Deblurred images post-processing by Poisson warping

Alexandra Nasonova, and Andrey Krylov, *Member, IEEE*

Abstract

In this work we develop a post-processing algorithm which enhances the results of the existing image deblurring methods. It performs additional edge sharpening using grid warping. The idea of the proposed algorithm is to transform the neighborhood of the edge so that the neighboring pixels move closer to the edge, and then resample the image from the warped grid to the original uniform grid. The proposed technique preserves image textures while making the edges sharper. The effectiveness of the method is shown for basic deblurring methods on LIVE database images with added blur and noise.

Index Terms

Image sharpening, edge model, image grid warping, deblurring, post-processing

I. Introduction

THERE are some fairly powerful techniques for image deblurring [1], [2], [3]. The typical problem of image deblurring methods is finding optimal parameters for a compromise between smooth result with blurry edges and sharp result with artifacts like ringing or noise amplification. In this work we present a new post-processing algorithm for image deblurring with enhancement of edge sharpness.

We localize the area of interest to the neighborhood of the edges. The idea is to transform the neighborhood of the blurred edge so that the neighboring pixels move closer to the edge, and then resample the image from the warped grid to the original uniform grid.

The warping approach is related to the morphology-based sharpening [4] and shock filters [5], [6], [7]. But these methods make the image appear piecewise constant which is effective mostly for cartoon-like images. The proposed method is applied to edges locally so the textures are preserved a priori, also warping compresses the edge neighborhood at fixed rate and does not make the image piecewise constant.

The warping approach for image enhancement was introduced in [8]. The warping of the grid is performed according to the solution of a differential equation that is derived from the warping process constraints. The solution of the equation is used to move the edge neighborhood closer to the edge, and the areas between edges are stretched. The method has several parameters, and the choice of optimal values for the best result is not easy. Due to the global nature of the method the resulting shapes of the edges are sometimes distorted.

In [9] the warping map is computed directly using the values of left and right derivatives. In both methods [8] and [9] the pixel shifts are proportional to the gradient values. It results in oversharpening of already sharp and high contrast edges and insufficient sharpening of blurry and low contrast edges. Both methods also introduce small local changes in the direction of edges and produce aliasing effect due to calculation of horizontal and vertical warping components separately.

II. One-dimensional edge sharpening

In this section we describe the idea of a single edge enhancement using a coordinate grid transformation.

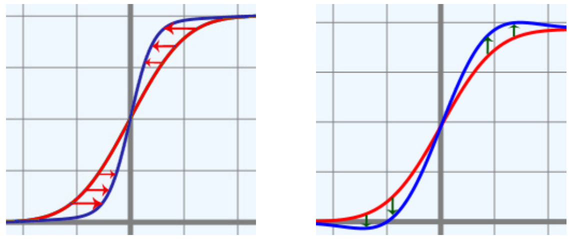
A. Grid warping

Compared to a sharp edge profile, the profile of a blurred edge is more gradual. So in order to make the edge sharper its transient width should be decreased (see Fig. 1).

A. Nasonova and A. Krylov are with the Faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University, Moscow, Russia, e-mail: nasonova@cs.msu.ru, kryl@cs.msu.ru.

The work was supported by RSCF grant 14-11-00308.

Manuscript received ...; revised ...



a. Proposed approach: pixels are shifted

b. Typical deblurring approach: pixel values are modified

Fig. 1. The idea of 1D edge sharpening

For any edge $g(x)$ centered at $x = 0$ its sharper version $h(x)$ can be obtained shifting the pixels from the neighborhood of the edge towards its center. The *displacement function* $d(x)$ describes the shift of a pixel with coordinate x to a new coordinate $x + d(x)$: $h(x + d(x)) = g(x)$.

The warped grid should remain monotonic (i.e. for any $x_1 < x_2$ new coordinates should be $x_1 + d(x_1) \leq x_2 + d(x_2)$), so the displacement function should match the following constraint:

$$d'(x) \geq -1. \tag{1}$$

Another constraint localizes the area of warping effect near the edge center:

$$d(x) \to 0, \qquad |x| \to \infty.$$

The displacement function $d(x)$ greatly influences the result of the edge warping. On the one hand, the edge slope should become steeper. On the other hand, the area near the edge should not be stretched over some predefined limit to avoid wide gaps between adjacent pixels in the discrete case.

B. The proximity function

The displacement function $d(x)$ is connected with the *proximity* of image pixels $p(x)$: $p(x) = 1 + d'(x)$.

The proximity is the distance between adjacent pixels after image warping. This value is inverse to the density value. If the proximity function $p(x)$ is less than 1, then the area is densified at the point x. If the proximity is greater than 1, then the grid is rarefied. For an unwarped image $p(x) \equiv 1$.

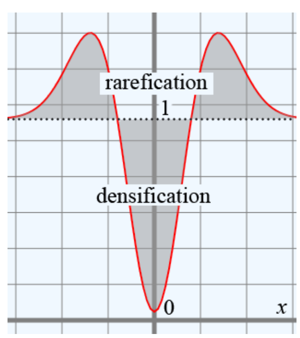


Fig. 2. Example of proximity function for edge sharpening

The constraint (1) leads to non-negativity of the proximity function. Also high values of the proximity function should be avoided, because it will be hard to perform interpolation in rarefication areas if the rarefication is too strong.

We use the proximity function $p(x)$ to calculate the displacement function: $d(x) = \int\limits_{-\infty}^{x} (p(y) - 1)dy$.

C. Edge model

The choice of the displacement function is based on the assumption that the blurred edge $E_\sigma(x)$ can be approximated by an ideal step edge $H(x)$ convolved with Gaussian filter with known parameter σ [10]:

$$E_\sigma(x) = [H * G_\sigma](x), \qquad H(x) = \begin{cases} 1, & x \geq 0, \\ 0, & x < 0, \end{cases}$$

where $G_\sigma(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$.

D. Choosing the proximity function

Consider the following physical model: the point x is a particle of fixed mass situated on the slope $E_\sigma(x)$. The particle is connected by a spring to a vertical axis. Due to the force of gravity the spring is deformed until the particle reaches equilibrium at the point $x + d(x)$. This model ensures that the shape of the edge is not distorted and the grid transformation is smooth.

This model produces the following displacement function:

$$d(x) = -\kappa G'_\sigma(x).$$

The constant $\kappa > 0$ controls the level of the grid warping. With the increase of the value of κ the edge becomes sharper. The value κ is limited by the constraint (1)

$$\kappa \leq 1/G''_\sigma(0).$$

According to this model, the proximity function takes the form:

$$p(x) = 1 - \kappa G''_\sigma(x) \tag{2}$$

This approach has a limitation: it controls the areas of densification and rarefication simultaneously.

We approximate the proximity function (2) with the difference of two Gaussian functions in order to control the areas of rarefication and densification independently:

$$p(x) = 1 + \kappa(G_{\sigma_1}(x) - G_{\sigma_2}(x)), \quad \sigma_2 > \sigma_1. \tag{3}$$

Parameter σ_1 controls the width of the densification area while parameter σ_2 controls the width of the rarefication area. When $\sigma_2 \to \sigma_1$, the proximity function (3) will converge to (2).

Observations show that with $\sigma_1 < \sigma$, where σ is the width of the edge, the width of the densification area becomes narrower (see Fig. 3c) while with $\sigma_1 > \sigma$ the width of the densification area becomes wider than the width of the edge (see Fig. 3d). Using $\sigma_1 \geq \sigma$ gives reasonable results. High values of σ_2 results in too wide rarefication area (see Fig. 3f).

Typically, images contain many edges, so it is desirable to localize the area of warping effect to the neighborhood of the edges, therefore the values of parameters σ_1 and σ_2 should be as low as possible.

For the edge blurred with σ, we use $\sigma_1 = \sigma$. Parameter σ_2 is taken as $k\sigma_1 = k\sigma$. Good results are obtained with $1.5 \leq k \leq 2$. To achieve the strongest sharpening effect, we use the maximal values of κ that matches the constraint (1):

$$\kappa = 1/\left(G_{\sigma_1}(0) - G_{\sigma_2}(0)\right).$$

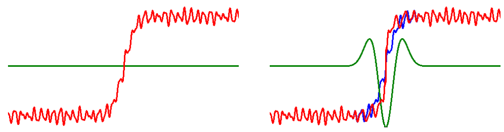
III. Image sharpening using warping

A. Two-dimensional extension

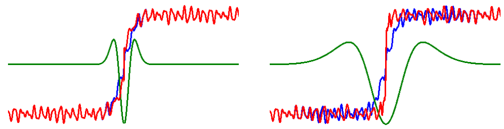
In the 2D case the displacement is a vector field $\vec{d}(x, y)$. There are some obvious constraints (see Fig. 4).

1) The shapes of the edges cannot be warped, so $\vec{d}(x_e, y_e) = 0$ for each edge point (x_e, y_e).

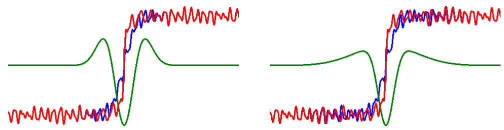
2) Also, there cannot be any turbulence: $\mathrm{rot}\ \vec{d} = 0$. Since $\mathrm{rot}\nabla u \equiv 0$, the displacement field is assumed to be gradient of some scalar function $u(x, y)$: $\vec{d}(x, y) = \nabla u(x, y)$.



a. Input edge b. Approach (2)



c. $\sigma_1 << \sigma$ d. $\sigma_1 >> \sigma$



e. $\sigma_1 \sim \sigma, \sigma_2 = 1.1\sigma_1$ f. $\sigma_1 \sim \sigma, \sigma_2 = 3\sigma_1$

Fig. 3. Edges warped using $p(x)$ (3) with different parameters. Blue function is the model edge with some noise. Red function is the result of warping. Green line is the proximity function.

3) The edge neighborhood points situated farther from the edge cannot move closer to the edge than the neighborhood points situated nearer to the edge. This condition can be described in terms of the constraint (1) where 1D derivative corresponds to a divergence of 2D vector field

$$\text{div } \vec{d} \geq \ -1$$

and the proximity function takes the form:

$$p(x,y) = \ 1 \ + \ \text{div } \vec{d}(x, \ y).$$

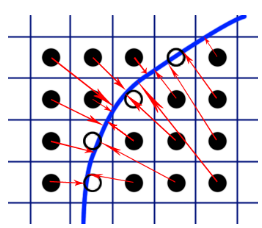


Fig. 4. Displacements directions for 2D grid warping. The blue line represents the exact edge location, white circles represent edge pixels, black circles represent pixels from the edge neighborhood.

Since $\text{div}\nabla \equiv \Delta$, where Δ is a Laplacian, the warping problem can be posed as a Dirichlet problem for the Poisson equation in the area of the image:

$$\begin{cases} \Delta u & = p(x,y) - 1, \\ u(x,y) & = 0 \text{ at image borders.} \end{cases} \tag{4}$$

The second constraint here is the boundary condition: the displacements at image borders should be equal to zero.

B. The warping algorithm

In order to get the same results as in the 1D case and to keep the edge pixels unwarped, the proximity value should be equal to the 1D proximity function depending on the distance to the edge. However, the distance to the closest edge ρ as an argument of the proximity function $p(\rho)$ is not efficient as it may produce gaps between close edges. Also it blurs edge ends.

We suggest the following method for calculating the proximity function:

$$p(x_0, y_0) = \frac{\sum\limits_{(x,y)\in E(x_0,y_0)} p(x_n) G_\sigma(x_t) |\vec{g}(x,y)|}{\sum\limits_{(x,y)\in E(x_0,y_0)} |\vec{g}(x,y)|}$$

where $E(x_0, y_0)$ is the set of edge points in the neighborhood of (x_0, y_0).

The values x_n and x_t are projections of the vector $(x_0 - x, y_0 - y)$ on the edge gradient vector $\vec{g}(x, y)$ and on the tangent to the edge respectively.

The function $p(x_n)$ is the 1D proximity function, $G_\sigma(x_t)$ is the weighting function with standard deviation equal to the edge's blur σ.

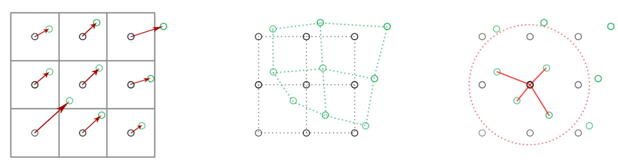
We solve the equation (4) by Gauss-Zeidel method and perform interpolation at the original pixel grid using the method described in [**?**].

C. Interpolation

The idea of interpolation from the warped grid to the uniform grid is as follows: the intensity of the image at pixel (i, j) is computed as a weighted sum of intensities of all points on the warped grid in the neighborhood of the pixel (see Fig. 5c): for a given radius r and all neighboring points $\{(x_k, y_k) : d_k = \sqrt{(i - x_k)^2 + (j - y_k)^2} \le r\}_{k=1}^{K}$ the intensity of a sharpened image I_s at (i, j) is computed as

$$I_s(i, j) = \sum_{k=1}^{K} \frac{1}{D_k} I(x_k, y_k), \text{ where } D_k = d_k / \sum_{l=1}^{K} d_l.$$

We use the interpolation radius $r = 1.5$.



a. displacements b. warped grid c. interpolation

Fig. 5. Grid warping and interpolation

IV. Results and experiments

The edge map at the input of the algorithm has a great influence on the result of grid warping as only detected edges will be sharpened. In our work we use the result of Canny edge detection [11] with zero thresholds and a predefined radius.

The example of the proposed method is shown on Fig. 6. It can be seen that the edges become better and the overall quality is improved. Nevertheless, small SSIM degradation regions exist. They correspond to the initially blurred regions of the image. Of course, an unwanted sharpening effect for the blurred areas of the original image can appear but it is a rare case.

The proposed method was tested on 29 images from LIVE database [12]. The images were blurred with Gaussian kernel with random radius in the range $[1, 6]$, then Gaussian white noise with random standard deviation in the range $[0, 10]$ was added. The degraded images were deblurred using standard MATLAB deblurring methods with known parameters of blur and noise levels used for degrading the original images. After that we warped the deblurred images using the known blur level. Table I represents the result. Instead of our method postprocessing by methods [8] and [9] did not improve the results of used deblurring algorithms.

Preliminary experiments with automatic estimation of the unknown edge width [13] also show the enhancement of deblurring methods.

Method	No warping	With warping
Blurred and noisy images	22.84	23.25
Unsharp masking	23.00	23.36
TV regularization	23.30	23.40
Low-frequency TV reg. [15]	23.08	23.18
TVMM [2]	23.31	23.48
Lucy-Richardson [14]	23.83	23.99
Wiener [14]	24.00	24.17
MatLab blind deconvolution	23.79	23.96

TABLE I
AVERAGE PSNR VALUES FOR IMAGES FROM LIVE DATABASE WITH ADDED BLUR AND NOISE

V. CONCLUSION

The advantages of the proposed algorithm are as follows:

1. It can be efficiently used as a post-processing method for global image deblurring methods.

2. No artifacts like ringing effect or noise amplification are introduced.

3. Unlike morphological methods and shock filters, the resulting images look natural and do not inevitably become piecewise constant.

4. It can be used as a standalone method of image sharpening. It is a good choice in the presence of strong noise and complex and non-uniform blur kernel.

The proposed algorithm is not posed as a global image deblurring method. It enhances only image edges neighborhood and practically does not affect image textures.

Demo software can be found at http://imaging.cs.msu.ru/soft

REFERENCES

[1] M. Almeida and M. Figueiredo, "Parameter estimation for blind and non-blind deblurring using residual whiteness measures," *IEEE Trans. Image Processing*, vol. 22, pp. 2751–2763, 2013.

[2] J. Oliveira, J. M. Bioucas-Dia, and M. Figueiredo, "Adaptive total variation image deblurring: A majorization-minimization approach," *Signal Process.*, vol. 89, pp. 1683–1693, 2009.

[3] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Variational bayesian blind deconvolution using a total variation prior," *IEEE Trans. Image Process.*, vol. 18, pp. 12–26, 2009.

[4] J. Schavemaker, M. Reinders, J. Gerbrands, and E. Backer, "Image sharpening by morphological filtering," *Pattern Recognition*, vol. 33, no. 6, pp. 997–1012, 2000.

[5] S. Osher and L. I. Rudin, "Feature-oriented image enhancement using shock filters," *SIAM Journal on Numerical Analysis*, vol. 27, no. 4, pp. 919–940, 1999.

[6] G. Gilboa, N. A. Sochen, and Y. Y. Zeevi, "Regularized shock filters and complex diffusion," *Lecture Notes in Computer Science*, vol. 2350, pp. 399–413, 2002.

[7] J. Weickert, "Coherence-enhancing shock filters," *Lecture Notes in Computer Science*, vol. 2781, pp. 1–8, 2003.

[8] N. Arad and C. Gotsman, "Enhancement by image-dependent warping," *IEEE Trans. Image Proc.*, vol. 8, pp. 1063–1074, 1999.

[9] J. Prades-Nebot et al., "Image enhancement using warping technique," *IEEE Electronics Letters*, vol. 39, pp. 32–33, 2003.

[10] A. V. Nasonov and A. S. Krylov, "Scale-space method of image ringing estimation," in *Proceedings of International Conference on Image Processing (ICIP'09)*, Cairo, Egypt, 2009, pp. 2794–2797.

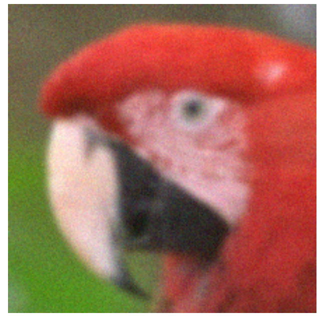
[11] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, pp. 679–698, 1986.

[12] H. Sheikh, M. Sabir, and A. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3440–3451, 2006.

[13] A. A. Nasonova and A. S. Krylov, "Determination of image edge width by unsharp masking," *Computational Mathematics and Modelling*, vol. 25, pp. 72–78, 2014.

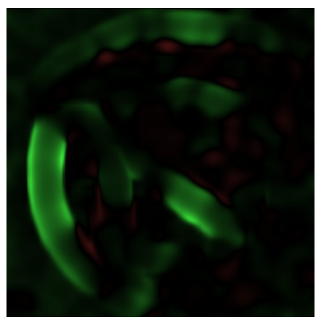
[14] J. G. Nagy, K. Palmer, and L. Perrone, "Iterative methods for image deblurring: A matlab object-oriented approach," *Numerical Algorithms*, vol. 36, no. 1, pp. 73–93, 2004.

[15] A. S. Krylov and A. V. Nasonov, "Adaptive image deblurring with ringing control," *Fifth International Conference on Image and Graphics (ICIG '09)*, pp. 72–75, 2009.

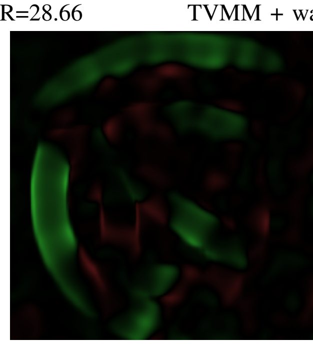
Reference image | Degraded image, PSNR=25.99


Wiener method [14], PSNR=27.05 | Wiener + warping, PSNR=27.21



SSIM difference after image warping for Wiener method
Green areas show improvement of SSIM, red areas show SSIM degradation.


TVMM [2], PSNR=28.66 | TVMM + warping, PSNR=28.72



SSIM difference after image warping for TVMM method.
Green areas show improvement of SSIM, red areas show SSIM degradation.

Fig. 6. Example of the proposed warping algorithm