Scientific Research and Essays Vol. 6(18), pp. 3834-3844, 1 September, 2011
Available online at http://www.academicjournals.org/SRE
DOI: 10.5897/SRE11.436
ISSN 1992-2248 ©2011 Academic Journals

Full Length Research Paper

Robust super-resolution using kernel regression with outliers-reduction scheme

Feng Xu[1], Lizhong Xu[1,2,3]*, Fengchen Huang[1,2], Chenrong Huang[4] and Aiye Shi[1,2,3]

[1]College of Computer and Information Engineering, Hohai University, 210098 Nanjing, P.R. China.
[2]Institute of Information and Control Engineering, Hohai University, 210098 Nanjing, P.R. China.
[3]Engineering Center of Telemetry and Remote Sensing and Information System, Hohai University, 210098 Nanjing, P.R. China.
[4]School of Computer Engineering, Nanjing Institute of Technology, 211167 Nanjing, P.R. China.

Accepted 27 June, 2011

In the process of recording a digital image, super-resolution (SR) is a feasible soft method for solving the limitation of device and effect of environment. During the last two decades, many researchers proposed various SR algorithms for image reconstruction. Among these algorithms, kernel regression is a helpful tool which considers not only spatial distance between center pixel and neighbor pixel but also structural information. However, the problem of removing noise and outlier in kernel regression can be further studied and solved. In this paper, we proposed a new idea of so called trilateral kernel regression. Besides the above factors considered, the new kernel regression with trilateral idea considers additional factor: Confident correlation of pixels, so it can obtain more accurate result. Experiments are carried out to demonstrate the effectiveness of our method. The index of RMSE can be reduced by 2 or even 4 in some severe case.

Key words: Super-resolution, kernel regression, trilateral kernel, outliers reduction, confident index.

INTRODUCTION

In many digital imaging applications areas, such as remote sensing, surveillance, medical, and video entertainment, the high-resolution (HR) images that give not only enough pixels density but also abundant details are desired to be convenient for post processing. If wanting to reach the above goal through modifying hardware, one must increase the number of sensor pixels in imaging device. But this technology has reach its limit beyond which will possibly cause many disgusting shot noises and increase the cost. In order to overcome these drawbacks, an alternatively economical and effective approach called super-resolution (SR) is widely researched and used. SR is technique that constructs HR images from several observed low-resolution (LR) images, thereby increasing the high-frequency components and removing the degradations caused by the imaging process of the low-resolution camera (Milanfar, 2010). In fact, the process of SR can be viewed as a reverse flow for LR images generative model which is demonstrated in Figure 1.

The concept SR was first mentioned by Tsai and Huang (1984), and then many researchers followed this trace to extend the method (Bose et al., 1993; Su and Kim, 1994; Tom et al., 1994). However they solved the resolution problem in frequency domain that cannot tackle complicated imaging model. For more flexibility in overcoming the difficulty in frequency domain, SR in spatial domain was proposed, adopted and researched by more and more researchers (Elad et al., 2004). There are many algorithms in spatial domain such as non uniform interpolation approach (Takeda et al., 2007), statistical approach (Pickup et al., 2009), set theoretic approach (Patti and Altunbasak, 2001), example-based approach (Elad and Datsenko, 2007) and others (Tian,

*Corresponding author. E-mail: xufengxz@gmail.com.

Abbreviations: SR, Super-resolution; **HR,** high-resolution; **LR,** low-resolution; **NC,** normalized convolution; **RMSE,** root of mean squared error.

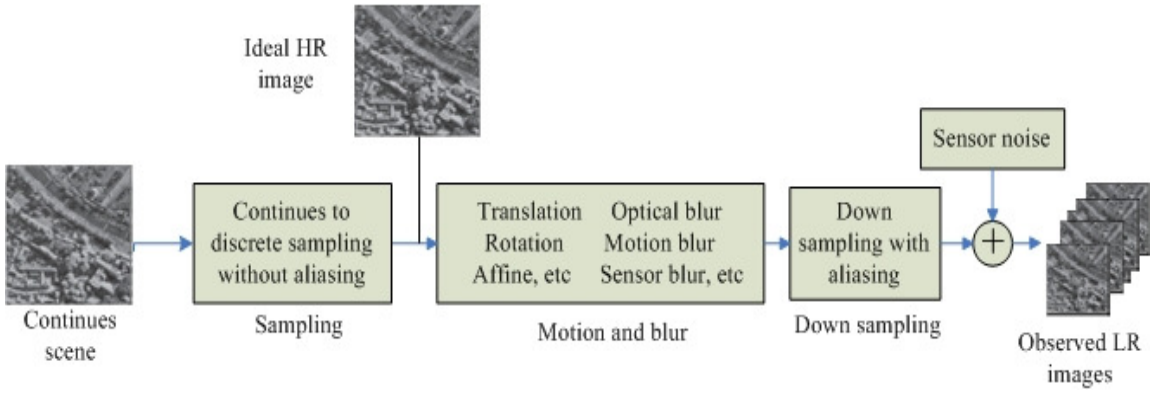

Figure 1. The generative model of a real imaging system relating a high resolution image to the low-resolution observation frames with motion between the scene and the camera.

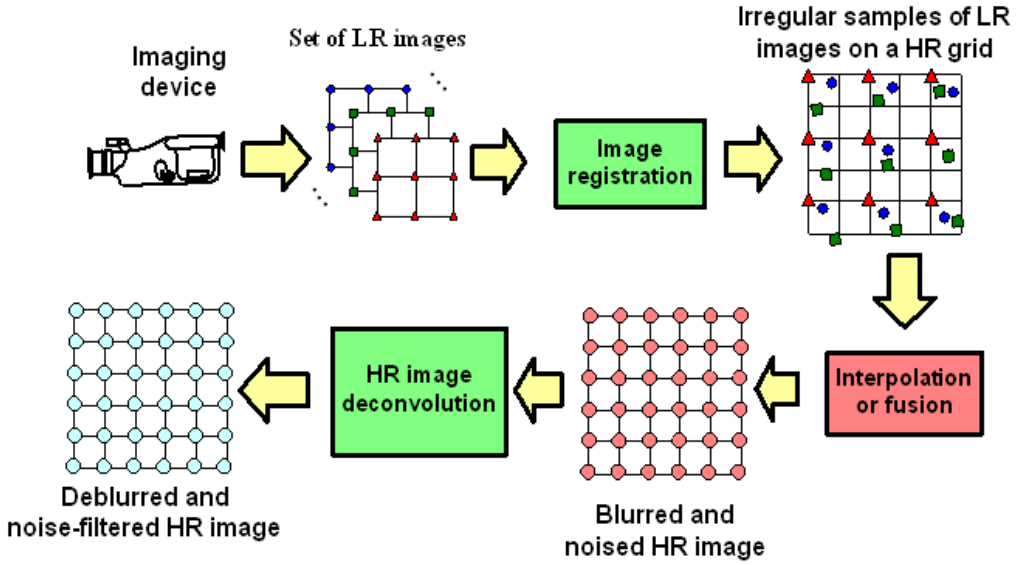

Figure 2. The SR reconstruction flow with three-step method from a set of noised and blurred LR images with motion to state-of-the-art image.

2010). Generally, all approaches may be divided into two categories according to processing flow: one is three-step method registration-fusion-deconvolution and the other is two-step method registration-restoration.

In this paper we focus on the three-step method for SR. Each step of this method can be described as follows. Firstly, the subpixel motion parameters of LR images are estimated, and then used to pursue registration that puts each LR image onto a HR (or subpixel spaced) grid. In most cases, the LR image positions on non-integral points, so there is a non uniform pixel distributed HR image on the HR grid. In order to form a uniform HR image, a kind of interpolation algorithm is employed to get intensities of all integral points. Above procedure, that is fusion, is the second step. Thirdly, for the noise and blurring in fused image are not removed in previous steps, deconvolution is a necessary process that can use one of the existing or novelly proposed restoration algorithm. Figure 2 illustrates the schematic representation of three-step method for SR.

An efficient wavelet-based interpolation using the interlacing sampling structure in LR images was proposed by Nguyen and Milanfar (2010) to reconstruct SR image from non uniform sampled data, but it only aimed at speeding up the computation. Alam et al. (2000) proposed a weighted nearest neighbors method to interpolate non uniform sampled images, and then use Wiener filtering to deconvolute. In order to remove degradation during generation of images, Elad and Hel-Or (2001) presented a very computationally efficient

algorithm, while they just consider special case of pure translation, space invariant blur, and additive Gaussian noise. Lertrattanapanich and Bose (2002) organized the irregularly sampled data as Delaunay triangles, and then interpolated the uniform data within each triangle. However this interpolation is very sensitive to noise. In fact, under the outlier circumstance, the surface fit interpolation is superior for fusion than conventional interpolation. Farneback (2002) proposed and extended the facet model for surface fit. But they all consider only uniform data and cannot adapt the method to structure of image. Pham et al. (2006) noticed the drawback of the above fusion method. Based on normalized convolution (NC), they introduced a Gaussian certainty and a structure adaptive applicability function to the polynomial facet model and applied it to fusion of irregularly sampled data. Recently, Takeda et al. (2010) proposed an adaptive steering kernel regression for interpolation on the HR image grid where the LR images are registered and mapped on. The adaptive steering kernel regression is valid when the noises follow a kind of distribution model. However, disgusting outliers usually appear on the image; thus influence the results severely. Unfortunately, Takeda's kernel regression cannot perfectly resolve this problem.

The aim of this paper is to improve the performance of kernel regression and enable it to remove outlier (perhaps it is salt and pepper noise or occasional mis-registration) more effective, we set up outlier detection and removing rule to propose a novel scheme of kernel regression. The new scheme considers not only local structure of image to be adapted but also the outlier removal making the real and valuable pixels to attend the kernel regression.

Kernel regression based super-resolution

In this section, we will review ideas of classic kernel regression and its extended adapted kernel regression. After that, we describe the procedure for application of kernel regression to super-resolution, as well as the weakness of this scheme, and motivate the improvement of conventional method.

Classic kernel regression

Kernel regression method, in fact, is a non-parametric interpolation or fitting computing which has been widely used in many science and engineering areas such as pattern discrimination and intelligent computing. Compared to the parametric interpolation which depends on the specific interested signal model with parameter, the kernel regression only depends on local data to determine the model structure without any parameter. Once the model is decided through explicit parametric method or implicit non-parametric method, the interpolation can be implemented. Apparently, the image is a kind of signal with two dimensions, so we formulate the kernel regression with image scenario as follows:
Assuming the real image function is $z(\mathbf{x})$, we can represent noisy version of $z(\mathbf{x})$ as:

$$y(\mathbf{x}) = z(\mathbf{x}) + \boldsymbol{\varepsilon}, \qquad \mathbf{x} = [x_1, x_2]^T, \tag{1}$$

Where, $y(\mathbf{x})$ is observed image function stained by noise, $\mathbf{x}$ is coordinates of pixels in two dimension image grid, and $\boldsymbol{\varepsilon}$ is independent identically distributed stochastic noise with zero mean distribution. The task of kernel regression is to make use of local (neighbor) observed pixels of $y(\mathbf{x})$ to estimate point-wise pixels of $z(\mathbf{x})$.

For a specific pixel $z(\mathbf{x}_i)$ sampled at position $\mathbf{x}_i$, we can expand it in a neighborhood centered at $\mathbf{x}$ using the Nth order Taylor series

$$z(\mathbf{x}_i) = z(\mathbf{x}) + \{\nabla z(\mathbf{x})\}^T (\mathbf{x}_i - \mathbf{x}) + (1/2)(\mathbf{x}_i - \mathbf{x})^T \{\boldsymbol{H} z(\mathbf{x})\}(\mathbf{x}_i - \mathbf{x}) + \cdots \tag{2}$$

$$= \beta_0 + \boldsymbol{\beta}_1^T (\mathbf{x}_i - \mathbf{x}) + \boldsymbol{\beta}_2^T \operatorname{vech}\{(\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^T\} + \cdots \tag{3}$$

Where, ∇ and $\boldsymbol{H}$ are the gradient (2 × 1) and Hessian (2 × 2) operators, respectively, and vech(•) is the half-vectorization operator which lexicographically orders the lower triangular portion of a symmetric matrix into a column-stacked vector, Equation 3 represents two examples for intuition.

$$\operatorname{vech}\left(\begin{bmatrix} a & b \\ b & c \end{bmatrix}\right) = [a \quad b \quad c]^T$$
$$\operatorname{vech}\left(\begin{bmatrix} a & b & c \\ b & d & e \\ c & e & f \end{bmatrix}\right) = [a \ b \ c \ d \ e \ f]^T \tag{4}$$

Comparison of Equations 2 and 3 make it clear that β_0 is the pixel value of interest, and the vectors $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$ are the first and second derivatives, respectively. that is,

$$\beta_0 = z(\mathbf{x}) \tag{5}$$

$$\boldsymbol{\beta}_1 = \nabla z(\mathbf{x}) = \left[\frac{\partial z(\mathbf{x})}{\partial x_1} \quad \frac{\partial z(\mathbf{x})}{\partial x_2}\right]^T \tag{6}$$

$$\boldsymbol{\beta}_2 = \frac{1}{2}\left[\frac{\partial^2 z(\mathbf{x})}{\partial x_1^2} \quad 2\frac{\partial^2 z(\mathbf{x})}{\partial x_1 \partial x_2} \quad \frac{\partial^2 z(\mathbf{x})}{\partial x_2^2}\right]^T \tag{7}$$

The $\boldsymbol{\beta}_n$ is computed from the following optimization problem:

$$\min_{\{\boldsymbol{\beta}_n\}} \sum_{i=1}^{P} \Big[y_i - \beta_0 - \boldsymbol{\beta}_1^T (\mathbf{x}_i - \mathbf{x}) - \boldsymbol{\beta}_2^T \operatorname{vech}\left\{(\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^T\right\} - \cdots \Big]^2 K_{\boldsymbol{H}}(\mathbf{x}_i - \mathbf{x}) \tag{8}$$

Where, *K* is the 2-D realization of the kernel function which assigns different weight for each residual error, and *H* is the 2 × 2 smoothing matrix which controls the scope of neighborhood. For classic kernel regression, $\boldsymbol{H} = h\boldsymbol{I}$, where *h* is scalar, $\boldsymbol{I}$ is an identity matrix. In general, K is chosen as type of Gaussian function which assigns weight according to distance between interested pixel and measured pixel. So the kernel function is formulated as:

$$K_{\boldsymbol{H}}(\mathbf{x}_i - \mathbf{x}) = \frac{\boldsymbol{1}}{2\pi\sqrt{\det\left(\boldsymbol{H}\boldsymbol{H}^T\right)}} \exp\left\{-\frac{1}{2}(\mathbf{x}_i - \mathbf{x})^T \left(\boldsymbol{H}\boldsymbol{H}^T\right)^{-1}(\mathbf{x}_i - \mathbf{x})\right\} \tag{9}$$

If ignoring the regression order (N) and the dimensionality of $z(\mathbf{x})$, we can see that the optimization problem (8) actually is a weighted least squares optimization problem. It can also be expressed as,

$$\hat{\mathbf{b}} = \arg\min_{\mathbf{b}} \left[(\mathbf{y} - \mathbf{X}\mathbf{b})^T \mathbf{K} (\mathbf{y} - \mathbf{X}\mathbf{b}) \right] \tag{10}$$

Where,

$$\mathbf{b} = \left[\beta_0 \ \boldsymbol{\beta}_1 \cdots \boldsymbol{\beta}_N \right]^T \tag{11}$$

$$\mathbf{y} = \left[\mathrm{y}_1 \ \mathrm{y}_2 \cdots \mathrm{y}_\mathrm{P} \right]^T \tag{12}$$

$$\mathbf{K} = \operatorname{diag}\left[K_{\boldsymbol{H}}(\mathbf{x}_1 - \mathbf{x}) \ K_{\boldsymbol{H}}(\mathbf{x}_2 - \mathbf{x}) \cdots K_{\boldsymbol{H}}(\mathbf{x}_P - \mathbf{x}) \right] \tag{13}$$

$$\mathbf{X} = \begin{bmatrix} \mathbf{1} & (\mathbf{x}_1 - \mathbf{x})^T & \operatorname{vech}^T\left\{(\mathbf{x}_1 - \mathbf{x})(\mathbf{x}_1 - \mathbf{x})^T\right\} & \cdots \\ \mathbf{1} & (\mathbf{x}_2 - \mathbf{x})^T & \operatorname{vech}^T\left\{(\mathbf{x}_2 - \mathbf{x})(\mathbf{x}_2 - \mathbf{x})^T\right\} & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{1} & (\mathbf{x}_P - \mathbf{x})^T & \operatorname{vech}^T\left\{(\mathbf{x}_P - \mathbf{x})(\mathbf{x}_P - \mathbf{x})^T\right\} & \cdots \end{bmatrix} \tag{14}$$

By setting zero to derivative of cost function, Equation 10 can be further solved and gives an explicit solution;

$$\begin{aligned} \hat{\mathbf{b}} &= \left[\hat{\beta}_0 \ \hat{\boldsymbol{\beta}}_1 \cdots \hat{\boldsymbol{\beta}}_N \right]^T \\ &= \left[\hat{z}(\mathbf{x}) \ \frac{\partial \hat{z}(\mathbf{x})}{\partial x_1} \ \frac{\partial \hat{z}(\mathbf{x})}{\partial x_2} \ \frac{\partial^2 \hat{z}(\mathbf{x})}{2\partial {x_1}^2} \ \frac{\partial^2 \hat{z}(\mathbf{x})}{\partial x_1 \partial x_2} \ \frac{\partial^2 \hat{z}(\mathbf{x})}{2\partial {x_2}^2} \cdots \right]^T \\ &= \left(\mathbf{X}^T \mathbf{K} \mathbf{X}\right)^{-1} \mathbf{X}^T \mathbf{K} \mathbf{y} \end{aligned} \tag{15}$$

To sum up so far, from the final result, (Equation 15) we can say that the classic kernel regression is actually the application of a linear weighted average of local data toestimate point-wise desired regression model. Obviously, the weights play a key role during estimating. The classic kernel regression assigns weight just according to spatial distance or correlation (that is, the farther the distance, the lower correlation thus lower weight and vice versa). However, in image the correlation of pixels involve not only spatial distance but also diverse relation such as photometric, radiometric, structure and so on. Consequently, in next subsection, we will formulate the adaptive kernel regression and briefly analyze its merit and drawback.

Structure-adapting kernel regression

In contrast with classic kernel regression, the most important difference of adaptive kernel regression is the kernel function $K_{\boldsymbol{H}}$ is not a radio symmetric (or Centro symmetric) recessionary mask that is relevant to local sample values. Therefore, from (Equation 15) we can derivate a non-linear combination of local data not as simple as previous but finer and more vivid.

Detailed size and shape of kernel function is decided based on the local data structure information which is the result of analyzing the radiometric (pixel value) differences locally.

Similar to previous derivation, adaptive kernel regression has the same form of optimization problem (Equation 8). But being different from classic kernel regression, the matrix ***H*** is not a scalar multiple of the identity with the global parameter *h*, thus has equal effect along the x1- and x2-directions. We define the matrix ***H*** as follow;

$$\boldsymbol{H}_i = h\boldsymbol{C}_i^{\frac{1}{2}} \tag{16}$$

Which is called the steering matrix by Takeda et al. (2007) and where, for each given sample *yi*, the matrix *Ci* is estimated as the local covariance matrix of the neighborhood spatial gradient vectors. A naive estimate of this covariance matrix may be obtained as;

$$\hat{\boldsymbol{C}}_i = \boldsymbol{J}_i^T \boldsymbol{J}_i \tag{17}$$

with;

$$\boldsymbol{J}_i = \begin{bmatrix} \vdots & \vdots \\ z_1(\boldsymbol{x}_j) & z_2(\boldsymbol{x}_j) \\ \vdots & \vdots \end{bmatrix} \tag{18}$$

Where, $z_1(\bullet)$ and $z_2(\bullet)$ are the first derivatives along x1- and x2-axes, the number of rows in this matrix equates to

the number of samples in the local analyzed neighborhood centered at interested point $\boldsymbol{x}_i$. Undoubtedly, the method in (Equation 17) is simple but often unstable or rank deficient. Consequently, sometimes an effective alternative is to use singular value decomposition (SVD) of $\boldsymbol{J}_i$ to construct covariance matrix $\boldsymbol{C}_i$.

Based on the estimated $\boldsymbol{H}_i$, (Equation 9) and (Equation 16), the adaptive kernel function (steering kernel) can be formulated as:

$$K_{\boldsymbol{H}_i}(\mathbf{x}_i - \mathbf{x}) = \frac{\sqrt{\det(\boldsymbol{C}_i)}}{2\pi h^2} \exp\left\{-\frac{1}{2h^2}(\mathbf{x}_i - \mathbf{x})^T \boldsymbol{C}_i (\mathbf{x}_i - \mathbf{x})\right\} \quad (19)$$

Thus, according to Equations 10 to 15, we can obtain the more accurate result.

Super-resolution using kernel regression

Because adaptive kernel regression is superior in interpolation than classic kernel regression for considering image structure apart from distance correlation, it can perform well in super-resolution.
In other word, either classic kernel regression or adaptive only aims at fusion task in super-resolution, and cannot implement the entire super-resolution process. The rest task must be added in the completed flow including registration and disconvolution. We represent a detailed flow of super-resolution using kernel regression as follow:
Step 1: Read the entire noisy and blurred LR image with stochastic motion.
Step 2: Estimate the motion parameters in sub pixel accurate precision for each LR image according to a reference (in general the 1st) LR image.
Step 3: Put every LR image onto a standard HR grid according to their motion parameter, that is, registration.
Step 4: Interpolate previous irregular HR grid image onto integer point using classic kernel regression.
Step 5: Compute the covariance matrix $\boldsymbol{C}_i$ for each point using the derivatives obtained in the step 4
Step 6: According to Equation 19 compute the adaptive kernel function
Step 7: Implement the adaptive kernel regression interpolation
Step 8: Deblur, that is, deconvolute the up scaled image thus get the HR image approach ideal image as better as possible.

NEW FRAMEWORK FOR SUPER-RESOLUTION

In traditional adaptive kernel regression, the rule for determining valid weight corresponding to pixels is based on two factors. One factor is spatial correlation which assigns high weights to pixels with short distance between interested pixel and measured irregular positioned sample and assigns low weights to pixels with long distance. The other factor is photometric or structural correlation which assigns high weights to pixels with low residual error between interested measurement and neighbor measurement in one local analyzed window and assigns low weights to pixels with high residual error. We notice that this scheme is similar to the spatial difference weights of bilateral filter, in ideally assumed case such as Gaussian or Laplacian noise, this scheme performed well. While in most case, the observed LR images contain some outliers due to the poignant electromagnetic disturb, the error caused by registration or blur kernel, occlusion and so on, there is no corresponding effective mechanism to remove these outliers in the kernel regression. Kernel regression does not consider the pesky bad pixels detection and reduction even removal, so there is a little bit pity in the frame of kernel regression.

As that discussed above, in order to process them properly we consider existing of outliers in observed images and proposed an outlier-reduction scheme incorporating kernel regression for SR with robustness. According to the similar derivation from Equations 8 to 15, we present a little bit different formulation. The primary modifying lies in Equations 8 and 13. We give a new version of optimal problem (Equation 8) as follow:

$$\min_{\{\boldsymbol{\beta}_n\}} \sum_{i=1}^{P} \Big[y_i - \beta_0 - \boldsymbol{\beta}_1^T (\mathbf{x}_i - \mathbf{x}) - \boldsymbol{\beta}_2^T \operatorname{vech}\left\{(\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^T\right\} - \cdots \Big]^2 K_{\boldsymbol{H}}(\mathbf{x}_i - \mathbf{x}) \boldsymbol{S}_i \quad (20)$$

And change Equation 13 into the follow:

$$\mathbf{K} = \operatorname{diag}\left[K_{\boldsymbol{H}}(\mathbf{x}_1 - \mathbf{x}) \boldsymbol{S}_1 \quad K_{\boldsymbol{H}}(\mathbf{x}_2 - \mathbf{x}) \boldsymbol{S}_2 \quad \cdots \quad K_{\boldsymbol{H}}(\mathbf{x}_P - \mathbf{x}) \boldsymbol{S}_P \right] \quad (21)$$

The difference consisted in Equations 8 and 13 is the symbol $\boldsymbol{S}_i$ which is a tool to detect and remove outlier. In fact, $\boldsymbol{S}_i$ still is a weight assigned to correspond pixel residual error according to the pixel reliability. When the value of y_i is in low confidence, $\boldsymbol{S}_i$ will be low; the other way round, when the value of y_i is in high confidence, $\boldsymbol{S}_i$ will be high. A typical definition of $\boldsymbol{S}_i$ is formulated as follow:

$$\boldsymbol{S}_i = \exp\left(-\frac{(y_i - \hat{y}_i)^2}{\sigma^2} \right) \quad (22)$$

Where, the $\hat{y}_i$ is the estimated initialization which can be obtained with any interpolation including nearest, bilinear, even the kernel regression discussed in this paper. The

σ defines a tolerant degree of $(y_i - \hat{y}_i)$ which is set artificially by experience. Samples with residual error less than σ get weight close to one, whereas those with larger residual error get extremely low confidence.

It can be found that the scheme we proposed is similar to the idea of bilateral filter, but it considers three factors: the first is spatial correlation; the second is photometric or structural correlation; besides above two factors included in idea of bilateral filter, the third is the confidence correlation between real and estimated value. Thus due to the three factors and enlightened by the name bilateral filter, we called this optimization scheme as trilateral scheme.

Now we rewrite the flow of super-resolution using trilateral kernel regression as follow:

Step 1: Read the entire noisy and blurred LR image with stochastic motion

Step 2: Estimate the motion parameters in sub pixel accurate precision for each LR image according to a reference (in general the 1st) LR image

Step 3: Put every LR image onto a standard HR grid according to their motion parameter, that is, registration

Step 4: Interpolate previous irregular HR grid image onto integer point using classic kernel regression

Step 5: Compute the covariance matrix $\boldsymbol{C}_i$ for each point using the derivatives obtained in the step 4

Step 6: Compute the confidence $\boldsymbol{S}_i$ for each point using the derivatives obtained in the step 4

Step 7: According to Equation 19, compute the adaptive kernel function

Step 8: Implement the adaptive kernel regression interpolation

Step 9: Deblur, that is, deconvolute the up scaled image; thus get the HR image approach ideal image as better as possible

EXPERIMENTS

In order to demonstrate the validity of proposed methods, we carry out four groups of experiments. The former two is SR reconstruction using several related methods for blurred LR 'Lena' test images contaminated with different level Gaussian noise. In order to prove the robustness of proposed scheme, the last two is SR reconstruction using several related methods for another blurred LR test images contaminated with different level outliers (in this paper we choose salt and pepper noise).

The first experiment is carried out to demonstrate the effectiveness of the trilateral kernel regression for high noisy images. First, ten images are made by randomly shifting an original image. Then these translated images are blurred by Gaussian blur mask sized 10 with sigma 1, down sampled by 3 for each dimension, corrupted by Gaussian noise with STD = 10 and 1% of salt and pepper noise as outlier. The original image is shown in Figure 3 (a) and one of ten corrupted LR shifting images is shown in Figure 3 (b) which has been zoomed in as same size as original image for observation. We apply proposed method to the generated sequence comparing with three related SR methods: classic kernel regression based SR, adaptive kernel regression based SR, classic kernel regression based SR with confidence weight, and trilateral kernel regression based SR we proposed.

The evaluation of these results is performed by the index of root of mean squared error (RMSE) which isdefined as:

$$\mathrm{RMSE}(f, \hat{f}) = \sqrt{\frac{1}{N}\sum\left(f - \hat{f}\right)^2} \tag{23}$$

Where, N is the number of samples. Observing from results shown in Figure 3, we can say (e) and (f) are clearer than (c) and (d). Specially, on the edge of hat (e) and (f) are preserved better, whereas the (c) and (d) are a bit like curtained images. In addition, on outlier processing, (f) is better than (e). The few black and white spots in (e) and (f) can approve this judgment clearly. At the same time, the RMSE also confirms effectiveness of proposed scheme.

Using the same method as described in first experiment, we generated another simulated sequence with different Gaussian noise with STD = 5. This sequence will be used for another experiment to prove trilateral kernel regression performed more excellent for low level Gaussian noise. The original image is shown in Figure 4(a) and one of ten simulated images is shown in Figure 4(b).

In Figure 4, we find that (e) and (f) contain fewer curtains than (c) and (d), because the confident weight can detect and restrain more outliers. Specially, (f) is closer to origin than (e) as the (f) used trilateral kernel regression described previously.

In order to show the robustness of trilateral kernel regression, we choose another test image to carry out last two experiments. Other SR methods for comparison are as same as the former two experiments.

The third experiment is carried out to demonstrate the effectiveness of the trilateral kernel regression for low lever outlier contaminated images. The method of generating LR is as same as the second experiment. In Figure 5, we can find the trilateral kernel regression still performs best comparing other SR method. The RMSE also prove this result.

Last experiment, we modified the third experiment by contaminating with higher lever outlier (that is, 3% salt and pepper noise) to illustrate the effect of proposed trilateral kernel regression SR. As expected, Figure 6 (f) gives a best result from view of either visual or index.

CONCLUSION

This paper analyzed the drawback of the classic kernel

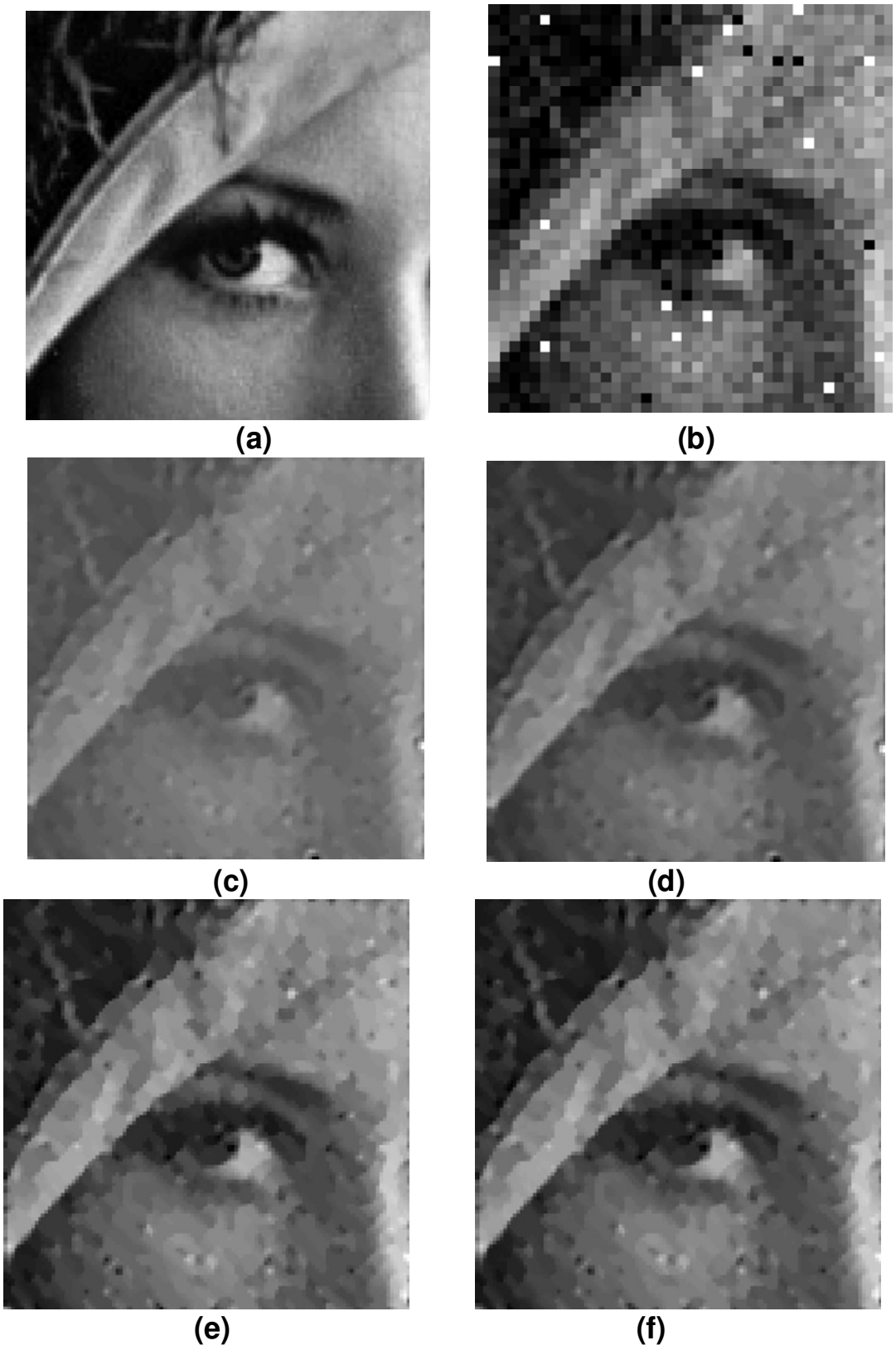

Figure 3. SR reconstruction of 10 corrupted LR images with STD = 10 Gaussian noise. (a) Original image; (b) 1 of 10 inputs (zooming in for observation); (c) classic kernel regression based SR [RMSE = 14.8652]; (d) adaptive kernel regression based SR [RMSE = 13. 9266]; (e) classic kernel regression based SR with confidence weight [RMSE = 13.4770]; (f) kernel regression based SR we proposed [RMSE = 13.3652].

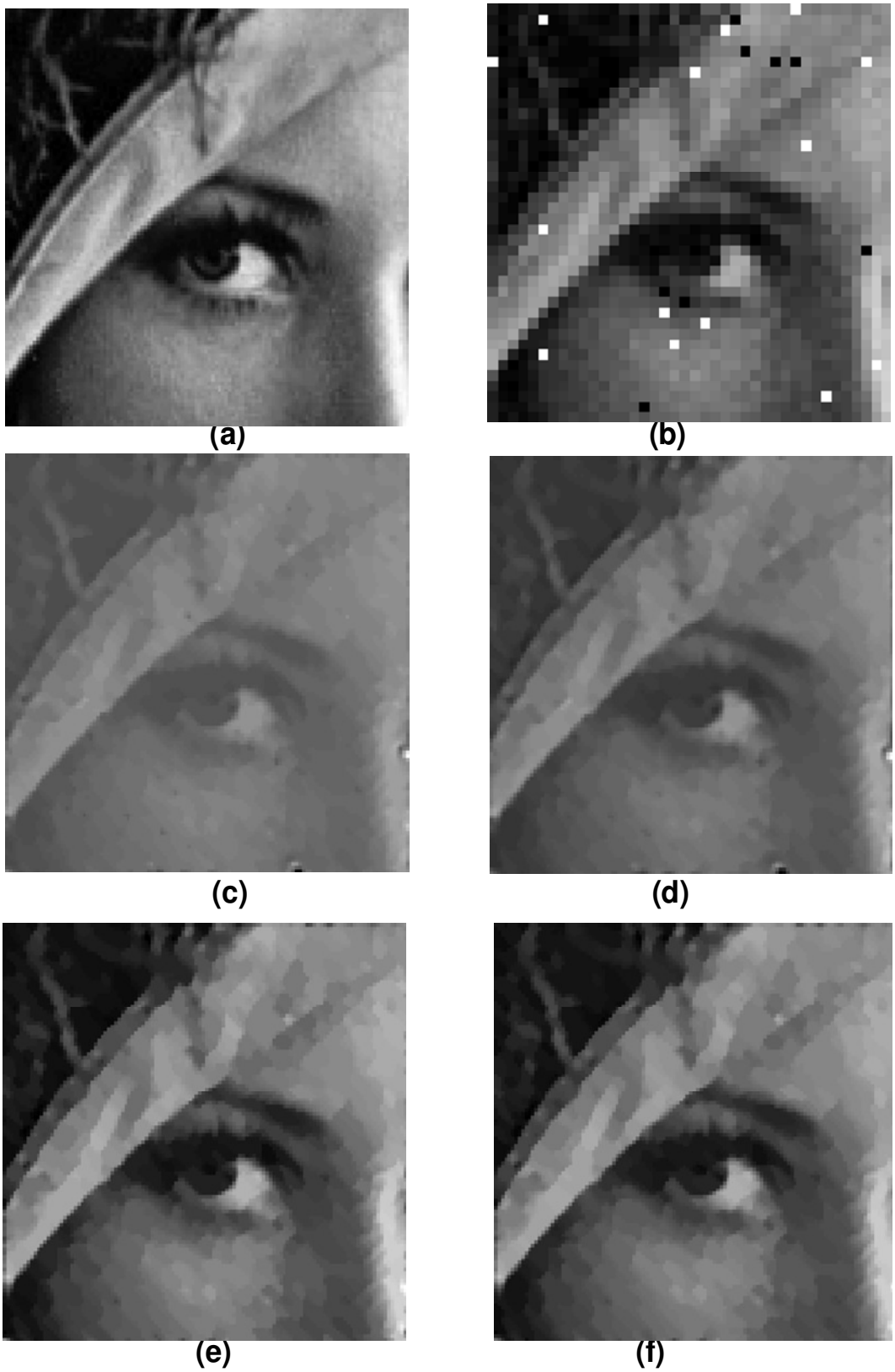

Figure 4. SR reconstruction of 10 corrupted LR images with STD = 5 Gaussian noise. (a) Original image; (b) 1 of 10 inputs (zooming in for observation); (c) classic kernel regression based SR [RMSE = 10.9703]; (d) adaptive kernel regression based SR [RMSE = 10.3455]; (e) classic kernel regression based SR with confidence weight [RMSE = 9.5000]; (f) kernel regression based SR we proposed [RMSE = 9.3993].

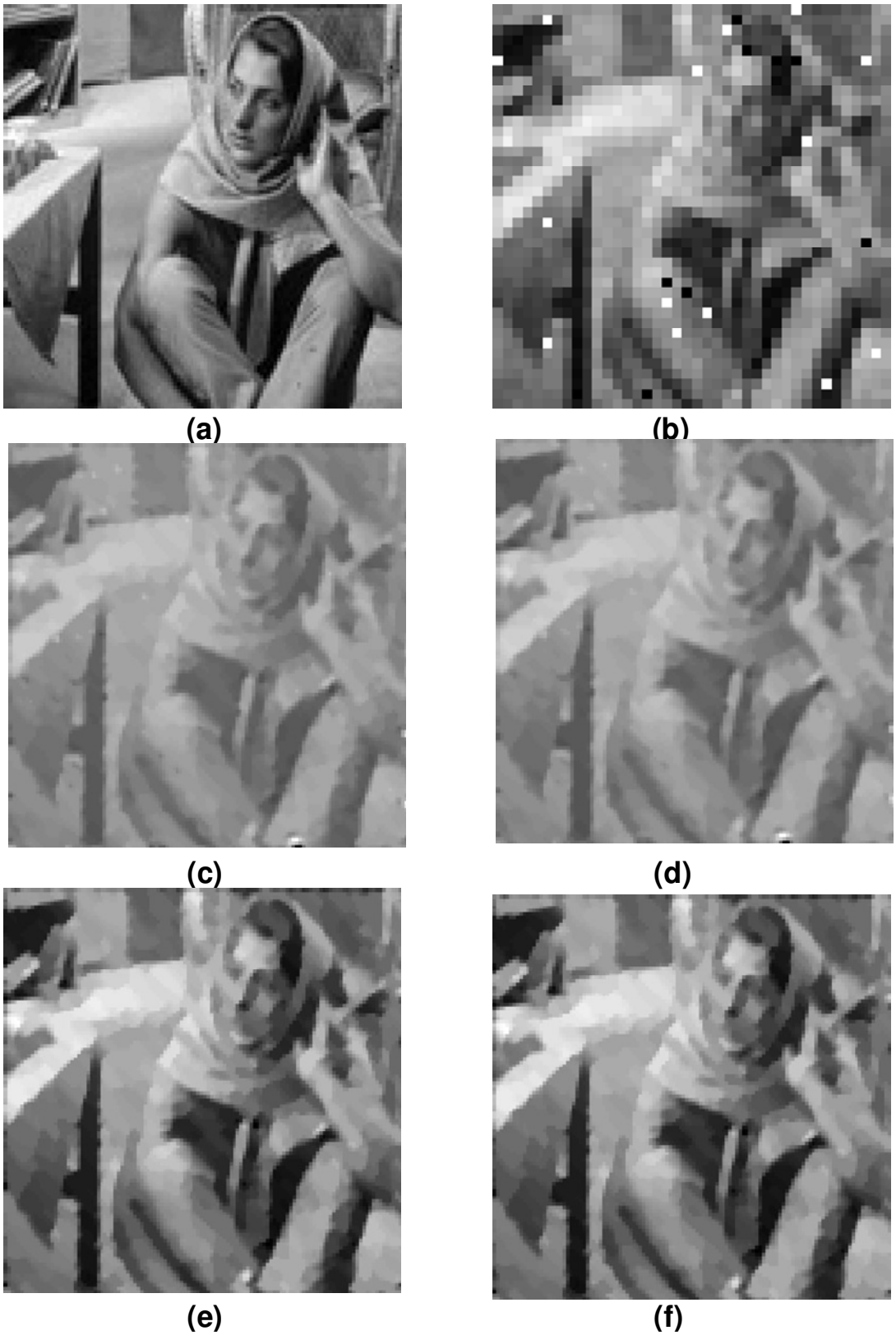

Figure 5. SR reconstruction of 10 corrupted LR images with 1% salt and pepper noise. (a) Original image; (b) 1 of 10 inputs (zooming in for observation); (c) classic kernel regression based SR [RMSE = 15.8266]; (d) adaptive kernel regression based SR [RMSE = 15.5067]; (e) classic kernel regression based SR with confidence weight [RMSE = 14.6025]; (f) kernel regression based SR we proposed [RMSE = 14.5031].

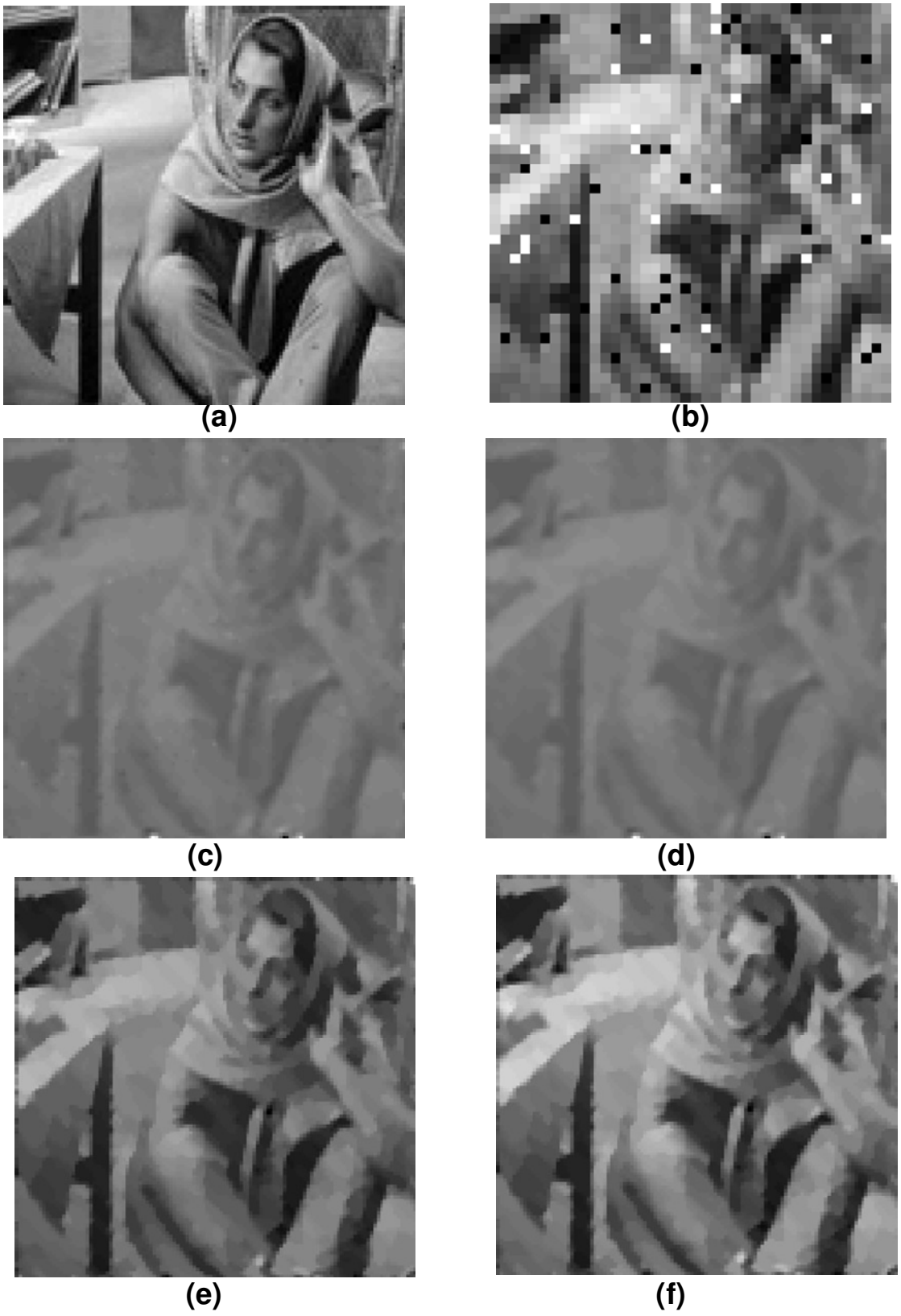

Figure 6. SR reconstruction of 10 corrupted LR images with 3% salt and pepper noise. (a) Original image; (b) 1 of 10 inputs (zooming in for observation); (c) classic kernel regression based SR [RMSE = 20.4670]; (d) adaptive kernel regression based SR [RMSE = 19.7877]; (e) classic kernel regression based SR with confidence weight [RMSE = 15.1208]; (f) kernel regression based SR we proposed [RMSE = 14.8410].

regression and adaptive kernel regression. Motivated by the need of removing more outliers and SR reconstructing more general LR sequence, we modified the kernel regression and suggested a new trilateral kernel regression. The proposed trilateral kernel regression considers not only spatial distance and photometric difference but also confidence of pixels, so it can obtain more accurate result. On the other hand, this paper also applies idea of trilateral kernel regression to SR, and gives an algorithm flow. Analysis and experiment indicated that the proposed method is effective to remove more noise and outliers than other method in SR.

ACKNOWLEDGMENTS

This work is supported partly by “National Natural Science Foundation of China” (No. 60872096, by “the Natural Science Foundation of Jiangsu Province of China” (No. BK2009352), and by “the Fundamental Research Funds for the Central Universities” of China (No. 2010B16414)

REFERENCES

Alam MS, Bognar JG, Hardie RC, Yasuda BJ (2000). Infrared image registration and high-resolution reconstruction using multiple translationally shifted aliased video frames. IEEE Transactions on Instrumentation and Measurement. 49(5): 915-923.

Bose NK, Kim HC, Valenzuela HM (1993). Recursive implementation of total least squares algorithm for image reconstruction from noisy, undersampled multiframes. In Proceedings of the IEEE Conference on Acoustics, Speech Signal Process., 5: 269–272.

Elad M, Datsenko D (2007). Example-based regularization deployed to super-resolution reconstruction of a single image. Comput. J., 52(1): 15-30.

Elad M, Farsiu S, Robinson D, Milanfar P (2004). Advances and challenges in super-resolution. Int. J. Imaing Syst. Technol., 14(2): 47-57.

Elad M, Hel-Or. Y (2001). A fast super-resolution reconstruction algorithm for pure translational motion and common space invariant blur. IEEE Transactions Image Process., 10(8): 1187-1193.

Farneback G (2002). Polynomial expansion for orientation and motion estimation. Ph.D. thesis. Linkoping University, Linkoping, Sweden.

Lertrattanapanich S, Bose NK (2002). High resolution image formation from low resolution frames using delaunay triangulation. IEEE Transactions Image Process., 11(12): 1427-1441.

Milanfar P (ed.) (2010). SUPER-RESOLUTION IMAGING. CRC Press, New York, USA.

Nguyen N, Milanfar P (2010). A wavelet-based interpolation-restoration method for superresolution. Cir. Syst. Signal Process, 19: 4.

Patti AJ, Altunbasak Y (2001). Aritifact reduction for set theoretic super resolution image reconstruction with edge adaptive constraints and higher-order interpolants. IEEE Transaction Image Process., 10(1): 179-186.

Pham TQ, van Vliet LJ, Schutte K (2006).Robust fusion of irregularly sampled data using adaptive normalized convolution. EURASIP J. Appl. Signal Process., pp. 1-12.

Pickup LC, Capel DP, Roberts SJ, Zisserman A (2009). Bayesian methods for image super-resoluti on. Comput. J., 52(1): 101-113.

Su W, Kim SP (1994). High-resolution restoration of dynamic image sequences. Int. J. Imaging Syst. Technol., 5(4): 330-339.

Takeda H, Farsiu S, Milanfar P (2007). Kernel regression for image processing and reconstruction. IEEE Transactions Image Process., 16(2): 349-366.

Takeda H (2010). Locally Adaptive Kernel Regression Methods for Multi-dimensional Signal Processing, Ph.D. Thesis, Electrical Engineering, UC Santa Cruz.

Tian J, Ma KK (2010). Stochastic super-resolution image reconstruction. Journal of Visual Commun. Image Representation, 21(3): 232-244.

Tom BC, Katsaggelos AK, Galatsanos NP (1994). Reconstruction of a high resolution image from registration and restoration of low resolution images. In Proceedings of IEEE International Conference Image Process., 553–557.

Tsai RY, Huang TS (1984). Multipleframe image restoration and registration. In: Advances in Computer Vision and Image Processing. JAI Press Inc, Greenwich, CT. pp. 317-339.