Learning a Convolutional Neural Network for Non-uniform Motion Blur Removal

Jian Sun[1], Wenfei Cao[1], Zongben Xu[1], Jean Ponce[2,3]
[1]Xi'an Jiaotong University. [2]École Normale Supérieure / PSL Research University. [3] Willow Team, ENS/Inria/CNRS UMR 8548.

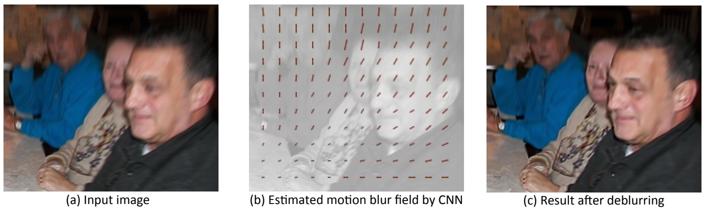

Figure 1: An example illustrating our approach. Given an image with non-uniform motion blur (left). We first estimate the field of non-uniform motion blur kernels by a convolutional neural network (middle), then deconvolve the blurred image (right).

Non-uniform deblur has been a challenge in computer vision. Methods in [4, 5, 8] work on non-uniform blur caused by camera rotations, in-plane translations or forward out-of-plane translations. They are effective for removing non-uniform blur consistent with these motion assumptions. Another category of approaches works on non-uniform motion blur caused by object motion. They estimate blur kernels by analyzing image statistics [7], blur spectrum [1], or with a learning approach using hand-crafted features [2]. Other approaches [6, 9] jointly estimate sharp image and blur kernels using sparsity prior.

In this work, we propose a novel deep learning-based approach to estimating non-uniform motion blur, followed by a patch statistics-based deblurring model adapted to non-uniform motion blur, as illustrated in Fig. 1. We estimate the probabilities of motion kernels at the patch level using a convolutional neural network (CNN) [3], then fuse the patch-based estimations into a dense field of motion kernels using a Markov random field (MRF) model. To fully utilize the CNN, we propose to extend the candidate motion kernel set predicted by CNN using an image rotation technique. Due to the strong feature learning power of CNNs, we can well predict the challenging non-uniform motion blur that can hardly be well estimated by the state-of-the-art approaches.

We next briefly introduce our approach. Given a blurry image I, we represent the local motion blur kernel at an image pixel $p \in \Omega$ (Ω is the image region) by a ***motion vector*** $\mathbf{m_p} = (l_p, o_p)$, which characterizes the length and orientation of the motion field in p when the camera shutter is open. Each motion vector determines a motion kernel with non-zero values only along the motion trace. The blurry image can be represented by $I = k(M) * I_0$, i.e., the convolution of a latent sharp image I_0 with the non-uniform motion blur kernels $k(M)$ determined by the motion field $M = \{\mathbf{m_p}\}_{p\in\Omega}$.

To predict motion blur kernels (or equivalently, the motion vector) at the patch level, we decompose the image into overlapping patches of size 30×30. Given a blurry patch Ψ_p centered at pixel p, we aim to predict the probabilistic distribution of motion kernels $P(\mathbf{m} = (l, o)|\Psi_p)$, for all $l \in S^l$ and $o \in S^o$, S^l and S^o are the sets of motion lengths and orientations respectively. We call this distribution as ***motion distribution***. In our implementation, we discretize the range of motion length into 13 samples from $l = 1$ to 25 with interval of two, and discretize the range of motion orientation $[0, 180^\circ)$ into 6 samples from 0° to 150° with interval of 30°.

Taking the problem of motion kernel estimation as a learning problem, we utilize convolutional neural network to learn the effective features for predicting motion distributions. The CNN is constructed as shown in Fig. 2. To train the CNN model, we generate a large set of training data $T = \{\Psi_k, \mathbf{m}_k\}_{k=1}^K$, which are composed of around 1.4 million blurry patch / motion kernel pairs. Using Caffe [3], we train the CNN model in one million iterations with batches of 64 patches in each iteration. Because the final layer is a soft-max layer, we can predict the probabilities of motion kernels given an observed blurry patch Ψ as

This is an extended abstract. The full paper is available at the Computer Vision Foundation webpage.

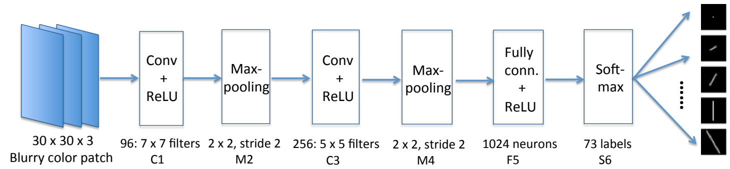

Figure 2: Structure of CNN for motion kernels prediction. It is composed of 6 layers of convolutional layers and fully connected layers. It outputs the probability of each candidate motion kernel using soft-max layer.

$$P(\mathbf{m} = (l, o)|\Psi) = \frac{\exp((w_c^{S6})^T \phi_{F5}(\Psi))}{\sum_n \exp((w_n^{S6})^T \phi_{F5}(\Psi))}, \qquad (1)$$

where w_c^{S6} is the vector of weights on neuron connections from F5 layer to the neuron in $S6$ layer representing the motion kernel (l, o), c is the index of (l, o) in S. $\phi_{F5}(\Psi)$ is the output features of $F5$ layer of a blurry patch Ψ, which is a 1024-dimensional feature vector.

Extending the Motion Kernel Set of CNN. Our learned CNN model can predict the probabilities of 73 candidate motion kernels that were used as labels in CNN training. We further extend this motion kernel set to enable the prediction for motion kernels beyond them. We achieve this goal by feeding the original patch and its rotated versions into CNN, then we can estimate the probabilities of motion kernels that may be not belonged to the motion kernel set of CNN. By this rotation technique, our CNN can predict motion distribution of image patch over 361 motion kernel candidates.

Dense Motion Field Estimation by MRF. Given the motion distribution at patch level, we then estimate the dense motion field $M = \{\mathbf{m}_p = (l_p, o_p)\}_{p\in\Omega}$ over image I by optimizing the following MRF model:

$$\min_M \sum_{p\in\Omega} [-C(\mathbf{m}_p = (l_p, o_p)) + \sum_{q\in N(p)} \lambda[(u_p - u_q)^2 + (v_p - v_q)^2], \qquad (2)$$

where (u_p, v_p) and (u_q, v_q) are motion vectors $\mathbf{m}_p$ and $\mathbf{m}_q$ in Cartesian coordinates, $C(\mathbf{m}_p)$ is the confidence of motion kernel $\mathbf{m_p}$ at pixel p . $N(p)$ is the neighborhood of p. By minimizing the energy function, we can estimate a smooth motion vector field over the blurry image.

Non-Uniform Motion Deblurring. With the dense non-uniform motion kernels estimated by CNN, we deconvolve the blurry image by adapting the uniform deconvolution approach in [10] to the non-uniform deconvolution problem. The non-uniform deconvolution is modeled as optimizing: $\min_I \frac{\lambda}{2}||k(M) * I - O||_2^2 - \sum_{i\in\Omega} \log(P(R_i I))$ where O is the observed blurry image, R_i is an operator to extract the patch located at i from an image. $P(\cdot)$ is the prior distribution of natural image patches, which is modeled as a Gaussian mixture model learned from natural image patches [10]. By optimizing this energy function, we can estimate the final deblurred image.

[1] A. Chakrabarti, T. Zickler, and W.T. Freeman. Analyzing spatially-varying blur. In *CVPR*, pages 2512–2519, 2010.

[2] Florent Couzinié-Devy, Jian Sun, Karteek Alahari, and Jean Ponce. Learning to estimate and remove non-uniform image blur. In *CVPR*, 2013.

[3] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. *arXiv:1310.1531*, 2013.

[4] A. Gupta, N. Joshi, C. Lawrence Zitnick, M. Cohen, and B. Curless. Single image deblurring using motion density functions. In *ECCV*, 2010.

[5] M. Hirsch, C.J. Schuler, S. Harmeling, and B. Scholkopf. Fast removal of non-uniform camera shake. In *ICCV*, 2011.

[6] Tae Hyun Kim and Kyoung Mu Lee. Segmentation-free dynamic scene deblurring. In *CVPR*, 2014.

[7] A. Levin. Blind motion deblurring using image statistics. In *NIPS*, 2007.

[8] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce. Non-uniform deblurring for shaken images. *IJCV*, 98(2):168–186, 2012.

[9] Li Xu, Shicheng Zheng, and Jiaya Jia. Unnatural l0 sparse representation for natural image deblurring. In *CVPR*, 2013.

[10] Daniel Zoran and Yair Weiss. From learning models of natural image patches to whole image restoration. In *ICCV*, 2011.