LEARNING-BASED SHADOW RECOGNITION AND REMOVAL FROM MONOCHROMATIC NATURAL IMAGES

Seminar Report

Submitted in the partial fulfilment of the award of the degree of

Master of Technology

in

Computer Science with Specialization in Image Processing

of

APJ Abdul Kalam Technological University

by

JICKCY ANNA KOSHY

November, 2017

Department of Computer Engineering

College of Engineering, Chengannur Kerala-689121

Phone: (0479) 2454125, 2451424; Fax: (0479) 2451424

COLLEGE OF ENGINEERING, CHENGANNUR
KERALA

Department of Computer Engineering

CERTIFICATE

This is to certify that the seminar entitled

LEARNING-BASED SHADOW RECOGNITION AND REMOVAL FROM MONOCHROMATIC NATURAL IMAGES

Submitted by

JICKCY ANNA KOSHY

is a bonafide record of the work done by him.

Co-ordinator | **Head of The Department**

ACKNOWLEDGEMENT

I am greatly indebted to God Almighty for being the guiding light throughout with his abundant grace and blessings that strengthened me to do this endeavour with confidence.

I express my heartfelt gratitude towards **Dr. Jacob Thomas V.**, Principal, College of Engineering Chengannur for extending all the facilities required for doing my seminar. I would also like to thank **Dr. Smitha Dharan**, Head, Department of Computer Engineering, for providing constant support and encouragement.

Now I extend my sincere thanks to my seminar co-ordinators **Mr. Gopakumar G.**, Associate Professor in Computer Engineering and **Ms. Meera Varma** , Assistant Professor in Computer Engineering for guiding me in my work and providing timely advices and valuable suggestions.

Last but not the least, I extend my heartfelt gratitude to my parents and friends for their support and assistance.

ABSTRACT

Learning based approach is used for shadow recognition and removal from monochromatic natural images. Without chromatic information, shadow recognition and removal are extremely challenging in the literature, mainly due to the missing of invariant color cues. Natural scenes make this problem even harder due to the complex illumination condition and ambiguity from many near-black objects. In this paper, a learning based shadow recognition and removal scheme is proposed to tackle the challenges above. Firstly, we propose to use both shadow-variant and invariant cues from illumination, texture and odd order derivative characteristics to recognize shadows. Such features are used to train a classifier via boosting a decision tree and integrated into a Conditional Random Field, which can enforce local consistency over pixel labels. Secondly, a Gaussian model is introduced to remove the recognized shadows from monochromatic natural scenes. The proposed scheme is evaluated using both qualitative and quantitative results based on a novel database of hand-labeled shadows, with comparisons to the existing state-of-the-art schemes. We show that the shadowed areas of a monochromatic image can be accurately identified using the proposed scheme, and highquality shadow-free images can be precisely recovered after shadow removal.

Contents

1 INTRODUCTION **1**

2 MONOCHROMATIC BASED FEATURE EXTRACTION **3**

2.1 Dataset 3

2.2 Feature Extraction 4

2.2.1 Shadow-Variant Features 4

2.2.2 Shadow-Invariant Features 5

2.2.3 Near-Black Features 6

3 LEARNING TO RECOGNIZE SHADOW **8**

3.1 Logistic Random Field 8

3.2 Learning LRF to Recognize Shadows 9

4 PRACTICAL ISSUES **10**

4.1 Feature Boosting 10

4.2 Parallel Training 11

5 EVALUATION ON RECOGNITION RESULTS **12**

6 SHADOW REMOVAL IN MONOCHROMATIC NATURAL IMAGES **14**

6.1 Linear Shadow Model 14

6.2 Parametric Illumination Change Model 14

6.3 Monochromatic Shadow Removal 15

6.4 Result 17

7 CONCLUSION **19**

8 REFERENCES **20**

List of Figures

1.1 Ambiguity of shadow recognition in monochromatic domain 1

2.1 Example of shadows in the dataset. 3

2.2 Mean Histograms of log illumination and local max 5

2.3 Two cues used to distinguish shadows from near-black objects 6

2.4 Feature visualization. 6

4.1 Examples of conditional probability. 10

6.1 Results of shadow removal 18

6.2 Results of shadow removal from monochromatic images. 18

1 INTRODUCTION

A shadow is a local change in illumination intensity and (often) illumination colour. Shadows are among the most noticeable effects on understanding the structure and semantic of a natural scene. A shadow occurs when an object partially or totally occludes direct light from a source of illumination. Shadows are one of the principal factors affecting computer vision performance in outdoor scenes. In image segmentation, the shadow boundaries may be confused with real object boundaries. In motion estimation, the computation of optical flow is susceptible to illumination changes caused by shadows. In supervision, for the influence of shadow, erroneous target may be tracked. Identifying and removing shadows in image is of great practical significance in image segmentation, and object tracking and supervision, etc.

While shadows can provide useful cues regarding scene properties such as object size, shape, and movement, their existence has also complicated various image recognition tasks for natural images, ranging from object recognition to scene parsing and removal[2]. While promising progress has been made in the literature, most existing works still retains on chromatic information for shadow recognition and removal.

One fundamental assumption for chromatic-based shadow recognition and removal is that the chromatic appearance of image regions does not change across shadow boundaries, while the intensity component of a pixel does. Such approaches work well given color images where object surfaces are still discernible inside the shadow. However, such schemes cannot be deployed to monochromatic images where shadows also prevalently exist. To the best of our knowledge, shadow detection and removal for monochromatic images is left unexploited. In particular, there exist two particular challenges:


(a) Diffusion (b) Specularity (c) Self-shading

Figure 1.1: Ambiguity of shadow recognition in monochromatic domain

- Objects in monochromatic images are more likely to appear black or near black, as shown in Figure 1.1. Such black objects exhibit similar intensity statistics to the shadow regions to be removed, which increases the difficulty in recognizing and removing shadows.

- Shadow recognition highly relies on separating changes in albedo from illumination change. Therefore, chromatic information is the most important cue, which is however missing in monochromatic images.

Despite the above difficulties, detecting and removing shadow in monochromatic images are still highly demanded in real-world applications. On one hand, color is not always available in all types of sensors, such as sensors tuned to specific spectra, or designed for specific usages like aerial, satellite, and celestial images, as well as sensors to capture images from very high dynamic scenes. On the other hand, it is widely known that humans are still able to identify shadows in monochromatic images , which inspires us to exploit the underlying principles in shadow modeling from monochromatic images.

In this paper, we present a learning-based approach for detecting and removing shadows from a single monochromatic image. The shadow recognition approach is based on boosted decision tree classifiers and integrated into a Conditional Random Field (CRF). Finally, we further present a Gaussian model based method for removing shadows, which uses less parameters than previous work. Different from the existing works, we target at capturing the unique statistics of shadows within monochromatic images by exploiting both shadow-variant and invariant cues in a data-driven, learning-based manner. Given sufficient training data, our approach is more general than previous works that exploited grayscale cues to remove general illumination effects[7], as quantitatively evaluated on vast natural images ranging from highways, parking lots to indoor office scenes. Comparing to the most related work proposed in, the training set improved or separating shading and reflectance changes is still limited to small objects, while our work focuses on shadows in large-scale outdoor scenes.

2 MONOCHROMATIC BASED FEATURE EXTRACTION

2.1 Dataset

We have built a database consisting of 355 images. For each image, the shadows have been handcraft-labeled at the pixel level, which are further cross-validated by a group of volunteers to ensure the label precision. In this database, approximately 50% images were collected from various outdoor environments such as campus and downtown areas. To ensure a wide variety of lighting conditions, we also collected images at different times throughout the day. The dataset includes additional 74 aerial images with a size of 257 x 257 from Overhead Imagery Research Dataset (OIRDS) [3] and another 54 image with a size of 640 x 480 from LabelMe dataset

Compared to the database used in our previous work, the shadows in the newly added 110 images are extremely challenging for recognition (see the last row in Figure 2). Our motivation is to check how good the learning methods introduced in this paper perform on this challenging scenes. Nevertheless, results from the previous work can be also used as a baseline for comparison purpose.

Figure 2.1: Example of shadows in the dataset.

2.2 Feature Extraction

Given a single monochromatic natural image, our goal is to identify those pixels that are associated with shadows. Without chromatic information, the features should be able to identify illumination, texture and odd order derivative characteristics. Rather than using pixel values alone, we also include the features from homogeneous regions generated by over-segmentation using intensity values. To reliably capture the cues across shadow boundaries, following [5], we collect statistics of neighboring pairs of shadow/non-shadow segments from all individual images in the dataset. These statistics are represented as histograms from shadow and non-shadow segments given equal bin centers.

We propose three types of features: shadow-variant features that describe different characteristics in shadows and in non-shadows; shadow-invariant features that exhibit similar behaviors across shadow boundaries and near-black features that distinguish shadows from near-black pixels. The above complimentary cues are further integrated to provide strong predictions of shadows jointly. For example, if the segment has invariant texture (compared with its neighbor) but has variant intensities, it is more likely that this segment belongs to shadows. Furthermore, each type of feature is characterized by scalar values to provide information relevant to shadow properties.

2.2.1 Shadow-Variant Features

- **Intensity:** Intensity is the measure of amount of light. Since shadows are expected to be relatively dark, we gather statistics (Figure 2.2 (a)) about the intensity of pixel values, based on which a histogram is calculated and further augmented with the averaged intensity value and the standard deviation inside a segment.

- **Local Max:** Local Max measures the maximum value of the neighbours. In a local patch, shadows have very low intensity values. Therefore, its local max value is expected to be small. On the contrary, non-shadows often have high intensity values and its local max value is expected to be large (Figure 2.2 (b)). We capture this cue by a local max completed at each segment.

- **Smoothness:** Shadows are often smoothed among neighborhood pixels. This is because shadows tend to suppress local variations on the underlining surfaces. The method subtracts a smoothed version from the original image. To measure the smoothness, we also use the standard deviation of the smoothed version from each segment. Smoothing is used to reduce noise or blurring.

- **Skewness:** We gathered several statistical variables (standard deviation, skewness and kurtosis) to find a mean value of 1.77 for shadows and -0.77 for non-shadows in skewness empirically. It indicates that the asymmetries in shadows and in non-shadows are different, which helps to locate shadows. This odd order statistic is also found to be useful in extracting reflectance and gloss from natural scenes.

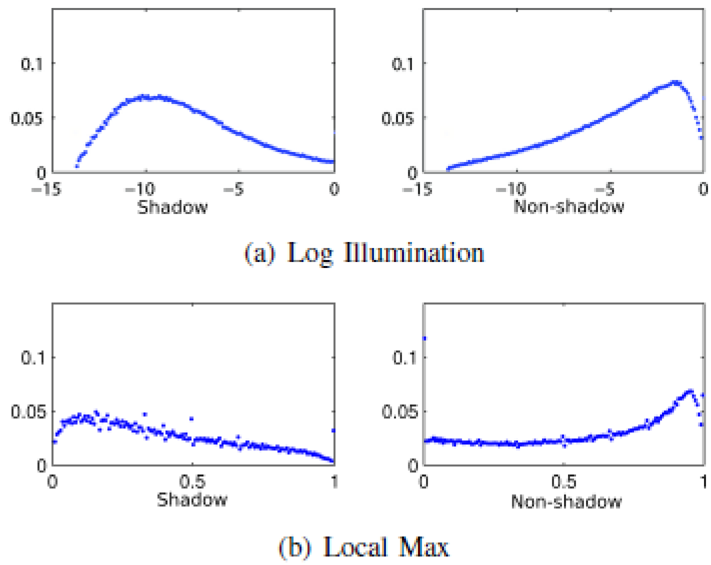
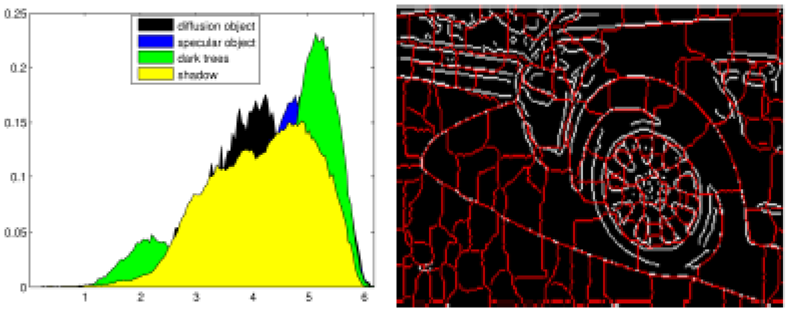


Figure 2.2: Mean Histograms of log illumination and local max

2.2.2 Shadow-Invariant Features

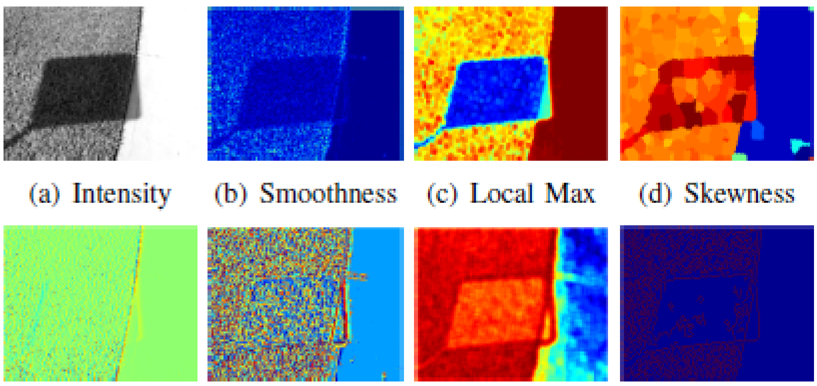
- **Gradient Similarity:** We have observed that transforming the image with a pixel-wise log transformation makes the shadow being an additive offset to the pixel values in the scene. Therefore, the distribution of image gradient values is expected to be invariant across shadow boundaries. To capture this cue, we measure the similarity between the distributions of several first order derivatives of Gaussian filters.

- **Texture Similarity:** We have also observed that the textural properties of surfaces are changed little across shadow boundaries. We measure the textural properties of an image region. The method filters a database of images with a bank of Gaussian derivative filters with 8 orientations and 3 scales, and then applies clustering to form 128 discrete centers. Given a new image, the texon is assigned as the histograms binned at these discrete centers. As shown in Figure 2.4(f) where the color indicates the texton index selected at each point, the distributions of textons inside and outside shadows are similar. The primary difference is that the distortion artifacts in the darker portions of the image lead to a slight increase in the number of lower-index textons, as indicated by more blue pixels.



(a) Entropy (b) Segment-based edge response

Figure 2.3: Two cues used to distinguish shadows from near-black objects



(a) Intensity (b) Smoothness (c) Local Max (d) Skewness

(e) Gradient (f) Texton (g) Entropy (h) Edge Map

Figure 2.4: Feature visualization.

2.2.3 Near-Black Features

As shown in Figure 2.2(a), the pixel intensity is a good indicator of the shadow, which is usually dark. Unfortunately, this heuristic alone cannot be used to reliably identify shadows as it fails when facing dark objects. We have found that objects with a dark albedo are some of the most difficult image regions to separate from shadows. Similarly, tree regions are also difficult because of the complex self-shading caused by the leaves. Due to the complexity of hand-labeling, the self-shading within a tree is not considered to be a shadow. We refer to these objects as near-black objects (see Figure 1.1). To correctly distinguish shadows from near-black objects, we introduce two additional features:

- **Discrete Entropy:** We have observed that the entropy of patches inside shadows differs from patches inside objects. The entropy is compute for each segment using the formula below:

$$Ei = \sum_{i \in w} pi \times log_2(pi), \quad (1)$$

  where denotes all the pixels inside the segment, pi is the probability of the histogram counts at pixel i.

  Figure 2.3(a) shows the distribution of the entropy values in patches from shadows and near-black objects respectively. These histograms show that the entropy of diffuse objects with dark albedos are relatively small, as most black objects are textureless in natural scenes. Figure 2.3(a) also shows histograms of the entropy of patches from specular objects(shown in blue) and image patches from trees(shown in green). There is a small mode at lower entropy values in the histogram from tree patches. Explanation lies in that some parts of trees are over-exposured to the sun, leading to saturated areas with low entropy.

- **Edge Response:** Another feature we found useful is the edge response, which is often small in shadows. Figure 2.3(b) shows an example where segments in shadows have near zero edge response, while that of specular object (the body of the car) has a strong edge response. We quantize this cue by summing up edge responses (computed using the Canny edge with a threshold setting to 0.01) inside a segment. Figure 2.4 shows a typical scene to visualize all 8 proposed features.

3 LEARNING TO RECOGNIZE SHADOW

From the dataset, we randomly selected 125 images as training data, and use the remaining 120 images as test data. We formulate the shadow detection problem as a per-pixel binary labeling problem, where every pixel is classified as either being inside shadow or not. Formally speaking, our goal is to assign $y_i \in \{-1,+1\}$, for all pixels i. A Logistic Random Field model is adopted, whose parameters are trained in the learning process to specify the decision boundary.

3.1 Logistic Random Field

The Logistic Random Field is essentially a logistic regression model, which is generalized to the conditional random field. If we regard each pixel as a sample, the probability that a pixel should be labeled as shadows +1 given its observation x is:

$$p(+1|x) = \sigma(wTf) \tag{2}$$

The vector w is a weighting vector that defines a decision boundary in the feature space and f is the feature of this pixel. The logistic function is:

$$\sigma(a) = 1(\div(1+exp(-a))) \tag{3}$$

Logistic regression can be viewed as converting the linear function $w^T f$, which ranges from $-\infty$ to ∞, into a probability, which ranges from 0 to 1.To generalize Logistic Regression to a random field model, we introduce response image,$r*$. $r*$ is a linear response map computed by $w^T f$. We express the probability of a pixel belong to shadow as l_i, which is calculated from the output of the logistic function

$$l_i = \sigma(r_{i^*}). \tag{4}$$

We use a pairwise CRF to model the labeling problem, which provides an elegant means to enforce local consistency and smoothness. The conditional distribution over labels l, given an input image o, has a form:

$$P(\mathbf{l}|\mathbf{o}) = \frac{1}{Z}\prod_i \phi_i(l_i|o_i) \prod_{<i,j>} \psi_{i,j}(l_i, l_j|o_i),$$

where $\prod_{<i;j>}$ denotes the product of all pairs of neighbouring pixels, both horizontal and vertical. The constant Z is the normalizing factor of the distribution. The LRF model discriminatively estimates the marginal distribution over each pixels label. Essentially, a Gaussian CRF is used to estimate the response image $r*$ which is passed in a pixel-wise fashion through the logistic function $\sigma()$. Formally, the likelihood of pixel i taking the label +1(denoted as l_i) is expressed as: $l_i = \sigma(r_i^*)$

where $r* = argmin, C(r;o)$ where C(r; o) is a quadratic cost function that captures the same types of propagation behavior desired from the CRF model. We then introduce how to formulate C(r; o) below.

3.2 Learning LRF to Recognize Shadows

The cost function C(r; o) is based upon interactions between the expected response r and the current observations o. It expresses the relationship between responses in a neighborhood jointly to recognize shadows. We define C(r; o) as

$$C(r;o) = \sum_i w(o;\theta 1)(r_i - 10)^2 + w(o;\theta 2)(r_i + 10)^2 + w(o;\theta 3) \sum_{<i;j>} (r_i - r_j)^2. \tag{5}$$

Each term ri refers to the entry pixel i in the response image $r*$. These two terms pull each pixel to either -10 or +10 in the response image $r*$. While the response image should technically vary between $-\infty$ to $+\infty$, ranging the values between -10 and +10 is sufficient in practice.

The weight w(o; θk), k$\in$ 1, 2, 3 at pixel i is given by $w(o_i;\theta_k) = \sum_{j \in N_f} \exp(\theta_k^j f_k^j)$

whereθ_k^j is the parameter vector associated with the jth feature fi for term k, N_f is the total number of features. By concatenating$\theta 1, \theta 2, \theta 3$into a vector θ, we show thatθ can be found by minimizing the negative log-likelihood of the training dataset. We define the negative log-likelihood of model response over a training image by $L(\theta)$ as

$$L(\theta) = \sum_i log(1 + exp(t_i r_i^*)) + \lambda \sum_{j \in N} \theta_j^2, \tag{6}$$

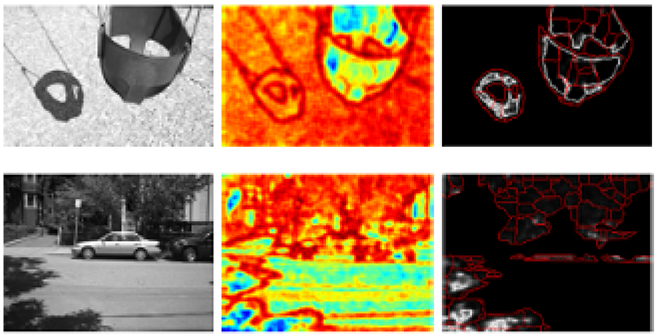
where t_i is the ground-truth probability of each pixellabeled to shadows, and the second term is a quadratic regularization term to avoid overfitting. λ is manually set to 10^{-4}. We use a standard gradient descent method to iteratively update the parameters θ which are all initialized at zero. Note that $L(\theta)$ depends onθ via r^*. The regularizer penalizes the model parameters uniformly, corresponding to imposing a uniform variance onto θ. This motivates a normalization of each type of the feature into [-1,1].

4 PRACTICAL ISSUES

In this section, we introduce two practical issues in our model. In Feature Boosting, we show that by integrating Boosted Decision Tree instead of the raw features, the recognition accuracy of LRF can be further improved. In Parallel Training, we introduce a low cost acceleration method (without using of cluster servers) to update the gradient.

4.1 Feature Boosting

The LRF model works well if the marginal distribution of a pixel belonging to shadows can be defined using the proposed features. Our results show that the linear classifier trained from the LRF achieves acceptable results (75.2%), but it also misclassifies many dark pixels such as near black objects (as shown in Figure 8). One reason is that the case to distinguish shadows from near black objects are very complex. Figure 2.5 shows such a case where a pixel is likely to be shadow according to the entropy cue varies in different images.



(a) Reference (b) Entropy Map (c) Mask

Figure 4.1: Examples of conditional probability.

To tackle this issue, Boosted Decision Tree (BDT) [4] is adopted, which builds a number of hierarchy trees by increasing the weights of the misclassified training samples. We sample each segment from the training dataset and learn the BDT estimators. For each pixel in the image, BDT estimator returns a probability distribution over the shadow labels. We use 40 weak classifiers, and ensemble their outputs as new features in the LRF model. In the experiment section, we show that by integrating BDT with LRF, we achieve the best shadow recognition rate. Note that, combining the labeling distributions from decision trees and random fields, where improved segmentation results are observed by enforcing the layout consistency.

4.2 Parallel Training

. We further implement a parallel version of learning parameters from the LRF model with two considerations:

- the memory requirement for loading all the training data exceeds a commodity PC memory.
- gradient updating requires tremendous computing resources.

Traditional methods to solve the first problem is to load a subset of training data or to compute the features on the fly. Online training approaches such as Stochastic Gradient Descent, can be also introduced to accelerate the convergence by updating the parameters from each training sample. In this paper, we handle the efficiency challenge from an architecture level, i.e. parallel training without special requirements of a cluster.

5 EVALUATION ON RECOGNITION RESULTS

To evaluate how well the classifiers can locate the shadows, we select a subset of the images (126) including high quality (can be easily labeled by people) shadows as the training dataset. We predict the shadow label at every pixel for another 124 images with ground truth shadow labels.

The pixels that identified as shadows are then compared with the shadow masks associated with each image. True Positives TP are measured as the number of pixels inside the mask. False Positives FP are measured as the number of pixels outside the mask. False Negatives FN are measured as the number of pixels falsely located outside the mask.

Our results are divided into two main groups. The first group includes three comparisons using monochromatic based features:

- different types of features
- different classification models
- different levels of over-segmentations

In the second group, we compared the performance between monochromatic and chromatic based features. For all the comparisons, we used the accuracy computed by $TP \div TP + FP + FN$ where TP is the true positive, FP is the false positive and FN is false negative. Overall, our results show that features expressing illumination, textural and odd derivative characteristics can successfully identify shadows. Our results show BDT integrated with LRF using two combined levels of segments achieves highest recognition rate at 38.1, which is 6.9% higher than LRF alone; 3.7% higher than Support Vector Machine (SVM) and 2.2% higher than BDT. Our results also show the chromatic features is superior than the monochromatic feature, and their combination contributes a 1.8% accuracy improvement.

It is also interesting to see that several combination of the features, such as all variant features with edge and all variant features wit Entropy and Edge, achieve best accuracy in our dataset. We would like to point out that although our dataset includes variance illumination and places where shadow is casted, but they are not complete and perfect. This leads to training results biased towards some features such as the edge and the entropy.

6 SHADOW REMOVAL IN MONOCHROMATIC NATURAL IMAGES

In this section, we introduce our approach to remove the recognized shadow in monochromatic natural images. Our method is based on the observation that illumination, caused from shadow, across shadow boundary is smoothly changed. We find that in the gradient domain, this leads to a Gaussian approximation to cancel the shadows. We also find that a Gaussian distributed assumption of the underlining texture across shadow boundary is helpful to obtain high quality shadow-free images.

6.1 Linear Shadow Model

We assume that all the images for shadow removal are taken in the raw format using DSLR cameras. We introduce two assumptions that make shadows separable from underlining surfaces in a linear model. We make our first assumption that the input shadow image, I(x, y), can be expressed as the product of a reflectance image, R(x, y), and the shadow image, S(x, y)

$$I(x,y) = R(x,y)S(x,y) \tag{7}$$

The reflectance of a scene is the material of the surfaces in the scene, and the shadow of a scene refer to the illumination that is occluded from objects in the scene. Considering the images in the log domain, the derivatives of the input image are the sum of the derivatives of the reflectance and the illumination. We find that significant shadow boundaries and reflectance edges is unlikely occur at the same point. Although this assumption is not consistent to occlusion shadows, where edges indicate both a shadow and a material edge[7], we find such cases are actually very few in natural scenes.

Consequently, we make the second assumption that every image derivative is either caused by shadow or reflectance. A similar assumption of shading and reflectance is reported in . This reduces the problem of assuming that the images x and y derivative changes on the shadow boundary are mainly from illumination changes, in our case, the shadows.

6.2 Parametric Illumination Change Model

We parameterize the linear shadow model for each pixel across the shadow boundary by sampling horizontal and vertical line segments to model the illumination change $\Delta I(x,y)$. Line segments provides easily parameterizable illumination profile, which is also used. The difference between our approach and we model $\Delta I(x,y)$ in the log derivative domain instead of the RGB domain as we observed that a simple Gaussian model, with fewer parameters, can well describe the shadow canceling process by favoring the smoothness inside a neighborhood. Therefore, our approach actually models $\nabla(\Delta I(x,y))$, the derivative of the illumination change model.

To sample the line segments, we first use Canny edge detection method to locate the shadow edge from a binary version of the shadow probability map obtained by the recognition process. And then, we dilate p pixels inward and outward given the Canny edge map and obtain boundary that is sufficiently across the shadow boundary. The dilation parameter p is chosen based on three factors: the position of the occlusion object, the direction of the illumination and the image size.

6.3 Monochromatic Shadow Removal

To cancel the monochromatic shadow effect in the gradient domain, we first fit a local Gaussian function, for each line segment, to simulate illumination changes. To further preserve the underlining texture across the shadow boundary, we formulate a Markov Random Field (MRF) model to globally optimize the parameters of the Gaussian function. Although other advanced techniques, such as Belief Propagation, Graph Cut and Variational Model can also be used to optimize the parameters, we find standard gradient descent works also well.

- **Shadow Canceling**

  For each line segment, the Gaussian function, f, is fitted by the gradients difference to the mean

$$f = A * exp(-(x-u)^2/(2*\sigma^2)) \tag{8}$$

  where u is the center of the derivative values computed as

$$u = 1 \div n \sum_i (y_i * x_i). \tag{9}$$

  yi and xi forms a local coordinate system with xi denotes the line segments horizontal coordinates and yi denotes its vertical coordinates. n is the total number of the pixels in the line segment. σ is the standard deviation of the distribution and A is a constant scaling the whole distribution. We initialize A = 0.1 and σ= 0.1.

- **Texture Preserving**

To preserve the texture across the shadow boundary after canceling the shadows, we assume the texture distribution is a Gaussian distribution in the gradient domain. We formulate a Markov Random Field (MRF) to optimize the parameters A, u, σ of the Gaussian function f by forcing the texture distribution as a Gaussian. In the MRF, each line segment is viewed as a node, and its two nearest line segments are edges. The energy function is formulated as

$$E = \sum_i \phi(y, f) + \lambda \sum_{<i;j>} \Psi(A, u,) \tag{10}$$

where y is the observed derivative values. The data term ϕ is designed as a function to measure the texture distribution of the cancelled derivatives yf.

For natural images, this texture distribution can be well expressed as a Gaussian distribution with mean u_0 and standard deviation σ_0.

$$\phi(y, f) = exp((y - fu_0)^2 \div 2\sigma_0^2) \tag{11}$$

In our approach, u_0, σ_0 are calculated using additional 6 pixels by extending the line segment on both sides. The smoothness term is simply a summation of L2 function that penalties consistency within neighboring area for each parameter A, u, σ. The weight from the smoothness term λ is set to 0.1 in all the tests.

We minimize the Equation (5) by standard gradient descent method. Although this method does not guarantee global optimum, we find high quality results can be obtained within 500 iterations of gradient updates. In all our tests, to recover the shadow-free image, we re-integrate the canceled derivatives yf (note that we do horizontal and vertical derivative canceling separately) by iteratively solving a Laplace equation given the horizontal and vertical derivatives for 5000 times.

Algorithm Gaussian-based Shadow Removal

1) Thresholding the shadow probability map returned from the recombination process.

2) Detecting the edges from the binary map.

3) Dilate and erode the edge map to sufficiently include shadow boundary.

4) For each pixel on the edge map, we sample one horizontal line segment and one vertical line segment from the boundary map.

5) In the log domain, for each line segment, we fit a Gaussian function given their gradients.

6) We formulate a MRF model to globally refine the underling texture by forcing its gradient as a Gaussian distribution and penalty the consistency in a neighboring area.

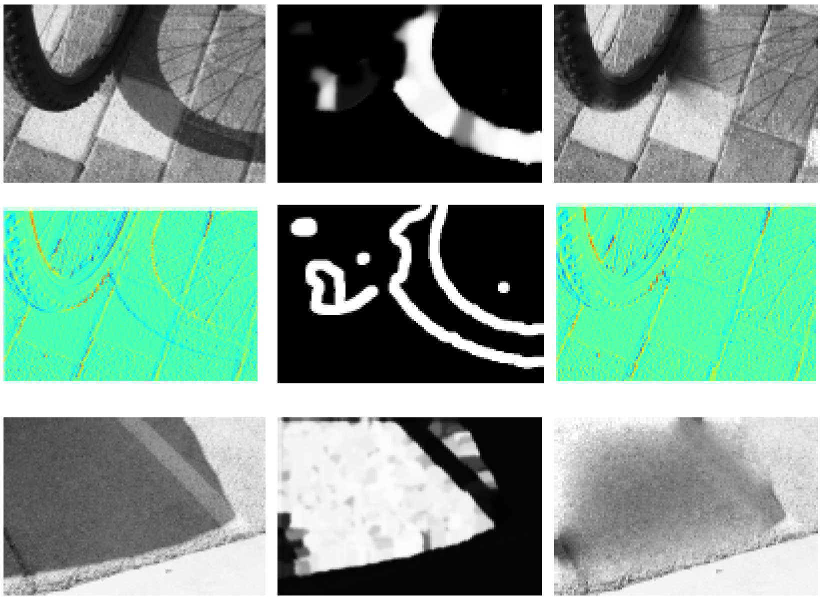
7) Finally, we recover the shadow-free image by solving a Laplace equation given the shadow canceled derivatives.

6.4 Result

We show part of the shadow removal results in Figure 6.2. The first four rows show results from the images taken by a Cannon Digital XTI camera, with a lens size of 50mm, and the last two rows show results from the overhead image library. We can see, overall, the local method works well while the global method performs better than local method. We also see saturation artifacts exist in local results (in 1rd and 2rd rows). We believe the artifacts near the right corner of 1rd row is because of the imperfection line sampling due to the acute angle.

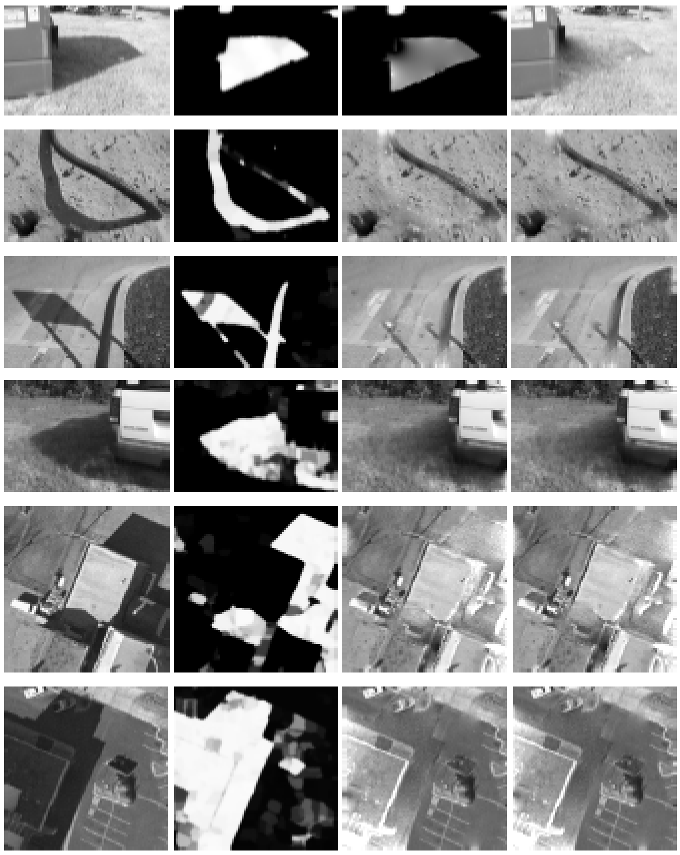
It is known that typical digital photos have more noise in shadow (dark) areas. This artifact can be clearly seen from two overhead images showed in Figure 6.2. By successfully removing the shadows, we show that the recovered shadow free images are more semantically useful missing context such as the cars is enhanced.

One interesting result is shown in the first row in Figure 6.1 where shadow and reflectance edge are very complex. Although this may violate our assumption that significant shadow boundary and reflectance edges are unlikely happen at the same point, the recovered result is still acceptable. Although our shadow removal approach works well for most of the scene, it still relies on the shadow recognition results, especially the correctness of the shadow boundary.We show a failed example in the second row in Figure 6.1.



(a) Shadow (b) Probability (c) Recovered

Figure 6.1: Results of shadow removal



(a) Shadow Images (b) Probability Map (c) Local Method (d) Gloabl Refinement

Figure 6.2: Results of shadow removal from monochromatic images.

7 CONCLUSION

We have proposed a system for recognizing and removal shadows from monochromatic natural images. Using the cues presented in this paper, our method can successfully identify shadows in real-world data where color is unavailable. We found that while single-pixel classification strategies work well, a Boosted Decision Tree integrated into a CRF-based model achieves the best results. We also found that a simple assumption of Gaussian illumination changes in the gradient domain across the shadow boundary can help to remove the shadows.

8 REFERENCES

[1] Z. Jiejie, S. Kegan G.G., M. Syed Z., and T. Marshall F. Learning based shadow recognition from monochromatic natural images. CVPR, 2010. 2, 6, 8, 93

[2]M. Bell and W. Freeman. Learning local evidence for shading and reflection. In ICCV, 2001.

[3]F. Tanner, B. Colder, C. Pullen, D. Heagy, C. Oertel, and P. Sallee. Overhead imagery research data set (oirds) an annotated data library and tools to aid in the development of computer vision algorithms, 2009.

[4] M. Collins, R. E. Schapire, and Y. Singer. Logistic regression, adaboost and bregman distances. In MACHINE LEARNING, pages 158169, 2000.

[5] R. Dror, A. Willsky, and E. Adelson. Statistical characterization of real-world illumination. Journal of Vision, 4(9):821837, 2004.

[6] G. Finlayson, C. Fredembach, and M. S. Drew. Detecting illumination in images. In ICCV, 2007.

[7] Y. Weiss. Deriving intrinsic images from image sequences. In ICCV, pages 6875, 2001

[8] http://ieeexplore.ieee.org/document/8003487/ [Accessed on:20-09-2017]