Dictionary Replacement for Single Image Restoration of 3D Scenes

T M Nimisha
ee13d037@ee.iitm.ac.in
M Arun
ee14s002@ee.iitm.ac.in
A N Rajagopalan
raju@ee.iitm.ac.in

Image Processing and Computer Vision Lab
Indian Institute of Technology, Madras
Chennai, India

Abstract

In this paper, we address the problem of jointly estimating the latent image and the depth/blur map from a single space-variantly blurred image using dictionary learning. The approach taken is based on the central idea of dictionary replacement viz. the sparse representation of a blurred image over a blurred dictionary is equivalent to that over a clean dictionary. While most of the dictionary-based deblurring methods consider planar scenes with space-invariant blur, we handle 3D scenes with space-variant blur caused by either camera motion or optical defocus. For a given blurred image, the dictionary blurred with the corresponding blur kernel provides the best representation with the least error. We formulate our problem of blur map and latent image estimation as a multi-label MRF and solve it using graph-cut. Experimental results on defocus as well as motion blur depict the effectiveness of our scheme.

1 Introduction

Technological advancements have revolutionized the imaging industry. New cameras in the market are now equipped with motion sensors, image stabilizer and high processing power. Yet it has not gone so far as to completely alleviate blurring distortions caused by camera motion or defocus effects. To capture a well-exposed frame, there is typically a compromise between aperture and exposure time. Increasing the aperture reduces the exposure time thereby minimizing motion blur during capture. However, it leads to reduced depth of field which can result in some regions being optically defocused in the image. On the other hand, a smaller aperture creates a larger depth of field but warrants increase in exposure leading possibly to motion blur.

Image deblurring has been widely studied to deal with defocus [19, 29] and motion blur [22, 25, 27]. Non-blind deblurring techniques assume that the distortion function is known and proceed with restoring the latent image using Weiner filtering, Richardson-Lucy [20], etc. On the other hand, blind techniques iteratively solve for the latent image as well as the blur kernels using priors (on both) for convergence. Since blind deblurring is highly ill-posed, most of these methods work with multiple input images that are degraded by different degrees of blur. Amongst the methods that deal with single image motion deblurring, most

Pages 32.1-32.12
DOI: https://dx.doi.org/10.5244/C.30.32

(a) (b) (c) (d) Zoomed-in patches

Figure 1: Result of our proposed dictionary based restoration of a defocused image: (a) Input image. Deblurred result using (b) [9] and (c) our method. (d) Our estimated blur map (higher intensity indicates higher blur). Zoomed-in patches of (a), (b) and (c), respectively.

consider space-invariant blur that are represented as mere convolutions [14]. These only hold for planar scenes with camera motion restricted to in-plane translations. Some of the issues with these methods are that they are quite complex and suffer from deblurring artifacts.

Dictionary-based deblurring methods were proposed to circumvent the explicit need for a regularizer. Defocus deblurring using conventional deconvolution methods is not quite effective due to zero crossings in the frequency spectra for a Gaussian aperture [29]. Dictionary-based deblurring methods have shown promising results in this direction, but are restricted to space-invariant blur. These methods work on the central idea that natural images have a sparse representation in an over-complete dictionary. These dictionaries can be learned using K-SVD [1]. Zhang et al. [28] propose an incremental deblurring method which can handle motion blur. A centralized sparse representation based deblurring method is discussed in [6]. Lou et al. [17] propose the idea of a blur-invariant sparse representation for the image to restore space-invariant Gaussian blurred image. However, all these methods are non-blind and require knowledge of the blur kernel. Coupled dictionary-based deblurring in [26] requires the blur dictionary to be learned along with the clear dictionary. For inputs blurred with multiple blur kernels, the dictionary pair need to be learned for each blur kernel separately, which makes it computationally complex. There are also dictionary-based blind deblurring methods which estimate the kernel and latent image iteratively [8, 9]. These are restricted to convolution models and fail in the case of 3D scenes where the blur varies with depth. Fig. 1 depicts the failure of convolutional model when the input image has space-variant defocus blur induced by depth variations. In the far focused image (Fig. 1(a)), the leaf in the front is defocused while the background is in focus. The dictionary-based method in [9] which assumes global blur cannot account for these blur variations (Fig. 1(b)).

There also exist works that deal with space-variant blur. Multiple image based depth recovery and deblurring from motion blurred scenes has been attemped in [12]. Paramanand et al. [18] propose a method to recover depth from a space-variantly blurred image pair but restricted to bilayer scenes. Depth from defocused images has been handled in [2, 29], but requires at least a pair of defocused images.

Different from previous works, we attempt here dictionary-based space-variant deblurring from a *single image*. The main challenge encountered in such a situation is that there are too many unknowns (the latent image intensity and depth/blur at each pixel location) to solve for from a single-blurred image. We elegantly exploit blur-depth dependency and sparseness of natural images within a dictionary to jointly estimate the depth and the latent image. Although the work in [10] has attempted a similar problem, it considers only camera motion blur. They solve it as a segment-wise depth estimation problem and with a user-assisted segmentation step where each segment corresponds to one depth layer. In contrast, our method is automatic (user-input-free). It does not require any initial segmentation and

works irrespective of the blur type. In the case of motion blur, we restrict camera motion to in-plane translations to preserve blur-depth dependency. The output of our method for the defocusing example is given in Fig. 1(c). Our method not only provides a good estimate of the latent image but also yields the blur/depth map of the scene (Fig. 1(d)).

The main contributions of our work are summarized below.

- To the best of our knowledge, this is the first work of its kind to attempt space-variant deblurring and depth estimation from a single image using the notion of dictionary replacement.
- Our generic dictionary based deblurring algorithm works irrespective of the blur type (motion/defocus) and for depth varying scenarios.

2 Blur-Invariant Sparse Representation

Sparsity model-based representations assume that any signal can be modeled as a linear combination of a few atoms from a pre-learned dictionary. The dictionary that sparsifies a given set of signals can be learned apriori using K-SVD techniques. Given a dictionary $D_{(n\times L)}$ learned from a set of training patches of size $(\sqrt{n},\sqrt{n})$, the image patch x can be represented using the sparse coefficient vector (α) as $x_{n\times 1} = D_{n\times L}\alpha_{L\times 1}$. The sparsest solution for α can be found by posing the problem as an l_0 minimization problem which is relaxed to l_1 minimization for simplicity. For the p^{th} image patch x_p, α_p can be estimated as

$$\arg\min_{\alpha_p} \quad ||R_p X - D\alpha_p||_2^2 + \lambda||\alpha_p||_1 \tag{1}$$

where $R_{p_{(n\times N)}}$ represents the matrix that extracts the p^{th} patch from the image $X_{(N\times 1)}$. This can be solved by Lasso [23] or by using greedy algorithms [24]. We use the greedy method of orthogonal matching pursuit. From the estimated sparse coefficients for each overlapping patch, the image X can be approximated as

$$X^* \approx \left(\sum_p R_p^T R_p\right)^{-1} \sum_p R_p^T D\alpha_p = D\circ\Lambda \tag{2}$$

where Λ is the set of sparse coefficients at all spatial locations and $\circ$ denotes patch-wise reconstruction and merging operation at all spatial locations.

An image degraded by space-invariant blur can be modeled by convolution as

$$Y = h\otimes X = h\otimes D\circ\Lambda = D_b\circ\Lambda \tag{3}$$

where h represents the blur kernel, Y is the observed blurred image, X is the latent image and $\otimes$ represents the 2D convolution operation. $h\otimes D$ is denoted as D_b, a blurred version of dictionary D, where each atom of D is blurred using the kernel h. This implies that when kernel h is known (as in non-blind deblurring), the signal X can be recovered from Y using the blur-invariant representation Λ. Dictionary replacement-based deblurring techniques, in fact, work on this principle. A visualization of the blur-invariant sparse representation is provided in Fig. 2.

Existing works on dictionary replacement [15, 17] deal with space-invariantly blurred images of fronto-parallel planar scenes where the blur is modeled as a simple convolution

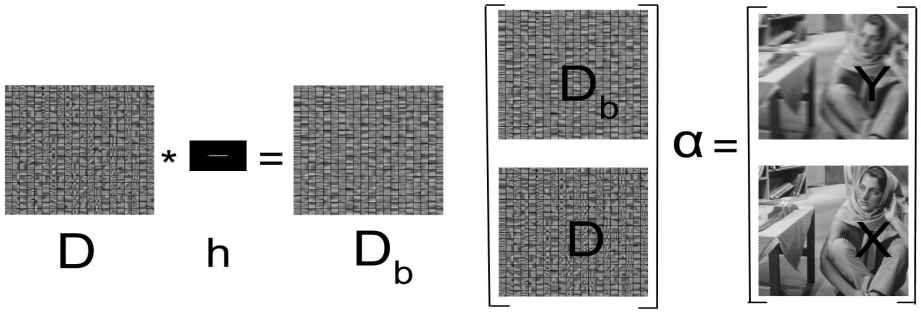


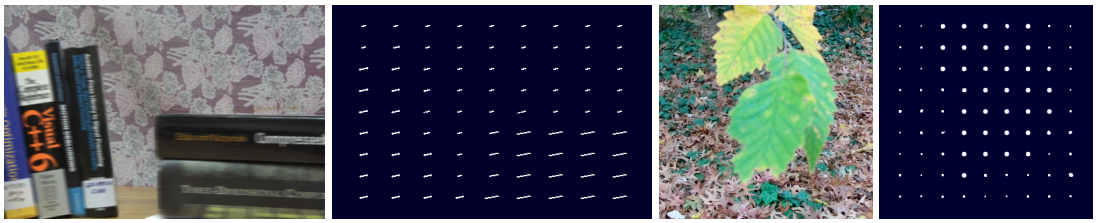
Figure 2: Dictionary replacement and blur-invariant sparse representation.

throughout the image with a single kernel. But the assumption of space-invariant blur is too restrictive and does not hold in most real scenarios. In this work, we extend the concept of dictionary replacement to space-variantly blurred images of 3D scenes where the blur and depth are inter-related. As discussed in the following section, we elegantly exploit this blur-depth dependency in our work to estimate the depth as well as a clean latent image.

3 Dictionary-based Depth Estimation and Deblurring

For a scene with varying depth, the blur experienced at each location is a function of depth. In the case of a 3D scene and when camera motion is restricted to in-plane translations, the objects closer to the camera experience larger motion when compared to the farther ones. This gives rise to space-variant blurring. For motion blur, the blur magnitude is inversely related to the depth from the camera center. In the case of defocus blur, if the object in the front is focused, then the blur-depth relation is direct, i.e., blurring increases with depth. For a far focused image, the relation is inverse. Our goal is to produce a clean all-focused image irrespective of the nature of blur.

The blur variations with depth in the case of motion blur and defocus effect are depicted in Fig. 3. The blur kernel at different depths are scaled versions of one another where the scaling depends on depth values. Since dictionary-based reconstruction is a patch-wise operation, we can effectively capture local changes within an image. We utilize this dictionary localization to estimate the depth at each location, as discussed subsequently.



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

Figure 3: Scaling of blur kernels with depth. (a-b) Motion blurred image and corresponding kernel grid. (c-d) Defocused image and corresponding kernel grid.

Let Y be the observed blurred image of a 3D scene and h_0 be the blur kernel corresponding to the most blurred region in the image. From the blur-depth relation, we know that the blur at any other position is a scaled down version of h_0. Hence, the problem of depth estimation boils down to estimating the scale of the blur kernel at each location. With the underlying idea that for a given sparsity, the dictionary blurred with the correct scale will

represent the blurred patch with minimum error, we formulate this as a Markov Random Field (MRF) problem. We employ α-expansion graphcut [4, 5, 11] to solve for the scale at each location by minimizing the following energy.

$$\arg\min_{i} \quad DC_i(k) + \sum_{k' \in \mathcal{N}} SC(\bar{i}_{k'}, i_k) \tag{4}$$

where $DC_i(k)$ is the data cost of assigning scale i to the pixel position k and SC denotes the smoothness cost. We assign an edge aware smoothness cost with value zero at the edges and $SC(\bar{i}, i) = (1 - r^{|\bar{i}-i|})$ at other pixel locations where r is chosen as 0.8.

Given an initial kernel h_0, we scale it using a range of values $S \in \{0, 1\}$ with a step-size 0.1 to produce kernels h_i's which represent the blur at different scales. With these blur kernels, we produce the blurred dictionary $D_{b_i} = h_i \otimes D$ from the pre-learned sparsifying dictionary $D_{(n \times L)}$. In our experiment, we design our dictionary with n =256 and L =1024. The dictionary is trained on 16×16 patches of sharp images from the dataset provided in [7]. It is worth mentioning here that unlike [26], we do not learn the blurred dictionaries for each blur kernel. The learning is done only once for the clear dictionary. All the blurred dictionaries are produced from this dictionary D.

For a particular sparsity, the dictionary D_{b_i} that gives the best representation (i.e., the least error) for the blurred image Y at each patch is used to choose the correct scale at that patch location. Thus, the data cost for assigning a scale i at pixel location k is formulated as

$$DC_i(k) = ||Y(k) - \bar{Y}_i(k)||_2^2 \quad \text{where} \quad \bar{Y}_i = D_{b_i} \circ \Lambda_i \tag{5}$$

where $\bar{Y}_i$ is an approximation of Y obtained from the sparse representation (Λ_i) and the blurred dictionary (D_{b_i}) corresponding to blur kernel h_i. Along with the data cost calculation, the deblurred result corresponding to i^{th} scale is estimated by dictionary replacement as $\hat{X}_i = D \circ \Lambda_i$.

It is to be noted that in the estimated $\hat{X}_i$, only the regions corresponding to blur level h_i will be restored properly. These deblurred results are stacked up in $\mathbf{X}$. Once the graphcut based scale is estimated for the entire image, the deblurred latent image $\bar{X}$ is formed by picking pixels from the stack $\mathbf{X}$ according to the scale. For a blurred image with two depth layers in Fig. 4(a), the estimated blur map is shown in Fig. 4(b). Figs. 4(c-d) show the deblurred results $\hat{X}_i$'s for the two scale values. $S = 0.2$ corresponds to background and $S = 1$ the foreground. The deblurred image (Fig. 4(c)) at scale 0.2 has the background intact, but the leaf in the foreground is not recovered. The deblurred result at scale 1 (Fig. 4(d)) has the leaf deblurred but the background has become over-deblurred. Hence, picking the pixels corresponding to the blur map from these two deblurred results yields the final deblurred result of Fig. 4(e).

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

Figure 4: Dictionary-based deblurring: (a) Input image. (b) Estimated blur map. (c-d) Deblurred results corresponding to two different scales. (e) Estimated latent image.

3.1 Initial kernel estimation

The above discussion assumes that the kernel h_0 corresponding to the highly blurred region is known. In the case of defocus or directional motion blur, the assumption of known kernel is reasonable since, h_0 can be well-approximated with a Gaussian or linear motion blur which can be appropriately scaled to produce blur at different depth levels. However, in the case of non-uniform blur, the initial kernel must be estimated since it can be of arbitrary shape. Hence, we include a kernel estimation step prior to deblurring.

Given the blurred image, we first estimate a highly blurred region using singular value features [21]. From this region, we pick a blurred patch P_b and estimate the kernel and the corresponding latent patch P using conventional blind deblurring as

$$\arg\min_{h,P} \quad ||P_b - h \otimes P||_2^2 + \lambda_1 ||h||_2^2 + \lambda_2 ||\nabla P||_1 \tag{6}$$

The above problem being non-convex is solved using an iterative approach, wherein at each iteration, one of the variables is fixed and the other is solved for. The estimated kernel h using (6) corresponds to h_0. This is then scaled down to produce the kernels at other depths.

3.2 Refinement of Latent Image

As discussed in Section 3, in our approach of dictionary-based deblurring, we pick pixels from the stack of deblurred images, each corresponding to a single scale value. In doing so, there is a possibility that when deblurring is carried out on the entire image with a single blur kernel to produce $\hat{X}_i$'s, the patches that do not match with the blur will introduce artifacts. This when averaged can result in reduction of quality at pixels that actually coincide with the blur at that scale. Hence, picking the pixels from the stack can produce mild degradations. To refine this result, we use the estimated deblurred image from dictionary learning ($\bar{X}$) as a prior for performing non-blind deblurring as

$$\arg\min_{X} \quad ||MX - Y||_2^2 + \lambda ||X - \bar{X}||_2^2 + \gamma ||\nabla X||_1 \tag{7}$$

Here M is a large but sparse matrix representing space-variant blurring. Each row of M is derived from the blur kernel at a pixel position which depends on the depth. We use alternating direction method of multipliers (ADMM) [3] to solve for X in (7). The smoothness of the recovered image is controlled by the value γ. High value of γ suppresses the edges and resultant image will be oversmoothened and lower value leads to ringing artifacts. Similarly, the value of λ decides how much details from the initial estimate of the latent image using dictionary has to be retained. In our experiments, we choose γ to be in the range $(0.002, 0.008)$ and λ to be in the range $(0.025, 0.05)$.

The complete algorithm of our proposed depth and latent image estimation from a single space variantly blurred image is given in Algorithm 1.

4 Experimental Results

To validate the effectiveness of our proposed method, we conducted experiments for both synthetic and real scenarios. In our experiments, we included space-variant blur caused by defocus effect as well as motion blur. We also provide comparisons with other relevant

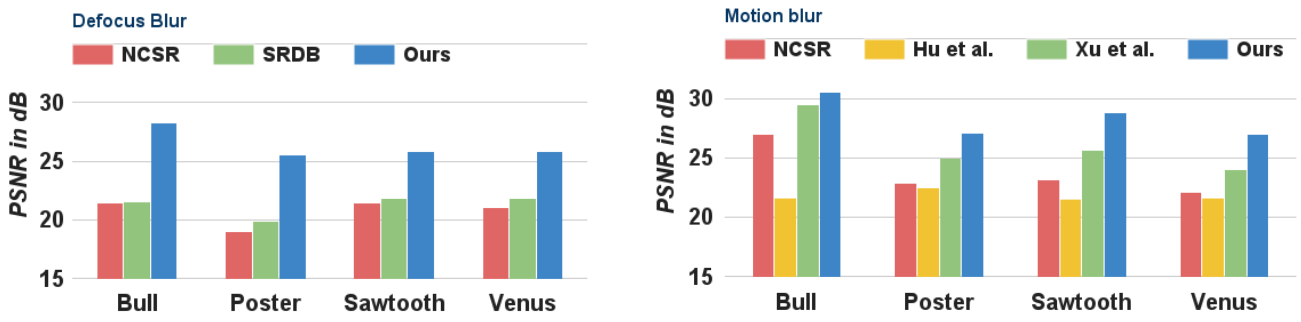
Algorithm 1 : Dictionary based deblurring and depth estimation.

Input: Single motion blurred image Y.
Output: Depth map and latent image X.

Estimate an initial kernel h_0 using (6).
for Each scale $i \in S$ **do**
  Produce blur kernel h_i at scale level i from h_0.
  Form the blurred dictionaries $D_{b_i} = h_i \otimes D$.
  Evaluate the sparse representation Λ_i of Y in D_{b_i} using (1).
  Reproduce Y as $\bar{Y}_i = D_{b_i} \circ \Lambda_i$.
  Form the data term by stacking up the error $||Y - \bar{Y}_i||_2^2$.
  Estimate latent image $\hat{X}_i$ as $\hat{X}_i = D \circ \Lambda_i$ and place it in a stack $\mathbf{X}$.
end for
Estimate the depth using graphcut.
Estimate the latent image $\bar{X}$ by picking pixels from $\mathbf{X}$ according to the estimated depth.
Refine the output using (7).

methods. Codes used for all comparisons are downloaded from the corresponding author's web page. Additional results and comparisons are provided in the supplementary material.

We first discuss synthetic examples for the Middlebury dataset. We took four inputs, one each from Bull, Poster, Sawtooth and Venus. These images were synthetically blurred using (i) Gaussian defocus blur, and (ii) directional uniform motion blur in accordance with their ground truth depth maps. In the case of defocus blur, we compared with dictionary-based non-blind methods SRDB [28] and NCSR [6]. Since these methods are restricted to space-invariant blur, deblurring the image with a single kernel introduces artifacts in the regions where the blur does not correspond to the same blur. This results in a lower PSNR when compared to our method as shown in Fig. 5(a). For non-blind space-invariant deblurring methods, we deblurred using ground truth (GT) kernels at various scales separately. The deblurred result with highest PSNR is used for comparison.



(a) (b)

Figure 5: Quantitative analysis for (a) defocus, and (b) motion blur.

For motion blur, we compared our method with dictionary-based NCSR [6] (non-blind), Hu et al. [9] (blind) and natural prior-based [27] deblurring techniques. Both the dictionary-based methods [6] and [9] handle only space-invariant blur and hence cannot deblur the space-variantly blurred image using a single global blur kernel. The method of Xu et al. [27] accommodates space-variant blur but cannot account for depth changes. In all the four sets of images, our method delivers the best result and this is reflected in the PSNR values plotted in Fig. 5(b). Due to space constraint, qualitative discussion of these results are given in the

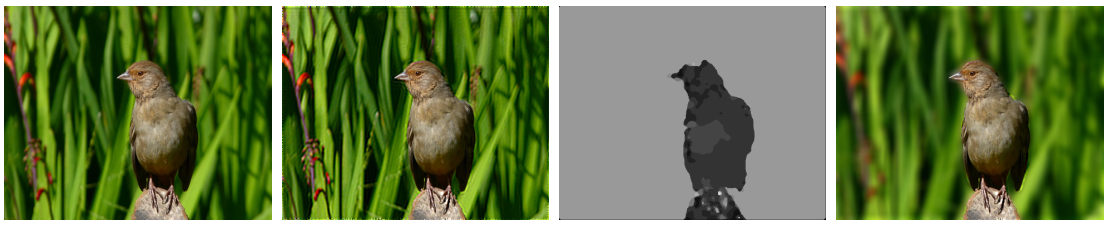
supplementary material.

Real Examples (Defocus blur): Fig. 6(a) shows an image of a 3D scene with the foreground object blurred and background layer focused. The image was taken from the dataset provided in [16]. Since the image is blurred due to defocus effect, we assume a Gaussian kernel of size $(15,15)$ and σ=3 as our initial kernel h_0. We then form the data cost for the graph-cut using scaled kernel dictionaries following steps 3 to 8 of Algorithm 1. Finally, the blur map is estimated using graph-cut, and the corresponding latent image is formed and refined using (7). The deblurred result and the estimated blur map using the proposed method are given in Figs. 6(b) and (c).

We next demonstrate refocusing as an application of the proposed work. The input image has the background in focus and the bottle in the foreground is blurred. From the estimated depth and known kernel, we blurred the background such that the foreground blur now reverts to the background. The refocused image is shown in Fig. 6(d). Here the foreground is clear and legible whereas the background appears blurred.

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

Figure 6: Refocusing application: (a) Input image along with zoomed-in patches. (b) Deblurred result. (c) Blur map. (d) Refocused image along with zoomed-in patches.



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

Figure 7: Blur magnification: (a) Input image focused in the front. (b) Deblurred result. (c) Estimated blur map. (d) Background blur magnified.

The next example on defocus blur shows an application of our method for blur magnification to increase the aesthetics of an image. Specifically, the background is over-blurred to bring out the beauty of the foreground. In Fig. 7(a), the input image already has blur in the background. We deblur the image (Fig. 7(b)) and estimate the blur map (Fig. 7(c)). Using this blur map, we create a trimap and segment out the foreground alone using alpha matting [13]. The background is then subjected to high defocus blur, and the foreground is placed back to get the blur-magnified image of Fig. 7(d). In the resultant image, the bird projects out from the background.

Real examples (Motion blur): A real indoor example of a 3-layered scene taken with a DSLR camera is shown in Fig. 8. The depth dependency of motion blur is clearly visible from the input image of Fig. 8(a). The books in the front are heavily blurred when compared to the textured wall in the rear end. The output latent image and the estimated depth using our proposed algorithm are shown in Figs. 8(c) and (d), respectively. Fig. 8(b) shows the

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

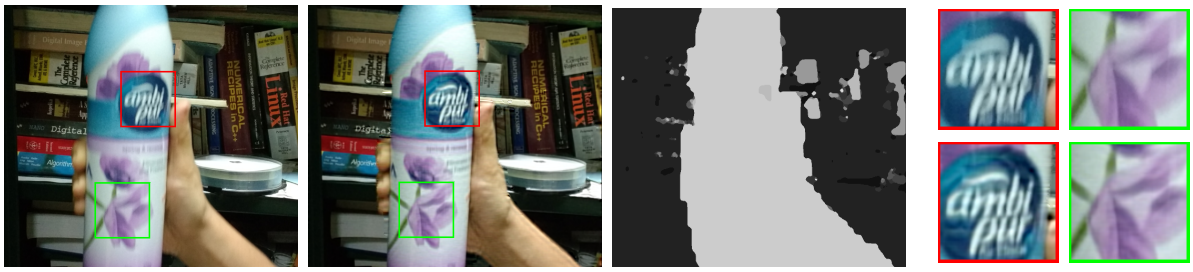
Figure 8: (a) Blurred input image. Deblurred using (b) [27] and (c) our method. Zoomed-in patches are also shown. (d) Estimated depth map.

result obtained from [27] which is not satisfactory since it does not handle depth variations. It is quite clear from the zoomed-in patches that our method outperforms [27].

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

Figure 9: Comparison with Hu et. al [10]: (a) Input image. (b) Deblurred result of [10]. (c) Our result. (d) Zoomed-in patches from (a), (b) and (c), respectively.

The method in [10] addresses a similar problem of depth estimation and deblurring from a single image based on segmentation. Since the code is not available, we ran our algorithm on the image provided in that paper. The image in [10] has coloured boxes to indicate patches. Hence, we cropped the portion without these boxes for testing. The cropped input blurred image along with the result of [10] and our output are given in Fig. 9. Each row of Fig. 9(d) corresponds to the respective zoomed-in patches from (a), (b) and (c). Even though our final deblurring has introduced slight ringing at the borders, the overall deblurring is better than that of [10]. The zoomed-in patches from the sign boards clearly reveal the deblurring efficiency of our proposed method over [10].



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

Figure 10: Object motion: (a) Blurred input image. (b) Our result. (c) Recovered blur map. (d) Zoomed-in patches from (a) and (b), respectively.

We finally considered the case of blur due to object motion. As our method is local and does not assume any global camera motion constraint, it performs well in these scenario too. However, the object blur should be consistent throughout the blurred region. Consider the example in Fig. 10. The input image is taken with a static camera and the blur is due to the

moving foreground object alone. The estimated blur map here corresponds to the velocity of the object in motion (rather than depth values). The deblurred output of our method is shown in Fig. 10(b). The estimated scale can be further used for moving object segmentation.

4.1 Computational Complexity

Creating a clean dictionary D using K-SVD is a time consuming process. But this is done apriori and no more learning is done in the entire algorithm. Hence this computation is only a one time process. Apart from this the computations involved for each iteration are, L convolutions for creating the blurred dictionary D_b, L correlations (computational complexity of $O(Ln)$) with the input signal for finding the sparse representation of the image in D_b, nL^2 and $n(L-1)$ scalar multiplication and additions to project back the sparse representation in D to recover the clean patch. In addition to this, the graphcut framework using α expansion to find the blur scale at each pixel position requires a computational time of $O(m|S|)$ where m is total image size and $|S|$ is the number of scales. Run-time of our algorithm for an image size 432×432 using an unoptimized MATLAB code on a 3.6GHz PC with 16GB RAM is around 156 seconds.

5 Conclusions

In this work, we addressed the challenging problem of deblurring a space-variantly blurred image from a single observation. We exploited the concept of dictionary representation in the blurred domain to estimate scaled kernels at each pixel location thereby arriving at the depth map of the image. An impulse kernel dictionary reconstruction provided the underlying clean latent image. Comparisons reveal that our method outperforms the state-of-the-art. As future work, we plan to handle even rotational motion which would entail decoupling of translation from rotation to exploit scaling of kernel as a function of depth.

References

[1] M. Aharon, M. Elad, and A. Bruckstein. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *Signal Processing, IEEE Transactions on*, 54(11):4311–4322, 2006.

[2] Arnav V Bhavsar and A. N Rajagopalan. Depth estimation with a practical camera. In *BMVC*, pages 1–11, 2009.

[3] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[4] Yuri Boykov and Vladimir Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(9):1124–1137, 2004.

[5] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23 (11):1222–1239, 2001.

[6] Weisheng Dong, Lei Zhang, and Guangming Shi. Centralized sparse representation for image restoration. In *ICCV*, pages 1259–1266. IEEE, 2011.

[7] Li Fei-Fei and Pietro Perona. A bayesian hierarchical model for learning natural scene categories. In *CVPR*, volume 2, pages 524–531. IEEE, 2005.

[8] Y. Zhang H. Zhang, J. Yang and T. S Huang. Sparse representation based blind image deblurring. In *Proc. IEEE ICME*, pages pp. 1–6, 2011.

[9] Zhe Hu, Jia-Bin Huang, and Ming-Hsuan Yang. Single image deblurring with adaptive dictionary learning. In *ICIP*, pages 1169–1172. IEEE, 2010.

[10] Zhe Hu, Li Xu, and Ming-Hsuan Yang. Joint depth estimation and camera shake removal from single blurry image. In *CVPR*, pages 2893–2900. IEEE, 2014.

[11] Vladimir Kolmogorov and Ramin Zabin. What energy functions can be minimized via graph cuts? *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(2): 147–159, 2004.

[12] Hee Lee and Kuoung Lee. Dense 3D reconstruction from severely blurred images using a single moving camera. In *CVPR*, pages 273–280, 2013.

[13] A. Levin, D. Lischinski, and Y. Weiss. A closed-form solution to natural image matting. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(2):228–242, 2008.

[14] A. Levin, Y. Weiss, F. Durand, and W. T Freeman. Understanding and evaluating blind deconvolution algorithms. In *CVPR*, pages 1964–1971. IEEE, 2009.

[15] Haisen Li, Yanning Zhang, Feng Duan, and Yu Zhu. Blind image deblurring based on dictionary replacing. In *Intelligent Science and Intelligent Data Engineering*, pages 357–364. Springer, 2011.

[16] Nianyi Li, Jinwei Ye, Yu Ji, Haibin Ling, and Jingyi Yu. Saliency detection on light field. In *CVPR*, pages 2806–2813. IEEE, 2014.

[17] Y. Lou, A. L. Bertozzi, and S. Soatto. Direct sparse deblurring. *Journal of Mathematical Imaging and Vision*, 39(1):1–12, January 2011. ISSN 0924-9907.

[18] C Paramanand and A. N Rajagopalan. Non-uniform motion deblurring for bilayer scenes. In *CVPR*, pages 1115–1122. IEEE, 2013.

[19] A. N Rajagopalan and S. Chaudhuri. An MRF model-based approach to simultaneous recovery of depth and restoration from defocused images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(7):577–589, 1999.

[20] William Hadley Richardson. Bayesian-based iterative method of image restoration. *JOSA*, 62(1):55–59, 1972.

[21] Bolan Su, Shijian Lu, and Chew Lim Tan. Blurred image region detection and classification. In *Proceedings of the 19th ACM international conference on Multimedia*, pages 1397–1400. ACM, 2011.

[22] Yu-Wing Tai, Ping Tan, and Michael S Brown. Richardson-lucy deblurring for scenes under a projective motion path. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(8):1603–1618, 2011.

[23] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society*, pages 267–288, 1996.

[24] Joel A Tropp. Greed is good: Algorithmic results for sparse approximation. *Information Theory, IEEE Transactions on*, 50(10):2231–2242, 2004.

[25] Oliver Whyte, Josef Sivic, Andrew Zisserman, and Jean Ponce. Non-uniform deblurring for shaken images. *International journal of computer vision*, 98(2):168–186, 2012.

[26] Shiming Xiang, Gaofeng Meng, Ying Wang, Chunhong Pan, and Changshui Zhang. Image deblurring with coupled dictionary learning. *International Journal of Computer Vision*, 114(2-3):248–271, 2015.

[27] Li Xu, Shicheng Zheng, and Jiaya Jia. Unnatural l0 sparse representation for natural image deblurring. In *CVPR*, pages 1107–1114, 2013.

[28] Yanning Zhang and Yanning Zhang. Sparse representation based iterative incremental image deblurring. In *ICIP*, pages 1293–1296. IEEE, 2009.

[29] Changyin Zhou, Stephen Lin, and Shree K Nayar. Coded aperture pairs for depth from defocus and defocus deblurring. *International journal of computer vision*, 93(1):53–72, 2011.