MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Robust Subspace Learning for Motion Deblurring in Images

Lopez, O.; Mansour, H.

TR2018-208 March 29, 2019

Abstract

We developed a framework for motion deblurring that finds a low rank approximation of the sharp image patches from a collection of blurry image patches. The approach relies on the notion that each blurry patch has undergone a different type of blur compared to the other patches. As a result, the low rank approximation of the group of patches recovers a sharp image component without the misalignment artifacts associated with a rank one approximation.

MERL website

Robust Subspace Learning for Motion Deblurring in Images

Oscar López and Hassan Mansour
Mitsubishi Electric Research Laboratories

March 13, 2019

Abstract

We developed a framework for motion deblurring that finds a low rank approximation of the sharp image patches from a collection of blurry image patches. The approach relies on the notion that each blurry patch has undergone a different type of blur compared to the other patches. As a result, the low rank approximation of the group of patches recovers a sharp image component without the misalignment artifacts associated with a rank one approximation.

1 Methodology

We organize our desired collection of "blur-free" images $\{x_k\}_{k=1}^N \subset \mathbb{R}^{n\times m}$ as the columns of a low-rank matrix $X \in \mathbb{R}^{nm\times N}$, i.e., using matlab notation

$$X(:,k) = \text{vec}(x_k) \qquad k \in \{1,2,...,N\}.$$

Similarly, organize our given blurry observations (which we assume are "aligned") $\{b_k\}_{k=1}^N \subset \mathbb{R}^{n\times m}$ as the columns of a matrix $B \in \mathbb{R}^{nm\times N}$,

$$B(:,k) = \text{vec}(x_k) \qquad k \in \{1,2,...,N\}$$

and our desired blur kernels $\{h_k\}_{k=1}^N \subset \mathbb{R}^{n\times m}$ as the columns of $H \in \mathbb{R}^{nm\times N}$,

$$H(:,k) = \text{vec}(h_k) \qquad k \in \{1,2,...,N\}.$$

Then if $\mathcal{F} \in \mathbb{C}^{nm\times nm}$ is the 2D FFT that acts on a vectorized 2D image, our measurements can be modeled as

$$B = \mathcal{F}^{-1}\left((\mathcal{F}H) \circ (\mathcal{F}X)\right),$$

where $\circ$ denotes the Hadamard matrix product. In this was we may attempt to

approximate X, H as

$$(X,H) \approx \text{argmin}_{Z,W\in\mathbb{R}^{nm\times N}} \ \ \|Z\|_* + \gamma\|W\|_1 + \frac{\lambda}{2}\|B - \mathcal{F}^{-1}\left((\mathcal{F}W)\circ(\mathcal{F}Z)\right)\|_F^2 \tag{1}$$

$$= \text{argmin}_{Z,W\in\mathbb{R}^{nm\times N}} \ \ \|Z\|_* + \gamma\|W\|_1 + \frac{\lambda}{2}\|\mathcal{F}B - (\mathcal{F}W)\circ(\mathcal{F}Z)\|_F^2, \tag{2}$$

where $\|W\|_1 = \sum_{i=1}^{nm}\sum_{j=1}^{N}|W_{ij}|$ is the ℓ_1 norm acting on matrices.

Our prior on the images is the low-rank structure of X, while we retain our sparsity prior on the kernels. In particular notice that this approach is very similar to RASL, but we posses the advantage of linearity in the constraint

$$\mathcal{F}B = (\mathcal{F}H)\circ(\mathcal{F}X),$$

i.e., with H fixed the operator $(\mathcal{F}H)\circ\mathcal{F} : \mathbb{C}^{nm\times N} \mapsto \mathbb{C}^{nm\times N}$ is linear. This is due to the nice structure of our convolution model in Fourier domain. In contrast the work in RASL requires the introduction of the nonlinear operator τ, which complicates the optimization process.

We may therefore proceed as in the RASL work, but without the need for "iterative linearization" for the operator updates (the blur kernels in this case, rather than the image deformations). We may solve (1) via augmented Lagrange multiplier methods (as in the RASL paper) or by alternating minimization as we have been doing.

2 Preliminary Results

The following are results of the methodology above for the TUM dataset link and the UBC dataset link.

Program (1) is implemented via Alternating minimization, i.e., with initial kernel estimates H^0 we solve

$$X^1 := \text{argmin}_{Z\in\mathbb{R}^{nm\times N}} \ \ \|Z\|_* \ \text{ s.t. } \ \frac{\lambda}{2}\|\mathcal{F}B - (\mathcal{F}H^0)\circ(\mathcal{F}Z)\|_F^2 \le \sigma, \tag{3}$$

to obtain our first estimate of the images. We then update our kernel estimates via

$$H^1 := \text{argmin}_{W\in\mathbb{R}^{nm\times N}} \ \ \gamma\|W\|_1 \ \text{ s.t. } \ \frac{\lambda}{2}\|\mathcal{F}B - (\mathcal{F}W)\circ(\mathcal{F}X^1)\|_F^2 \le \sigma, \tag{4}$$

and so on to obtain final signal estimates (X^K, H^K) given $K \in \mathbb{N}$ as a parameter to indicate the number of alternations to update each variable.

We begin with the face data from the TUM dataset:

We now run this methodology on the full TUM and UBC datasets. Here, the full image is divided into 64×64 size patches. A block matching algorithm is then run for each of these patches to find the best matching patches in the

Figure 1: Total of 8 blurry input images of individual's face, ran with 5 alternations. Here are shown the least blurry output and input (smallest ℓ_1, ℓ_2 norm ratio). Our methodology produces a sharper output, arguably better than the blurry input images and an improvement relative to our previous deblurring methodologies that use TV and TV2 regularization.

neighboring frames. These set of blurry images are then deblurred via our methodology.

As expected, we see favorable results on the datasets that consists of frame sequences with multiple motions of blur (Figures 2, 3, 4).

On the other hand, the methodology does not work so well on frame sequences whose blur effect derives from a single direction of motion (Figures 5, 6). In this case, the observations do not provide enough information of the scene and effectively becomes an ill-posed problem, along the lines of blind deconvolution from a single input image. However, while we have not satisfyingly removed the effect of blur in these examples, the output does seem much sharper than the input image. Arguably, in these cases, our output is still favorable for post-processing in comparison to the blurry input.

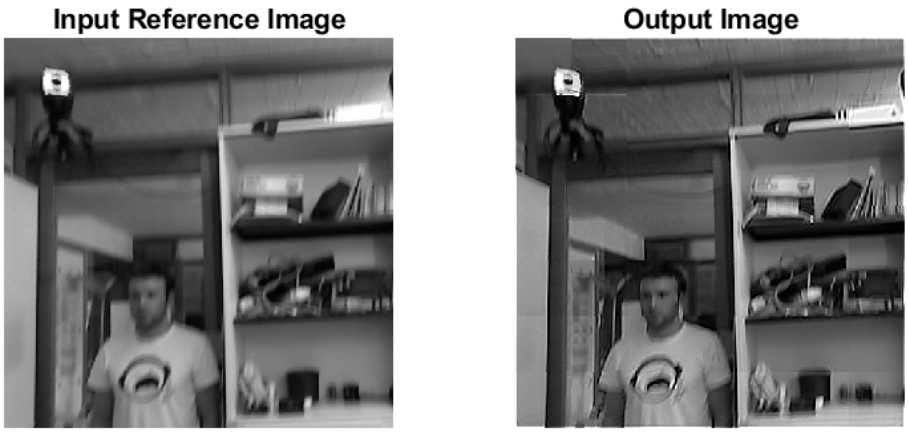

Figure 2: Total of 9 blurry inputs from TUM dataset, ran with 5 alternations and linearly decreasing rank parameter. Here are shown the input reference image (left) and the output reference image (right). The sharpness of our output is clear, and is an improvement when compared to our previous methods of multi-image deblurring.

Figure 3: Total of 9 blurry inputs from UBC dataset, ran with 5 alternations and linearly decreasing rank parameter. Here are shown the input reference image (left), the output reference image (middle) and the ground truth (right).

Figure 4: Total of 9 blurry inputs from UBC dataset, ran with 5 alternations and linearly decreasing rank parameter. Here are shown the input reference image (left), the output reference image (middle) and the ground truth (right).

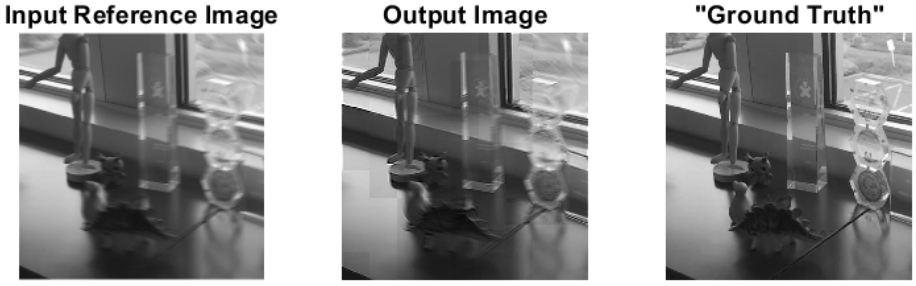

Figure 5: Total of 9 blurry inputs from UBC dataset, ran with 5 alternations and linearly decreasing rank parameter. Here are shown the input reference image (left), the output reference image (middle) and the ground truth (right).

Figure 6: Total of 9 blurry inputs from UBC dataset, ran with 5 alternations and linearly decreasing rank parameter. Here are shown the input reference image (left), the output reference image (middle) and the ground truth (right).