

Microsoft Research Asia

---

This paper addresses the problem of *natural shadow matting*: the removal or extraction of natural shadows from a single image. Because textures are maintained in the shadowless image after the extraction process, our approach produces some of the best results to date among shadow removal techniques. Using the image formation equation typical of computer vision, we advocate a new model for shadow formation where shadow effect is understood as light attenuation instead of a mixture of two colors governed by the conventional matting equation. This leads to a new shadow equation with fewer unknowns to solve, where a three-channel shadow matte and a shadowless image are considered in our optimization. Our problem is formulated as one of energy minimization guided by user-supplied hints in the form of a quadmap which can be specified easily by the user. This formulation allows for robust shadow matte extraction while maintaining texture in the shadowed region by considering color transfer, texture gradient, and shadow smoothness. We demonstrate the usefulness of our approach in shadow removal, image matting and compositing.

Categories and Subject Descriptors: I.3.6 [Computer Graphics]: Methodology and Techniques—*Interaction techniques*; I.4.8 [Image processing and computer vision]: Scene Analysis—*color*

General Terms: Algorithms

Additional Key Words and Phrases: Shadow removal and extraction, shadow matting, interactive extraction.

---

## 1. INTRODUCTION

Matting and compositing techniques are invaluable in creating realistic visual effects in television, film and video production. In matting, a user-specified foreground object is isolated (or cut) from an image. In compositing, the image cut-out is combined with a new background image to produce a composited image. Because shadows provide

---

Authors' address: Chi-Keung Tang and Tai-Pang Wu, Department of Computer Science and Engineering, the Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong.

Michael S. Brown, School of Computer Engineering, Nanyang Technological University, Singapore, 639798

Heung-Yeung Shum, Microsoft Research Asia, 5/F, Beijing Sigma Center, No.49 Zhichun Road, Hai Dian District, Beijing, China 100080.

Permission to make digital/hard copy of all or part of this material without fee for personal or classroom use provided that the copies are not made or distributed for profit or commercial advantage, the ACM copyright/server notice, the title of the publication, and its date appear, and notice is given that copying is by permission of the ACM, Inc. To copy otherwise, to republish, to post on servers, or to redistribute to lists requires prior specific permission and/or a fee.

© 20YY ACM 0730-0301/20YY/0100-0001 \$5.00

such strong perceptual cues to an object’s depth and relationship to its background, it is often necessary to cut and paste an object’s corresponding shadow to create a visually convincing composite. Composites without shadows can appear unnatural and artificial (see Fig. 1).

**Shadow compositing equations** Inspired by the image compositing equation [Porter and Duff 1984]:

$$I = \alpha F + (1 - \alpha)B \quad (1)$$

where  $F$ ,  $B$ , and  $\alpha$  are the foreground image, background image, and fractional alpha respectively, the shadow compositing equation proposed in [Chuang et al. 2003] is defined as:

$$I = \beta L + (1 - \beta)S \quad (2)$$

where  $L$  is the lit image,  $S$  is the shadow image, and  $\beta$  is the shadow matte. In [Chuang et al. 2003],  $\beta$  was modeled as fractional occlusion with respect to the light source. Note that the image and shadow compositing equations are identical. In the case of a color image,  $L$  and  $S$  have three color channels (Red, Green, and Blue) and with the shadow matte,  $\beta$ , being the same for all three channels. Assuming a point-light source and uniform chromatic response to shadows, the authors [Chuang et al. 2003] proposed a method that used a video sequence to estimate the terms of Eqn. 2. From a single color input image, however, solving for the terms of these equations is ill-posed as the equations are under-constrained with three channel equations and seven unknowns.

Recent techniques such as Bayesian matting [Chuang et al. 2001] and Poisson matting [Sun et al. 2004] have provided state-of-the-art solutions in addressing the image compositing equation given a single input image. These techniques, referred to as *natural image matting*, employ user-supplied hints, in the form of a trimap, to help constrain the equations in order to produce a solution. There are several problems, however, when matting techniques such as Bayesian matting are applied to extract a shadow matte from a single image (Fig. 2). First, in shadow matting, the shadow layer is semi-transparent over the shadow region, while in traditional matting the foreground layer is opaque over a substantial region. As a result, traditional matting does not provide the amount of attenuation caused by the shadow. Second, traditional matting does not provide a shadowless image. As shown in Fig. 2, textures within the shadowed region are also extracted using Bayesian matting because the Bayesian formulation cannot explicitly consider the effect of textures when color samples are collected using the user-defined trimap.

In this paper, we address the problem of natural shadow matting from a single image from a different perspective. Instead of modeling a shadow as a per-pixel fractional visibility with respect to a primary light source, we consider a *natural shadow* observed in the real world as the aggregate effect of light attenuation, possibly resulted by the complex interaction of various optical phenomena. Rather than modeling these complex phenomena, we adopt in this paper the image formation equation typically used in computer vision.

In [Barrow and Tenenbaum 1978], an image of a scene is the pixelwise product of its two corresponding intrinsic images defined as:

$$I = \mathcal{R}\mathcal{L} \quad (3)$$

where  $\mathcal{R}$  is the albedo image and  $\mathcal{L}$  is the illumination image.  $\mathcal{L}$  encapsulates illumination, including both shading and shadows. Working from Eqn. 3, we can further decompose  $\mathcal{L}$  into a multi-channel shadow matte  $\beta$  which encapsulates the combined effect attenuating the illumination at each pixel, and another image  $\mathcal{L}'$  which describes shading only, such that  $\mathcal{L} = \beta\mathcal{L}'$ . A multi-channel shadow matte is used in our equation to allow the three color channels to respond differently for different materials and illumination.

By decoupling  $\mathcal{L}$  in this manner, we can express Eqn. 3 as  $I = \beta\mathcal{R}\mathcal{L}'$  or simply

$$I = \beta\mathcal{F} \quad (4)$$



$I$  (input image)



$\mathcal{F}$  (shadowless image)



$\beta$  (shadow matte)



composite without shadow



composite with the shadow brightened

Fig. 1. Application in image composition: *Puzzled Boy*. Textures are maintained in the extracted shadowless image  $\mathcal{F}$ . Shadow smoothness is maintained in the extracted shadow matte  $\beta$ .

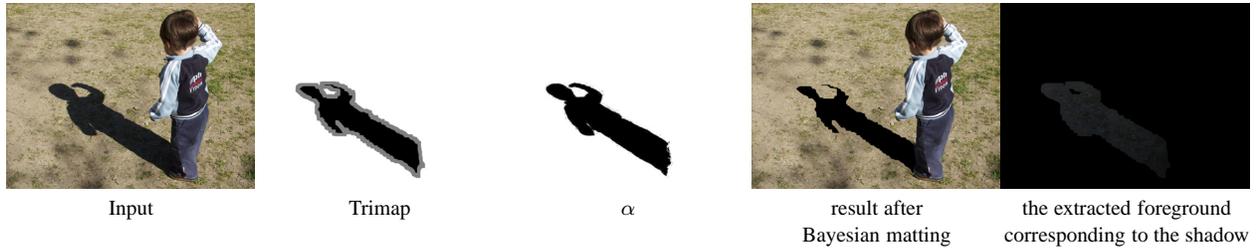


Fig. 2. Bayesian matting applied to natural shadow matting. We expect that global Poisson matting produces a similar result because the method assumes smooth matte gradient. Since nonsmooth texture gradients are not explicitly considered by Bayesian matting and global Poisson matting in their color sampling steps (or trimaps), the textures under the shadow are also extracted. Note that a hole is left in the resulting image, due to the use of trimap in specifying “definitely shadowed” region.

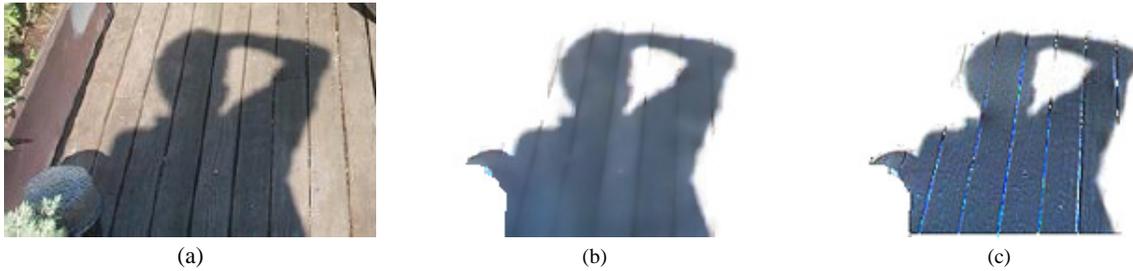


Fig. 3. Comparison of shadow matte extraction: (a) Input. (b) The shadow matte produced by our method. (c) The shadow produced by shadow removal method [Finlayson et al. 2004]. Note that the textures of the wood under the shadow are also extracted in (c).

where  $\mathcal{F}$  is the shadowless image. We call Eqn. 4 the *natural shadow equation*. This image model simply means that the observed image  $I$  is the shadowless image  $\mathcal{F}$  scaled by the fractional shadow matte image  $\beta$ . From this new equation several important observations can be made:

- (1) While Eqn. 4 is still under-constrained because each channel equation gives two unknowns (one for  $\beta$  and one for  $\mathcal{F}$ ), it has less unknowns than Eqn. 2 (three equations and seven unknowns).
- (2) Despite its simplicity, our image formation model for  $I$  does *not* assume any simplified lighting model.
- (3) We can multiply the shadowless  $\mathcal{F}$  by the shadow matte  $\beta$  to produce the observed shadowed image  $I$ .

**Shadow removal and shadow matting** Using this new shadow equation, it should be possible to apply shadow removal algorithms to obtain the shadowless  $\mathcal{F}$ , thus deriving the shadow matte by  $\beta = \frac{I}{\mathcal{F}}$ . In practice, however, this is only possible if a perfect  $I$  and a perfect  $\mathcal{F}$  are both available. Even very good shadow removal algorithms, such as [Finlayson et al. 2004], are unable to produce perfect  $\mathcal{F}$  and result in shadow mattes with unwanted noise that are unacceptable for compositing (Fig. 3). Because  $I$  is expressed as a pixelwise product of  $\mathcal{F}$  and  $\beta$  in our shadow equation, these two maps are interdependent and so both of them should be considered in the optimization, subject to a smoothness criterion that imposes the necessary shadow smoothness while maintaining nonsmooth texture gradients. As such, our shadow extraction algorithm goes beyond shadow matting, and produces some of the best results to date in shadow removal by requiring only simple user assistance.

One caveat in our image model for shadow extraction is in the situation where a severe amount of image noise is present. In this case, our shadow removal/matting may not successfully extract *both* a good  $\beta$  and a good  $\mathcal{F}$ , because the model  $I = \beta\mathcal{F}$  does not explicitly handle image noise. This will be further elaborated and illustrated in the result



Fig. 4. The quadmap:  $S$  = shadowed (red),  $\bar{S}$  = nonshadowed (blue),  $U$  = uncertain (pink), and  $E$  = excluded (yellow) regions marked by the user. The boundary  $\Omega$ , defined as  $\partial(S \cup U) \setminus E$ , is the pink curve not intersecting the  $E$  region.

section.

**Quadmap and trimap** Note that image matting techniques cannot be used to solve Eqn 4 because it is different from Eqn 2, and that we model a shadow as light attenuation instead of color composition typically employed in image matting. While techniques for image matting are inapplicable here, the *trimap* used by Bayesian and Poisson Matting provides inspiration to our UI design. User-supplied hints are crucial in natural shadow matting, because it is challenging to identify shadowed and nonshadowed regions given a single image captured by a normal camera.

In Bayesian and Poisson matting, the under-constrained system of equations is solved using pixel information specified via a trimap in which image regions marked by the user are defined as: “definitely foreground region” ( $\alpha = 1$ ), “definitely background region” ( $\alpha = 0$ ) and “uncertain region”. Local color statistics from the two definite regions are sampled and analyzed to derive an optimal  $\alpha \in [0, 1]$  in the uncertain region where the foreground and background layers blend together. The optimal  $\alpha$  is dependent on the numerical approach used, either Bayesian-based or Poisson-based.

In natural shadow matting, three regions, namely the “definitely shadowed region,” “definitely nonshadowed regions,” and “uncertain region” can be specified likewise. The fourth region, “excluded” region, is specified to exclude irrelevant colors from shadow calculation. These four regions together constitute a *quadmap* in our approach, aiming to provide relevant color statistics to the automatic optimization step to optimize for the shadow matte and the shadowless image.

**Our approach** Given an image of a complex scene, our approach provides a new interactive technique for shadow markup, followed by an energy minimization to solve for the terms of our shadow equation. The marking step relies on a user interface (UI), which specifies shadowed and nonshadowed regions by a few strokes on the image. Our technique then proceeds in two steps. We first compute an approximate shadowless image  $\hat{F}$  based on color statistics extracted from the marked-up regions. We then use this approximate shadowless image to define an optimization energy for  $\beta$  that includes a color term and a smoothness term. The color term seeks to respect our shadow equation for  $\beta$  and the approximate shadowless image  $\hat{F}$ , while the smoothness term seeks to impose shadow smoothness and respect texture detail.

Notation	Meaning
$\mathcal{S}$	Definitely shadowed region
$\overline{\mathcal{S}}$	Definitely nonshadowed region
$\mathcal{U}$	Uncertain region
$\mathcal{E}$	Excluded region
$\beta$	Shadow matte
$\mathcal{F}$	Shadowless image
$\hat{\mathcal{F}}$	Coarse estimation of $\mathcal{F}$
$\varphi_{\mathcal{S}}$	Shadowed affinity map for $\mathcal{S}$
$\varphi_{\overline{\mathcal{S}}}$	Nonshadowed affinity map for $\overline{\mathcal{S}}$

Table I. Notations used in this paper.

## 2. RELATED WORK

Shadow generation techniques typically used in the film industry either are manual [Wright 2001] or involve projecting the foreground object’s own alpha matte as its shadow. Manual shadow generation is time-consuming while the projection approach can distort the shadow shape producing unrealistic shadows. It is possible to use natural image matting techniques [Chuang et al. 2001; Sun et al. 2004], however, these approaches are designed to extract opaque foreground objects, and as a result, will extract the full shadow region with its underlying texture and structure rather than extracting a semi-transparent shadow layer. These approaches also suffer from the problems that the specified *trimap* is not sufficient in solving the natural shadow equation, and in distinguishing the shadow from the background when the object that casts the shadow is present. The only approach that has examined shadow matting directly [Chuang et al. 2003] makes several assumptions about the scene lighting and requires a video sequence to solve for the shadow matte.

Shadow removal techniques are related to shadow matting. In shadow removal, unwanted shadows are detected and eliminated from an image. Because the goal is to obtain a shadow-free image, these approaches do not produce shadow mattes that are necessarily usable for compositing. Using the retinex algorithm proposed in the classical paper [Land and McCann 1971], an algorithm was proposed in [Finlayson et al. 2002] to remove a shadow from a single image, where the goal is to extract meaningful information from an image rather than producing an aesthetic image free of shadows. Starting from a single image, in image-based modeling and photo editing [Oh et al. 2001], the authors used texture-illumination decoupling to remove soft shadows on textured surfaces, followed by bilateral filtering. Their method assumes large-scale luminance variations and small-scale textures. As long as a sharp shadow has high contrast it can be handled by this method. Note that a low-contrast sharp shadow boundary will be classified to the textures (as high frequency component) instead of the low frequency (shadow) component. In computer vision, the problem of shadow removal is addressed using intrinsic image estimation. Weiss [Weiss 2001] used a sequence of  $N$  images to derive one reflectance image and  $N$  illumination images. [Tappen et al. 2002] separated intrinsic images from a single image by classifying image derivatives as changes due to either reflectance or shading. In [Baba et al. 2004], an image is segmented into sunshine, penumbra and umbra, followed by color and brightness adjustment for shadow removal. Automatic shadow removal algorithm from one image [Finlayson et al. 2004] uses entropy minimization to derive an illumination invariant grayscale image for shadow removal, without resorting to any calibration.

## 3. NATURAL SHADOW MATTING

Our technique exploits the information provided by the user (a quadmap) to first compute an approximate shadowless image. For this, we estimate the probability that a pixel is part of a shadow and use the color histogram from the shadowed and unshadowed regions to deduce a color transfer that transforms colors in shadow into colors in lit regions. The transfer applied to a pixel is proportional to the estimated probability to be in shadow. In addition, we refine this

approximate shadowless image using an affinity map that reduces the effect of color transfer at shadow boundaries. Once this approximate shadowless image has been computed, our optimization uses energy minimization to solve for the shadow matte  $\beta$  while enforcing two desirable properties: the product of the shadow matte and the approximate shadowless image should reconstruct the input image (realized by an  $E_{color}$  energy term), and the shadow matte should be as smooth as possible where the shadow is smooth (by an  $E_{smooth}$  energy term). These energy terms will be described in detail in section 3.3.

A concise summary of *color transfer* is given in the following:

**Color transfer** The goal of color transfer between two images [Reinhard et al. 2001] is to synthesize an image by transferring some aspects of the distribution of the data points in color space from a source  $I_s$  to a target image  $I_t$ . Let  $T(I_t(x))$  be the synthesized image. Let  $\mu_s$  and  $\sigma_s$  be the mean and standard deviation of the intensities on the source image  $I_s$ .  $\mu_t$  and  $\sigma_t$  are similarly defined for the target image  $I_t$ . Then:

$$T(I_t(x)) = \mu_s + \frac{\sigma_t}{\sigma_s}(I_t(x) - \mu_t)$$

Our algorithm proceeds in the following steps:

- (1) Compute an estimate of the probabilities that pixels are in  $\mathcal{S}$ ,  $\overline{\mathcal{S}}$ ,  $\mathcal{U}$  and  $\mathcal{E}$ .
- (2) Compute a coarse estimate  $\hat{\mathcal{F}}$  for the shadowless image. Use it to define a color (or data) energy term  $E_{color}$ .
- (3) Use the shadow and nonshadow probabilities to define a spatial smoothness energy term  $E_{smooth}$  that enforces smoothness except at shadow boundaries.
- (4) Minimize a quadratic cost function to estimate  $\beta$  over the pixel regions  $\mathcal{S}$  and  $\mathcal{U}$ .  $\mathcal{F}$  can then be deduced from  $\beta$  and our shadow matting Eqn. 4.

Table I summarizes the notations used in this paper. To extract a shadow matte, the user specifies a *quadmap* consisting of four regions (Fig. 4). These four regions provide the necessary cues for the optimization process. We first define the four regions, describe the user interface (UI) used by the user to mark them, and then explain how the UI couples with the automatic optimization to extract color statistics where each region’s implications are detailed in Section 3.3 when the energy functions for optimization are introduced.

### 3.1 UI design for marking

Our simple drawing-type UI utilizes the user’s ability to identify shadows and nonshadows easily and does not require precise mark up. The extraction process is triggered once the user has specified a number of rough regions. Either loops or brush strokes (or a combination of the two) can be used to specify a region. A region is not limited to a single connected component. For simplicity, Fig. 4 shows image examples that have been marked up using only loops. For more complex shadows, the user may perform shadow matte extraction in multiple shadow regions to obtain the overall result.

In a *quadmap*, the marked regions are categorized into *definitely shadowed region*  $\mathcal{S}$ , *definitely nonshadowed region*  $\overline{\mathcal{S}}$ , *uncertain region*  $\mathcal{U}$  and *excluded region*  $\mathcal{E}$ .  $\mathcal{S}$  indicates where the darkest shadow occurs.  $\overline{\mathcal{S}}$  samples shadowless regions. For satisfactory results, we require the user to pick  $\overline{\mathcal{S}}$  and  $\mathcal{S}$  regions that are similar in textures/patterns but not in intensity (see Fig. 4). The similarity between these two regions helps the algorithm to identify and recover the shadowed pixels.  $\mathcal{U}$  can be easily marked except that it should enclose  $\mathcal{S}$  and therefore  $\mathcal{U}$  may contain both shadowed and nonshadowed pixels. So,  $\mathcal{S} \cup \mathcal{U}$  is the region where optimization is performed, that is, the region where natural shadow extraction operates.  $\mathcal{E}$  indicates the regions of pixels that should be excluded from optimization. For example,

the shadow casting object (the cat) in Fig. 4 should be carefully excluded because the object’s colors are irrelevant to shadow extraction. While this region needs to be carefully excluded from the extracted shadow, the user only needs to mark  $\mathcal{E}$  roughly, because  $\mathcal{E}$  will be automatically refined by identifying pixels at which both the shadow and nonshadow probabilities are close to zero.

Note that, although “quadmap” and “trimap” are related, our use of quadmaps is different from the use of trimaps in natural matting in two ways. In a trimap, the specified foreground and background regions capture the locality where colors are sampled to estimate the foreground color, background color and  $\alpha$  in the uncertain region within a local neighborhood of influence. In our quadmap, the information that each region captures consists of color statistics or color distributions and no locality is utilized. An energy function is derived from the color statistics, which will be optimized to obtain the solution.

### 3.2 Extraction of color statistics

From the user supplied quadmap, color statistics will be obtained to identify pixels from the shadowed and nonshadowed regions. The distribution of colors in these regions are described by their probability density functions (pdf) as a Gaussian Mixture Model (GMM) [Mitchell 1997], which is derived from the histogram of  $\mathcal{S}$  and  $\overline{\mathcal{S}}$  to give continuous pdfs. Simply speaking, one or more Gaussians are used to approximate the continuous representation of the histograms. The number of Gaussians necessary is related to the distribution of colors inside the marked shadowed and nonshadowed regions.

Given the histograms of  $\mathcal{S}$  and  $\overline{\mathcal{S}}$  (Fig. 5), the GMMs used for representing the pdfs of the color statistics in  $\mathcal{S}$  and  $\overline{\mathcal{S}}$  are respectively estimated using the Expectation-Maximization (EM) algorithm [Mitchell 1997]. The EM algorithm is widely used to estimate the continuous representation given the observations generated by certain distributions. To achieve better pixel classification as being shadowed or otherwise, we use 3D GMM where each model estimated in the algorithm is assumed to be 3D Gaussian in RGB color space. The EM algorithm consists of two steps: the E-step (expectation step) and the M-step (maximization step). Using  $\mathcal{S}$  as an example (the GMM for  $\overline{\mathcal{S}}$  can be computed similarly):

*E-step:* Suppose we have  $K$  Gaussians. The probability  $P_i(x)$  that a pixel color  $I(x)$  belongs to the  $i$ th Gaussian  $G_{\mathcal{S}}^i(\mu_{\mathcal{S}}^i, \sigma_{\mathcal{S}}^i, \pi_{\mathcal{S}}^i)$  is defined as:

$$P_i(x) = \frac{\pi_{\mathcal{S}}^i \exp\left(-\frac{\|I(x) - \mu_{\mathcal{S}}^i\|^2}{2\sigma_{\mathcal{S}}^i \sigma_{\mathcal{S}}^i}\right)}{\sum_{j=1}^K \pi_{\mathcal{S}}^j \exp\left(-\frac{\|I(x) - \mu_{\mathcal{S}}^j\|^2}{2\sigma_{\mathcal{S}}^j \sigma_{\mathcal{S}}^j}\right)} \quad (5)$$

where  $K$  is the total number of Gaussians.

*M-step:* the mean  $\mu_{\mathcal{S}}^i$ , standard deviation  $\sigma_{\mathcal{S}}^i$ , and the weight  $\pi_{\mathcal{S}}^i$  of each Gaussian  $G_{\mathcal{S}}^i$  are re-estimated by:

$$\mu_{\mathcal{S}}^i = \frac{1}{Z} \sum_{x \in \mathcal{S}} P_i(x) I(x) \quad (6)$$

$$\sigma_{\mathcal{S}}^i = \sqrt{\frac{\sum_{x \in \mathcal{S}} P_i(x) \|I(x) - \mu_{\mathcal{S}}^i\|^2}{Z}} \quad (7)$$

$$\pi_{\mathcal{S}}^i = \frac{Z}{N} \quad (8)$$

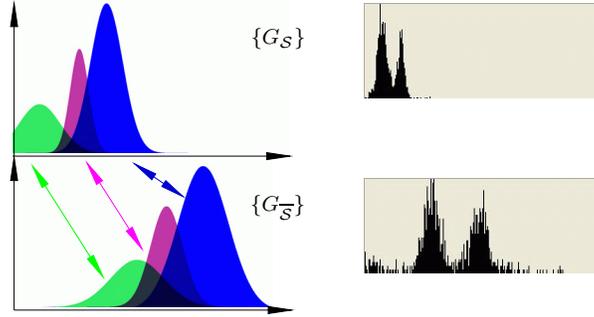


Fig. 5.  $\{G_S\}$  and  $\{G_{\bar{S}}\}$  for modeling color statistics. Typical real histograms of a complex scene are also shown. Color transfer  $\mathcal{T}$  is performed between corresponding Gaussians.

where  $Z = \sum_{x \in \mathcal{S}} P_i(x)$  is the normalization factor, and  $N$  is the total number of pixels in the region  $\mathcal{S}$ .

The E-step and M-step are iterated until convergence. To initialize the algorithm, K-means clustering [Duda et al. 2001] is a common choice and is used in our implementation, where  $K$  initial cluster centers are first generated by uniform distribution. Upon convergence, the estimated means and variances of the  $K$  clusters are treated as the initial guesses for the EM algorithm. The assignment of each pixel observation in  $\mathcal{S}$  to one of the  $K$  clusters (given by calculating the shortest distance from the observation to the cluster) and re-estimation of the  $K$  means (given by recalculating the means of pixel intensities assigned to each cluster) are performed alternately until convergence.

After the EM algorithm has been used to find the GMM for the  $\mathcal{S}$  and  $\bar{\mathcal{S}}$  regions, the “excluded” region,  $\mathcal{E}$  is refined as follows: all pixels that do not belong to the union of  $\mathcal{S}$  and  $\bar{\mathcal{S}}$  (indicated by their respective probabilities being close to zero) are excluded from  $\mathcal{S} \cup \bar{\mathcal{S}}$ .

### 3.3 Energy minimization for automatic optimization

After refining the excluded region (and thereby finalizing the region  $\mathcal{S} \cup \bar{\mathcal{S}}$ ), the shadow is now ready to be extracted. We formulate the shadow extraction problem into one of optimization and use energy minimization to solve for the optimal  $\beta$  and  $\mathcal{F}$ . The energy terms used in the optimization are defined here.

Recall that  $I = \beta\mathcal{F}$  where  $I$ ,  $\beta$  and  $\mathcal{F}$  consists of three color channels, and that the multiplication is done in the independent R, G, and B channels, respectively. This provides the implementation simplicity and we perform optimization in each individual channel and combine the results. In the following equations, the term  $\beta$  is used to represent any one of the three color channels. Since  $\mathcal{F}$  can be expressed in terms of  $\beta$ , i.e. ( $\mathcal{F} = \frac{I}{\beta}$ ), we focus on how to estimate  $\beta \in (0, 1]$ . Because we perform energy minimization in each channel, the EM algorithm is run again, now on each channel, to produce 1D GMMs for each color channel for  $\mathcal{S}$  and  $\bar{\mathcal{S}}$  respectively. To simplify notations, we still use  $\mu$  and  $\sigma$  to denote the mean and standard deviation for each channel (scalars) in the rest of this section.

The energy function for estimating the  $\beta$  matte consists of:

$$E(\beta) = E_{color}(\beta) + E_{smooth}(\beta) \quad (9)$$

where  $E_{smooth}(\beta)$  controls the smoothness of the estimated  $\beta$  matte, and  $E_{color}$  measures the difference between the observed image and the estimated shadow applied to a coarse shadowless image. The coarse shadowless image is estimated by exploiting the color statistics (1D GMM) of the definitely shadowed and definitely nonshadowed regions to perform color transfer that turns shadowed colors into lit colors.



Fig. 6. Some examples of the coarse shadowless image  $\hat{\mathcal{F}}$  we estimated using the measure of model affinity. Note that it provides a very crude estimations of the shadowless image but a good indicator for encoding  $E_{color}$ .

The differences between our color transfer approach and those proposed in [Reinhard et al. 2001; Welsh et al. 2002] will be detailed shortly and an illustrative example will be given in the result section.

In the following, we detail how the energy terms in Eqn. 9 can effectively address the issues in shadow extraction from a complex image, namely, preserving texture details (nonsmooth texture) in the estimation of  $\mathcal{F}$  image while maintaining the smoothness of the extracted shadow matte  $\beta$ .

**3.3.1 Color energy ( $E_{color}$ ).** Recall that our  $E_{color}$  term measures the difference between the observed image and the estimated shadow  $\beta$  applied to a coarse estimate  $\hat{\mathcal{F}}$  of shadowless image  $\mathcal{F}$ . In order to obtain  $\hat{\mathcal{F}}$ , we develop a new color transfer that exploits the color statistics to “illuminate” shadowed colors. Our color transfer strategy extends Reinhard et al.’s work to selectively apply color transfer to pixels that are in shadow. We use both a *model affinity* that roughly identifies pixels in shadow based on color statistics, and a *spatial affinity* which excludes pixels at the shadow boundary  $\Omega$  and enforces a smooth application of transfer in the neighborhood of the  $\Omega$ . The detail of these two considerations will be discussed later.

Specifically, for each pixel  $x \in \mathcal{S} \cup \mathcal{U}$ :

- (1) Apply the modified color transfer to  $I(x)$  which takes its model affinity into consideration.
- (2) Calculate the spatial affinity of pixel  $x$ .
- (3) Compute  $\hat{\mathcal{F}}(x)$  which is a function of spatial affinity and the modified color transfer.
- (4) Define  $E_{color}$  as the squared difference between  $I(x)$  and  $\beta(x)\hat{\mathcal{F}}(x)$ .

The details of the above steps will be described in the following. Recall that  $\mathcal{S}$  and  $\bar{\mathcal{S}}$  should be marked such that they are similar in textures or patterns except in intensity. It is easy to mark such  $\mathcal{S}$  and  $\bar{\mathcal{S}}$ , as shown in Fig. 4.

Let the respective GMMs be  $\{G_{\bar{\mathcal{S}}}^i(\mu_{\bar{\mathcal{S}}}^i, \sigma_{\bar{\mathcal{S}}}^i, \pi_{\bar{\mathcal{S}}}^i)\}$  and  $\{G_{\mathcal{S}}^i(\mu_{\mathcal{S}}^i, \sigma_{\mathcal{S}}^i, \pi_{\mathcal{S}}^i)\}$ ,  $i = 1, 2, \dots, K$ .

Using our image model  $I = \beta\mathcal{F}$  and that  $\beta$  is nearly constant in  $\mathcal{S}$  (darkest shadow), the histogram of  $\mathcal{S}$  should be approximately a scaled and shifted version of that of  $\bar{\mathcal{S}}$ . Based on this observation,  $G_{\bar{\mathcal{S}}}^i(\mu_{\bar{\mathcal{S}}}^i, \sigma_{\bar{\mathcal{S}}}^i, \pi_{\bar{\mathcal{S}}}^i)$  and  $G_{\mathcal{S}}^i(\mu_{\mathcal{S}}^i, \sigma_{\mathcal{S}}^i, \pi_{\mathcal{S}}^i)$ ,  $i = 1, 2, \dots, K$  are *corresponding*, if we take respectively from the two GMMs the  $K$  Gaussians with the largest weights and sort them by their  $\mu$ ’s. Fig. 5 shows an example. In practice, we find that  $K \leq 2$ . We now derive the color energy term by considering the model affinity.

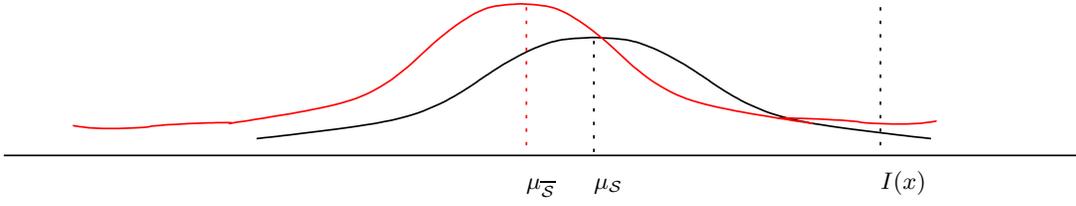


Fig. 7. The use of the weight  $d$  to measure model affinity.

**Model affinity** We want to infer a coarse shadowless image  $\hat{\mathcal{F}}$  by estimating the light attenuation induced by the shadow matte. Fig. 6 shows a few examples of  $\hat{\mathcal{F}}$  and note that they are crude estimation only but provide adequate information to encode the  $E_{color}$  energy (Eqn 18).

Consider a *shadowed* pixel  $x$  in  $\mathcal{S} \cup \mathcal{U}$ . Suppose  $I(x)$  is generated by Gaussian model  $G_{\mathcal{S}}^i$  for some  $i = 1, \dots, K$ . The corresponding Gaussian,  $G_{\mathcal{S}}^i$ , should therefore be used to transfer the color (intensity) for estimating the attenuation at  $x$ . Therefore, to estimate a coarse shadowless image, one way is to directly use the estimated GMM  $\{G_{\mathcal{S}}\}$  and  $\{G_{\mathcal{U}}\}$  to measure the respective probabilities of a pixel being shadowed or otherwise. While this can be used, we adopt an easier alternative in our implementation that works well in practice. Our alternative is to assign a larger weight to the Gaussian with mean closer to the pixel’s color, and hence we use the term *model affinity*. The underlying reason for this is illustrated in Fig. 7 where  $I(x)$  falls onto the “tail” part of both Gaussians. While both probabilities are close to zero, it is more reasonable to assign a larger weight to the Gaussian with mean closer to the pixel’s color. Hence, *model affinity* is measuring the affinity of the pixel’s color to a particular Gaussian.

Refer to Fig. 8. Let  $x$  be a pixel in  $\mathcal{S} \cup \mathcal{U}$ . Then

$$T(I(x)) = \frac{1}{\sum_i d_{\mathcal{S}}^i} \sum_{i=1}^K d_{\mathcal{S}}^i T_i(I(x)) \quad (10)$$

is the transfer function weighted by  $d_{\mathcal{S}}^i$ , where

$$d_{\mathcal{S}}^i = \frac{1}{\left| \frac{I(x) - \mu_{\mathcal{S}}^i}{\sigma_{\mathcal{S}}^i} \right| + \varepsilon} \quad (11)$$

is the weighted inverse color distance to each Gaussian in  $\{G_{\mathcal{S}}\}$ , where  $\varepsilon$  is a small constant ( $\varepsilon = 0.1$  for all of our examples).  $T_i(I(x))$  is defined as

$$T_i(I(x)) = \mu_{\mathcal{S}}^i + \frac{\sigma_{\mathcal{S}}^i}{\sigma_{\mathcal{U}}^i} (I(x) - \mu_{\mathcal{S}}^i) \quad (12)$$

In other words,  $T(I(x))$  in Eqn. 10 is the integrated color transfer result based on the corresponding  $\{G_{\mathcal{S}}^i, G_{\mathcal{U}}^i\}$  pairs. That is, for each pixel in  $\mathcal{S} \cup \mathcal{U}$ , we perform color transfer between the  $\{G_{\mathcal{S}}^i, G_{\mathcal{U}}^i\}$  pair, and the overall result is the integration of all transfers, weighted by the weighted inverse color distance  $d_{\mathcal{S}}^i$ , which measures the Gaussian model affinity.

It remains to identify the shadowed pixels in  $\mathcal{S} \cup \mathcal{U}$  where  $T$  in Eqn. 10 is applied. Note that  $\mathcal{S} \cup \mathcal{U}$  contains both shadowed and nonshadowed pixels (see Fig. 4), and we need to determine in a probabilistic sense how much influence  $T$  should have on the pixel.

First, we compute the shadowed affinity map  $\wp_{\mathcal{S}}(x)$ , defined as:

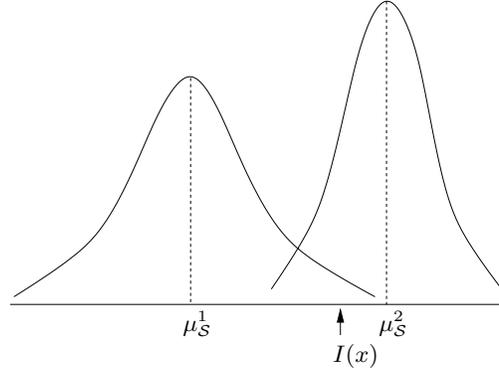


Fig. 8. Model affinity. Given an intensity value  $I(x)$  of a pixel  $x$ , model affinity measures how close  $I(x)$  to a given Gaussian model, which is defined by the weighted color distance  $d_S^i(x)$  where  $i = 1, 2$  here. The definition for  $d_S^i(x)$  is similar.

$$\wp_S(x) = \frac{1}{2} \sum_{i=1}^K (\pi_S^i + \pi_{\bar{S}}^i) \wp_S^i(x) \quad (13)$$

where  $\pi_S^i, \pi_{\bar{S}}^i$  are the respective weights of the Gaussian, with each  $\wp_S^i(x)$  encoding the shadow probability of  $x$  in an ad-hoc sense by a weighted color distance:

$$\wp_S^i(x) = \frac{d_S^i(x)}{d_S^i(x) + d_{\bar{S}}^i(x)} \quad (14)$$

where  $d_S^i(x) = 1/(|\frac{I(x) - \mu_S^i}{\sigma_S^i}| + \varepsilon)$  (defined earlier) and

$$d_{\bar{S}}^i(x) = \frac{1}{\left| \frac{I(x) - \mu_{\bar{S}}^i}{\sigma_{\bar{S}}^i} \right| + \varepsilon} \quad (15)$$

Note that we multiply the mean  $\frac{(\pi_S^i + \pi_{\bar{S}}^i)}{2}$  in Eqn. 13 instead of  $\pi_S^i$  to alleviate the bias of the weights of the Gaussian when using  $\mathcal{S}$  to calculate the affinity. This is because  $\pi_S^i$  and  $\pi_{\bar{S}}^i$  are supposed to be similar based on the assumption that the histogram of  $\mathcal{S}$  should be a scaled and shifted version of that of  $\bar{\mathcal{S}}$ .

Given the normalized shadowed affinity map  $\wp_S(x)$  (divided by a normalization factor) the overall color transfer for a pixel  $x$  where model affinity is taken into account is given by

$$T^*(I(x)) = I(x)(1 - \wp_S(x)) + T(I(x))\wp_S(x) \quad (16)$$

Some examples of  $T^*$  are shown in Fig. 9.

**Spatial affinity** In addition to the model affinity, we introduce a spatial affinity that improves our approximate shadowless image around the shadow boundary. In particular, the intensity  $I(x)$  of a pixel lying on  $\Omega = \partial(\mathcal{S} \cup \mathcal{U}) \setminus \mathcal{E}$  should remain unchanged after the transfer, and the effect of the transfer should gradually increase towards the area in shadow, that is, when the distance to  $\Omega$  increases.

In our implementation, we compute an image  $\mathbf{D}$  (Figs 10 and 11) where the pixel intensity inside  $\mathcal{S} \cup \mathcal{U}$  is equal to the intensity of the nearest pixel along  $\Omega$ . Note that all the pixel intensities in  $\mathbf{D}$  are directly derived from  $\Omega$  in order to



Fig. 9. An example  $T^*$ . Notice the band of bright pixels enclosing the shadow, which corresponds to the pixel regions where the effect of color transfer should be minimized.

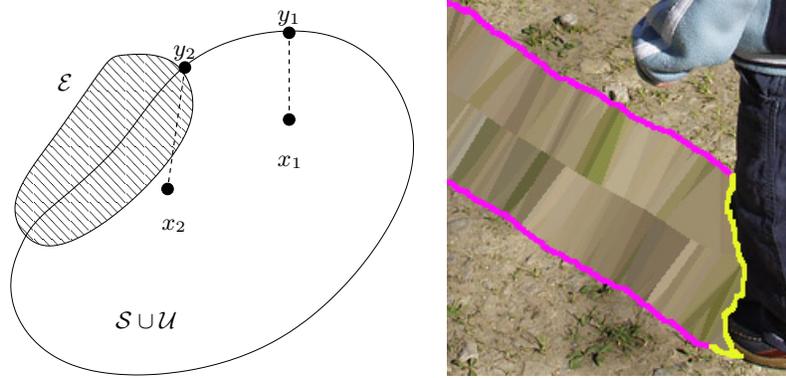


Fig. 10. Spatial affinity is encoded by the image  $\mathbf{D}$ . Here,  $\mathbf{D}(x_1) = I(y_1)$  and  $\mathbf{D}(x_2) = I(y_2)$  where  $y_1$  and  $y_2$  are the nearest pixels of  $x_1$  and  $x_2$  along  $\Omega$ , respectively. A real image of  $\mathbf{D}$  (*Cat*) is also shown. [CHANGE] Notice that the yellow portion of the boundary is automatically refined so that the color statistics of the shadow-casting object are excluded from shadow calculation.

maintain the boundary which should be unchanged (so if  $x$  lies on  $\Omega$  it will not be processed). We use a linear distance function to model the gradually increasing effect of the color transfer with increasing distance from  $\Omega$ : let  $D(x)$  be the distance from pixel  $x$  to its nearest pixel on  $\Omega$ , and  $D_{max}$  be the maximum  $D(x)$  among all  $x \in (S \cup U)$ . Our linear weight function is  $\frac{D(x)}{D_{max}} \in [0, 1]$ .

Finally, we define the coarse approximation to  $\mathcal{F}$ ,  $\hat{\mathcal{F}}(x)$ , which takes into account the spatial affinity of  $x$  to pixels on  $\Omega$  (see Fig. 6 for examples of  $\hat{\mathcal{F}}$ ):

$$\hat{\mathcal{F}}(x) = (1 - \frac{D(x)}{D_{max}})\mathbf{D}(x) + (\frac{D(x)}{D_{max}})T^*(I(x)) \tag{17}$$

Fig. 12 shows the relative importance of color transfer  $T^*$  and  $\mathbf{D}$  in defining  $\hat{\mathcal{F}}$ , where we set  $\hat{\mathcal{F}} = \mathbf{D}(x)$ . As shown in the figure, the results are sensitive to the strip-like structure inherent in  $\mathbf{D}(x)$ .

To conclude this section, we define the  $E_{color}$  term to be minimized by the residual equation:



Fig. 11. An example  $\mathbf{D}$ . Note that  $\mathbf{D}$  is a very coarse image with a lot of strip-like structures, indicating that making  $\mathbf{D}$  as  $\hat{\mathcal{F}}$  is not appropriate.



Fig. 12. Shadow removal result produced with  $\frac{D(x)}{D_{max}} = 0$ , that is, when  $T^*(I(x))$  does not produce any effect. These results show that  $T^*(I(x))$  is an important component for defining  $\hat{\mathcal{F}}$ . The  $\lambda$ 's used in generating this result are the same as for Fig. 1

$$E_{color}(\beta) = \sum_{x \in \mathcal{U} \cup \mathcal{S}} [I(x) - \beta(x)\hat{\mathcal{F}}(x)]^2 \quad (18)$$

3.3.2 *Smoothness energy ( $E_{smooth}$ )*. We observe that the gradient of  $\beta$  is small in general, except along the shadow's boundary. This implies that a good estimation for  $\beta$  should be one that is smooth inside umbra and nonshadowed regions. Recall that the spatial affinity reduces artifacts due to color transfer in the shadowed region, but not the smoothness of the extracted shadow, which is controlled by the smoothness energy.

The energy function  $E_{smooth}^\beta$  can be defined as follows:

$$E_{smooth}^\beta(\beta) = \lambda_s \sum_{\{x,y\} \in \mathcal{N}} (\beta(x) - \beta(y))^2 k(x,y) \quad (19)$$

where  $\lambda_s$  is the weight of  $E_{smooth}$  in  $E$ , and  $\mathcal{N}$  is the set of adjacent pixel pairs in  $x$ 's neighborhood in  $\mathcal{S} \cup \mathcal{U}$ . We define  $k(x,y) = \exp - \left( \frac{(\varphi_{\mathcal{S}}(x) - \varphi_{\mathcal{S}}(y))^2}{2\sigma_p^2} \right)$ , where  $\sigma_p$  controls the smoothness imposed by  $\varphi_{\mathcal{S}}$ , and  $k(x,y)$  is included in

Eqn. 19 to weaken the effect of  $\beta$  smoothness along the shadow boundary. We set  $\sigma_p^2 = 0.0005$  for all our experiments.

The above works well unless a very hard shadow boundary is present. To recover a better  $\mathcal{F}$ , the hard shadow boundary should be localized, along which more smoothness should be imposed. Noting that  $\mathcal{F}$  can be expressed in terms of  $\beta$ , to achieve such localized smoothing, we make use of the following energy term  $E_{smooth}^\Omega$ :

$$E_{smooth}^\Omega(\beta) = \lambda_b \sum_{\{x,y\} \in \mathcal{N}} \wp_S(x)\wp_{\overline{S}}(y)(\beta(y)I(x) - \beta(x)I(y))^2 \quad (20)$$

where  $\lambda_b$  is the weight of  $E_{smooth}^\Omega$  in  $E$ .  $\wp_{\overline{S}}$ , the opposite of  $\wp_S$ , is defined analogously.

The anisotropic smoothing effect imposed by Eqn. 20 operates by identifying large  $\wp_S(x)$  and large  $\wp_{\overline{S}}(y)$ , which together indicates hard shadow boundary where more smoothness should be imposed in recovering  $\mathcal{F}$ . At the pixels identified as close to the hard boundary, we minimize  $(\mathcal{F}(x) - \mathcal{F}(y))^2$ , or equivalently  $(\frac{\beta(y)I(x) - \beta(x)I(y)}{\beta(x)\beta(y)})^2$ . For simplicity, in our implementation, we minimize the numerator only (Eqn 20). We acknowledge that it is a severe approximation that assumes that the denominator remains roughly constant, but it has worked well for removing hard shadows in our examples. More formal study is needed to fully understand the effect of this simplification and is left to future work.

The final smoothness term becomes:

$$E_{smooth}(\beta) = E_{smooth}^\beta(\beta) + E_{smooth}^\Omega(\beta) \quad (21)$$

**3.3.3 Effect of energy parameters.** There is only one weight and one binary parameter in our energy formulation above that is crucial in controlling the energy function  $E$ :  $\lambda_s$  and  $\lambda_b$  respectively. The weight  $\lambda_b$  is set only when a very hard shadow is present. In most cases it is simply ignored by setting  $\lambda_b = 0$ .

The parameter  $\lambda_s$  explicitly controls the smoothness of  $\beta$ , and therefore *implicitly* controls the smoothness of  $\mathcal{F}$ : Using our image compositing model  $I = \beta\mathcal{F}$ , the smoother  $\beta$  is, the larger the amount of textures  $\mathcal{F}$  can be maintained. When both  $\beta$  and  $\mathcal{F}$  are smooth,  $E$  will be minimized gradually. Fig. 13 shows the effects of a spectrum of varying  $\lambda_s$  on  $\mathcal{F}$ , where various amount of shading is maintained in the recovered  $\mathcal{F}$ . The corresponding shadow  $\beta$  extracted are shown.

As shown in Table II, the parameter  $\lambda_s$  is often set to 10. This means that  $E_{smooth}$  is very important in general. However, it does not imply that  $E_{color}$  produces no effect at all. Fig. 14 shows two results on shadow removal produced by removing the  $E_{color}$  term from the optimization, which practically produces no effect as a result. From the optimization point of view, this can be explained by the fact that  $E_{color}$  is the data term: without the data term, there is no observation or information for incomplete data inference.

**3.3.4 Optimization.** To optimize the function  $E$ , we perform energy minimization on all pixels. The method described here minimizes the error along each axis (or  $\beta(x)$ ) sequentially by traveling down the error surface with maximum step size. We initialize  $\beta = 0$ , noting that any initialization produces the same optimization result because our optimization function is convex. In fact, it is quadratic in the unknown and any linear least square solver can be used such as conjugate gradient. For simplicity, we have implemented the simple iterative Gauss-Seidel algorithm. In each iteration, for each pixel  $x$ , we fix the values of all other pixels and apply  $\frac{\partial E}{\partial \beta} = 0$  to solve for  $\beta(x)$ . We only perform optimization for pixels inside  $(\mathcal{S} \cup \mathcal{U}) \setminus \mathcal{E}$ .

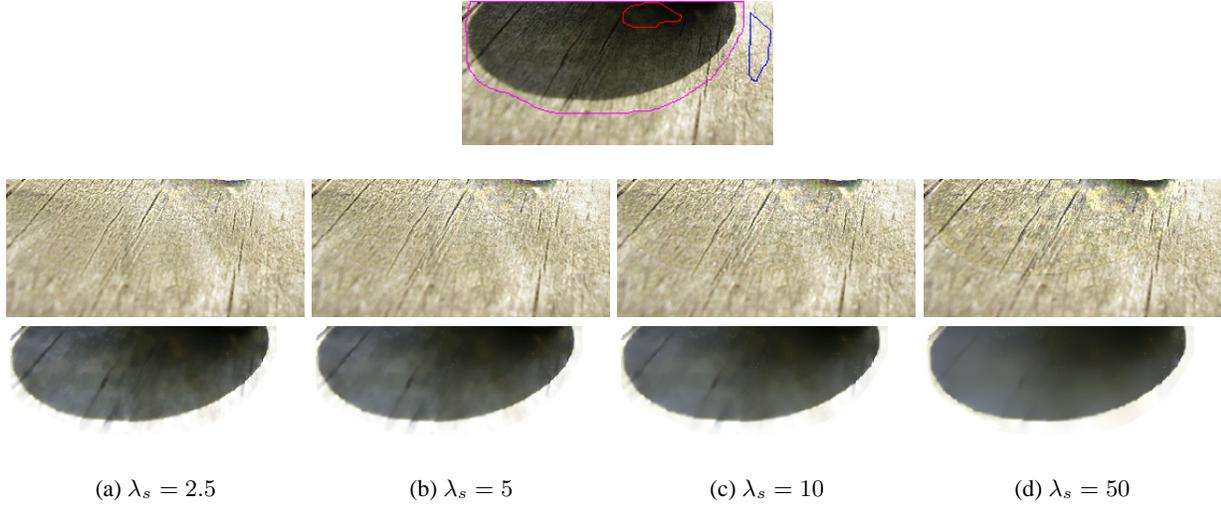


Fig. 13. *Stone*. The top is input image with the quadmap marking the regions of interest. By adjusting  $\lambda_s$  from the smallest in (a) to the largest in (d), we can retain different amount of texture in the recovered  $\mathcal{F}$ . The corresponding extracted shadows are shown.



Fig. 14. A shadowless image produced by removing the  $E_{color}$  term. Notice that the optimization does not produce any observable effect.

Setting  $\frac{\partial E}{\partial \beta} = 0$  produces:

$$\beta(x) = \frac{A_1(x) + A_3(x) + A_5(x)}{A_2(x) + A_4(x) + A_6(x)} \quad (22)$$

$$\begin{aligned} A_1(x) &= I(x)\hat{\mathcal{F}}(x) \\ A_2(x) &= \hat{\mathcal{F}}(x)^2 \\ A_3(x) &= \lambda_s \sum_{y \in \mathcal{V}} \beta(y)k(x, y) \end{aligned}$$

$$A_4(x) = \lambda_s \sum_{y \in \mathcal{V}} k(x, y)$$

$$A_5(x) = \lambda_b I(x) \varphi_{\mathcal{S}}(x) \sum_{y \in \mathcal{V}} \varphi_{\overline{\mathcal{S}}}(y) \beta(y)$$

$$A_6(x) = \lambda_b \varphi_{\mathcal{S}}(x) \sum_{y \in \mathcal{V}} \varphi_{\overline{\mathcal{S}}}(y) I(y)$$

where  $\mathcal{V}$  is the first-order neighborhood (top, bottom, left and right neighbors) of  $x$ . When the image region  $(\mathcal{S} \cup \mathcal{U}) \setminus \mathcal{E}$  enters a stationary state, the optimization ends.

## 4. RESULTS AND DISCUSSION

We first demonstrate our main results on natural shadow matting and compositing. We then compare our recovered shadowless images to the results obtained from the best existing shadow removal algorithms that use only a single image input. Finally, we conclude this section by discussing the limitations of our method and future work.

### 4.1 Experimental results

We have used our system to separate complex shadows from a single input image for shadow matting and compositing. All results generated in this section were automatically optimized after the user had supplied a simple quadmap. Some of the separated shadows were composited with their corresponding foregrounds onto novel background images to create special effects, where the extracted shadow  $\beta$  are sometimes blurred (indicated by  $\beta'$  in the figures). We first tabulate the running times for all examples in this paper in Table II. A Pentium-IV PC 3.2 GHz with 512 M RAM was used to perform the experiments. In most cases, it took less than 15 secs for user marking. Note that the running time also depends on the size of the shadow.

Example	Image resolution	$\lambda_s$	$\lambda_b$	Running time (secs)
<i>Puzzled Boy</i>	755 x 563	10	0	65
<i>Stone</i>	279 x 129	10	1	25
<i>Snowy Slope</i>	256 x 384	10	0	15
<i>Fairy</i>	384 x 256	0.001	1	4
<i>Running Boy</i>	800 x 511	10	1	50
<i>Tree</i>	531 x 325	10	0	29
<i>Flower</i>	598 x 507	50	0	50
<i>Lamp</i>	480 x 397	10	0	44
<i>Bird</i>	600 x 485	10	0	32
<i>Wall</i>	154 x 147	150	0	10
<i>Texture</i>	344 x 164	10	0	22
<i>Elephants</i>	384 x 256	1	0	4
<i>Two balls on pebbles</i>	128 x 89	10	0	9
<i>Plank (man only)</i>	340 x 436	10	1	14
<i>Lawn</i>	153 x 154	10	0	12

Table II. Running times. Typically, a user can achieve the results in this paper by adjusting  $\lambda$ 's in less than 3 times.

**$\beta$  and  $\mathcal{F}$  separation for shadow matting and compositing** In Fig. 15 we first show the comparison with an available shadowless image  $\mathcal{F}$ , which is used as the ground truth. The input shown in the figure is obtained by compositing an available shadow matte (extracted from another scene using our method) onto the ground truth  $\mathcal{F}$ . Note that the shadow matte  $\beta$  extracted from this input is very smooth. Textures in the estimated shadowless image  $\mathcal{F}$  are

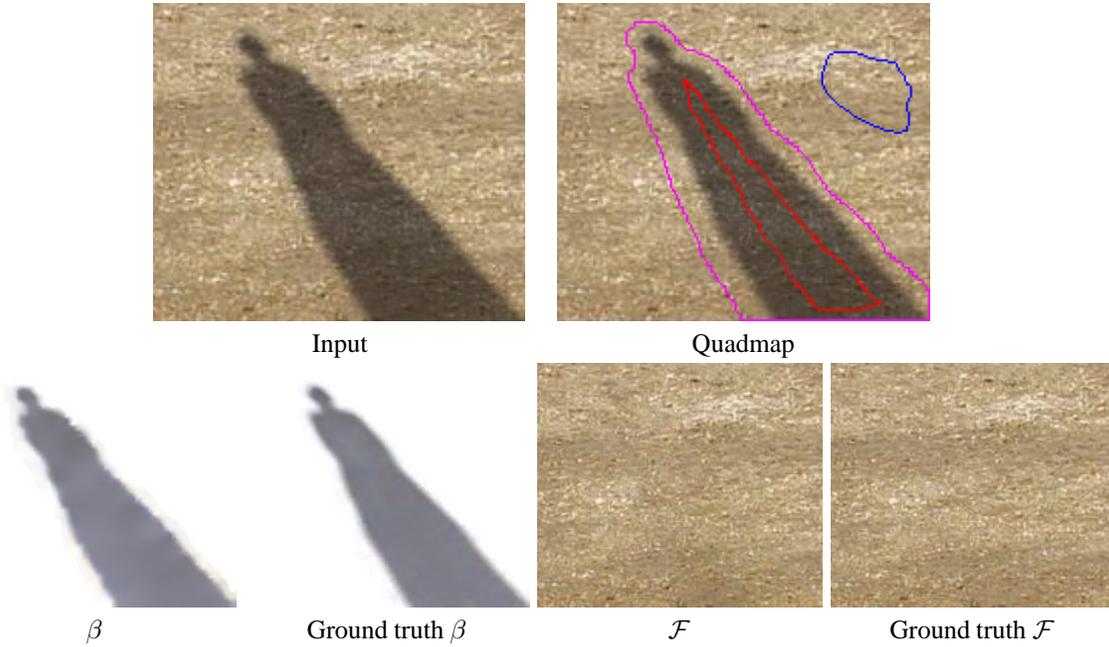


Fig. 15. Comparison with an available ground truth. The extracted natural shadow (for shadow matting) and the recovered shadowless image (for shadow removal) are visually acceptable. Note that the extracted shadow is smooth and no texture remains.

maintained. Figs 16 to 18 show the result of separating  $\beta$  and  $\mathcal{F}$  for shadow matting and compositing. In *Snowy Slope*, we can remove large shadows while maintaining textures and patterns of the snow under the removed shadow. Note that the affected area is large and cannot be faithfully recovered by non-parametric texture synthesis techniques, because no samples are present to match the long and oriented snow lines. Note that our extracted  $\beta$  matte can be composited seamlessly with a new background, as shown in our final composite. Another special visual effect is shown in the *Running Boy*, where the boy and his shadow are composited on the clouds image.

The shadow matte extracted from the *Fairy* shows both the advantages and some limitations of our method. While we can easily extract the complex and transparent shadow using a few clicks, two small spots corresponding to the leaves that should have been left on  $\mathcal{F}$  are extracted in the  $\beta$  matte. These small artifacts are due to the lack of color samples, because the user-supplied  $\bar{\mathcal{S}}$  only provides samples for the shadowless sand but not the leaves. Also, in this example, very ambiguous color statistics between the shadowed and nonshadowed regions are caused by the similarity of the color between the sand and brown leaves.

**Complex shadow extraction from noisy images** Since our image model  $I = \beta\mathcal{F}$  does not incorporate any consideration of image noise, the results generated by our shadow removal algorithm may suffer from undesired noise. Specifically, a satisfactory  $\beta$  and  $\mathcal{F}$  may not be obtained at the same time.

To deal with such situation, we can adjust the smoothness term  $\lambda_s$  as follows. If the goal is to extract a shadow matte for compositing, more smoothness on the shadow matte  $\beta$  can be enforced by using a large  $\lambda_s$ , thus sacrificing the  $\mathcal{F}$  by sending most image noise to the shadowless image.

In Figs 19–22, we show complex shadows we extracted from noisy images for shadow compositing and so a good

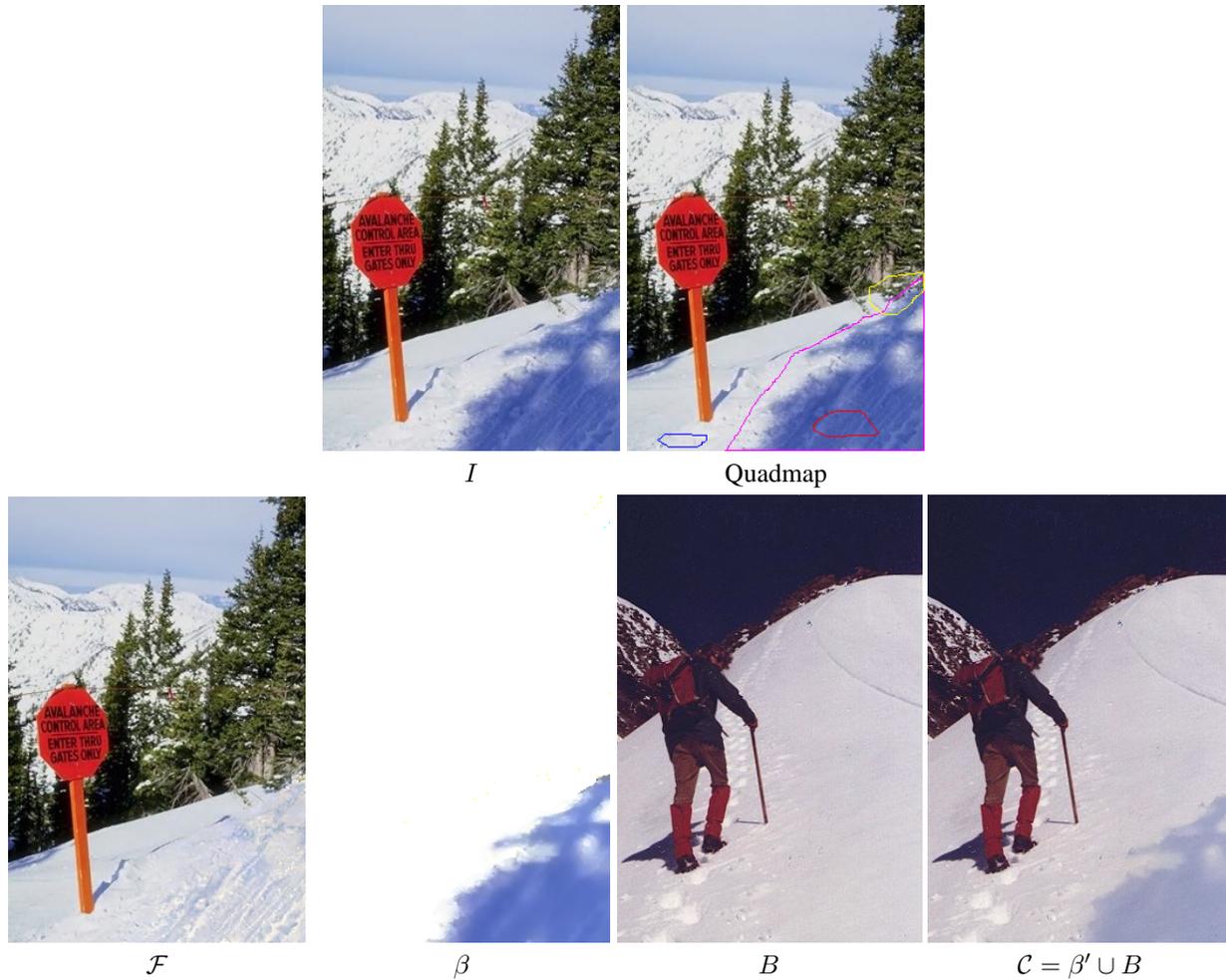


Fig. 16.  $\beta$  and  $\mathcal{F}$  separation for *Snowy Slope*:  $\mathcal{C}$  is the image composited by blending the extracted  $\beta$  matte with a new background image  $B$ .  $\beta'$  is obtained by adjusting the intensity of the extracted  $\beta$  to make the shadow look natural in  $\mathcal{C}$ .

shadow matte is preferred. In *Tree*, the shadow has a great deal of fine structures. Note that the black spots on the shadow matte are not artifacts; rather they are part of the shadow arising from the small rocks under the tree. In *Flower*, notice that the color statistics on the gravel road is spatially varying. The complex soft shadow is faithfully extracted by our method. In *Lamp*, the hard shadow extracted is caused by a complex and partially transparent object. Note the different shadow responses for different colors. Notice that the line remains in this shadow matte result (shown in (d)), which is due to image noise and our optimization which preserves only texture gradient, but not structures such as a salient line in the recovered shadowless image. In *Bird*, the extracted shadow is soft, which is caused by motion blur. Notice the use of Bayesian matting produces neither a good shadow matte nor a good shadowless image while the trimap specification is more complicated than that of a quadmap.

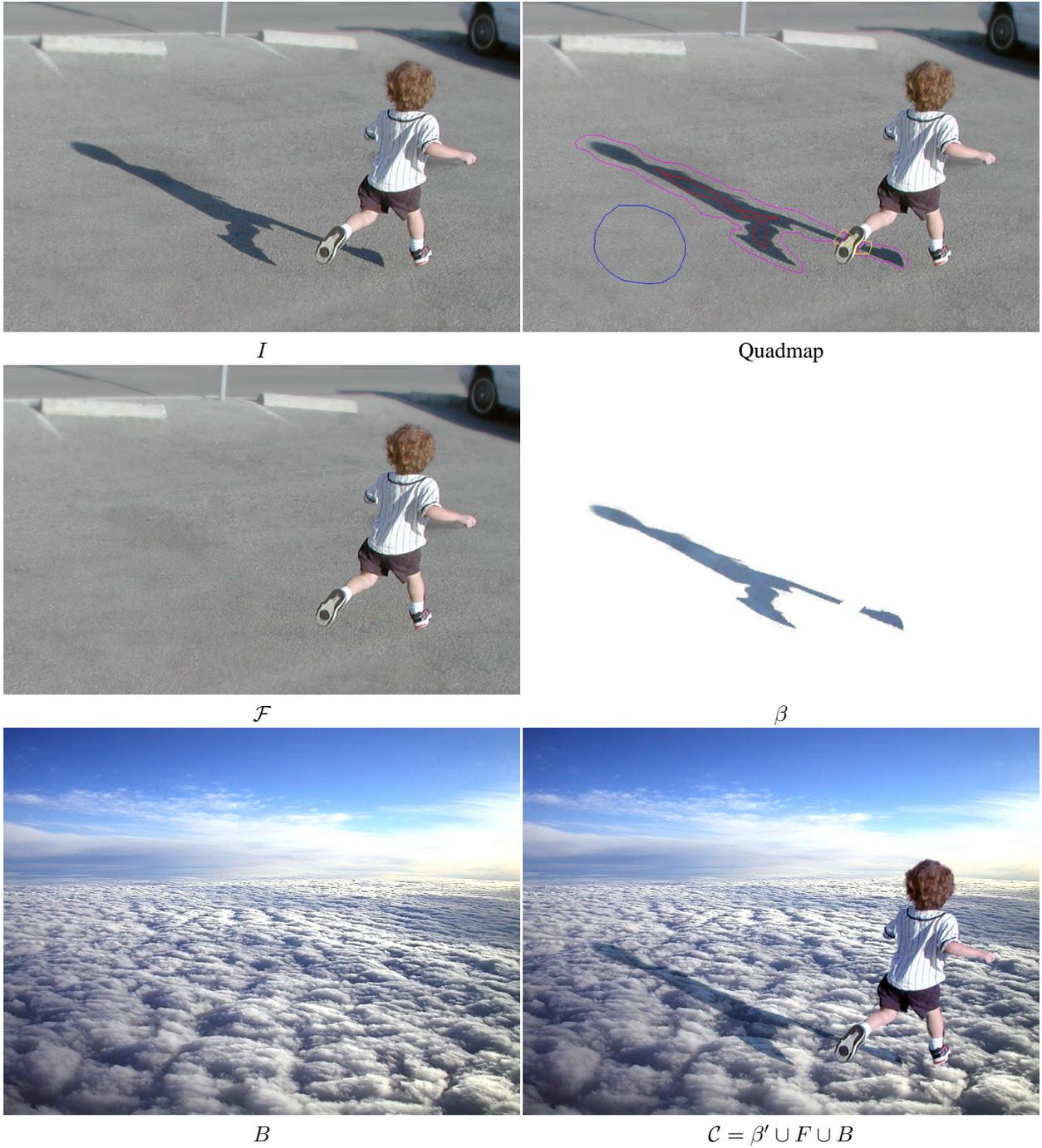


Fig. 17.  $\beta$  and  $\mathcal{F}$  separation for *Running Boy*:  $\mathcal{C}$  is the image composited by blending the extracted  $\beta$  matte with a new background image  $B$  and the foreground matte  $F$ .  $\beta'$  is obtained by adjusting the intensity of the extracted  $\beta$  to make the shadow look natural in  $\mathcal{C}$ .



Fig. 18.  $\beta$  and  $\mathcal{F}$  separation for and *Fairy*.  $C$  is the image composited by blending the extracted  $\beta$  matte with a new background image  $B$ . The reflection on the water is the produced by compositing the mirror image of the foreground  $F$ .  $\beta'$  is obtained by adjusting the intensity of the extracted  $\beta$  to make the shadow look natural in  $C$ .

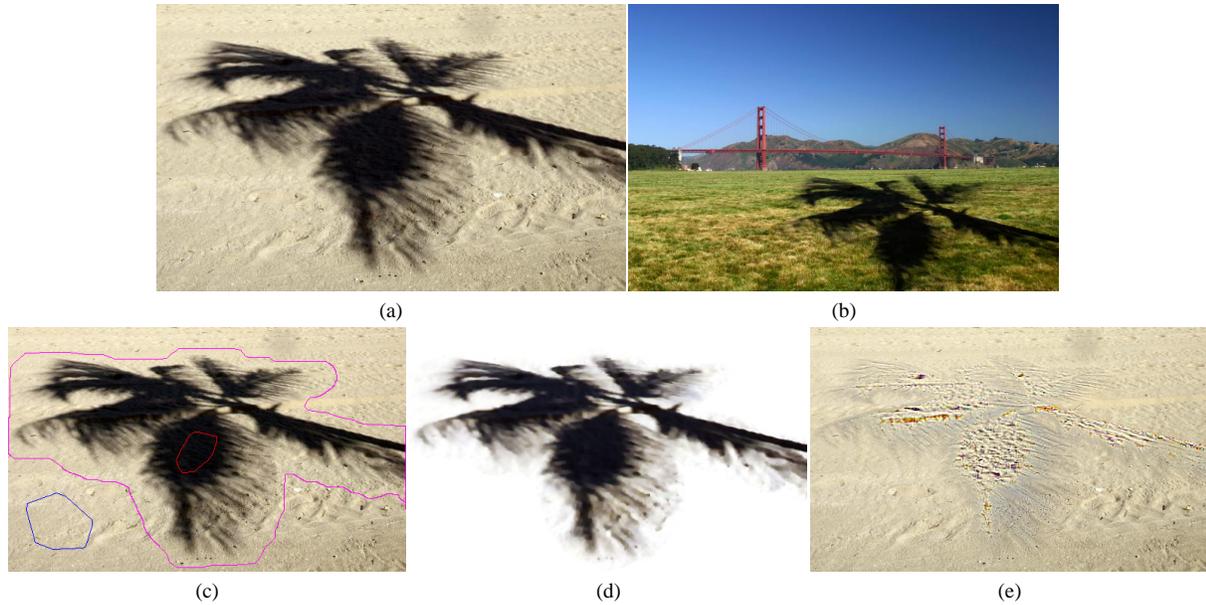


Fig. 19. *Tree*: Extracting a shadow matte from a noisy image and compositing it on a new background. Although the extracted shadow in (d) looks good, the shadowless image in (e) does not. (a) Input. (b) image composite with shadow intensity adjusted, (c) quadmap, (d) the extracted  $\beta$  and (e) the extracted  $\mathcal{F}$  using our method.

## 4.2 Comparison and limitations

**Preservation of textures** Fig. 23 shows that our method can preserve multiple textures after shadow removal. The figure also shows that we can remove the shadows and reflections of the elephants while maintaining the ripples on the water.

**Comparison with shadow removal** In Figs 24, 25, and 26, we compare our results with those in [Finlayson et al. 2004], which assumes Planckian lighting (e.g., sunlight) and narrow-band camera [Finlayson et al. 2002]. Although their method is excellent for automatic shadow detection, the texture is blurred where the shadow was removed. In contrast, our method produces significantly better results using only a few user-marked strokes, and the rest of processing is fully automatic. Our technique allows us to remove complex shadows, preserve structures and textures.

**Comparison with image completion and repairing** In situations where uncorrectable image noise dominates inherent color statistics, the shadow should be treated as a hole, and image inpainting, completion or repairing [Bertalmio et al. 2000; Drori et al. 2003; Jia and Tang 2004] should be applied to infer missing pixel colors. This situation is demonstrated here by removing the shadow cast by a lamp of the left side in Fig. 25, where the shadowed region is quite noisy. The small shadow is repaired. In other situations where the region under a shadow is too large to be handled by the above methods, or there simply does not exist any samples for non-parametric texture synthesis [Efros and Leung 1999] to properly operate, our method is an effective alternative to remove or extract this kind of shadow. Except for the lamp shadow, all the complex shadows in Fig. 25 were removed by our method.

**Comparison with pure color transfer** With the consideration of both texture gradient and shadow smoothness, our color transfer constrained by model and spatial affinities maintains color and texture consistency after shadow extraction. Fig. 27 compares our result with that generated by the color transfer model in [Reinhard et al. 2001], demonstrating that our color transfer that considers model and spatial affinities performs better than pure color transfer

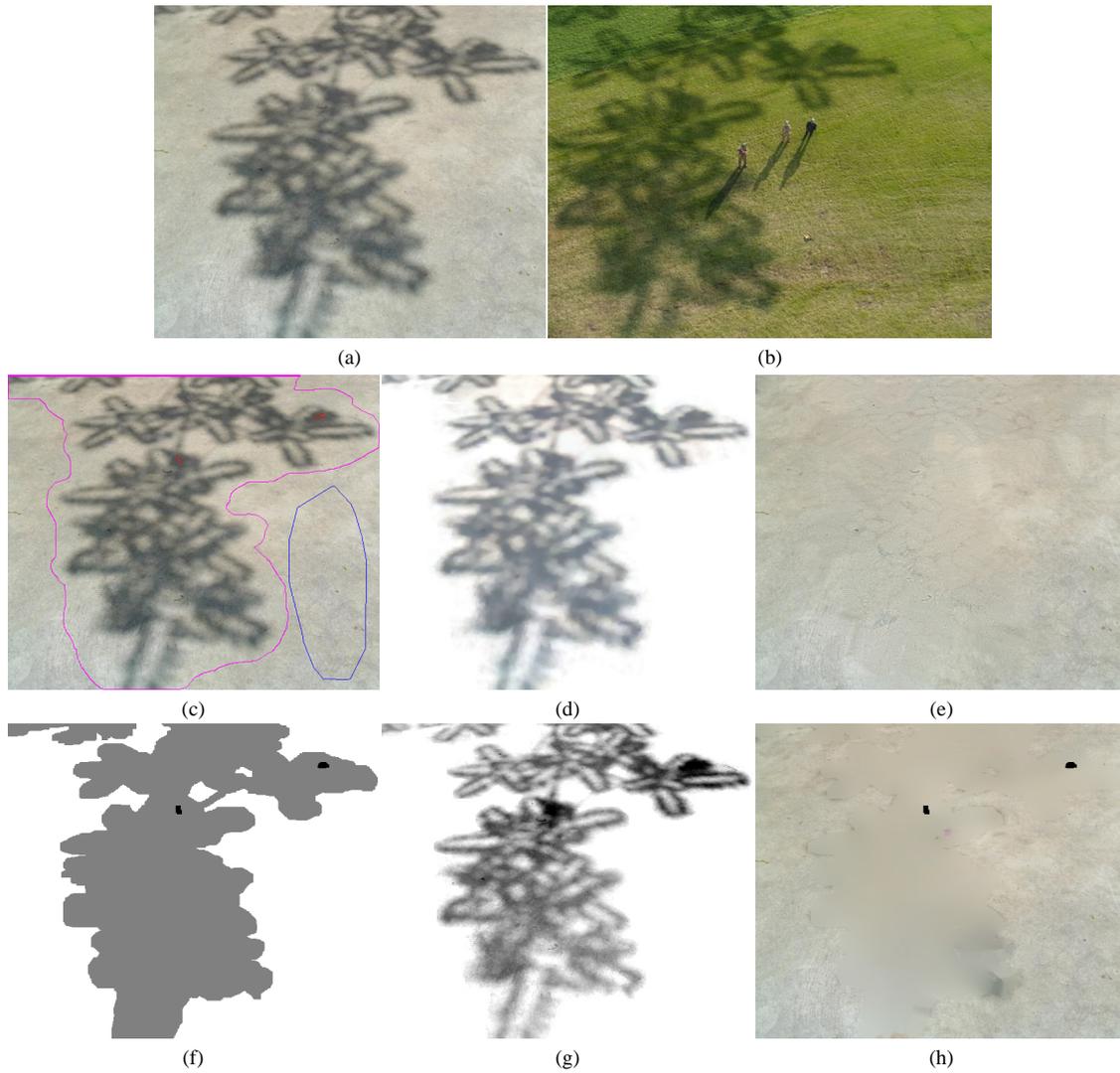


Fig. 20. *Flower*. Extracting a shadow matte from a noisy image and compositing it on a new background. Although the extracted shadow in (d) looks good, the shadowless image in (e) does not. More smoothness on the shadow matte  $\beta$  is enforced during the extraction to obtain a good shadow for matting and compositing as shown here, at the expense of sending most image noise to the shadowless image  $\mathcal{F}$  during the optimization. Notice that the natural matting technique such as Bayesian matting cannot produce good results while specifying a trimap is more difficult than a quadmap. (a) Input, (b) image composite with shadow intensity adjusted, (c) quadmap, (d) the extracted  $\beta$  and (e) the extracted  $\mathcal{F}$  using our method. (f) Trimap, (g) the extracted alpha and (h) the image after extracting the shadow using Bayesian matting.



Fig. 21. *Lamp*. Extracting a shadow matte from a noisy image and compositing it on a new background. Although the extracted shadow in (d) looks good, the shadowless image in (e) does not. Notice that the natural matting technique such as Bayesian matting cannot produce good results while specifying a trimap is more difficult than a quadmap. (a) Input, (b) image composite with shadow intensity adjusted, (c) quadmap, (d) the extracted  $\beta$  and (e) the extracted  $\mathcal{F}$  using our method. (f) Trimap, (g) the extracted alpha and (h) the image after extracting the shadow using Bayesian matting.

in shadow matte extraction.

**Limitations of our approach** By definition, our shadow compositing equation cannot distinguish shading and shadows. The “distribution” of the shading effect between  $\beta$  and  $\mathcal{F}$  can be controlled to a certain extent. For instance, increasing  $\lambda_s$  if the shadow is known to be smooth, thus distributing more shading effect to the estimated  $\mathcal{F}$ .

While our method preserves highly textured regions and some structures well, highly structured objects are still problematic if the shadowed region is too dark. In such cases, more user interaction is necessary. For instance, the image can be divided into several small regions and shadow matting is done on each region individually.

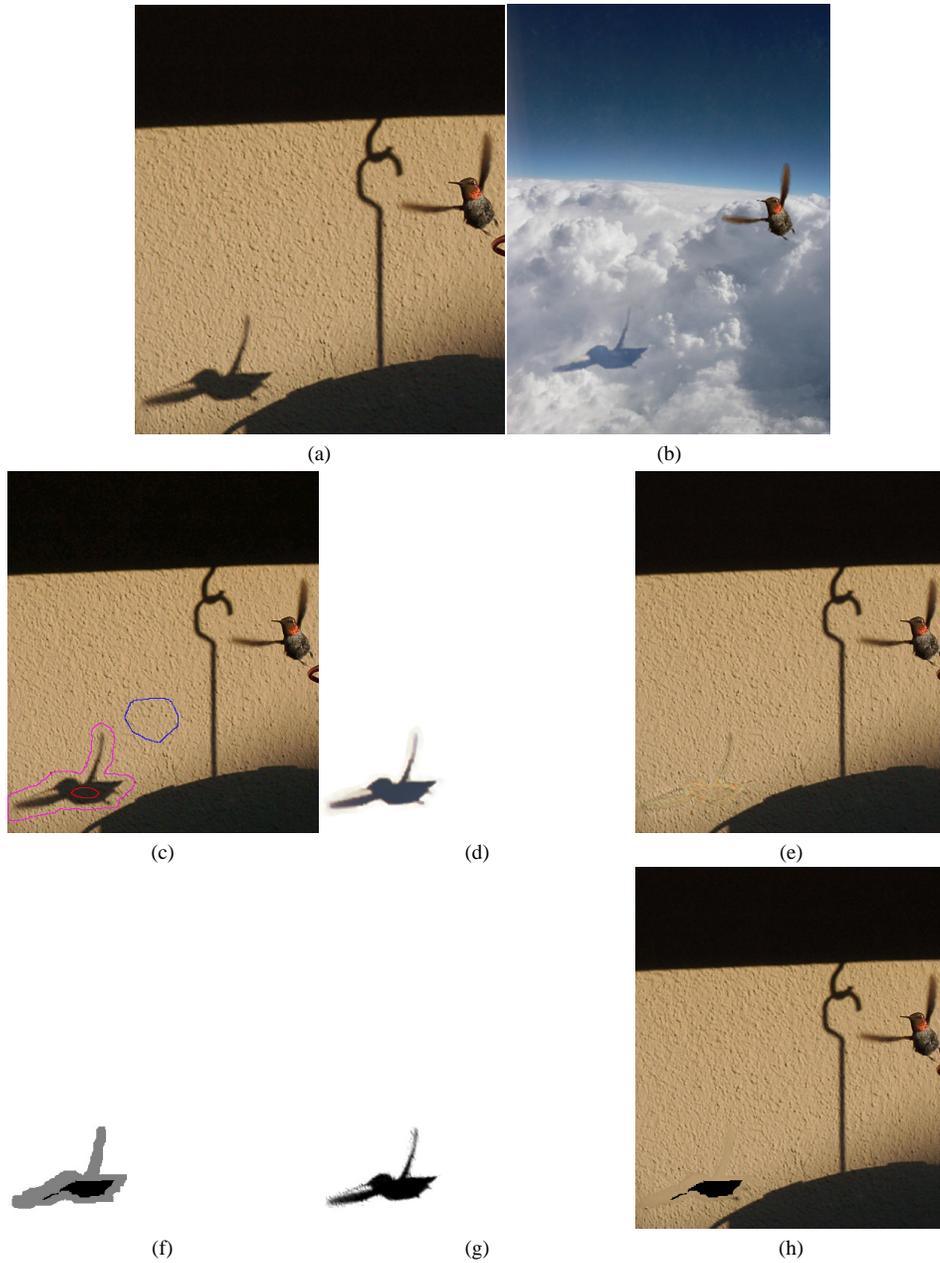


Fig. 22. *Bird*. Extracting a shadow matte from a noisy image and compositing it on a new background. Although the extracted shadow in (d) looks good, the shadowless image in (e) does not. Notice that the natural matting technique such as Bayesian matting cannot produce good results while specifying a trimap is more difficult than a quadmap. (a) Input, (b) image composite with shadow intensity adjusted, (c) quadmap, (d) the extracted  $\beta$  and (e) the extracted  $\mathcal{F}$  using our method. (f) Trimap, (g) the extracted alpha and (h) the image after extracting the shadow using Bayesian matting.

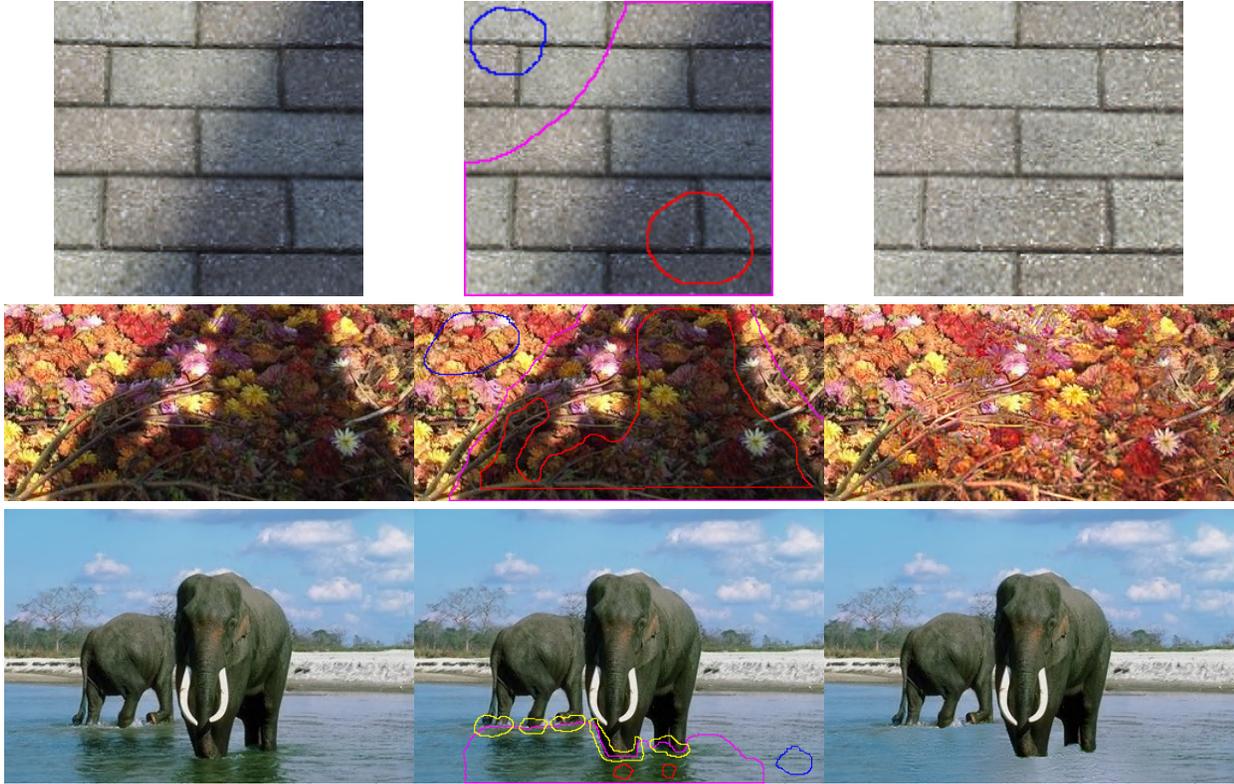


Fig. 23. Shadow removal on textured objects. Top: *Wall*, soft shadow on a brick wall. Middle: *Texture*, hard shadows on flowers. Bottom: *Elephants*, soft shadows and reflections on the rippling water.

We require that the user coarsely specify the  $\bar{\mathcal{S}}$  and  $\mathcal{S}$  with similar textures or patterns but with dissimilar intensities. It is inevitable that user's choice is sometimes inexact, or  $\bar{\mathcal{S}}$  is confused by similar shadow and nonshadow colors. This will result in an incorrect estimation of  $\varphi_{\mathcal{S}}(x)$ , which is based on GMM on color statistics. In this case, we suggest the user to edit  $\varphi_{\mathcal{S}}$  using one of the available brushes, such as eraser and clone brushes. Intelligent scissor [Mortensen and Barrett 1995], Grabcut [Rother et al. 2004], and Lazy Snapping [Li et al. 2004] may be used to segment the foreground object before applying our shadow matting, if the object's color is very similar to its cast shadow. In practice we note this is typically not necessary.

### 4.3 Application in image and shadow compositing

In *Puzzled Boy* (see Fig. 1), the big boy is puzzled, but the scene looks surprisingly realistic. Note the tundra in the input is under a very dark shadow, which can be seamlessly recovered in  $\mathcal{F}$  by our method. The special effect would have looked unrealistic if the shadows were left out in the final image composite. The shadow in the composite shown here should be warped according to the geometry and lighting captured by the target image, which has already been addressed in [Chuang et al. 2003].

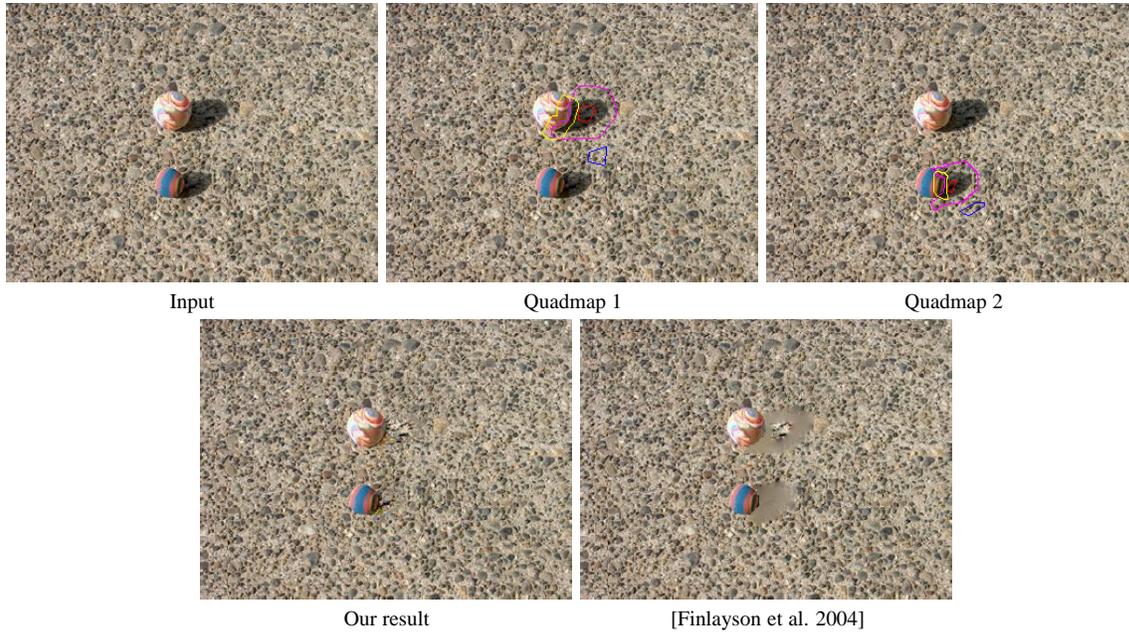


Fig. 24. *Two balls on Pebbles*: Comparison.

## 5. CONCLUSION

By proposing a new shadow equation, we presented a novel approach to address the problem of natural shadow removal and matting from a single image. Our user-assisted approach consists of two steps. First, the user performs a mark up to create a *quadmap* with four regions: *definitely shadowed*, *definitely nonshadowed*, *uncertain*, and *excluded* regions. Second, an automatic optimization step based on energy minimization is used to extract the complex shadows while maintaining texture gradient and shadow smoothness. By exploiting the statistics provided by the quadmap to achieve color transfer between the shadowed and nonshadowed region, the shadow matte and shadowless image are obtained. More importantly, our approach is capable of removing a shadow from an input image based on the new shadow equation. We have compared our method with state-of-the-art shadow removal methods and other related techniques, and found that our method produces some of the best visual results to date. Our technique has been applied to remove unwanted shadows and extract shadows for compositing. Future work consists of incorporating useful constraint (such as the bluish-white-yellowish colors of a shadow) on shadow matte extraction, addressing the current limitations described in the previous section, and extending this approach to process video data.



Fig. 25. *Plank*: Comparison. Here, we removed three dominant shadows: The shadows for the man and the plant are removed by applying the natural shadow matting. Due to the large amount of image noise, the background under the shadow of the lamp (circled region) is repaired by image repairing [Jia and Tang 2004] (see text for discussion).

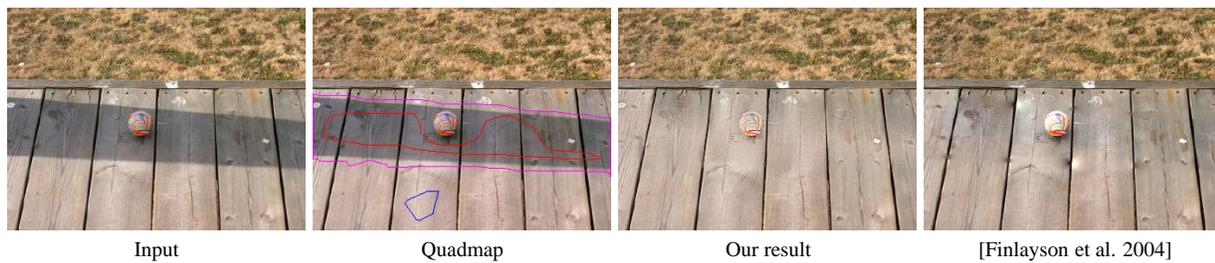


Fig. 26. *A ball on Planks*: Comparison. input. (b) Automatic result from [Finlayson et al. 2004]. (c) Note that there is no shadowless sample on the ball. The shadowless image for the planks are seamlessly recovered where the subtle textures on the planks are maintained.

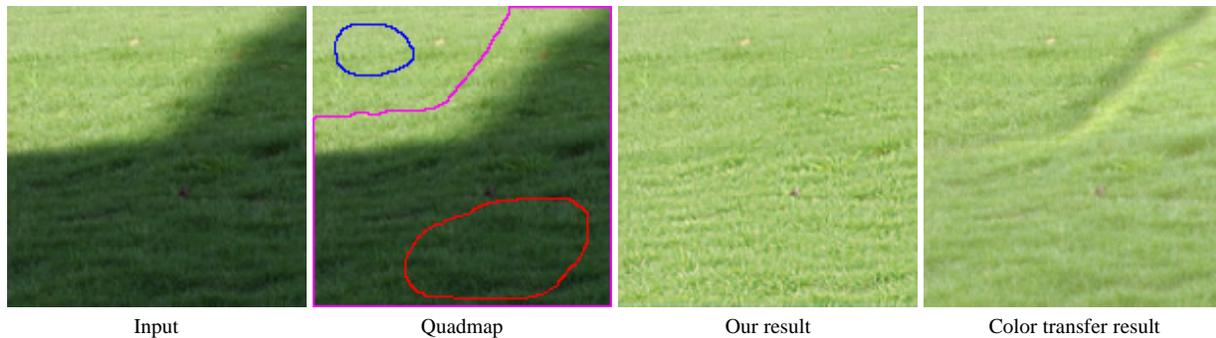


Fig. 27. The *Lawn* example: pure color transfer for shadow removal may not preserve textures and result in over-smooth texture. Our approach considers both texture gradient and smoothness. (a) Input. (b) color transfer result. (c) our result.

## Acknowledgment

This research was supported in part by the Hong Kong Special Administrative Region under AoE/E-01/99 and HKUST6175/04E. Tai-Pang Wu and Chi-Keung Tang would like to thank Microsoft Research Asia, for the timely and fruitful discussions as well as the computing resources during the preparation for the SIGGRAPH 2005 submission. The authors would like to thank the Associate Editor and all anonymous SIGGRAPH reviewers for their constructive comments and recommendation of this paper to the ACM Transactions on Graphics.

## REFERENCES

- BABA, M., MUKUNOKI, M., AND ASADA, N. 2004. Shadow removal from a real image based on shadow density. In *Proceedings of the SIGGRAPH'2004 Sketches and Applications*.
- BARROW, H. AND TENENBAUM, J. 1978. Recovering intrinsic scene characteristics from images. In *CVS78*. 3–26.
- BERTALMIO, M., SAPIRO, G., BALLESTER, C., AND CASELLES, V. 2000. Image inpainting. In *SIGGRAPH'2000*. 417–424.
- CHUANG, Y.-Y., CURLESS, B., SALESIN, D. H., AND SZELISKI, R. 2001. A bayesian approach to digital matting. *Proceedings of CVPR'01, Vol. II*, 264–271.
- CHUANG, Y.-Y., GOLDMAN, D. B., CURLESS, B., SALESIN, D. H., AND SZELISKI, R. 2003. Shadow matting and compositing. *ACM Trans. Graph.* 22, 3, 494–500.
- DRORI, I., COHEN-OR, D., AND YESHURUN, H. 2003. Fragment-based image completion. *ACM Trans Graph.* 22, 303–312.
- DUDA, R., HART, P., AND STORK, D. 2001. *Pattern Classification*. Wiley.
- EFROS, A. AND LEUNG, T. 1999. Texture synthesis by non-parametric sampling. In *ICCV99*. 1033–1038.
- FINLAYSON, G., DREW, M., AND LU, C. 2004. Intrinsic images by entropy minimization. In *ECCV04, Vol III*: 582–595.
- FINLAYSON, G. D., HORDLEY, S. D., AND DREW, M. S. 2002. Removing shadows from images. *Proceedings of ECCV'02, Vol. IV*, 823–836.
- JIA, J. AND TANG, C.-K. 2004. Inference of segmented color and texture description by tensor voting. *PAMI* 26, 6 (June), 771–786.
- LAND, E. AND MCCANN, J. 1971. Lightness and the retinex theory. *J. Opt. Soc. Am.*, 1–11.
- LI, Y., SUN, J., TANG, C.-K., AND SHUM, H.-Y. 2004. Lazy snapping. *ACM Trans. Graph.* 23, 3, 303–308.
- MITCHELL, T. 1997. *Machine Learning*. McGraw Hill.
- MORTENSEN, E. N. AND BARRETT, W. A. 1995. Intelligent scissors for image composition. In *SIGGRAPH '95*. 191–198.
- OH, B. M., CHEN, M., DORSEY, J., AND DURAND, F. 2001. Image-based modeling and photo editing. In *SIGGRAPH '01*. 433–442.
- PORTER, T. AND DUFF, T. 1984. Compositing digital images. *Proceedings of ACM SIGGRAPH '84*, 253–259.
- REINHARD, E., ASHIKHMIN, M., GOOCH, B., AND SHIRLEY, P. 2001. Color transfer between images. *IEEE CG&A* 21, 34–41.
- ROTHER, C., KOLMOGOROV, V., AND BLAKE, A. 2004. "grabcut": interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.* 23, 3, 309–314.
- SUN, J., JIA, J., TANG, C.-K., AND SHUM, H.-Y. 2004. Poisson matting. *ACM Trans. Graph.* 23, 3, 351–321.
- TAPPEN, M., FREEMAN, W., AND ADELSON, E. 2002. Recovering intrinsic images from a single image. In *MIT AIM*.

- WEISS, Y. 2001. Deriving intrinsic images from image sequences. In *ICCV01*. II: 68–75.
- WELSH, T., ASHIKHMEN, M., AND MUELLER, K. 2002. Transferring color to greyscale images. *ACM Trans. Graph.* 21, 277–280.
- WRIGHT, S. 2001. *Digital Compositing for Film and Video*. Focal Press.