Total Variation Blind Deconvolution: The Devil is in the Details
Technical Report

Daniele Perrone
University of Bern
Bern, Switzerland
perrone@iam.unibe.ch

Paolo Favaro
University of Bern
Bern, Switzerland
paolo.favaro@iam.unibe.ch

In this technical report we provide proofs for the theorems presented in [8] and additional experimental results.

1. Proofs

We first prove the following Lemma.

Lemma 1.1 *Let f be a 1D discrete signal of the following form*

$$f[x] = \begin{cases} \delta_1 - U & x = -L \\ -U & x \in [-L+1, -2] \\ \delta_0 - U & x = -1 \\ -\delta_1 + U & x = 0 \\ U & x \in [1, L-2] \\ -\delta_0 + U & x = L-1 \end{cases} \tag{1}$$

for some positive constants $\delta_0,\ \delta_1 > 0$, $U > \max\{\delta_0, \delta_1\}$, and $L \geq 2$. If $\lambda \geq \max((L-1)\delta_0 - \delta_1, (L-1)\delta_1 - \delta_0)$, then the solution $\hat{u}[x]$ to the following problem

$$\hat{u}[x] = \arg\min_u \frac{1}{2} \sum_{x=-L}^{L-1} (u[x] - f[x])^2 + \lambda \sum_{x=-L}^{L-2} |u[x+1] - u[x]|. \tag{2}$$

is

$$\hat{u} = \begin{cases} -\hat{U} & x \in [-L, -1] \\ \hat{U} & x \in [0, L-1] \end{cases} \tag{3}$$

where $\hat{U} = -\frac{\lambda + \delta_0 + \delta_1}{L} + U$. Also, if $\lambda \geq UL - \delta_0 - \delta_1$ then $\hat{U} = 0$.

Proof. The solution of problem (2) can also be written as $\hat{u}[x] = \hat{s}[x] - \hat{s}[x-1]$, $x \in [-L, L-1]$, where $\hat{s}$ is found by solving the *taut string* problem (*e.g.*, see Davies and Kovac [2])

$$\hat{s}[x] = \arg\min_s \sum_{x=-L}^{L-1} \sqrt{1 + |s[x] - s[x-1]|^2} \tag{4}$$
$$\text{s.t.} \max_{x \in [-L, L-1]} |s[x] - r[x]| \leq \lambda \quad \text{and}$$
$$s[-L-1] = 0, \quad s[L-1] = r[L-1]$$

where $r[x] = \sum_{y=-L}^{x} f[y]$ with $x \in [-L, L-1]$.

Given the explicit form of f in eq. (1) we obtain that

$$r[x] = \begin{cases} \delta_1 - U & x = -L \\ \delta_1 - U(x+L+1) & x \in [-L+1, -2] \\ \delta_0 + \delta_1 - UL & x = -1 \\ \delta_0 - U(L-1) & x = 0 \\ \delta_0 - U(L-1-x) & x \in [1, L-2] \\ 0 & x = L-1. \end{cases} \tag{5}$$

First, notice that the smallest value[1] of r is $\min_{x \in [-L, L-1]} r[x] = r[-1] = \delta_0 + \delta_1 - UL$ and occurs at $x = -1$. Next, consider solving the taut string problem by enforcing only the constraint $|s[-1] - r[-1]| \leq \lambda$. The cost of the taut string problem is minimum for the shortest path s through a point at $x = -1$. We can decompose such path into the concatenation of the shortest path from $x = -L-1$ to $x = -1$ and the shortest path from $x = -1$ to $x = L-1$. Given that each of these paths are only constrained at the end points, a direct solution will give a line segment between the end points, *i.e.*,

$$s[x] = \begin{cases} \frac{x+L+1}{L} s[-1] & x \in [-L-1, -2] \\ \frac{L-1-x}{L} s[-1] & x \in [-1, L-1]. \end{cases} \tag{6}$$

The value $s[-1]$ that yields the shortest path and satisfies the constraint

$$r[-1] - \lambda \leq s[-1] \leq r[-1] + \lambda \tag{7}$$

[1] Because of the constraints on U, δ_0 and δ_1, we have that f, the derivative of r, satisfies $f[x] < 0$ for $x < 0$ and $f[x] > 0$ for $x \geq 0$.

is $s[-1] = \delta_0 + \delta_1 - UL + \lambda$ when $\lambda \leq UL - \delta_0 - \delta_1$ and $s[-1] = 0$ otherwise.

Now, we will show that, given $\lambda \geq \max((L-1)\delta_0 - \delta_1, (L-1)\delta_1 - \delta_0)$, the above shortest path s is also the solution $\hat{s}$ to the taut sting problem (4) with all the constraints. It will suffice to show that this path satisfies all the constraints in the taut string problem. Then, since it is the shortest path with a single constraint, it must also be the shortest path for problem (4). To verify all the constraints, we only need to consider 4 cases:

$$\begin{aligned} x = -L &\rightarrow \left|\tfrac{1}{L}(\delta_0 + \delta_1 - UL + \lambda) - \delta_1 + U\right| < \lambda \\ x = -2 &\rightarrow \left|\tfrac{1}{L}(\delta_0 + \delta_1 - UL + \lambda) - \tfrac{\delta_1}{L-1} + U\right| < \tfrac{\lambda}{L-1} \\ x = 0 &\rightarrow \left|\tfrac{1}{L}(\delta_0 + \delta_1 - UL + \lambda) - \tfrac{\delta_0}{L-1} + U\right| < \tfrac{\lambda}{L-1} \\ x = L-2 &\rightarrow \left|\tfrac{1}{L}(\delta_0 + \delta_1 - UL + \lambda) - \delta_0 + U\right| < \lambda \end{aligned}$$

as all the others are directly satisfied when these are. By direct substitution, one can find that $\lambda \geq \max((L-1)\delta_0 - \delta_1, (L-1)\delta_1 - \delta_0)$ satisfies all the above constraints as long as $L \geq 2$. We can then obtain $\hat{u}[x] = \hat{s}[x] - \hat{s}[x-1]$ from eq. (6) and write

$$\hat{u}[x] = \begin{cases} \frac{1}{L}(\delta_0 + \delta_1 - UL + \lambda) & x \in [-L, -2] \\ -\frac{1}{L}(\delta_0 + \delta_1 - UL + \lambda) & x \in [-1, L-1] \end{cases} \tag{8}$$

and hence $\hat{U} = -\frac{\delta_0 + \delta_1 + \lambda}{L} + U$.

Finally, as already mentioned, when $\lambda \geq UL - \delta_0 - \delta_1$ we have $s[-1] = 0$. Thus, $s[x] = 0, \forall x \in [-L-1, L-1]$ and also $\hat{u}[x] = 0, \forall x \in [-L, L-1]$.

■

Theorem 3.2. Let f be a 1D discrete noise-free signal, such that $f = k_0 * u_0$, where u_0 and k_0 are two unknown functions and $*$ is the circular convolution operator. Let us also constrain k_0 to be a blur of support equal to 3 pixels, and u_0 to be a step function

$$u_0[x] = \begin{cases} -U & x \in [-L, -1] \\ U & x \in [0, L-1] \end{cases} \tag{9}$$

for some parameters U and L. We impose that $L \geq 2$ and $U > 0$. Then f will have the following form

$$f[x] = \begin{cases} \delta_1 - U & x = -L \\ -U & x \in [-L+1, -1] \\ \delta_0 - U & x = -1 \\ -\delta_1 + U & x = 0 \\ U & x \in [1, L-2] \\ -\delta_0 + U & x = L-1 \end{cases} \tag{10}$$

for some positive constants δ_0 and δ_1 that depend on the blur parameters. Then, there exists $\lambda \geq \max((L-1)\delta_0 - \delta_1, (L-1)\delta_1 - \delta_0)$ such that the PAM algorithm estimates the true blur $k = k_0$ in two steps, when starting from the no-blur solution (f, δ).

Proof. By assuming that the initial blur is a Dirac delta, the first step of the PAM algorithm solves the following problem

$$\hat{u} = \arg\min_u \frac{1}{2}||u - f||_2^2 + \lambda||u||_{BV}. \tag{11}$$

The solution u of the problem (11) when $\lambda = \max((L-1)\delta_0 - \delta_1, (L-1)\delta_1 - \delta_0)$ is given by Lemma 1.1, and it is

$$\hat{u} = \begin{cases} -\hat{U} & x \in [-L, -1] \\ \hat{U} & x \in [0, L-1] \end{cases} \tag{12}$$

with $\hat{U} = -\frac{\lambda + \delta_0 + \delta_1}{L} + U$. Notice that $\hat{u}$ is also a zero-mean signal.

Since we can always express a zero-mean step as another scaled zero-mean step, we can write

$$\hat{u} = a u_0 \tag{13}$$

for some constant a. We then solve the second step of the PAM algorithm

$$\hat{k} = \arg\min_k ||k * \hat{u} - f||_2^2 \tag{14}$$

and since we can write

$$\begin{aligned} \|k * \hat{u} - f\|_2^2 = \|k * a u_0 - f\|_2^2 = \|k * a u_0 - k_0 * u_0\|_2^2 \\ = \|(ak - k_0) * u_0\|_2^2 \end{aligned} \tag{15}$$

we have $\hat{k} = k_0/a$. Finally, by applying the last two steps of the PAM algorithm one obtains $\hat{k} = k_0$. ■

Theorem 3.3. Let f, u_0 and k_0 be the same as in Theorem 3.2. Then, for any $\max((L-1)\delta_0 - \delta_1, (L-1)\delta_1 - \delta_0) < \lambda < UL - \delta_0 - \delta_1$ the AM algorithm converges to the solution $k = \delta$. For $\lambda \geq UL - \delta_0 - \delta_1$ the AM algorithm is unstable.

Proof. For $\max((L-1)\delta_0 - \delta_1, (L-1)\delta_1 - \delta_0) < \lambda < UL - \delta_0 - \delta_1$ from Lemma (1.1) we have that the minimization of the problem 11 results in the following $\hat{u}$

$$\hat{u} = \begin{cases} -\hat{U} & x \in [-L, -1] \\ \hat{U} & x \in [0, L-1] \end{cases} \tag{16}$$

where $\hat{U} = -\frac{\lambda+\delta_0+\delta_1}{L} + U$. The cost $||k * \hat{u} - f||_2^2$, can be then split in 6 regions

$$\begin{array}{l} ||k * \hat{u} - f||_2^2 = \sum_x ((k * \hat{u})[x] - f[x])^2 = \\ (k[3]\hat{U} - (k[2] + k[1])\hat{U} - \delta_1 + U)^2 \\ +(L-2)(-\hat{U} + U)^2 + (-(k[3] + k[2])\hat{U} + k[1]\hat{U} + U - \delta_0)^2 \\ +(-k[3]\hat{U} + (k[2] + k[1])\hat{U} - U + \delta_1)^2 \\ +(L-2)(\hat{U} - U)^2 \\ +((k[3] + k[2])\hat{U} - k[1]\hat{U} - U + \delta_0)^2 \end{array} \tag{17}$$

The second and fifth terms do not depend on k, so only the other terms contribute to the estimation of k. We then compare the cost of each term for the no-blur solution $k[1] = 0,\ k[2] = 1,\ k[3] = 0$ with any other solution that satisfies the constraints $k \geq 0$ and $\mathbf{1}^T k = 1$. For the no-blur solution we have

$$\begin{aligned} (k[3]\hat{U} - (k[2] + k[1])\hat{U} - \delta_1 + U)^2 &= (-\hat{U} - \delta_1 + U)^2 \\ (-(k[3] + k[2])\hat{U} + k[1]\hat{U} + U - \delta_0)^2 &= (-\hat{U} + U - \delta_0)^2 \\ (-k[3]\hat{U} + (k[2] + k[1])\hat{U} - U + \delta_1)^2 &= (\hat{U} - U + \delta_1)^2 \\ ((k[3] + k[2])\hat{U} - k[1]\hat{U} - U + \delta_0)^2 &= (\hat{U} - U + \delta_0)^2 \end{aligned}$$

For a feasible solution k we now show that each of terms that contribute to the estimation of k in (17) is strictly larger than the corresponding one for the trivial solution.

Following from $\lambda > \max((L-1)\delta_0 - \delta_1, (L-1)\delta_1 - \delta_0)$ and from the definition of $\hat{U}$ the inequalities $-\hat{U} > -U + \delta_0$, $\hat{U} < U - \delta_1$, $\hat{U} < U - \delta_0$ and $-\hat{U} > -U + \delta_1$ hold.

From $k[2] + k[1] \leq 1$ and $-\hat{U} > -U + \delta_1$ we have $-(k[2]+k[1])\hat{U} \geq -\hat{U} > -U+\delta_1$, and therefore $-(k[2]+k[1])\hat{U} - \delta_1 + U > 0$ and $-\hat{U} - \delta_1 + U > 0$. The following inequality then holds

$$\begin{array}{l} (k[3]\hat{U} - (k[2] + k[1])\hat{U} - \delta_1 + U)^2 > \\ (-k[3]\hat{U} - (k[2] + k[1])\hat{U} - \delta_1 + U)^2 = (-\hat{U} - \delta_1 + U)^2. \end{array} \tag{18}$$

Similarly, using $(k[3]+k[2]) \leq 1$ and $-\hat{U} > -U+\delta_0$ we derive the inequalities $-(k[3] + k[2])\hat{U} \geq -\hat{U} > -U + \delta_0$, $-(k[3] + k[2])\hat{U} + U - \delta_0 > 0$ and $-\hat{U} + U - \delta_0 > 0$, that result in

$$\begin{array}{l} (-(k[3] + k[2])\hat{U} + k[1]\hat{U} + U - \delta_0)^2 > \\ (-(k[3] + k[2])\hat{U} - k[1]\hat{U} + U - \delta_0)^2 = (-\hat{U} + U - \delta_0)^2. \end{array} \tag{19}$$

In the other two cases, from $k[2] + k[1] \leq 1$ and $\hat{U} < U - \delta_1$ we have $(k[2]+k[1])\hat{U} \leq \hat{U} < U - \delta_1$, that implies $(k[2]+k[1])\hat{U} - U + \delta_1 < 0$, $\hat{U}_1 - U + \delta_1 < 0$. and therefore

$$\begin{array}{l} (-k[3]\hat{U} + (k[2] + k[1])\hat{U} - U + \delta_1)^2 > \\ (k[3]\hat{U} + (k[2] + k[1])\hat{U} - U + \delta_1)^2 = (\hat{U} - U + \delta_1)^2. \end{array} \tag{20}$$

Finally, from $(k[3] + k[2]) \leq 1$ and $\hat{U} < U - \delta_0$ we have $(k[3]+k[2])\hat{U} \leq \hat{U} < U - \delta_0$, $(k[3]+k[2])\hat{U} - U + \delta_0 < 0$, $\hat{U} - U + \delta_0 < 0$, and

$$\begin{array}{l} ((k[3] + k[2])\hat{U} - k[1]\hat{U} - U + \delta_0)^2 > \\ ((k[3] + k[2])\hat{U} + k[1]\hat{U} - U + \delta_0)^2 = (\hat{U} - U + \delta_0)^2. \end{array} \tag{21}$$

The above inequalities show how $||\hat{u} - f||_2^2 \leq ||k * \hat{u} - f||_2^2$, for any k such that $k \geq 0$ and $\mathbf{1}^T k = 1$. For $\lambda \geq UL - \delta_0 - \delta_1$ we have $\hat{U} = 0$, the cost becomes $||k * \hat{u} - f||_2^2 =$ const. Since any feasible k is a solution of $||k * \hat{u} - f||_2^2$ the AM algorithm becomes unstable. ■

References

[1] S. Cho and S. Lee. Fast motion deblurring. *ACM Trans. Graph.*, 2009.

[2] P. L. Davies and A. Kovac. Local extremes, runs, strings and multiresolution. *The Annals of Statistics*, 2001.

[3] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman. Removing camera shake from a single photograph. *ACM Trans. Graph.*, 2006.

[4] A. Goldstein and R. Fattal. Blur-kernel estimation from spectral irregularities. In *ECCV*, 2012.

[5] M. Hirsch, C. Schuler, S. Harmeling, and B. Schölkopf. Fast removal of non-uniform camera shake. In *ICCV*, pages 463–470, 2011.

[6] D. Krishnan, T. Tay, and R. Fergus. Blind deconvolution using a normalized sparsity measure. In *CVPR 2011*, 2011.

[7] A. Levin, Y. Weiss, F. Durand, and W. Freeman. Efficient marginal likelihood optimization in blind deconvolution. In *CVPR*, 2011.

[8] D. Perrone and P. Favaro. Total variation blind deconvolution: The devil is in the details. In *CVPR*, 2014.

[9] Q. Shan, J. Jia, and A. Agarwala. High-quality motion deblurring from a single image. In *ACM Trans. Graph.*, 2008.

[10] O. Whyte, J. Sivic, and A. Zisserman. Deblurring shaken and partially saturated images. In *Proceedings of the IEEE Workshop on Color and Photometry in Computer Vision, with ICCV 2011*, 2011.

[11] L. Xu and J. Jia. Two-phase kernel estimation for robust motion deblurring. In *ECCV*, 2010.

[12] L. Xu, S. Zheng, and J. Jia. Unnatural l0 sparse representation for natural image deblurring. In *CVPR*, 2013.

[13] L. Zhong, S. Cho, D. Metaxas, S. Paris, and J. Wang. Handling noise in single image deblurring using directional filters. In *CVPR*, 2013.


Figure 1. Blurry Input.

Figure 2. Restored image and blur with Xu and Jia [11].

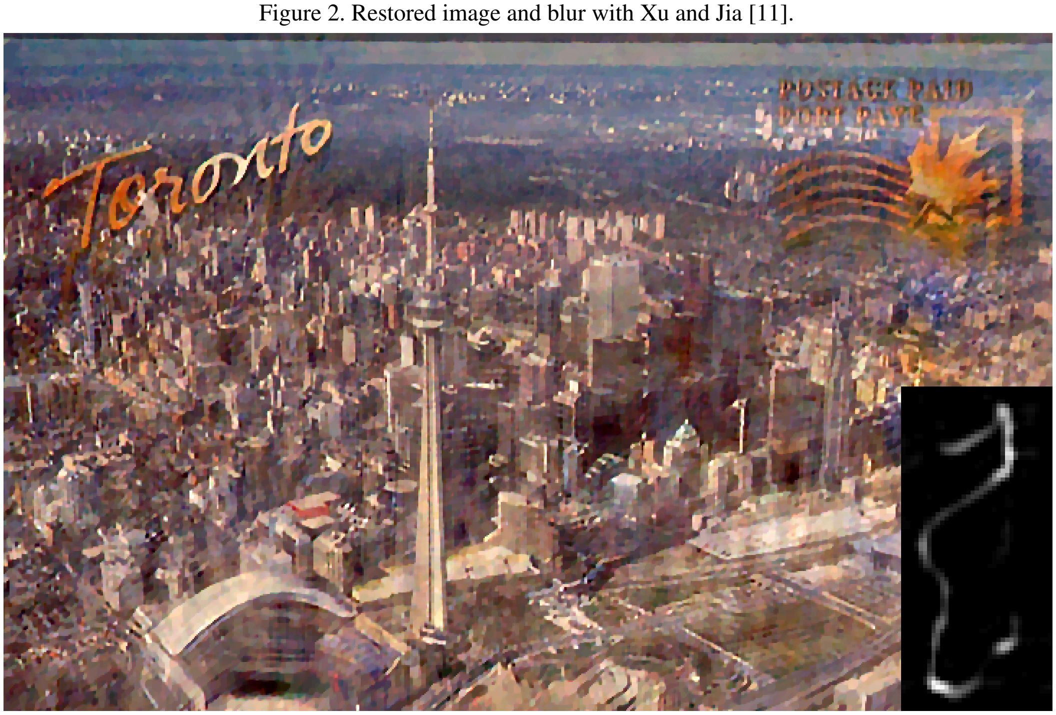

Figure 3. Restored image and blur with our algorithm.

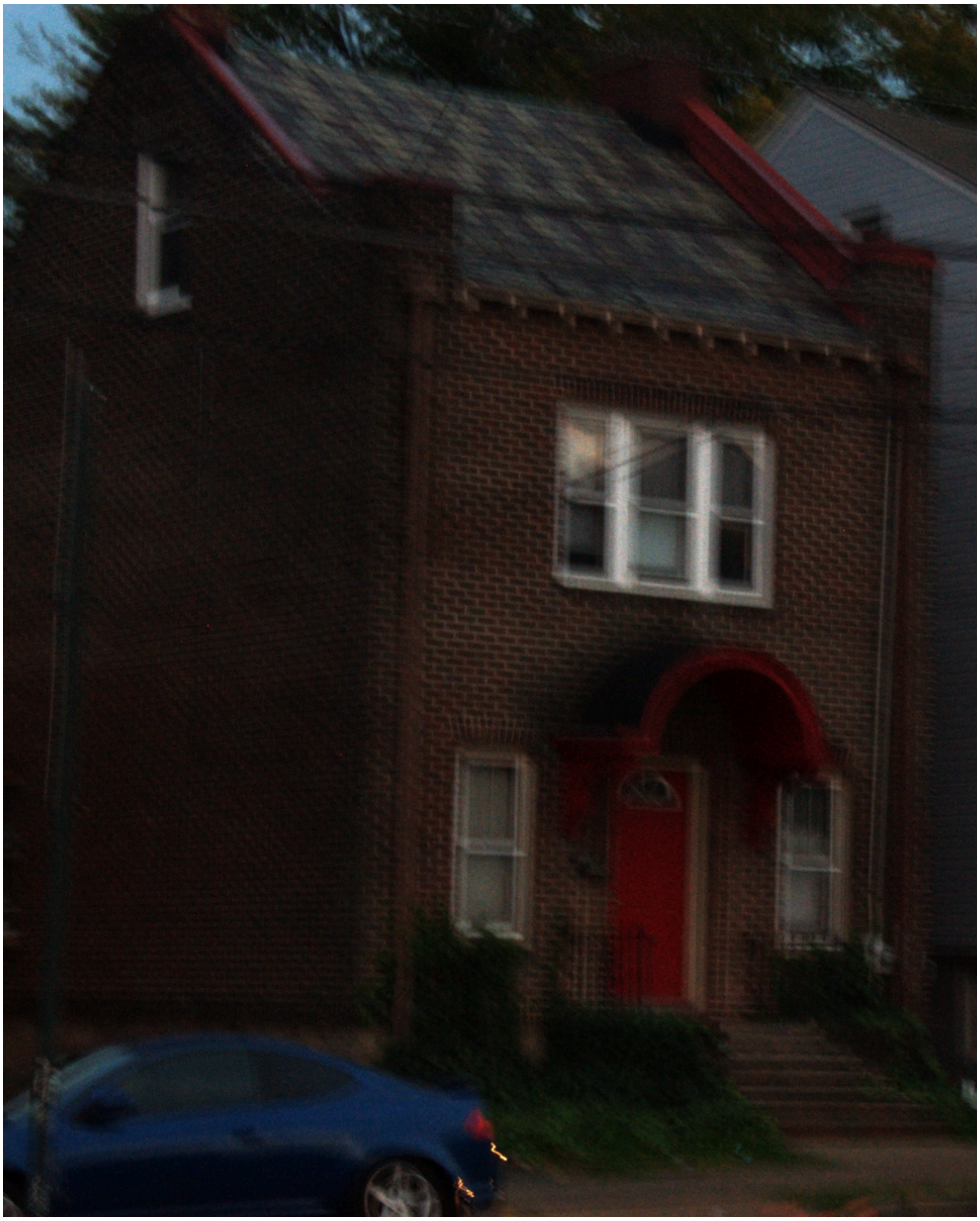

Figure 4. Blurry Input.

Figure 5. Restored image with Zhong *et al.* [13].

Figure 6. Restored image with our algorithm.

Figure 7. Blurry Input.

Figure 8. Restored image with Cho and Lee [1].

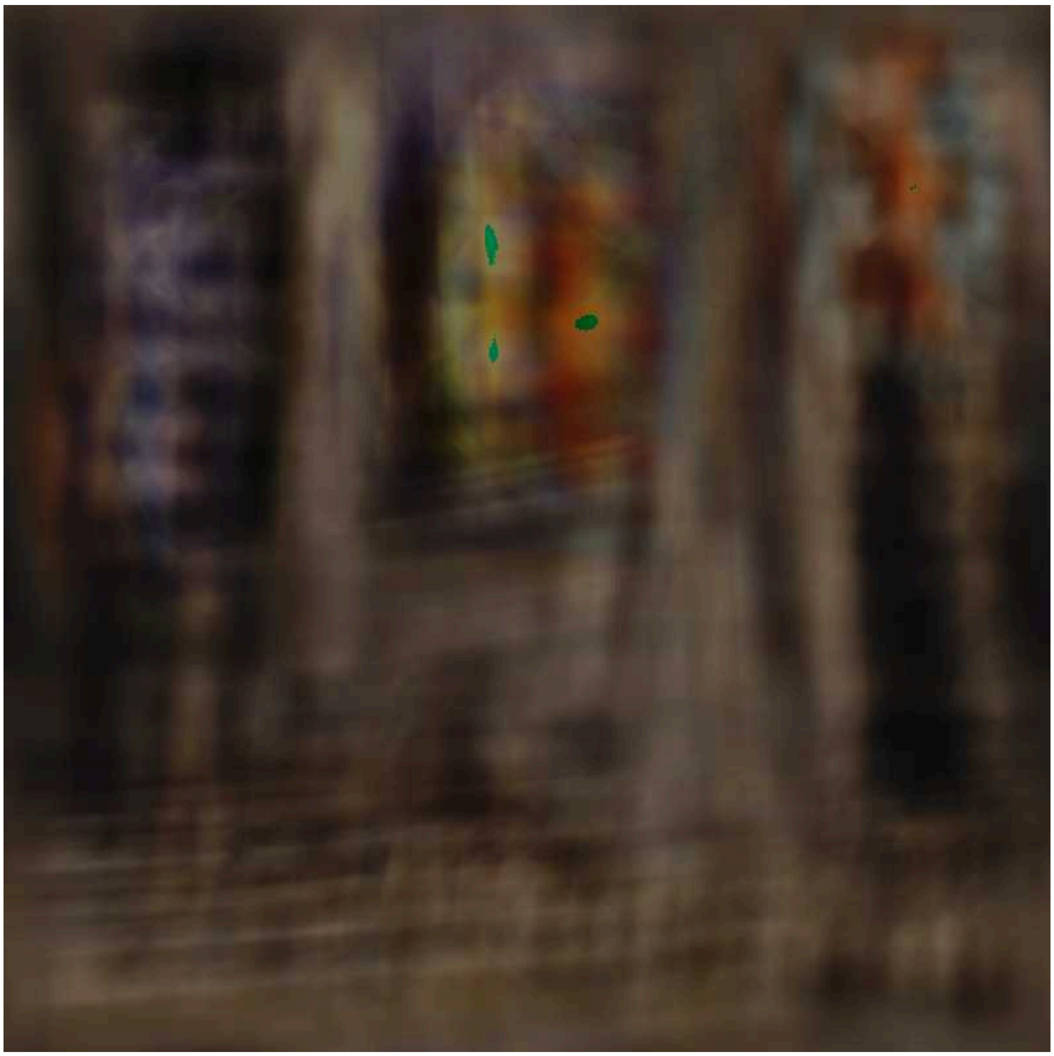

Figure 9. Restored image with Fergus *et al.* [3].

Figure 10. Restored image with Krishnan *et al.* [6].

Figure 11. Restored image with Xu and Jia [11].

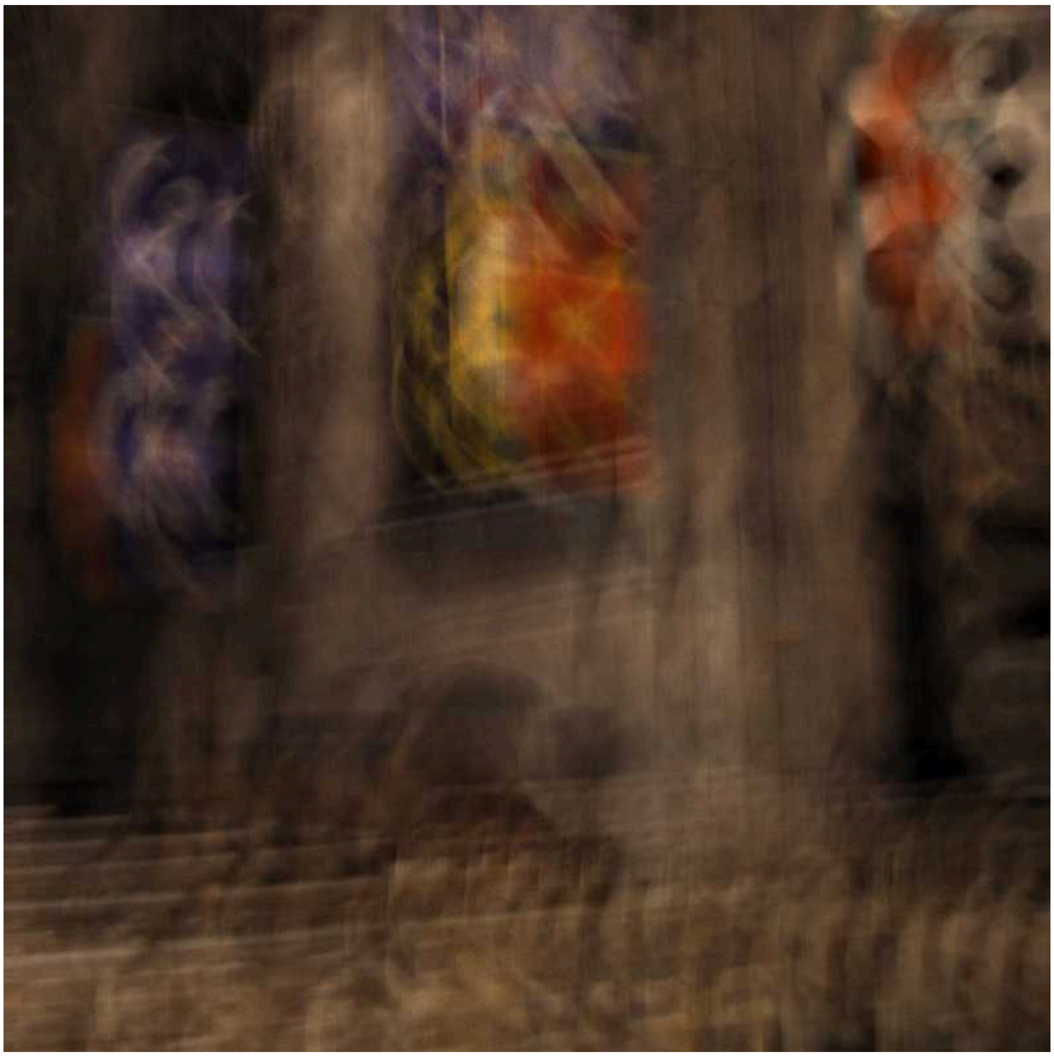

Figure 12. Restored image with Whyte *et al.* [10].

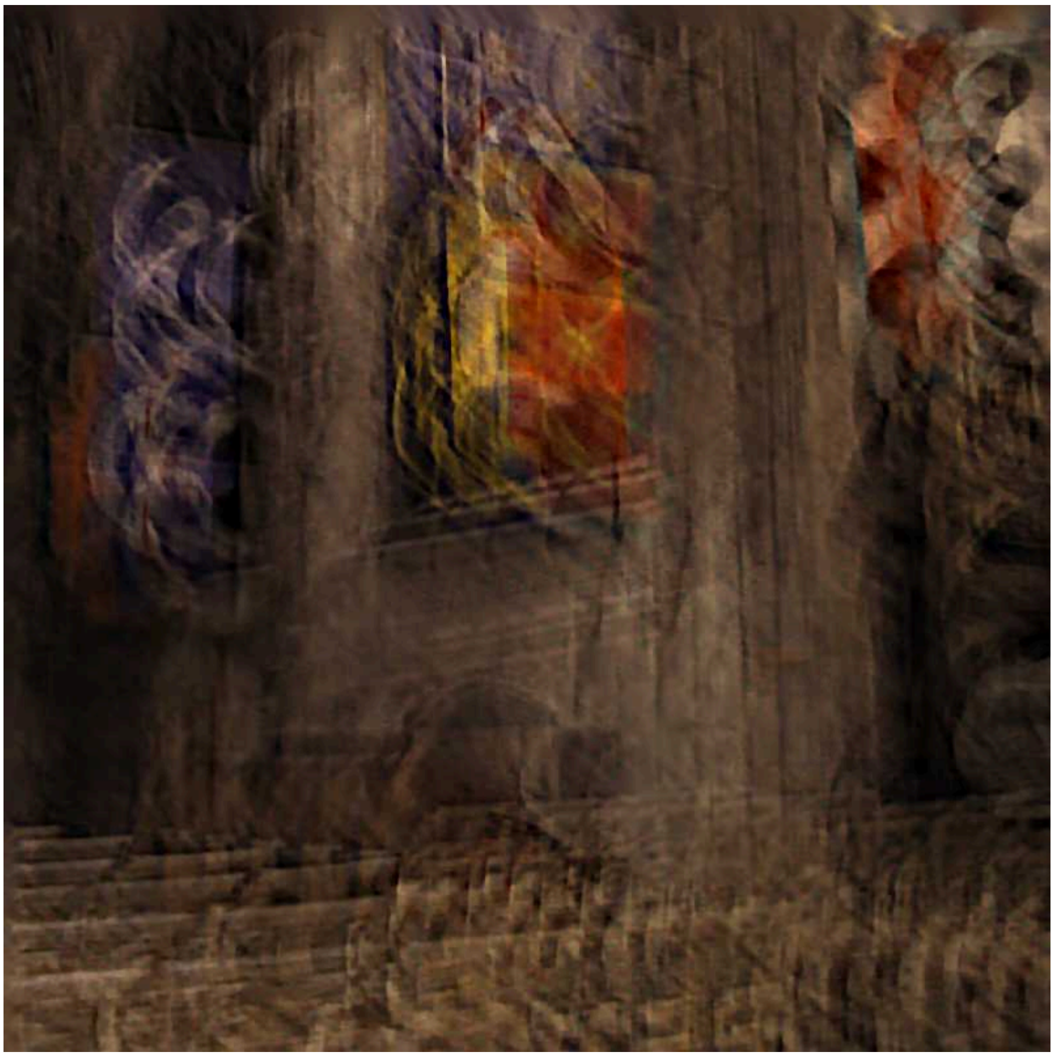

Figure 13. Restored image with Shan *et al.* [9].

Figure 14. Restored image with Hirsch *et al.* [5].

Figure 15. Restored image with Xu *et al.* [12].

Figure 16. Restored image with our algorithm.

Figure 17. Blurry Input.

Figure 17. Blurry Input.

Figure 18. Restored image with Cho and Lee [1].

Figure 19. Restored image with Levin *et al.* [7].

Figure 20. Restored image with Goldstein and Fattal [4].

Figure 21. Restored image with Zhong *et al.* [13].

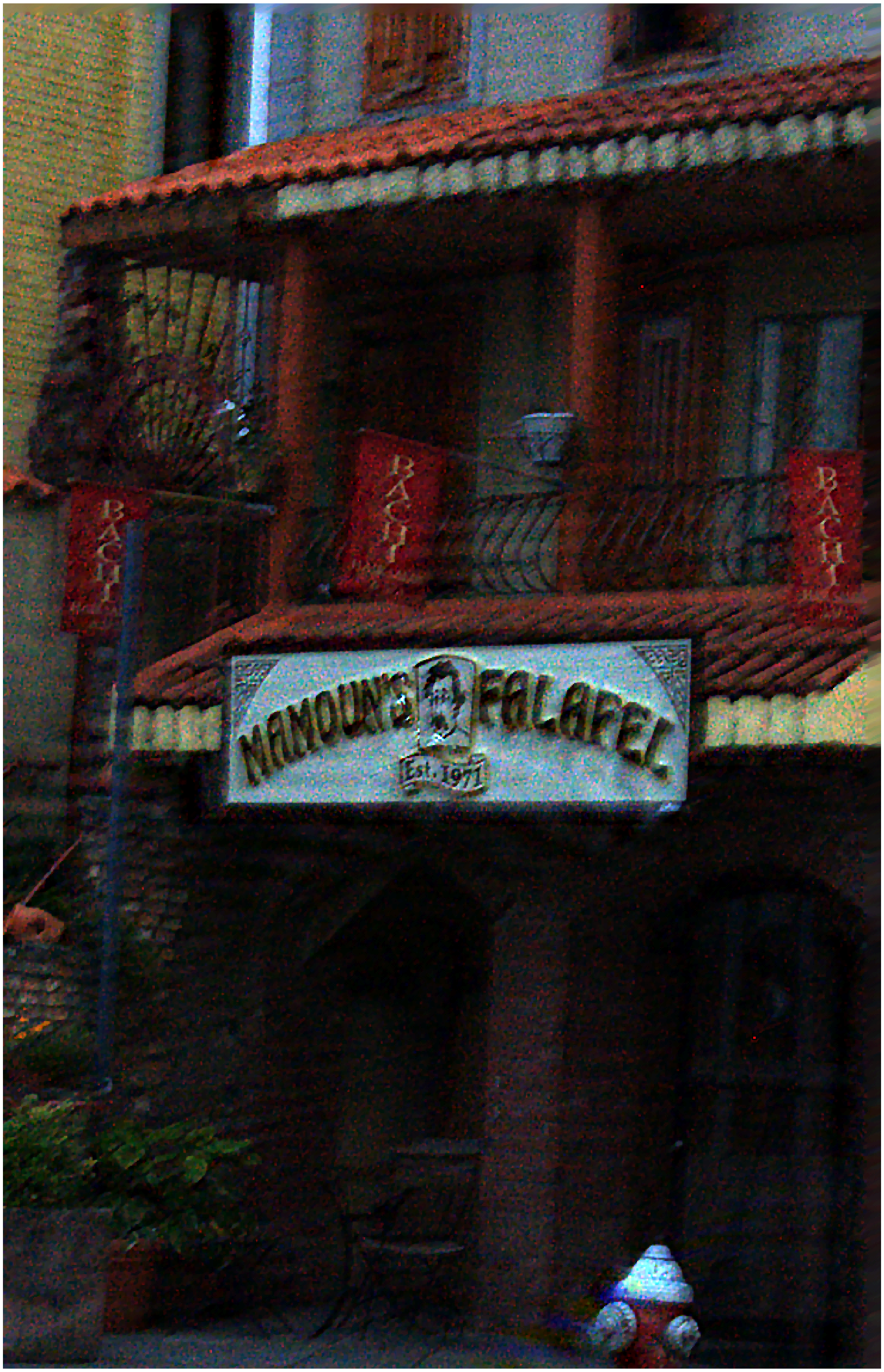

Figure 22. Restored image with our algorithm.