Evaluation of image deblurring algorithms for real-time applications

Giuseppe Airò Farulla, Marco Indaco, Daniele Rolfo, Ludovico Orlando Russo, Pascal Trotta
Politecnico di Torino
Dipartimento di Automatica e Informatica
Corso Duca degli Abruzzi 24, I-10129, Torino, Italy
Email: {name.familyname}@polito.it
Telephone: (+39) 011.090-7191

***Abstract*—Camera shake is a well-known source of degradation in digital images, as it introduces motion blur. Taking satisfactory photos under dim lighting conditions or using a hand-held camera is challenging. Same problems arise when camera is connected to mechanical equipments, that transfer vibrations to the camera itself. Since decades, many different theories and algorithms have been proposed with the aim of retrieving latent images from blurry inputs; most of them work quite well, but very often incur in large execution times. There are cases in which images have to be analyzed looking for features to be extracted; in this cases, it may be useful to consider deblurring as a pre-processing stage, that should not affect the performances of the whole image processing architecture, in terms of throughput.**

In this paper, an extensive survey of the deblurring algorithms that have been developed during the last 40 years is provided. Aim of this paper is to highlight software approaches that are able to quickly process input images and obtain good quality outcomes, analyzing the possibility of an hardware implementation to meet real-time requirements.

I. Introduction

Nowadays, computer vision is one of the most evolving areas of Information Technology (IT). Image processing is increasingly used in several application fields, such as medical [1], [2], aerospace [3], or automotive [4].

In every computer vision application, one or several images are taken from a camera, and processed, in order to extract information, used, for instance, for features identification [5], edge detection [6], or image registration [3].

However, there are cases in which it is not possible to rely on images' quality, as they may be affected by noise [7] or motion blur [8].

Restoring a blurry image has long been a challenging problem in digital imaging. It has been studied in depth from several points of view; a simple and quick research over the literature reveals that many studies have been done on this argument and hundreds of papers have been written during the last four decades (for instance [9], [10], [11]).

Several studies have focused on the task of recovering a latent image starting from an input blurry one (as in [12]).

Very often, authors have modeled the task as a two dimensional deconvolution process [13]. This simplification holds on when the blur is considered spatially invariant (or shift-invariant), meaning that every point in the original image spreads out the same way in forming the blurry image [14].

In fact, in this case, the blurry image is the result of the 2-D convolution of the real scene image with the blur kernel, also known as Point Spread Function (PSF) [15].

However, even in this simplistic case, to accomplish the deblurring task it is necessary to deal with 2-D deconvolution, that is well known to be an ill-conditioned and heavy task [16]. As 2-D convolution cannot be directly inverted (unless some restrictive hypothesis are verified [17]), it is necessary to perform complex mathematical operations to retrieve the real image hidden behind the blurry input [18].

For this reason, deblur algorithms are usually unable to achieve real-time performances, above all when high quality recovered outcomes are expected; this chance is common, because other subsequent algorithms after deblurring may succeed to extract information they need only from very sharp and detailed images [19].

When dealing with deblurring techniques, it must be taken into account that several kinds of agents can influence the process of taking and storing a photo. Very often, the blurring effect given by the atmosphere, rapid zooming or by aberrations on camera lenses can not be discarded if high-quality results are expected from the restoration process [20].

Photographical defocusing is another common type of blurring, known as out-of-focus blur, mainly due to the finite size of camera aperture [16].

In addition, camera shake is a common source of degradation in photographs [21]. It is well known how a picture can be severely degraded even by small movements of the camera, accentuated when the scene is poorly lighted and so a greater exposure time is required. In many situations there is simply not enough light to avoid using a long shutter speed, and so the result is inevitably blurry and disappointing [12] (as in Fig. 1).

Restoring blurred pictures is a challenging, and often severely underconstrained, problem, especially when both the blur kernel and the sharp image are unknown [21]. Curiously, it seems that researches in this field have always been more attracted by developing software solutions to the problem of deblurring (also proposing interesting but slow solutions as in [22]) rather than present ideas related to possible hardware implementations.

In fact, even if a deblurring approach aimed at successive hardware accelerations may reach real-time performances, still providing good quality outcomes, only few works have been proposed about this topic.

Aim of this paper is to analyze the current state-of-the-art providing a comprehensive survey on it, and highlight a software approach suitable for a successive hardware optimization, to

(a) Original image

(b) Blurry image

Figure 1: Example of the blur effect

be employed when real-time performances are needed.
The paper is organized as follows: Section II presents a comprehensive overview about existing deblurring approaches, Section III compares performances of selected algorithms, and Section IV summarizes the contributions and possible future works.

II. Deblurring Algorithms Overview

Some of the most important contributions and solutions to the problem of developing algorithms to recover latent images from blurry input ones are here listed.
The task of recovering the original image from an observed blurry one can be described by a two-dimensional image deconvolution problem. A commonly used notation is [13]:

$$f = g * p + n \quad (1)$$

where $*$ is the discrete 2-D convolution operator, g is the original image to recover (i.e., the one that would have been observed if no blur or noise occurred), f is the observed blurry and noisy image, p is the blur kernel (or PSF), and n is the noise affecting the image. It is common to model n as a White Gaussian noise, uncorrelated with the image g, although this consideration does not always hold [16].
In this paper, noise is not taken into consideration, knowing that in literature there have been proposed many different denoising algorithms (an interested reader may refer to [23], [24], [25]) and real-time denoising hardware architectures [7].
Depending on the availability of some previous knowledges acquired about p (that can also be obtainable from other sources, like from camera motion estimation), there are two categories of image deconvolution problems. If the blur kernel is given as a prior information, or it is somehow obtainable, recovering the original image becomes a *non-blind* deconvolution problem [26].
Non blind deconvolution requires the usage of some deconvolution operators, since convolution is (under certain assumptions) an invertible operation. However, it is widely known that deconvolution is an ill-conditioned inverse problem, as also tiny perturbations of the inputs may cause the direct solution from (1) being heavily distorted [27].
One of the very first works on this topic is presented in [28], where an iterative procedure is used for recovering a latent image that has been blurred by a known point spread function. Instead, if the blur kernel is unknown, reversing the effects of the convolution operator is a very challenging ill-conditioned inverse problem. In this case, the task is not only very sensitive to image noise, but it is also severely under-constrained, because the informations needed to estimate the blur kernel should be extracted from the blurred image itself. This issue is known as the problem of *blind* deconvolution [29].
Removing motion blur from images is a typical blind deconvolution problem [30], as the relative motion between the camera and the scene varies, and it is generally unknown. Instead, the task of removing blur induced by camera lens deformations could be expressed as a non-blind deconvolution problem, as informations about the blur kernel can be derived from previous knowledge about the technical characteristics of the lens and the camera.
The algorithm presented in [28] represents the basics for other successive works (e.g., [31], [32]). Its software implementations have been included as two built-in functions, named *deconvlucy* and *deconvblind*, in the Matlab tool [33]. *Deconvlucy* is an optimized implementation of the non-blind deconvolution algorithm proposed by Lucy, while *deconvblind* also includes an iterative kernel refinement process, implementing a simple deblurring algorithm that only need an initial estimation of the PSF (instead of its exact value).
On the other hand, some researchers have rised doubts about the model expressed by (1), arguing that it is inadequate to address all the components of a *natural* (i.e., not artificially induced) blur effect, and so it would lead to unacceptable errors and artefacts in the restored image.
In particular, the aforementioned model is accurate only if a shift-invariant motion blur is considered [34]. This exemplification holds where there is no significant parallax, any image-plane rotation of the camera is small and no parts of the scene are moving relative to one another during the exposure [35].
In order to address these issues, several more complex models have been discussed in literature. For example, in [9] authors deal with a different geometric model of the blur process that is expressed in terms of the rotational velocity of the camera during exposure. Authors state that even small rotations (less than a degree) of the camera may severely affect the task of taking a photo, causing large and non-uniform blur. For this reason, they propose a kernel estimation algorithm that does not rely on the assumption of blur effect uniformity.
Although considering non-uniform blur models may produce better results, this assumption introduces very complex operations, thus leading to performances that are very far from real-time requirements.
Another, totally different, approach to deblurring consists of formulating the blind deconvolution task as a joint minimization problem on both the blur kernel and the latent image. Algorithms that work on this direction aim at minimizing an energy cost function (e.g., the quadratic error between the latent image and the blurred input one) to produce a deblurred outcome; moreover, they intensively use regularization factors, to introduce additional information in order to reduce the ill-posedness of the problem [36] [37]. Among these methods, Total Variation (TV) norm and other derived methods have

been applied to solve the blind deblurring problem (e.g., [38]). TV regularization was introduced in [39]. This approach has represented a huge stride in the field of deblurring single images. In fact, algorithms based on it (e.g., [12] [40]) can produce particular good outcomes, almost regardless the type of blur considered (i.e., they can recover high quality images from input ones ruined by motion blur or out-of-focus blur…). In particular, it is possible to achieve impressive results when these approaches are used to remove modest motion blurring from images without rich textures. On the other hand, it seems that today recover high-quality latent images is still impossible if the blur kernel is big (e.g., greater than one hundred pixels). In this direction goes the approach described in [41], where authors propose to solve TV deconvolution problems by using the Alternating Direction Method (ADM). This approach can be applied to both single- and multi-channel images with either Gaussian or impulsive noise.

Despite the high quality of the outcomes, solving the optimization problem requires heavy computational cost on both the kernel and the image. For example, typical implementations of the minimization approach require up to eight minutes. For these reasons, the minimization approach cannot be slavishly pursued when a rapid and low-consuming approach to deblur images is needed.

Moreover, this class of approaches may results very expensive also from memory consumption point of view.

The motivation for regularizing with the TV $\mathbf{L_1}$ norm is that it is extremely effective for recovering edges of images, usually assuring very sharp outcomes.

From the development of the TV-norm optimization techniques, researchers have started analyzing more complex algorithms, taking into considerations typical characteristics of real word images. For example, natural images are intrinsically sparse in gradient domain. Authors in [42] state that since the contents of real-world images can vary significantly across different frames or different patches in the same image, they propose to learn various sets of bases from a precollected dataset of sample image patches. Then, for a given patch to be processed, adapt one set of bases to characterize the local sparse domain.

On the other hand, while working on sparse images and discontinuous blur, motion-basded blur or Gaussian blur (i.e., PSFs that does not show very sharp edges), the convergence rate of the regularization process is much slower (this has been highlighted by the experiments we have realized on this approach, and reported in the next Section).

The minimization approach is not suitable when real-time image processing is needed; the only possible solution is to use the model represented by (1) and accept the necessary simplifications it poses.

However, the necessity of simplification typically comes with some drawbacks. As reported in [26], in fact, deconvolved image usually contains unpleasant deconvolution visual artifacts (e.g., ringing artifacts) due to the ill-posedness of the restoration problem. Such disturbs, preventing the production of a high-quality outcome, are also known as *Gibbs phenomena*, and may be caused by problems related with zero and near-zero values in frequencies responses of blur kernels. "Kernels are often band-limited with a sharp frequency cut off; so, there will be zero (or near-zero) values in its frequency response. At those frequencies, the direct inverse of the kernel usually has a very large magnitude, causing excessive amplification of signal" [26]. Blur near edges produce, when amplified, ringing artifacts, because periodic overshoots and undershoots around the edge are introduced, which decay spatially away from the edge itself.

Ringing artifacts affect many of the already existing approaches to recover latent images based on deconvolution operations. As high frequency image components and details that have been destroyed in the blur process cannot be recovered by any algorithm, we have focused our efforts on analyzing the current literature looking for both non-blind and blind deblurring algorithms that can ensure the fulfillment of real-time requirements and the production of sharp and high-quality outcomes, with no ringings or unpleasant effects (that may prevent subsequent processing algorithm to succeed in their tasks).

Literature works can be grouped according to some common characteristics, like the usage of complex operations (that may not be resolved in real-time) and the number of input images required. In particular, this last feature is used to discriminate algorithms working on single images from ones that need several input images of the same subject to produce a high quality outcome.

Classical approaches to blind deblurring usually work by acquiring a single input image and trying to deblur it. In this case, prior assumptions have to be done, in order to reduce the number of unknowns that the approach have to calculate. Early works on this topic usually model the blur kernel using simple shapes and priors, as in [43].

Similar assumptions are very common in literature, because they ease the process of recovering latent images. On the other hand, these exemplifications may lead to poor results when applied to natural images, since distribution of gradients in natural scenes rarely follow precise parametric form [44].

Linear motion blur kernel model used in many works is very often overly simplified for true motion blurring [37]. To consider more complex motion blurring models, during the last years several multi-image based approaches have been proposed, aiming at obtaining information about the blur kernel by analysing multiple images of the same scene [45], [46]. Although these approaches have the advantage of discarding too simplistic (and often unrealistic) assumptions, they cannot be applied when it is necessary to work on single input images. Moreover, they sometimes require the usage of more than one camera, or to develop hardware solutions to contemporaneously take more than one picture of a scene.

[47] presents a hybrid camera system equipped with two imaging sensors. It can simultaneously captures high-resolution video together with a low-resolution video that has denser temporal sampling. Frames captured with higher temporal frequency are more resistance to blur, since the smaller camera occlusion time is, the fewer relative movements between camera and scene are. Using the different information retrievable at the same moment from the two sensors, this method aims at deblurring the frames in the high resolution video and at contemporaneously estimating new high-resolution video frames, producing a high-quality video with a higher temporal sampling density.

Although authors state that it is possible to achieve good results with their deblurring approach, it requires the usage of the hybrid camera they developed, and this is not always feasible. Similarly, in [48], authors propose to insert a patterned occluder within the aperture of the camera lens, creating a coded

aperture, using a criterion for depth discriminability. Using a statistical model of images, authors state their method can recover both depth information and an all-focus image from single photographs taken with the modified camera.
An interesting single-image deblurring approach, based on Hyper-Laplacian priors, is presented in [44]. Theoretical basis behind this method rely on the fact that typical gradients distributions in natural scene images have been proven to be well modelled by a Hyper-Laplacian distribution.
Authors present a minimization scheme that splits the problem into two separated sub-problems, to quickly solve the deconvolution task in the frequency domain. Both the two phases aim at minimizing a cost function to retrieve the most probable latent image.
The first sub-problem is separable among pixels. Authors propose an algorithm to quickly solve the first sub-problem by using a large Lookup Table (LUT) in which are stored pre-computed data that allows solving in a fast but approximate way (through simple linear interpolation) the minimization problem.
The second sub-problem is represented by a quadratic minimization problem. When circular boundary conditions are supposed, this can be easily translated in the frequency domain. Actually, this assumption does not always hold for real scene images, and may cause boundary artefacts to appear in the recovered image; on the other hand, input images can be enlarged and padded creating new borders, and so these effects can be easily neglected [26].

III. PERFORMANCES COMPARISONS

Previous Section has highlighted how a deblurring algorithm that works well in different circumstances is yet to be found. Some algorithms are fast but not robust, and so may be applied only on special classes of images, or when the exact PSF is known; others are very precise and obtain excellent results most of the times, but are impracticably slow and may require human interactions for large tuning phases.
To demonstrate this assertion with experimental results, we have developed a test environment in which, given a sharp 640x480 pixels grayscale images as input, a corresponding blurry one is obtained by using 2-D convolution (as in (1)) with a motion kernel. The test environment is based on Matlab R2012b, running on Windows 7 x64 on a Notebook PC equipped with an Intel Core i5-2450M @2.50GHz CPU and 8 GB of RAM.
Blur kernel is obtained by a filtering operation (implemented within the Matlab built-in command *fspecial*) that approximates a camera linearly moving on a straight line (31 pixels long with an angle with the horizontal axis of 11 degrees).
The test environment then invokes a deblurring function, giving the blurry image and the kernel matrix as inputs (for the algorithms that requires the PSF, also). It waits for the recovered image, and profiles the elapsed execution time of the deblurring algorithm. At the end, it computes the Root Mean Square Error (RMSE) and the Peak Signal-to-Noise Ratio (PSNR) between the restored image and the sharp initial image, to provide quantitative metrics of comparisons among algorithms.
For a given original image O and the correspondent latent image L, both of size $M \cdot N$ pixels, RMSE and PSNR are computed as:

$$\mathrm{RMSE}(L,O) = \sqrt{\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}(L_{ij}-O_{ij})^2}{M \cdot N}}, \tag{2}$$

$$\mathrm{PSNR}(L,O) = 10\ log_{10}\left(\frac{M \cdot N}{\sum_{i=1}^{M}\sum_{j=1}^{N}(L_{ij}-O_{ij})^2}\right). \tag{3}$$

The deblurring function is every time different, but all the functions have been uniformed to share the same interface and provide the restored image at the same way. All the algorithms have been optimized and tuned at the best of our efforts. Table I summarizes average results for the tested algorithms among 1,000 different VGA inputs, where * indicates that sometimes ringing artifacts appear in the retrieved image (they should disappear using information from coded camera architecture proposed by authors), ** refers to the Matlab implementation, that cause many unpleasant ringing artifacts in the outcomes, and *** refers to the Numipad implementation[1].

Table I: Results of the experiments

Algorithm	Avg El. Time (s)	Avg RMSE	Avg PSNR	Notes
[44]	0.830161	0.0652	23.7111	
[48]	0.963790	0.1176	18.5905	*
Deconvlucy	1.368123	0.0531	25.4981	**
[40]	1.873266	0.0708	22.9963	***
Deconvblind	4.868641	0.0534	25.4492	**
[41]	25.61191	0.1186	18.5205	
[42]	150.2101	0.0257	31.7895	

As discussed previously, TV norm-based algorithms [41] ensure the best results, in terms of RMSE and PSNR, but are not suitable for real-time applications, requiring a huge amount of time for their execution.
Similar considerations can be done on algorithms using other adaptive regularizations [42].
Deconvlucy, *Deconvblind*, and [40] still provide good quality outcomes, while drastically reducing the execution time, however still far from achieving real-time perfomances. In addition, approaches based on [28] (as *deconvlucy* and *deconvblind*) cause unpleasant artifacts, that may prevent subsequent algorithms of feature extractions, or edge detection, from obtaining the information they need.
On the other hand, approaches based on simpler operations [44] [48] obtain very fast, still acceptable results. Their execution times are below the second, while providing RMSE and PSNR values comparable to the other methods. However, [48] can be successful applied only when cameras with coded occlusure are available. It is worth noting that tests for this algorithm have been performed without the coded occlusure. It is expected that RMSE and PSNR values will improve by using the camera suggested by the authors, while the execution time remains constant.
Nonetheless, this approach is outperformed (in terms of execution time) by the approach proposed in [44] that seems the best compromise achievable between performances and quality of the outcomes. Moreover, it only uses simple operation in the frequency domain, and so it is a good candidate to be implemented in hardware, in order to achieve real-time

[1] http://numipad.sf.net

(a) Original image

(b) Blurry input image

(c) Image restored using [42]

(d) Image restored using [44]

Figure 2: Recovered images using [42] and [44]

performances.
Fig. 2 shows an example of blurry image restored using [44], the fastest algorithm, and [42], which provides the best results, in terms of quality.
From Fig. 2d and Fig. 2c it can be noted that the fast algorithm proposed in [44] provides acceptable results, if compared to the ones provided by [42], that incurs in a huge execution time.

IV. Conclusion

This paper presents a short, but comprehensive, overview about deblurring algorithms. The problem of retrieving latent images from single blurry inputs is of relevance in many scientific fields, ranging from medicine to spatial exploration. More generally, deblur may be of importance in all the computer vision applications.
This paper highlights how, even if deblur is studied since 40 years and a lot of works have been produced on this topic (and hundreds of algorithms have been discussed and patented), a definitive solution is yet to be found, above all when real-time performances are required and high quality outcomes are expected.

After several tests and considerations, this paper proposes the deblurring approach based on Hyper-Laplacian priors that is presented in [44] as a good solution for non-blind image deblurring, as it is a fast and itherative algorithm capable of producing high-quality outcomes. Moreover, this approach relies on simple operation in the Fourier domain, and iterative loop may be easily unrolled, so that this algorithm is suitable for an hardware implementation. Hardware accelerationon applied to this algorithm can surely ad to achieve real-time performances.

References

[1] S. Kumar and R. Amutha, "Edge detection of angiogram images using the classical image processing techniques," in *Proc. of International Conference on Advances in Engineering, Science and Management (ICAESM)*, pp. 55 – 60, 2012.

[2] V. Varghees, M. Manikandan, and R. Gini, "Adaptive MRI image denoising using total-variation and local noise estimation," in *Proc. of International Conference on Advances in Engineering, Science and Management (ICAESM)*, pp. 506 – 511, 2012.

[3] G. Troglio, J. Le Moigne, J. Benediktsson, G. Moser, and S. Serpico, "Automatic extraction of ellipsoidal features for planetary image registration," *IEEE Geoscience and Remote Sensing Letters*, vol. 9, no. 1, pp. 95 –99, 2012.

[4] A. Aponso and N. Krishnarajah, "Review on state of art image enhancement and restoration methods for a vision based driver assistance system with de-weathering," in *Proc. of International Conference of Soft Computing and Pattern Recognition (SoCPaR)*, pp. 135 – 140, 2011.

[5] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proc. of the 4th Alvey Vision Conference*, pp. 147 – 151, 1988.

[6] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679 – 698, 1986.

[7] S. Di Carlo, P. Prinetto, D. Rolfo, and P. Trotta, "AIDI: An adaptive image denoising fpga-based ip-core for real-time applications," in *Adaptive Hardware and Systems (AHS), 2013 NASA/ESA Conference on*, pp. 99–106, June 2013.

[8] L. O. Russo, G. Airò Farulla, M. Indaco, S. Rosa, D. Rolfo, and B. Bona, "Blurring prediction in monocular slam," in *International Design & Test Symposium. 2013. Proceedings. 2013 International IEEE Conference on*.

[9] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce, "Non-uniform deblurring for shaken images," *International Journal of Computer Vision*, vol. 98, no. 2, pp. 168–186, 2012.

[10] W. Wang and M. K. Ng, "On algorithms for automatic deblurring from a single image," *Journal of Computational Mathematics*, vol. 30, no. 1, pp. 80–100, 2012.

[11] X. Chen, X. He, J. Yang, and Q. Wu, "An effective document image deblurring algorithm," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 369–376, IEEE, 2011.

[12] Q. Shan, J. Jia, and A. Agarwala, "High-quality motion deblurring from a single image," in *ACM Transactions on Graphics (TOG)*, vol. 27, p. 73, ACM, 2008.

[13] J.-F. Cai, H. Ji, C. Liu, and Z. Shen, "Framelet-based blind motion deblurring from a single image," *Image Processing, IEEE Transactions on*, vol. 21, no. 2, pp. 562–572, 2012.

[14] M. M. Sondhi, "Image restoration: The removal of spatially invariant degradations," *Proceedings of the IEEE*, vol. 60, no. 7, pp. 842–853, 1972.

[15] M. Cannon, "Blind deconvolution of spatially invariant image blurs with phase," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 24, no. 1, pp. 58–63, 1976.

[16] P. Campisi and K. Egiazarian, *Blind image deconvolution: theory and applications*. CRC press, 2007.

[17] R. Rice, "Inverse convolution filters," *Geophysics*, vol. 27, no. 1, pp. 4–18, 1962.

[18] D. Krishnan, T. Tay, and R. Fergus, "Blind deconvolution using a normalized sparsity measure," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 233–240, IEEE, 2011.

[19] A. Rosenfeld and J. S. Weszka, "Picture recognition and scene analysis," *Computer*, vol. 9, no. 5, pp. 28–38, 1976.

[20] Z. Myles and N. da Vitoria Lobo, "Recovering affine motion and defocus blur simultaneously," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 20, no. 6, pp. 652–658, 1998.

[21] T. S. Cho, S. Paris, B. K. Horn, and W. T. Freeman, "Blur kernel estimation using the radon transform," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 241–248, IEEE, 2011.

[22] Z. Hu, J.-B. Huang, and M.-H. Yang, "Single image deblurring with adaptive dictionary learning," in *Image Processing (ICIP), 2010 17th IEEE International Conference on*, pp. 1169–1172, IEEE, 2010.

[23] A. Buades, B. Coll, and J.-M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Modeling & Simulation*, vol. 4, no. 2, pp. 490–530, 2005.

[24] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of gaussians in the wavelet domain," *Image Processing, IEEE Transactions on*, vol. 12, no. 11, pp. 1338–1351, 2003.

[25] J.-L. Starck, E. J. Candès, and D. L. Donoho, "The curvelet transform for image denoising," *Image Processing, IEEE Transactions on*, vol. 11, no. 6, pp. 670–684, 2002.

[26] L. Yuan, J. Sun, L. Quan, and H.-Y. Shum, "Progressive inter-scale and intra-scale non-blind image deconvolution," in *ACM Transactions on Graphics (TOG)*, vol. 27, p. 74, ACM, 2008.

[27] R. Neelamani, H. Choi, and R. Baraniuk, "Forward: Fourier-wavelet regularized deconvolution for ill-conditioned systems," *Signal Processing, IEEE Transactions on*, vol. 52, no. 2, pp. 418–433, 2004.

[28] L. Lucy, "An iterative technique for the rectification of observed distributions," *The astronomical journal*, vol. 79, p. 745, 1974.

[29] D. Kundur and D. Hatzinakos, "Blind image deconvolution," *Signal Processing Magazine, IEEE*, vol. 13, no. 3, pp. 43–64, 1996.

[30] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 1964–1971, IEEE, 2009.

[31] D. Fish, A. Brinicombe, E. Pike, and J. Walker, "Blind deconvolution by means of the richardson–lucy algorithm," *JOSA A*, vol. 12, no. 1, pp. 58–65, 1995.

[32] Y.-W. Tai, P. Tan, and M. S. Brown, "Richardson-lucy deblurring for scenes under a projective motion path," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 8, pp. 1603–1618, 2011.

[33] MATLAB, *R2013b*. The MathWorks Inc.

[34] S. Cho and S. Lee, "Fast motion deblurring," in *ACM Transactions on Graphics (TOG)*, vol. 28, p. 145, ACM, 2009.

[35] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman, "Removing camera shake from a single photograph," in *ACM Transactions on Graphics (TOG)*, vol. 25, pp. 787–794, ACM, 2006.

[36] A. Blumer, A. Ehrenfeucht, D. Haussler, and M. K. Warmuth, "Occam's razor," *Information processing letters*, vol. 24, no. 6, pp. 377–380, 1987.

[37] J.-F. Cai, H. Ji, C. Liu, and Z. Shen, "Blind motion deblurring from a single image using sparse approximation," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 104–111, IEEE, 2009.

[38] T. F. Chan and C.-K. Wong, "Total variation blind deconvolution," *Image Processing, IEEE Transactions on*, vol. 7, no. 3, pp. 370–375, 1998.

[39] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, no. 1, pp. 259–268, 1992.

[40] P. Rodriguez and B. Wohlberg, "An l1-tv algorithm for deconvolution with salt and pepper noise," 2009.

[41] M. Tao, J. Yang, and B. He, "Alternating direction algorithms for total variation deconvolution in image reconstruction," *TR0918, Department of Mathematics, Nanjing University*, 2009.

[42] W. Dong, L. Zhang, G. Shi, and X. Wu, "Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization," *Image Processing, IEEE Transactions on*, vol. 20, no. 7, pp. 1838–1857, 2011.

[43] G. Pavlovic and A. M. Tekalp, "Maximum likelihood parametric blur identification based on a continuous spatial domain model," *Image Processing, IEEE Transactions on*, vol. 1, no. 4, pp. 496–504, 1992.

[44] D. Krishnan and R. Fergus, "Fast image deconvolution using hyper-laplacian priors," in *Advances in Neural Information Processing Systems*, pp. 1033–1041, 2009.

[45] J. Chen, L. Yuan, C. Tang, and L. Quan, "Robust dual motion deblurring," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pp. 1–8, IEEE, 2008.

[46] A. Rav-Acha and S. Peleg, "Two motion-blurred images are better than one," *Pattern Recognition Letters*, vol. 26, no. 3, pp. 311–317, 2005.

[47] Y.-W. Tai, H. Du, M. S. Brown, and S. Lin, "Correction of spatially varying image and video motion blur using a hybrid camera," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 6, pp. 1012–1028, 2010.

[48] A. Levin, R. Fergus, F. Durand, and W. T. Freeman, "Image and depth from a conventional camera with a coded aperture," *ACM Transactions on Graphics (TOG)*, vol. 26, no. 3, p. 70, 2007.