Automatic Shadow Detection and Removal from a Single Image

S. H. Khan, M. Bennamoun, Member, IEEE, F. Sohel and R. Togneri, Senior Member, IEEE

Abstract—We present a framework to automatically detect and remove shadows in real world scenes from a single image. Previous works on shadow detection put a lot of effort in designing shadow variant and invariant hand-crafted features. In contrast, our framework automatically learns the most relevant features in a supervised manner using multiple convolutional deep neural networks (ConvNets). The features are learned at the super-pixel level and along the dominant boundaries in the image. The predicted posteriors based on the learned features are fed to a conditional random field model to generate smooth shadow masks. Using the detected shadow masks, we propose a Bayesian formulation to accurately extract shadow matte and subsequently remove shadows. The Bayesian formulation is based on a novel model which accurately models the shadow generation process in the umbra and penumbra regions. The model parameters are efficiently estimated using an iterative optimization procedure. Our proposed framework consistently performed better than the state-of-the-art on all major shadow databases collected under a variety of conditions.

Index Terms—Feature Learning; Bayesian shadow removal; Conditional Random Field; ConvNets; Shadow detection; Shadow matting

1 INTRODUCTION

Shadows are a frequently occurring natural phenomenon, whose detection and manipulation are important in many computer vision (e.g., visual scene understanding) and computer graphics applications. As early as the time of Da Vinci, the properties of shadows were well studied [1]. Recently, shadows have been used for tasks related to object shape [2, 3], size, movement [4], number of light sources and illumination conditions [5]. Shadows have a particular practical importance in augmented reality applications, where the illumination conditions in a scene can be used to seamlessly render virtual objects and their casted shadows. Contrary to the above mentioned assistive roles, shadows can also cause complications in many fundamental computer vision tasks. For instance, they can degrade the performance of object recognition, stereo, shape reconstruction, image segmentation and scene analysis. In digital photography, information about shadows and their removal can help to improve the visual quality of photographs. Shadows are also a serious concern for aerial imaging and object tracking in video sequences [6].

Despite the ambiguities generated by shadows, the Human Visual System (HVS) does not face any real difficulty in filtering out the degradations caused by shadows. We need to equip machines with such visual comprehension abilities. Inspired by the hierarchical architecture of the human visual cortex, many deep representation learning architectures have been proposed in the last decade. We draw our motivation from the recent successes of these deep learning methods in many computer vision tasks where learned features out-performed hand-crafted features [7]. On that basis, we propose to use multiple convolutional neural networks (ConvNets) to learn useful feature representations for the task of shadow detection. ConvNets are biologically inspired deep network architectures based on Hubel and Wiesel’s [8] work on the cat’s primary visual cortex. Once shadows are detected, an automatic shadow removal algorithm is proposed which encodes the detected information in the likelihood and prior terms of the proposed Bayesian formulation. Our formulation is based on a generalized shadow generation model which models both the umbra and penumbra regions. To the best of our knowledge, we are the first to use ‘learned features’ in the context of shadow detection, as opposed to the common carefully designed and hand-crafted features. Moreover, the proposed approach detects and removes shadows automatically without any human input (Fig. 1).

Our proposed shadow detection approach combines local information at image patches with the local information across boundaries (Fig. 1). Since the regions and the boundaries exhibit different types of features, we split the detection procedure into two respective portions. Separate ConvNets are consequently trained for patches extracted around the scene boundaries and the super-pixels. Predictions made by the ConvNets are local and we therefore need to exploit the higher level interactions between the neighboring pixels. For this
As an example, chromatic cues assume that the texture and color ratios across boundaries and the computation of texton features on both sides of the edges [11, 16]. Although these feature representations are useful, they are computed across image boundaries, such as intensity and color ratios across boundaries and the computation of texton features on both sides of the edges [11, 16]. Although these feature representations are useful, they are based on assumptions that may not hold true in all cases. As an example, chromatic cues assume that the texture of the image regions remains the same across shadow boundaries and only the illumination is different. This approach fails when the image regions under shadows are barely visible. Moreover, all of these methods involve a considerable effort in the design of hand-crafted features for shadow detection and feature selection (e.g., the use of ensemble learning methods to rank the best features [10, 11]). Our data-driven framework is different and unique: we propose to use deep feature learning methods to ‘learn the most relevant features’ for shadow detection.

Owing to the challenging nature of the shadow detection problem, many simplistic assumptions are commonly adopted. Previous works made assumptions related to the illumination sources [5], the geometry of the objects casting shadows and the material properties of the surfaces on which shadows are cast. For example, Salvador et al. [14] consider object cast shadows while Lalonde et al. [11] only detect shadows that lie on the ground. Some methods use synthetically generated training data to detect shadows [17]. Techniques targeted for video surveillance applications take advantage of multiple images [18] or time-lapse sequences [19, 20] to detect shadows. User assistance is also required by many proposed techniques to achieve their attained performances [21, 22]. In contrast, our shadow detection method makes absolutely ‘no prior assumptions’ about the scene, the shadow properties, the shape of objects, the image capturing conditions and the surrounding environments. Based on this premise, we tested our proposed framework on all of the publicly available databases for shadow detection from single images. These databases contain common real world scenes with artifacts such as noise, compression and color balancing effects.

## 2 Related Work and Contributions

### Shadow Detection

One of the most popular methods to detect shadows is to use a variety of shadow variant and invariant cues to capture the statistical and deterministic characteristics of shadows [10, 11, 12, 13, 14]. The extracted features model the chromatic, textural [10, 11, 13, 14] and illumination [12, 15] properties of shadows to determine the illumination conditions in the scene. Some works give more importance to features computed across image boundaries, such as intensity and color ratios across boundaries and the computation of texton features on both sides of the edges [11, 16]. Although these feature representations are useful, they are based on assumptions that may not hold true in all cases. As an example, chromatic cues assume that the texture of the image regions remains the same across shadow boundaries and only the illumination is different. This approach fails when the image regions under shadows are barely visible. Moreover, all of these methods involve a considerable effort in the design of hand-crafted features for shadow detection and feature selection (e.g., the use of ensemble learning methods to rank the best features [10, 11]). Our data-driven framework is different and unique: we propose to use deep feature learning methods to ‘learn the most relevant features’ for shadow detection.

Owing to the challenging nature of the shadow detection problem, many simplistic assumptions are commonly adopted. Previous works made assumptions related to the illumination sources [5], the geometry of the objects casting shadows and the material properties of the surfaces on which shadows are cast. For example, Salvador et al. [14] consider object cast shadows while Lalonde et al. [11] only detect shadows that lie on the ground. Some methods use synthetically generated training data to detect shadows [17]. Techniques targeted for video surveillance applications take advantage of multiple images [18] or time-lapse sequences [19, 20] to detect shadows. User assistance is also required by many proposed techniques to achieve their attained performances [21, 22]. In contrast, our shadow detection method makes absolutely ‘no prior assumptions’ about the scene, the shadow properties, the shape of objects, the image capturing conditions and the surrounding environments. Based on this premise, we tested our proposed framework on all of the publicly available databases for shadow detection from single images. These databases contain common real world scenes with artifacts such as noise, compression and color balancing effects.

### Shadow Removal and Matting

Almost all approaches that are employed to either edit or remove shadows are based on models that are derived from the image formation process. A popular choice is to physically model the image into a decomposition of its intrinsic images along with some parameters that are responsible for the generation of shadows. As a result, the shadow removal process is reduced to the estimation of the model parameters. Finlayson et al. [23, 24] addressed this problem by nullifying the shadow edges and reintegrating the image,
which results in the estimation of the additive scaling factor. Since such global integration (which requires the solution of a 2D Poisson equation [23, 25]) causes artifacts, the integration along a 1D Hamiltonian path [26] is proposed for shadow removal. However, these and other gradient based methods (such as [27, 28]) do not account for the shadow variations inside the umbra region. To address this shortcoming, Arbel and Hel-Or [29] treat the illumination recovery problem as a 3D surface reconstruction and use a thin plate model to successfully remove shadows lying on curved surfaces. Alternatively, information theory based techniques are proposed in [25, 30] and a bilateral filtering based approach is recently proposed in [31] to recover intrinsic (illumination and reflectance) images. However, these approaches either require user assistance, calibrated imaging sensors, careful parameter selection or considerable processing times. To overcome these shortcomings, some reasonably fast and accurate approaches have been proposed which aim to transfer the color statistics from the non-shadow regions to the shadow regions (‘color transfer based approaches’ e.g., [21, 32, 33, 34, 35]). Our proposed shadow removal algorithm also belongs to the category of color transfer based approaches. However, in contrast to previous related works, we propose a **generalized image formation model** which enables us to deal with non-uniform umbra regions as well as soft shadows. Color transfer is also made at **multiple spatial levels**, which helps in the reduction of noise and color artifacts. An added advantage of our approach is our ability to separate smooth shadow matte from the actual image.

Several assumptions are made in the shadow removal literature due to the ill-posed nature of recovering the model parameters for each pixel. The camera sensor parameters are needed in [23, 31]. Multiple narrow-band sensor outputs for each scene are required in [31], while [2] employs a sequence of images to recover the intrinsic components. Lambertian surface and Planckian lightening assumptions are made in [31]. Though several approaches work just on a single image, they require considerable user interaction to identify either tri-maps [36], quad-maps [33, 34], gradients [37] or exact shadow boundaries [27, 28]. Su and Chen [38] tried to minimize the user effort by specifying the complete shadow boundary from the user provided strokes. In contrast, our framework **does not require any form of user interaction** and makes no assumption regarding the camera or scene properties (except that the object surfaces are assumed to be Lambertian).

The key contributions of our work are outlined below:

- **We propose a new approach for robust shadow detection combining both regional and across-boundary learned features in a probabilistic framework involving CRFs (Sec. 3).**
- **Our proposed method automatically learns the most relevant feature representations from raw pixel values using multiple ConvNets (Sec. 3).**
- **We propose a generalized shadow formation model along with automatic color statistics modeling using only detected shadow masks (Sec. 4.1 and 4.2).**
- **Our proposed Bayesian formulation for the shadow removal problem integrates multi-level color transfer and the resulting cost function is efficiently optimized to give superior results (Sec. 4.3 and 4.4).**
- **We performed extensive quantitative evaluation to prove that the proposed framework is robust, less-constrained and generalisable across different types of scenes (Sec. 5).**

### 3 Proposed Shadow Detection Framework

Given a single color image, we aim to detect and localize shadows precisely at the pixel level (see block diagram in Fig. 2). If \( y \) denotes the desired binary mask encoding class relationships, we can model the shadow detection problem as a conditional distribution:

\[
P(y | x \; ; w) = \frac{1}{Z(w)} \exp(-E(y, x; w))
\]

where, the parameter vector \( w \) includes the weights of the model, the manifest variables are represented by \( x \) where \( x_i \) denotes the intensity of pixel \( i \in \{p_i\}_{i \in N} \) and \( Z(w) \) denotes the partition function. The energy function is composed of two potentials; the unary potential \( \psi_i \) and the pairwise potential \( \psi_{ij} \):

\[
E(y, x; w) = \sum_{i \in V} \psi_i(y_i, x; w_i) + \sum_{(i,j) \in E} \psi_{ij}(y_{ij}, x; w_{ij})
\]

In the following discussion, we will explain how we model these potentials in a CRF framework.

#### 3.1 Feature Learning for Unary Predictions

The unary potential in Eq. 2 considers the shadow properties both at the regions and at the boundaries inside an image.

\[
\psi_i(y_i, x; w_i) = \phi_i^r(y_i, x; w_i^r) + \phi_i^b(y_i, x; w_i^b)
\]

We define each of the boundary and regional potentials, \( \phi^r \) and \( \phi^b \) respectively, in terms of probability estimates from the two separate ConvNets,

\[
\phi_i^r(y_i, x; w_i^r) = -w_i^r \log P_{con1}(y_i | x_i) \\
\phi_i^b(y_i, x; w_i^b) = -w_i^b \log P_{con2}(y_i | x_i)
\]

This is logical because the features to be estimated at the boundaries are likely to be different from the ones estimated inside the shadowed regions. Therefore, we train two separate ConvNets, one for the regional potentials and the other for the boundary potentials.

The ConvNet architecture used for feature learning consists of alternating convolution and sub-sampling layers (Fig. 3). Each convolutional layer in a ConvNet consists of filter banks which are convolved with the input feature maps. The sub-sampling layers pool the incoming features to derive invariant representations. This
layered structure enables ConvNets to learn multilevel hierarchies of features. The final layer of the network is fully connected and comes just before the output layer. This layer works as a traditional MLP with one hidden layer followed by a logistic regression output layer which provides a distribution over the classes. Overall, after the network has been trained, it takes an RGB patch as an input and processes it to give a posterior distribution over binary classes.

ConvNets operate on equi-sized windows, so it is required to extract patches around desired points of interest. For the case of regional potentials, we extract super-pixels by clustering the homogeneous pixels\(^1\). Afterwards, a patch (\(I^r\)) is extracted by centering a \(r_s \times r_s\) window at the centroid of each superpixel. Similarly for boundary potentials, we first apply a Bilateral filter and then extract boundaries using the gPb technique \([40]\). We traverse each boundary with a stride \(\lambda_b\) and extract a \(r_s \times r_s\) patch at each step to incorporate local context\(^2\). Therefore, ConvNets operate on sets of boundary and super-pixel patches, \(X^b = \{ I^r(i,j) \}_{i,j} \times |F_{xb}(x)|\) and \(X^s = \{ I^r(i,j) \}_{i,j} \times |F_{xs}(x)|\) respectively, where \(|.|\) is the cardinality operator. Note that we include synthetic data (generated by artificial linear transformations \([41]\)) during the training process. This data augmentation is important not only because it removes the skewed class distribution of the shadowed regions but it also results in an enhanced performance. Moreover, data augmentation helps to reduce the overfitting problem in ConvNets (e.g., in \([42]\)) which results in the learning of more robust feature representations.

During the training process, we use stochastic gradient descent to automatically learn feature representations in a supervised manner. The gradients are computed using back-propagation to minimize the cross entropy loss function \([43]\). We set the training parameters (e.g., momentum and weight decay) using a cross validation process. The training samples are shuffled randomly before training since the network can learn faster from unexpected samples. The weights of the ConvNet were initialized with randomly drawn samples from a Gaussian distribution of zero mean and a variance that is inversely proportional to the \(fan-in\) measure of neurons. The number of epochs during the training of ConvNets is set by an early stopping criterion based on a small validation set. The initial learning rate is heuristically chosen by selecting the largest rate which resulted in the convergence of the training error. This rate is decremented by a factor of \(\nu = 0.5\) after every 20 epochs.

The ConvNet trained on boundary patches learn to separate shadow and reflectance edges while the ConvNet trained on regions can differentiate between shadow and non-shadow patches. For the case of the regions, the posteriors predicted by ConvNet are assigned to each super pixel in an image. However, for the boundaries, we first localize the probable shadow location using the local contrast and then average the predicted probabilities over each contour generated by the Ultra-metric Contour Maps (UCM) \([40]\).

\[ \phi_{p_1}(y_i, y_j) = \alpha 1_{y_i \neq y_j} \]

\[ \phi_{p_2}(y_i, y_j) = \alpha \]

\[ \psi_{ij}(y_{ij}, x; w_{ij}) = w_{ij} \phi_{p_1}(y_i, y_j) \phi_{p_2}(x) \]

3.2 Contrast Sensitive Pairwise Potential

The pairwise potential in Eq. 2 is defined as a combination of the class transition potential \(\phi_{p_1}\) and the spatial transition potential \(\phi_{p_2}\):

\[ \psi_{ij}(y_{ij}, x; w_{ij}) = w_{ij} \phi_{p_1}(y_i, y_j) \phi_{p_2}(x). \] (5)

The class transition potential takes the form of an Ising prior:

\[ \phi_{p_1}(y_i, y_j) = \alpha 1_{y_i \neq y_j} \]

\[ \phi_{p_2}(y_i, y_j) = \alpha \]

---

1. In our implementation we used SLIC \([39]\), due to its efficiency.
2. The step size is \(\lambda_b = r_s/4\) to get partially overlapping windows.
The spatial transition potential captures the differences in the adjacent pixel intensities:

$$\phi_{y_2}(x) = \exp\left(-\frac{||x_i - x_j||^2}{\beta_x ||x_i - x_j||^2}\right)$$

(7)

where $\langle \cdot \rangle$ denotes the average contrast in an image. The parameters $\alpha$ and $\beta_x$ were derived using cross validation on each database.

3.3 Shadow Contour Generation using CRF Model

We model the shadow contour generation in the form of a two-class scene parsing problem where each pixel is labeled either as a shadow or a non-shadow. This binary classification problem takes probability estimates from the supervised feature learning algorithm and incorporates them in a CRF model. The CRF model is defined on a grid structured graph topology, where graph nodes correspond to image pixels (Eq. 2). When making an inference, the most likely labeling is found using the Maximum a Posteriori (MAP) estimate $y^*$ upon a set of random variables $y \in \mathcal{L}^N$. This estimation turns out to be an energy minimization problem since the partition function $Z(w)$ does not depend on $y$:

$$y^* = \arg\max_{y \in \mathcal{L}^N} P(y|x; w) = \arg\min_{y \in \mathcal{L}^N} E(y, x; w)$$

(8)

The CRF model proved to be an elegant source to enforce label consistency and the local smoothness over the pixels. However, the size of the training space (labeled images) makes it intractable to compute the gradient of the likelihood. Therefore the parameters of the CRF cannot be found by simply maximizing the likelihood of the hand labeled shadows. Hence, we use the ‘margin rescaled algorithm’ to learn the parameters ($w$ in Eq. 8) of our proposed CRF model (see Fig 3 in [44] for details). Because our proposed energies are sub-modular, we use graph-cuts for making efficient inferences [45]. In the next section, we describe the details of our shadow removal and matting framework.

4 PROPOSED SHADOW REMOVAL AND MATTING FRAMEWORK

Based on the detected shadows in the image, we propose a novel automatic shadow removal approach. A block diagram of the proposed approach is presented in Fig. 4. The first step is to identify the umbra, penumbra and the corresponding non-shadowed regions in an image. We also need to identify the boundary where the actual object and its shadow meet. This identification helps to avoid any errors during the estimation of shadow/non-shadow statistics (e.g., color distribution). In previous works (such as [21, 29, 34]), this process has been carried out manually through human interaction. We, however, propose a simple procedure to automatically estimate the umbra, penumbra regions and the object-shadow boundary.

Heuristically, the object-shadow boundary is relatively darker compared to other shadow boundaries where differences in light intensity are significant. Therefore, given a shadow mask, we calculate the boundary normals at each point. We cluster the boundary points according to the direction of their normals. This results in separate boundary segments which join to form the boundary contour around the shadow. Then, the boundary segments in the shadow contour with a minimum relative change in intensity are classified to represent the object-shadow boundary. If $\phi_b$ denotes the mean intensity change along the normal direction at a boundary segment $b$ of the shadow contour $c$, all boundary segments s.t. $\phi_b / \phi_{max} \leq 0.5$ are considered to correspond to the segments which separate the object and its cast shadow. This simple procedure performs reasonably well for most of our test examples (Fig. 5). In the case where...
Fig. 6: Detection of Umbra and Penumbra Regions: With the detected shadow map (2nd image from left), we estimate the umbra and penumbra regions (rightmost image) by analyzing the gradient profile (4th image from left) at the boundary points.

Algorithm 4.1: Rough Estimation(S, N)

1: \( h_S, h_N \leftarrow \) Get histogram of color distribution in \( S, N \)
2: \( g_S, g_N \leftarrow \) Fit GMM on \( h_S, h_N \) using EM algorithm
3: for each \( j \in [0, J] \) do
   Channel wise color transfer between corresponding Gaussians using Eqs. 9, 10.
   Get probability of a pixel/super-pixel to belong to a Gaussian component using Eq. 11.
   Calculate overall transfer for each color channel using Eq. 12.
4: Combine multiple transfers:
   \( C^*(x, y) = \frac{1}{J} \sum_j C^j(x, y) \)
5: Calculate probability of a pixel to be shadow or non-shadow:
   \( p_S(x, y) = \frac{\sum_{k=1}^K \omega_k \frac{|\Sigma_k^S(x,y)|}{|\Sigma_k^S(x,y)| + |\Sigma_k^N(x,y)|}}{1} \)
6: Modify color transfer using Eq. 13
7: Improve result from above step using Eq. 14
return \( \hat{I}(x, y) \)

4.1 Rough Estimation of Shadow-less Image by Color-transfer

The rough shadow-less image estimation process is based on the one adopted by the color transfer techniques in [32] and [34]. As opposed to [32, 34], we perform a multilevel color transfer and our method does not require any user input. The color statistics of the shadowed as well as the non-shadowed regions are modeled using a Gaussian mixture model (GMM). For this purpose, a continuous probability distribution function is estimated from the histograms of both regions using the Expectation-Maximization (EM) algorithm. The EM algorithm is initialized using an unsupervised clustering algorithm (k-means in our implementation) and the EM iterations are carried out until convergence. We treat each of the R, G and B channels separately and fit mixture models to each of the respective histograms. It is considered that the estimated Gaussians, in the shadow and non-shadow regions, correspond to each other when arranged according to their means. Therefore, the color transfer is computed among the corresponding Gaussians using the following pair of equations:

\[
D^k_S(x, y) = \frac{\hat{I}(x, y) - \mu^k_S}{\sigma^k_S} \quad (9)
\]

\[
C^k(x, y) = \mu^k_S + \sigma^k_S D^k_S(x, y) \quad (10)
\]

where \( D(\cdot) \) measures the normalized deviation for each pixel, \( S \) and \( N \) denote the shadow and non-shadow regions respectively. The index \( k \) is in range \([1, K]\), where \( K \) denotes the total number of Gaussians used to approximate the histogram of \( S \). The probability that a pixel (with coordinates \( x, y \)) belongs to a certain Gaussian component can be represented in terms of its normalized deviation:

\[
p^k_G(x, y) = \left( \frac{1}{K} \sum_{k=1}^K \frac{1}{|D^k_S(x, y)|} + \epsilon \right)^{-1} \quad (11)
\]

The overall transfer is calculated by taking the weighted sum of transfers for all Gaussian components:

\[
c^{j=0}(x, y) = \sum_{k=1}^K p^k_G(x, y) C^k(x, y). \quad (12)
\]

The color transfer performed at each pixel location (i.e. at level \( j = 0 \)) using Eq. 12 is local, and it thus, does not accurately restore the image contrast in the shadowed regions. Moreover, this local color transfer is prone to noise and discontinuities in illumination. We
therefore resort to a hierarchical strategy which restores color at multiple levels and combines all transfers which results in a better estimation of the shadow-less image. A graph based segmentation procedure [46] is used to group the pixels. This clustering is performed at J levels, which we set to 4 in the current work based on the performance on a small validation set, where we noted an over-smoothing and a low computational efficiency when J ≥ 5. Since, the segment size is kept quite small, it is highly unlikely that the differently colored pixels will be grouped together. At each level j ∈ [1, J], the mean of each cluster is used in the color transfer process (using Eqs. 9, 10) and the resulting estimate (Eq. 12) is distributed to all pixels in the cluster. This gives multiple color transfers Cj(x, y) at J different resolutions plus the local color transfer i.e. Cj=0(x, y). At each level, a pixel or a super-pixel is treated as a discrete unit during the color transfer process. The resulting transfers are integrated to produce the final outcome:

\[ C^*(x, y) = \frac{1}{J} \sum_{j=0}^{J-1} C_j(x, y). \]

This process helps in reducing the noise. It also restores a better texture and improves the quality of the restored image. It should be noted that our hierarchical strategy helps in successfully retaining the self shading patterns in the recovered image compared to previous works (Sec. 5.3).

To avoid possible errors due to the small non-shadow regions that may be present in the selected shadow region S, we calculate the probability of a pixel to be shadowed using:

\[ p_S(x, y) = \sum_{k=1}^{K} \omega_k \delta_{S}(x, y), \]

where \( \omega_k \) is the weight of Gaussians (learned by the EM algorithm) and \( \delta_{S}(x, y) = |D_{S}^{\max}/(|D_{S}^{\min}| + |D_{S}^{\max}|) \). The color transfer is modified as:

\[ C_j(x, y) = (1 - p_S(x, y))I_S(x, y) + p_S(x, y)C^*(x, y) \quad (13) \]

However, the penumbra region pixels will not get accurate intensity values. To correct this anomaly, we define a relation which measures the probability (in a naive sense) of a pixel to belong to the penumbra region. Since the penumbra region occurs around the shadow boundary, we define it as: \( b_S(x, y) = d(x, y)/d_{\text{max}} \). The penumbra region is recovered using the exemplar based inpainting approach of Criminisi et al. [47]. The resulting improved approximation of the shadow-less image is:

\[ I(x, y) = (1 - b_S(x, y))I_S(x, y) + b_S(x, y)C^*(x, y) \quad (14) \]

where, \( I \) is the inpainted image.

In our approach, the crude estimate of a shadow-less image (Eq. 14) is further improved using Bayesian estimation (Sec. 4.3). But first we need to introduce the proposed shadow generation model used in our Bayesian formulation (Sec. 4.2).

### 4.2 Generalised Shadow Generation Model

Unlike previous works (such as [13, 21, 27, 34, 35]), which do not differentiate between the umbra and the penumbra regions during the shadow formation process, we propose a model which treats both types of shadow regions separately. It is important to make such distinction because the umbra and penumbra regions exhibit distinct illumination characteristics and have a different influence from the direct and indirect light (Fig. 6).

Let us suppose that we have a scene with illuminated and shadowed regions. A normal illuminated image can be represented in terms of two intrinsic images according to the image formation model of Barrow et al. [48]:

\[ I(x, y) = L(x, y)R(x, y) \quad (15) \]

where \( L \) and \( R \) are the illumination and reflectance respectively and \( x, y \) denote the pixel coordinates. The illumination intrinsic image takes into account the illumination differences such as shadows and shading. We assume that a single source of light is casting the shadows. The ambient light is assumed to be uniformly distributed in the environment due to the indirect illumination caused by reflections. Therefore,

\[ I(x, y) = (L_d(x, y) + L_i(x, y))R(x, y) \quad (16) \]

A cast shadow is formed when the direct illumination is blocked by some obstructing object resulting in occlusion. A cast shadow can be described as the combination of two regions created by two distinct phenomena, umbra (U) and penumbra (P). Umbra is surrounded by the penumbra region where the light intensity changes sharply from dark to illuminated. The occlusion which casts the shadow block all of the direct illumination and parts of the indirect illumination to create the umbra.
We can represent this as:

\[
I_u(x, y) = \beta'(x, y) \mathcal{L}_d(x, y) \mathcal{R}(x, y) \quad \forall x, y \in U
\]

where, \(\beta'(x, y)\) is the scaling factor for the U region.

Using Eq. 16 and 17, we have:

\[
I(x, y) = \frac{I_u(x, y)}{\beta'(x, y)} + \alpha(x, y)
\]

(18)

where, \(\alpha(x, y) = \mathcal{L}_d(x, y) \mathcal{R}(x, y)\).

For the case of the penumbra region, all direct light is not blocked, rather its intensity decreases from a fully lit region towards the umbra region. Since the major source of change is the direct light, we can neglect the variation caused by the indirect illumination in the penumbra region. Therefore,

\[
I_p(x, y) = (\beta''(x, y) \mathcal{L}_d(x, y) + \mathcal{L}_s(x, y)) \mathcal{R}(x, y)
\]

(20)

where, \(\beta''(x, y)\) is the scaling factor for the P region.

Using Eq. 16 and 20, we have:

\[
I_p(x, y) = I(x, y) - \alpha(x, y)(1 - \beta''(x, y)).
\]

(21)

\section*{4.3 Bayesian Shadow Removal and Matting}

Having formulated the shadow generation model, we can now describe the estimation procedure of the model parameters in probabilistic terms. We represent our problem in a well-defined Bayesian formulation and estimate the required parameters using maximum a posteriori estimate (MAP):

\[
\{\alpha^*, \beta^*\} = \arg\max_{\alpha, \beta} P(\alpha, \beta | U, P, N)
\]

(22)

\[
= \arg\max_{\alpha, \beta} \frac{P(U, P, N | \alpha, \beta) P(\alpha) P(\beta)}{P(U, P, N)}
\]

(23)

\[
= \arg\max_{\alpha, \beta} P_U(U, P, N | \alpha, \beta) + P_P(\alpha) + P_N(\beta) - P(U, P, N)
\]

(24)

where, \(P_U = \log P(\cdot)\) is the log likelihood and \(U, P, N\) represent the umbra, penumbra and non-shadow regions respectively. The last term in the above equation can be neglected during optimization because it is independent of the model parameters. Therefore:

\[
= \arg\max_{\alpha, \beta} P_U(U, P, N | \alpha, \beta) + P_P(\alpha) + P_N(\beta)
\]

(25)

Let \(I_u(x, y) \forall x, y \in \{U \cup P\}\) represent the complete shadow region. Then, the first term in Eq. 25 can be written as a function of \(I_u\) since the parameters \(\alpha\) and \(\beta\) do not affect the region \(N\), therefore:

\[
= \arg\max_{\alpha, \beta} P_U(I_u | \alpha, \beta) + P_P(\alpha) + P_N(\beta)
\]

(26)

The first term in Eq. 26 can be modeled by the difference between the current pixel values and the estimated pixel values, as follows:

\[
P_U(I_u | \alpha, \beta) = -\sum_{(x, y) \in S} \frac{I_u(x, y) - \hat{I}_u(x, y)^2}{2\sigma_u^2} - \sum_{(x, y) \in S} \frac{\pi(x, y)\eta(x, y)I(x, y) - \hat{I}_u(x, y)^2}{2\sigma_s^2}
\]

(27)

where, \(\eta(x, y) = 1 - \frac{\lambda(x, y)}{\lambda_{\max}}\) and \(\pi\) is an indicator function which switches on for the penumbra region pixels. \(\lambda(\cdot)\) is the distance metric which quantifies the shortest distance between a valid shadow boundary (i.e., excluding the object-shadow boundary). The estimated shadowed image \((\hat{I}_u)\) can be decomposed as follows using Eqs. 19 and 21.

\[
\hat{I}_u(x, y) = \begin{cases} 
(\hat{I}_u(x, y) - \alpha(x, y))\beta'(x, y) & \forall \{x, y\} \in U \subset S \\
\hat{I}_u(x, y) - \alpha(x, y)(1 - \beta''(x, y)) & \forall \{x, y\} \in P \subset S 
\end{cases}
\]

(28)

It can be noted that \(P_U(I_u | \alpha, \beta)\) models the error caused by the estimated parameters and encourages the recovered pixel values \((\hat{I}_u(x, y))\) to lie close to \((I_u(x, y))\) with variance \(\sigma^2\) following a Gaussian distribution. However, in the above formulation, there are nine unknowns for each pixel located inside the shadowed region. If we had a smaller scale problem (e.g., finding the precise shadow matte in the penumbra region by Chuang et al. [36]), we could have directly solved for the unknowns. But in our case, the large number of variables makes the likelihood calculation rather difficult and time consuming, especially when the number of shadowed pixels is large.

We therefore resort to optimize the crude shadow-less image \((\hat{I}_u(x, y))\) calculated in Sec. 4.1, Eq. 14.

The prior \(P_N(\beta)\) can be modeled as a Gaussian probability distribution centered at the mean \((\beta)\) of the
neighboring pixels. This helps in estimating a smoothly
varying beta mask. So,
\[
P(\beta) = -\sum_{(x,y)} \frac{[\beta(x,y) - \bar{\beta}(x',y')]^2}{2\sigma_\beta^2}, \quad (x',y') \in \mathcal{N}(x,y) \tag{28}
\]

The prior \( P(\alpha) \) can also be modeled in a similar
fashion. However, we require \( \alpha \) to model the variations
in the penumbra region as well. Therefore, an additional
term (called the ‘image consistency term’) is introduced in
the prior \( P(\alpha) \) to smooth the estimated shadow-less
image along the boundaries and to incorporate feedback
from the previously estimated crude shadowless image.
Therefore,
\[
P(\alpha) = -\sum_{(x,y)} \frac{|\alpha(x,y) - \bar{\alpha}(x',y')|^2}{2\sigma_\alpha^2} - \frac{1}{2\sigma_\alpha^2} \sum_{(x,y) \in \mathcal{S}} (1 - \frac{\lambda(x,y)}{\lambda_{\text{max}}})|I(x,y) - \hat{I}(x,y)|^2, \quad (x',y') \in \mathcal{N}(x,y)
\] \tag{29}

In the image consistency term (second term in Eq. 29),
\( \mathcal{I}(x,y) \) will take different values according to Eqs. 19 and
21:
\[
\mathcal{I}(x,y) = \begin{cases}
\mathcal{I}_u/\beta'(x,y) + \alpha(x,y) & \forall \{x,y\} \in \mathcal{U} \\
\mathcal{I}_p(x,y) + \alpha(x,y)(1 - \beta''(x,y)) & \forall \{x,y\} \in \mathcal{P}
\end{cases}
\]

4.4 Parameter Estimation

In spite of the crude shadow image estimation, it can be
seen from Eq. 27 that the objective function is not linear
or quadratic in term of the unknowns. To apply the gra-
dient based energy optimization procedure, we simplify
this procedure on each pixel in the shadow region until
convergence.

4.5 Boundary Enhancement in a Shadow-less Image

The resulting shadow-less image exhibits traces of
shadow boundaries in some cases. To remove these
artifacts, we divide the shadow boundary into a group
of segments, where each segment contains nearly similar
colored pixels. The boundary segments which belong to
the object shadow boundary are excluded from further
processing. For each non-object shadow boundary seg-
ment, we perform Poisson smoothing [49] to conceal the
shadow boundary artifacts.

5 Experiments and Analysis

We evaluated our technique on three widely used and
publicly available datasets. For the qualitative compari-
sion of shadow removal, we also evaluate our technique
on a set of commonly used images in the literature.

5.1 Datasets

UCF Shadow Dataset is a collection of 355 images
together with their manually labeled ground truths. Zhu
et al. have used a subset of 255/355 images for shadow
detection [10].

CMU Shadow Dataset consists of 135 consumer grade
images with labels for only those shadow edges which
lie on the ground plane [11]. Since our algorithm is not
restricted to ground shadows, we tested our approach on
the more challenging criterion of full shadow detection
which required the generation of new ground truths.

UIUC Shadow Dataset contains 108 images each of
which is paired with its corresponding shadow-free im-
age to generate a ground truth shadow mask [13].

Test/Train Split: For UCF and UIUC databases, we used
the split mentioned in [10, 13]. Since CMU database [11]
did not report the split, we therefore used even/odd
images for training/testing (following the procedure in
Jiang et al. [12]).
5.2 Evaluation of Shadow Detection

5.2.1 Results

We assessed our approach both quantitatively and qualitatively on all the major datasets for single image shadow detection. We demonstrate the success of our shadow detection framework on different types of scenes including beaches, forests, street views, aerial images, road scenes and buildings. The databases also contain shadows under a variety of illumination conditions such as sunny, cloudy and dark environments. For quantitative evaluation, we report the performance of our framework when only the unary term (Eq. 3) was used for shadow detection. Further, we also report the per-pixel accuracy achieved using the CRF model on all the datasets. This means that labels are predicted for every pixel in each test image and are compared with the ground-truth shadow masks. For the UCF and CMU datasets, the initial learning rate of $\eta_0 = 0.1$ was used, for the UIUC dataset we set $\eta_0 = 0.01$ based on the performance on a small validation set. After every 20 epochs the learning rate was decreased by a small factor $\beta = 0.5$ which resulted in a best performance.

Table 1 summarizes the overall results of our framework and shows a comparison with several state-of-the-art methods. The ROC curve comparisons are shown in Fig. 10. Some representative qualitative results are shown in Fig. 9 and Fig. 11. The proposed framework successfully detects shadows in dark environments (Fig. 9: 1st row, middle image) and distinguishes between dark non-shadow regions and shadow regions (Fig. 9: 2nd row, 3rd and 5th image from left). It performs equally well on satellite images (Fig. 9: last column) and outdoor scenes with street views (Fig. 9: 1st row, 3rd and 5th images; 2nd row, middle image), buildings (Fig. 9: 1st column) and shadows of animals and humans (Fig. 9: 2nd column).

<table>
<thead>
<tr>
<th>Methods/Uncertainty Priors</th>
<th>UCF Dataset</th>
<th>CMU Dataset</th>
<th>UIUC Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>BDT-BCRF (Zhu et al. [10])</td>
<td>85.70%</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>BDT-CRF-Scene Layout (Lalonde et al. [11])</td>
<td>90.20%</td>
<td>84.80%</td>
<td>89.10%</td>
</tr>
<tr>
<td>Unary SVM-Pairwise (Guo et al. [13])</td>
<td>85.90%</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>Bright Channel-MRF (Panagopoulos et al. [15])</td>
<td>83.50%</td>
<td>84.98%</td>
<td>--</td>
</tr>
<tr>
<td>ConvNet(Boundary+Region)</td>
<td>89.31%</td>
<td>87.02%</td>
<td>92.31%</td>
</tr>
<tr>
<td>ConvNet(Boundary+Region)-CRF</td>
<td>90.65%</td>
<td>88.79%</td>
<td>93.16%</td>
</tr>
</tbody>
</table>

TABLE 1: Evaluation of the proposed shadow detection scheme; All performances are reported in terms of pixel-wise accuracies.

<table>
<thead>
<tr>
<th>Methods/Datasets</th>
<th>Shadows</th>
<th>Non-Shadows</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>UCF Dataset</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BDT-CRF [Zhu et al. [10]]</td>
<td>85.50%</td>
<td>93.30%</td>
</tr>
<tr>
<td>Unary-Pairwise (Guo et al. [13])</td>
<td>73.3%</td>
<td>93.7%</td>
</tr>
<tr>
<td>Bright Channel-MRF (Panagopoulos et al. [15])</td>
<td>68.3%</td>
<td>89.4%</td>
</tr>
<tr>
<td>ConvNet(Boundary+Region)</td>
<td>72.5%</td>
<td>92.1%</td>
</tr>
<tr>
<td>ConvNet(Boundary+Region)-CRF</td>
<td>78.0%</td>
<td>92.6%</td>
</tr>
<tr>
<td><strong>CMU Dataset</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BDT-CRF [Lalonde et al. [11]]</td>
<td>73.3%</td>
<td>96.3%</td>
</tr>
<tr>
<td>ConvNet(Boundary+Region)</td>
<td>81.5%</td>
<td>90.5%</td>
</tr>
<tr>
<td>ConvNet(Boundary+Region)-CRF</td>
<td>83.3%</td>
<td>90.9%</td>
</tr>
<tr>
<td><strong>UIUC Dataset</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Unary-Pairwise (Guo et al. [13])</td>
<td>71.6%</td>
<td>95.2%</td>
</tr>
<tr>
<td>ConvNet(Boundary+Region)</td>
<td>83.0%</td>
<td>94.7%</td>
</tr>
<tr>
<td>ConvNet(Boundary+Region)-CRF</td>
<td>84.7%</td>
<td>95.5%</td>
</tr>
</tbody>
</table>

TABLE 2: Class-wise accuracies of our proposed framework in comparison with the state-of-the-art techniques. Our approach gives the highest accuracy for the class ‘shadows‘.

3. Relative increase in performance is calculated by: $100 \times \frac{(\text{our accuracy} - \text{previous best})}{\text{previous best}}$. This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication. The final version of record is available at http://dx.doi.org/10.1109/TPAMI.2015.2462355
Fig. 9: Examples of our results; Images (1st, 3rd row) and shadow masks (2nd, 4th row); Shadows are in white.

Fig. 10: ROC curve comparisons of proposed framework with previous works.

5.2.2 Discussion

The previously proposed methods (e.g., Zhu et al. [10], Lalonde et al. [11]) that use a large number of hand-crafted features, not only require a lot of effort in their design but also require long training times when ensemble learning methods are used for feature selection. As an example, Zhu et al. [10] extracted different shadow variant and invariant features alongside an additional 40 classification results from the Boosted Decision Tree (BDT) for each pixel as their features. Their approach required a huge amount of memory (~9 GB for 125 training images of average size of approximately 480 × 320). Even after parallelization and training on multiple processors, they reported 10 hours of training with 125 images. Lalonde et al. [11] used 48 dimensional feature vectors extracted at each pixel and fed these to a boosted decision tree in a similar manner as Zhu et al. [10]. Jiang et al. included illumination features on top of the features that are used by Lalonde et al. [11]. Although, enriching the feature set in this manner improves the performance, it not only takes much more effort to design such features but it also slows down the detection procedure. In contrast, our feature learning procedure is fully automatic and requires only ~1GB memory and approximately one hour training for each of the UCF, CMU and UIUC databases. The proposed approach is also efficient at test time because the ConvNet feature extraction and unary potential computation take an average of 1.3 ± 0.35 sec per image on the UCF, CMU and UIUC databases.

The graph-cut inference step used for the CRF energy minimization is also fast and takes 0.21 ± 0.03 sec per image on average. Overall, our technique takes 2.8 ± 0.81 sec per image for shadow detection. In comparison, the method by Guo et al. [13] takes 40.05 ± 10 sec per image for shadow detection.

We extensively evaluated our approach on all available databases and our proposed framework turned out to be fairly generic and robust to variations. It achieved the best results on all the single image shadow databases known to us. In contrast, previous techniques were only tested on a portion of database [11], one [10] or at most two databases [13]. Another interesting observation was that the proposed framework performed reasonably well when our ConvNets were trained on one dataset and tested on another dataset. Table 3 summarizes the results of cross-dataset evaluation experiments. These
performance levels show that the feature representations
learned by the ConvNets across the different datasets
were common to a large extent. This observation further
supports our claim regarding the generalization ability
of the proposed framework.

In our experiments, objects with dark albedo turned
out to be a difficult case for shadow detection. More-
over, some ambiguities were caused by the complex self
shading patterns created by tree leaves. There were some
inconsistencies in the manually labeled ground-truths,
in which a shadow mask was sometimes missing for
an attached shadow. Narrow shadowy regions caused
by structures like poles and pipes also proved to be a
challenging case for shadow detection. Examples of the
above mentioned failure cases are shown in Fig. 11.

5.3 Evaluation of Shadow Removal
For a quantitative evaluation of our shadow removal
framework, we used all images from the UIUC Shadow
dataset which come with their corresponding shadow-
free ground truths [13]. The qualitative results of our
method are evaluated against the common evaluation
images used in the literature for a fair comparison.
To further illustrate the performance of our algorithm,
we also included qualitative results on some example
images from UIUC, UCF and CMU shadow datasets.

5.3.1 Quantitative Evaluation
Table 4 presents the per pixel root mean square error
(RMSE) for the UIUC dataset, calculated in LAB color
space [13]. The first row gives the actual error between
the same image, with and without shadow. The dif-
ference between the two versions of the same image
is calculated for both the shadow and the lit regions.
Note that the error is large for the shadowed region (as
expected), but it is not zero for the lit regions for two
reasons: the shadow masks are not perfect and there
is a little difference in the light intensity due to the
change in the ambient light for the lit regions when

\[
\begin{array}{|c|c|c|c|}
\hline
\text{Methods} & \text{Shadow Reg.} & \text{Lit Reg.} & \text{All Reg.} \\
\hline
\text{Actual Error} & 42.9 & 4.8 & 14.7 \ \\
1. Removal (Wu et al. [34]) with & 28.2 & 7.6 & 12.6 \\
Automatic Shadow Detection & & & \\
1b. Removal (Wu et al. [34]) using GT & 21.3 & 5.9 & 9.7 \\
2a. Removal (Guo et al. [13]) & 13.9 & 5.4 & 7.4 \\
2b. Removal using GT (Guo et al. [13]) & 11.8 & 4.7 & 6.4 \\
3a. Removal without Bayesian Refinement & 15.2 & 5.5 & 7.9 \\
3b. Removal with Bayesian Refinement & 12.1 & 5.1 & 6.8 \\
3c. Removal using GT & 10.5 & 4.7 & 6.1 \\
\hline
\end{array}
\]

the object casting shadow is present. We achieved an
average RMSE error of 6.8 compared to 7.4 and 12.6
achieved by the methods of Guo et al. [13] and Wu et
al. [34], respectively. Following Guo et al. [13], we also
include the removal performance when the ground truth
(GT) shadow masks are used for removal. This gives a
more precise estimate of the performance of the recovery
algorithm. When we evaluated our method using GT
masks, our method achieved an error of 6.1 compared
to 6.4 and 9.7 reported by [13] and [34] respectively. We
also tested the removal results without the Bayesian opti-
mization, which resulted in an RMSE error of 7.9. This
is high compared to the results achieved after optimization.
In summary, our method achieved a reduction in error
of 8.1% (removal using the detected masks) and 4.6% (removal
using ground truths) compared to the approach of
Guo et al. in [13].

5.3.2 Qualitative Evaluation
For the qualitative evaluation, we show some example
images and their corresponding recovered images along
with the shadow masks in Fig. 12. It can be seen that our
method works well under different settings e.g., outdoor
images (first five images from the left) and indoor images
(first two images from the right). The complex texture
in the shadow regions is preserved and the arbitrary
shadow matte are precisely recovered. Note that while

Fig. 12: Qualitative Evaluation: Shadow recovery on sample images from UIUC, UCF databases and other images used in
literature. Given a original image with shadow mask (first row), our method is able to extract exact shadows (second row) and
to automatically recover the shadow-less images (third row). (Best viewed in color)
Fig. 13: Comparison with Automatic/Semi-Automatic Methods: Recovered shadow-less images are compared with the state-of-the-art shadow removal methods which are either automatic [13, 23] or require minimal user input [21, 35]. We compare our work with: (from left to right) Finlayson et al. [23], Shor and Lischinski [21], Xiao et al. [35] and Guo et al. [13] respectively. The results achieved using our method (second column from right) are comparable or better than the previous best results (columns 1-5 from left). Additionally, our method works without any user input and provides shadow matte (last column) which can be used to generate composite images. (Best viewed in color and enlarged)

our method can remove hard and smooth shadows (e.g., 1st, 5th and 6th image from left), it also works well for the soft and variable shadows (e.g., 2nd, 3rd and 4th image from left). Overall, the results are visually pleasing and the extracted shadow matte are smooth and accurate.

5.3.3 Comparisons

We provide a qualitative comparison with two distinct categories of shadow removal methods. First, we show comparisons (see Fig. 13) with the state-of-the-art shadow removal methods which are either fully automatic (e.g., [13, 23]) or require minimal user input (e.g., [21, 35]). From left to right we show the original image along with the results from Finlayson et al. [23], Shor and Lischinski [21], Xiao et al. [35], Guo et al. [13] and our technique. In comparison to the previous automatic and semi-automatic (requiring minimal user input) methods, our approach produces cleaner recovered images (second column from the right) along with an accurate shadow matte (right most column).

Since, there are only very few automatic shadow removal methods in the literature, we also compare our approach with the most popular approaches but which require user input (see Fig. 14). From left to right, we show our recovered images (bottom row) along with the results from Wu et al. [34], Liu and Gleicher [27], Arbel and Hel-Or [29], Vicente and Samaras [50], Fredembach and Finlayson [26] and Kwatra et al. [30]. For the ‘puzzled child’ image, it can be seen that the contrast of the recovered region is much better than the one recovered by Wu et al. [34]. The shadow-less image has no trace of strong shadow boundaries and the recovery in the penumbra region is smooth due to introduction of $\alpha$ in the model and the exclusion of the spatial affinity term [34] or boundary nullification [33] during the rough shadowless image estimation process. Similar effects can be seen with the other images; e.g., in 3rd image from the left, the result of Arbel and Hel-Or [29] has a high contrast while our result is smooth and successfully retains texture. Similarly, for the case of the 4th, 5th and 6th images from the left, our shadow removal result is visually pleasing and considerably better than the recent state-of-the-art methods. Note however that the recovery result of the 2nd image from the left has an over-smoothing effect, probably because the color distributions of differently colored shadowed regions could not be separated during the Gaussian fitting process. Overall, the results are quite reasonable considering that the algorithm does not require any user assistance and it does not make any prior assumptions such as a Planckian light source or a narrow-band camera.

5.3.4 Failure Cases and Limitations

Our shadow removal technique does not perform well on curved surfaces and in the case of highly non-uniform shadows (e.g., Fig. 15: 1st and 3rd image from left). Since, we apply a multi-level color transfer scheme, very fine texture details of image regions with similar appearance can be removed during this transfer process (e.g., Fig. 15: 2nd image from left). For the cases of shadows in dark environments, our method appears to increase the contrast of the recovered region. These limitations are due to the constraints imposed on the shadow generation model, where the higher order statistics are ignored during the shadow generation process (Eqs. 19 and 21).
Fig. 14: Comparison with Methods Requiring User Interaction: Recovered shadow-less images are compared with the state-of-the-art shadow removal methods (which require considerable amount of user input). We compare our work with: (from left to right in the second row) Wu et al. [34], Liu and Gleicher [27], Arbel and Hel-Or [29], Vicente and Samaras [50], Fredembach and Finlayson [26] and Kwatra et al. [30] respectively. The results achieved by our method (last row) are comparable or better than the previous best results (second row). Additionally, our method works without any user input and provides shadow matte (third row) which can be used to generate composite images. (Best viewed in color and enlarged)

5.3.5 Discussion

Our method does not require any user input and it automatically removes shadow after its detection. The proposed shadow removal approach makes comparatively fewer assumptions about the scene type, the type of light source or camera. The only assumptions are that of Lambertian surfaces and the correspondence between the shadow and the non-shadow region color distributions. The shadow removal method of [33, 34] cannot separate the shadow from shading. With the inclusion of the image consistency term in $P_l(I_s|\alpha, \beta)$, we are able to deal with the shading by introducing a penalty on the distribution of the shadow effect through the parameters $\beta$ and $\alpha$. The proposed shadow removal approach takes $82.2 \pm 25$ sec for each image on the UIUC database. The main overhead during the shadow removal process is the Bayesian refinement step (which is required mainly for shadow matting). It takes $73.6 \pm 20$ sec out of $82.2 \pm 25$ sec per image on the UIUC database. In comparison, the method by Guo et al. [13] takes $104.7 \pm 18$ sec for shadow removal. The main overhead in their removal process is also due to Levin et al.’s matting algorithm [51] which takes around $91.4 \pm 11$ sec per image.

5.4 Applications

Shadow detection, removal and matting have a number of applications. A direct application is the generation of visually appealing photographs and the removal of unwanted shadows. Some other applications include:

Shadow Compositing: Fig. 16(a) shows examples of shadow compositing. The extracted shadow matte can be used to depict a realistic image compositing. For example, the first image from the left did not originally contain the flying bird and its shadow. If we had added just the bird, it would have looked unrealistic. With the addition of a texture-free shadow matte, the photograph looks natural and realistic. In the remaining three im-
of boundary and region ConvNets incorporated in the CRF model provides the best performance. For shadow removal, the multi-level color transfer followed by the Bayesian refinement performs well on unconstrained images. The proposed framework has a number of applications including image editing and enhancement tasks.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve the quality of the paper. This research was supported by the IPRS awarded by The University of Western Australia and the Australian Research Council (ARC) grants DP110102166, DP150100294 and DE120102960. The Flickr images are used under a Creative Commons license from Flickr users: mountaintrekker2001, Ulrich J and Kanshian.

REFERENCES


Fig. 16: Different Applications of Shadow Detection, Removal and Matting. (Best viewed in color and enlarged)


Ferdous A Sohel received PhD degree majoring Image Processing from Monash University, Australia (in 2009). He joined the School of Computer Science and Software Engineering, The University of Western Australia in 2008 where he is currently a Research Assistant Professor. His research interests include computer vision, image processing, pattern recognition, and video coding. He has published more than 50 scientific articles. He is a recipient of prestigious Discovery Early Career Research Award (DECRA) funded by the Australian Research Council. He is a recipient of the Early Career Investigator award (UWA) and the best PhD thesis medal form Monash University.

Roberto Togneri (M89-SM04) received the B.E. degree in 1985, and the Ph.D degree in 1989 both from the University of Western Australia. He joined the School of Electrical, Electronic and Computer Engineering at The University of Western Australia in 1988, where he is now currently a Professor. Prof Togneri is a member of the Signals and Systems Engineering Research Group and heads the Signal and Information Processing Lab. His research activities include signal processing and robust feature extraction of speech signals, statistical and neural network models for speech and speaker recognition, audio-visual recognition and biometrics, and related aspects of communications, information retrieval, and pattern recognition. He has published over 150 refereed journal and conference papers in the areas of signals and information systems, was the chief investigator on two Australian Research Council Discovery Project research grants from 2010 to 2013, and is currently an Associate Editor for IEEE Signal Processing Magazine Lecture Notes and Editor for IEEE Transactions on Speech, Audio and Language Processing.