Fast image inpainting using exponential-threshold POCS plus conjugate gradient

S. Q. Wang[1] and J. H. Zhang*[2]

Image inpainting can remove unwanted objects and reconstruct the missing or damaged portions of an image. The projection onto convex sets (POCS) is a classical method used in image inpainting. However, the traditional POCS converges slowly due to the linear error threshold. We propose an exponential-threshold scheme, which greatly improves the convergence of the POCS. Although the exponential-threshold POCS can recover the image in about 20 iterations, it cannot reconstruct the image details very well even with hundreds of iterations. Thus, we append the non-local restoration to the exponential-threshold POCS to further refine the image details, and then we solve this objective function using the conjugate gradient. Numerical experiments show that for each iteration, the exponential-threshold POCS and the conjugate gradient have very similar computational efficiencies. For an image with various topologies of the missing areas, our scheme can recover missing pixels simultaneously and obtain a satisfied inpainting result in only 20 iterations of the exponential-threshold POCS and 20 iterations of the conjugate gradient. The proposed method can excellently restore damaged photographs and remove superimposed text. This method has less computational cost than the conjugate gradient and has a higher resolution than the POCS.

Keywords: Projection onto convex sets, Inpainting, Exponential threshold, Conjugate gradient, Non-local restoration

Introduction

Image inpainting is important for restoring an image that is blotted, contaminated or partially destroyed.[1–3] It is also popular for removing selected objects in a perfectly sampled image. Usually, we assume that some of the pixels in the entire image are credible, and they are defined as valid pixels; meanwhile, the other parts are partially credible or even completely unreliable, and they are defined as invalid pixels. The image inpainting automatically fills in the invalid pixels under the constraints from the valid pixels. The resulting image should be more unitary and legible than before and the changes should be undetectable to an observer.

The projection onto convex sets (POCS) is a general tool for convexly constrained parameter/function estimation. For a more detailed historical review of the POCS, readers may consult Combettes (1993),[4] Bauschke and Borwein (1996),[5] Deutsch (2001)[6] and Theodoridis et al. (2011).[7] The POCS is a classical method for image inpainting.[8–13] It is also an important part of advanced image restoration.[14,15] The main advantage of the POCS is that it is simple to implement. The POCS is also convenient for including a priori information as a constraint during the iterations. The main disadvantage of the POCS is that it has slow convergence and thus has high computational cost. In addition, its ability to restore detailed structures is still limited.

Although some quadratically convergent algorithms are proposed,[12,13,16] the reconstructed results of high frequency components are still not satisfactory. An important reason for this is that the cutoff error threshold or stop criterion is difficult to determine in advance in practical applications. However, the restoration of high frequency components is highly dependent on the cutoff error threshold. In addition, spectrum leakages are inherent to an image with missing pixels; unfortunately, these spectrum leakages are usually buried in low-energy spectrum and are highly mixed with the spectrum of the high frequency components. Therefore, it is difficult for the POCS to completely separate the spectrum leakages from the high frequency components, which means that the POCS would lose a lot of detailed structures.

In this paper, we propose a new scheme that accelerates the convergence of the POCS using an exponential error threshold rather than the traditional linear one. The idea comes from the exponential cooling rate[17,18] that is essential for a fast convergence of the simulated annealing algorithm.[19] The major advantage of the exponential error threshold is that it can greatly improve the convergence of the POCS;[16] in addition, it is very easy to implement and it maintains the algorithm structure of the traditional POCS. Compared to previous works on accelerating the convergence of the

[1]School of Information Engineering, Minzu University of China, Beijing, China
[2]Institute of Geology and Geophysics, Chinese Academy of Sciences, Beijing, China

*Corresponding author, email zjh@mail.iggcas.ac.cn

POCS, our scheme is one of the simplest in the literature, and only requires a small change to the existing code. The computational cost for each iteration is the same as that of the traditional POCS implementation. More importantly, our scheme can perform almost all iterations on the most urgent part by using small initial error threshold as well as small steps. Therefore, our scheme can reconstruct big gaps with much less iteration numbers compared with both traditional linear error thresholds and recently proposed exponential error threshold.[16]

Although our scheme makes the POCS converge much faster than before and only needs a very limited number of iterations (e.g. 20–50), it is still not able to improve the final results for the high-frequency components. Thus, we propose to only employ the most efficient part of the POCS to generate a low-resolution result. Then, this result is taken as an initial input for other methods that are good at recovering the detailed information. The non-local restoration[20–23] solved with the conjugate gradient (CG)[24–26] is selected for the purpose of refining the high frequency components, since it is powerful in reconstructing the details. The main drawback of this method is that it requires many more iterations to reconstruct wide areas of missing pixels (i.e. big holes).

The proposed hybrid scheme, the exponential-threshold POCS plus the CG, fully takes advantage of both and avoids their disadvantages as much as possible. The iteration time of the proposed hybrid scheme is usually about 20 for either the POCS or the CG. In contrast, the pure POCS could not obtain high resolution even with several hundreds of iterations, and the pure CG generally needs about 200 iterations to obtain a similar-quality image. Note that our experiments show that the computational cost of the POCS is quite similar for each iteration to that of the CG for various image sizes.

This paper is organised as follows: firstly, we give a brief review of the traditional POCS; secondly, we present a modified scheme that uses exponential error threshold; thirdly, we show the hybrid scheme of POCS and the CG; next, we show some examples to illustrate the effectiveness of our hybrid scheme; and finally, we draw some conclusions.

A brief review of traditional POCS

The basic idea of the POCS was first proposed by Bregman[27] and Gubin et $al.$.[28] Then it becomes popular in image inpainting,[8–13] motion deblurring,[29,30] video super-resolution reconstruction[31,32] and other fields.[33] In the POCS, all image constraints are presented as a series of closed convex sets in Hilbert space; then, starting from an arbitrary initial value, the image is iteratively projected onto the intersection of all closed convex sets using the projection operator; finally, the image in the intersection is regarded as the optimal solution.

The object image is represented by a two-dimensional $M \times N$ array $f(m,n)$, where $m = 1, 2, …, M$ and $n = 1, 2, …, N$. The two-dimensional discrete Fourier transform of $f(m,n)$ is given by

$$F(k,l) = \frac{1}{(MN)^{1/2}} \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} f(m,n) \exp\left[-2\pi j\left(\frac{mk}{M} + \frac{nl}{N}\right)\right]$$

(1)

and the inverse Fourier transform is given by

$$f(m,n) = \frac{1}{(MN)^{1/2}} \sum_{l=0}^{N-1} \sum_{k=0}^{M-1} F(k,l) \exp\left[2\pi j\left(\frac{mk}{M} + \frac{nl}{N}\right)\right]$$

(2)

where $j = (-1)^{1/2}$, $0 \le m, k \le M-1$ and $0 \le n, l \le N-1$. For simplicity, we express them as $F = \Im^+(f)$ and $f = \Im^-(F)$, respectively. In the ith iteration for $i = 0, 1, 2, …, I$, the POCS is expressed as

$$f_{i+1} = f_0 + (1 - W)\Im^-[T_i \Im^+(f_i)]$$

(3)

where $W \equiv W(m,n)$ is a binary mask whose value is 0 if the pixel is invalid and 1 otherwise, $f_0 = Wf$ is the input image, and

$$T_i \equiv T_i(k,l) = \begin{cases} 1, & \text{if} \quad \Im^+(f_i) \ge \varepsilon_i \\ 0, & \text{if} \quad \Im^+(f_i) < \varepsilon_i \end{cases}.$$

(4)

where ε_i is the error threshold. The iteration stops if the error threshold ε_i is smaller than a given error limitation of ε_{min} or if the total iteration times reach a given upper limit. If the original image f' is known, we define the evaluated error as

$$\frac{\|f' - \Im^-[T_i \Im^+(f_i)]\|^2}{\|f'\|^2}$$

(5)

If the original image f' is not known, we define the evaluated error as

$$\frac{\|f_0 - W\Im^-[T_i \Im^+(f_i)]\|^2}{\|f_0\|^2}$$

(6)

In the traditional POCS, the error threshold ε_i is basically a linearly decreasing error threshold, that is

$$\varepsilon_{i+1} = \varepsilon_i - \delta$$

(7)

where δ is a fixed step. For a large step δ, the iteration would stop quickly but would possibly make the POCS unusable, because the POCS with a large step usually can not successfully converge to the intersection. For a small step δ, we can guarantee the convergence of the POCS but would greatly increase the computational cost. Note that there are two two-dimensional Fourier transforms for each iteration, and the computational cost is proportional to the iteration times. Even with fast Fourier transforms, it is still extremely time-consuming when total iteration times are large. Therefore, we should significantly reduce the total iteration times to make the POCS feasible for practical applications especially for large images.

An exponential-threshold POCS

In literature of the simulated annealing algorithm,[19] a proper selection of the cooling rate (i.e. the error threshold) has been proven to be essential for convergence. Generally, an exponential cooling rate allows a much quicker convergence of the simulated annealing algorithm than the traditional linear one does.[17,18] In this paper, we introduce the exponential cooling rate to improve the convergence of the POCS. The error threshold ε_i of the modified POCS is changed from equation (7) into

$$\varepsilon_{i+1} = \alpha \varepsilon_i \qquad (8)$$

where $\alpha \in (0,1)$ is the factor of the exponential error threshold. In case of a constant α during the iteration, exponential error threshold ε_i in the ith iteration is related to the initial error threshold ε_0 with a factor of α^i. That is

$$\varepsilon_i = \alpha^i \varepsilon_0 \qquad (9)$$

In contrast, the traditional linear error threshold is

$$\varepsilon_i = \varepsilon_0 - i\delta \qquad (10)$$

Although the attenuation factor α is usually about 0·5–0·99, it makes the error threshold ε_i decay in the form of an exponential function starting from the initial error threshold ε_0. In contrast, although the attenuation factor δ is usually smaller than 0·001, it makes the error threshold ε_i decay in the form of a linear function and is subducted from the starting error threshold of ε_0.

According to numerical analyses, the initial error threshold ε_0 of the exponential-threshold POCS has a limited influence on the subsequent iteration. For example, ε_0 can take either 0·6 or 0·9 but will not destroy the convergence of the POCS. However, both the initial error threshold ε_0 and the step of the error threshold (i.e. the interval between two neighboring error thresholds, $\varepsilon_{i+1} - \varepsilon_i$ or $\delta_{i+1} - \delta_i$) are essential for the convergence of the POCS. In fact, the step should be small enough especially when the iteration times are increasing. This indicates that the error threshold should decrease gradually with increasing iteration times. However, the linear threshold has a globally constant step, which is unnecessarily fine for the beginning of the iteration but is too big when the iteration times become larger. In contrast, the exponential threshold has a variable (or dynamic) step. This dynamic step would be fairly large at the beginning of an iteration and becomes smaller and smaller with increasing iteration times, which is consistent with the error-decay trend of the POCS. Therefore, the exponential threshold is superior to the traditional linear threshold and achieves a much faster convergence.

Gao *et al.*[16] suggest a different kind of exponential threshold in reconstructing irregular seismic data. Their exponential threshold is defined as

$$\varepsilon_i = \exp[(i-1)\beta]\varepsilon_0 \qquad (11)$$

with

$$\beta = \frac{-1}{I-1}\ln\left(\frac{\varepsilon_{max}}{\varepsilon_{min}}\right) \qquad (12)$$

being the attenuation factor, where $i = 1, 2, \ldots, I$. Note that the attenuation factor β is associated with the maximum iteration number I; thus, one should provide an estimate I at the beginning. In contrast, our scheme does not need any pre-estimated parameter. Rewrite equation (9) as

$$\varepsilon_i = \alpha^i \varepsilon_0 = \exp(i \ln \alpha)\varepsilon_0 \qquad (13)$$

we see that our scheme is basically different from equation (11) because of using completely different attenuation factor.

For a possible range of $\varepsilon_{min} \in [0 \cdot 00005, 0 \cdot 005]$ and $I = 21, 22, \ldots, 100$, the corresponding range of β suggested by Gao *et al.*[16] is about $[0 \cdot 053, 0 \cdot 495]$, which is far below our suggested range around 0·7 (e.g. from 0·5 to 0·8). However, our tests show that a too small attenuation factor is harmful for the successful reconstruction of both large gaps and local details. On the other hand, we use a small initial error threshold ε_0 (e.g. 0·6–0·9), rather the big one suggested by Gao *et al.*,[16] to further reduce the total iteration numbers; that is, their initial error threshold is bigger than ours (i.e. $\varepsilon_{max} > \varepsilon_0$). In general, we suggest using small step of error threshold as well as small initial error threshold. Thus, our scheme can concentrate almost all iterations on the most urgent part and can achieve a significant improvement after each iteration. Therefore, our scheme can reconstruct big gaps with much less iteration numbers compared with that proposed by Gao *et al.*[16]

A hybrid scheme of the POCS + CG

The POCS has difficulty obtaining a perfect result with fine image details. With the help of our exponential threshold, it is still not able to recover all the details even with properly selected steps and tremendous iterations. That is, our exponential threshold is only helpful to obtain the same results much quicker than the traditional linear threshold. To further restore the image details, we have to use another method that is powerful for high resolution restorations.

In non-local regularisation,[20–22] an inpainting result $\hat{f}$ can be regarded as a quadratic minimisation problem as following

$$\hat{f} \in \arg\min_f \|Af - b\|^2 + \lambda J(f) \qquad (14)$$

where A is the ill-conditioned linear operator, λ is the regularisation parameter and $J(f)$ is the non-local regularisation functional. We set $A = W - \lambda\Delta$, $b = Wf$ and $\lambda = 0 \cdot 01$, and use the discretised Sobolev norm[22]

$$J_{Sob}(f) = \|\nabla f\|^2 \qquad (15)$$

where ∇f is the gradient of f, and Δ is the divergence operator of the gradient of f.

We minimise equation (14) using the CG as following

$$r_0 = Wf_0 - Af_0$$

$$d_0 = r_0$$
$$\text{for } i = 1, 2, \ldots, I$$

$$a_i = \frac{\|r_{i-1}\|^2}{\|Ad_{i-1}\|^2}$$

$$f_i = f_{i-1} + a_i d_{i-1}$$

$$r_i = r_{i-1} - a_i Ad_{i-1}$$

$$c_i = \frac{\|r_i\|^2}{\|r_{i-1}\|^2}$$
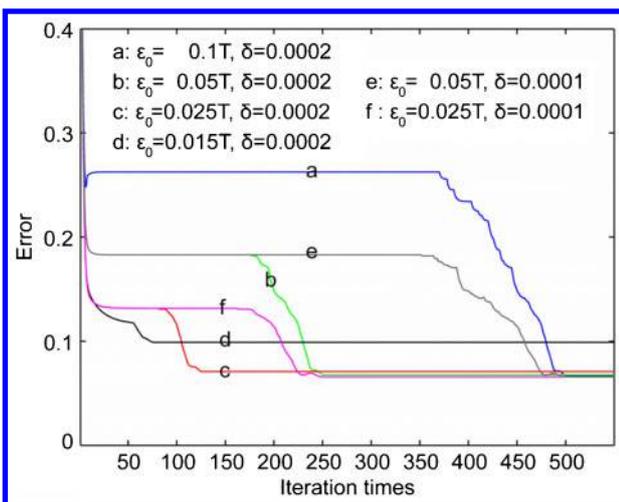
$$d_i = r_i + c_i d_{i-1}$$

end

1 Comparison of inpainting results between the traditional linear-threshold POCS and the exponential-threshold POCS: *a* **original image;** *b* **an image with randomly missing pixels and a big hole;** *c* **the traditional linear-threshold POCS with 200 iterations;** *d* **the traditional linear-threshold POCS with 300 iterations;** *e* **the exponential-threshold POCS with 20 iterations;** *f* **the exponential-threshold POCS with 30 iterations. For the traditional linear-threshold POCS, the initial threshold is** $\varepsilon_0 = 0{\cdot}025T$ **and the step** $\delta = 0{\cdot}0001$**, where** $\mathbf{T} = \max|\Im^{+}(f_0)|$ **is the maximum amplitude of the spectrum. According to Fig. 2, the selected** ε_0 **would make the cutoff error acceptable within a relatively small iteration time. If we select a bigger initial threshold** ε_0 **(e.g.** $0{\cdot}05T$**), the inpainting result at 200 and 300 iterations would become much worse, since the residual error is still very large at these two stages (see curve 'e' in Fig. 2). If we select a bigger step** δ **(e.g.** $0{\cdot}0002$**), the inpainting result would also become worse, since the residual error is larger (see curves 'c' and 'f' in Fig. 2). For the exponential-threshold POCS, we select the initial threshold** $\varepsilon_0 = 0{\cdot}5T$ **and the factor** $\alpha = 0{\cdot}7$ **according to Fig. 3**

The CG is very popular in numerical optimisations,[34] and is widely used in image restorations.[24–26] The main advantage of the CG is that it can minimise the objective function with a limited number of iterations, which is not bigger than the matrix rank. However, the non-local regularisation solved by the CG, named as CG for simplicity, is not good at reconstructing big area of missing pixels. Therefore, we propose to first employ the exponential-threshold POCS with very limited iterations to recover the low frequency components that are related to the macro image frame; then, we take this result as the input of the CG; finally, we only perform very limited iterations for the CG to recover the high frequency components that are related to the image details.
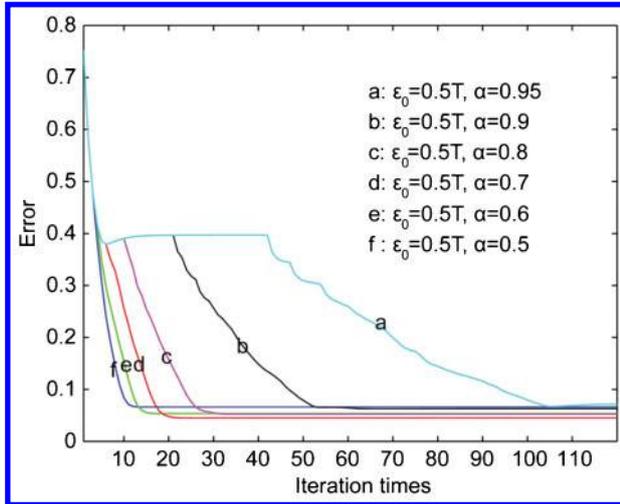
Numerical experiments

Figure 1a show an image of boy Yide. Figure 2b has some randomly missing pixels and a big hole. Figures 1c and d show inpainting results obtained by the traditional linear-threshold POCS with 200 and 300 iterations, respectively. Figures 1e and f shows inpainting results obtained by the exponential-threshold POCS with 20 and 30 iterations, respectively. Obviously, the exponential-threshold POCS with 20 iterations obtains a good result, while the traditional linear-threshold POCS with 200 iterations obtains a poor result. Therefore, the exponential-threshold POCS converges much faster than the traditional linear-threshold POCS.

Note that, we select $0{\cdot}025T$ as the initial error threshold for the traditional linear-threshold POCS, where $T = \max|\Im^{+}(f_0)|$ is the maximum amplitude of the spectrum. If we select a bigger initial threshold ε_0 (e.g. $0{\cdot}05T$), the inpainting result at 200 and 300 iterations would become much worse, since the residual error is still very large at these two stages (see the curve


2 Error curves of the traditional linear-threshold POCS. The images used are shown in Fig. 1a and b. The initial threshold ε_0 **and the step** δ **are different for each curve. Note that if we select** $\varepsilon_0 = 0{\cdot}5T$ **as the initial error threshold, the iteration times will have a several-fold increase**

3 Error curves of the exponential-threshold POCS. The images used are shown in Fig. 1a and b. The initial threshold $\varepsilon_0=0\cdot5T$, which is much looser than that (e.g. $0\cdot025T$) for the traditional linear-threshold POCS. The factor α is different for each curve

'e' in Fig. 2). If we use $\varepsilon_0=0\cdot5T$ for the traditional linear-threshold POCS, it will greatly increase the computational cost about 20 times. On the other hand, if we select a bigger step δ (e.g. $0\cdot0002$), the inpainting result also becomes worse, since the final residual error is larger (see curves 'c' and 'f' in Fig. 2). In contrast, we use $\varepsilon_0=0\cdot5T$ as the initial error threshold for the exponential-threshold POCS, which is much larger than that of the traditional linear-threshold POCS (i.e. $\varepsilon_0=0\cdot025T$); however, we do not see any independence of the initial error threshold and the step, as shown in Fig. 3.

Figure 2 shows the error curves of the traditional linear-threshold POCS for different initial error thresholds and steps. We see that the traditional linear-threshold POCS is very sensitive and highly dependent on the initial error threshold ε_0. In addition, a big step δ may make the algorithm divergent, but a small step with a big error threshold means that there were too many iterations. According to Fig. 2, we select the initial error
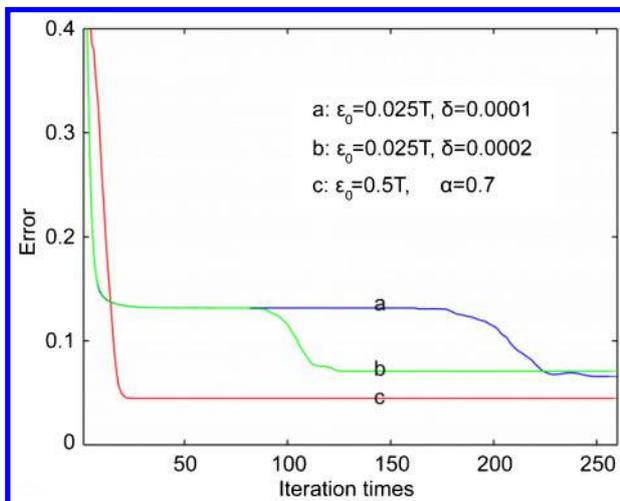
4 Comparison of error curves between the traditional linear-threshold POCS and the exponential-threshold POCS. These three curves are from Fig. 2 and 3
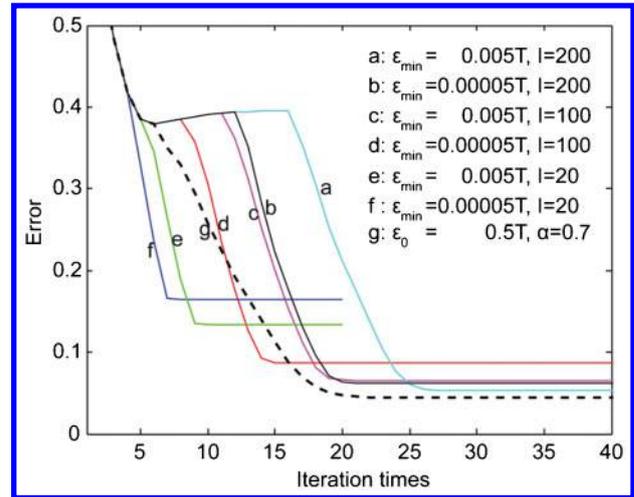
5 Comparison of error curves between Gao *et al.*'s exponential-threshold POCS and our exponential-threshold POCS. The initial error threshold used in Gao *et al.*[16] is always $\varepsilon_{max}=1\cdot0T$; the minimum error threshold is set to be $\varepsilon_{min}=0\cdot005T$ and $\varepsilon_{min}=0\cdot00005T$, respectively; and the iteration number I is set to be 20, 100 and 200, respectively. The two control parameters used in our scheme is set to be $\varepsilon_0=0\cdot5T$ and $\alpha=0\cdot7$, respectively. The solid lines show the results of Gao *et al.*[16] and the dashed line shows the result of our scheme

threshold as $\varepsilon_0=0\cdot025T$ and the step as $\delta=0\cdot0001$ to make the computational cost of the traditional linear-threshold POCS comparative to that of the exponential-threshold POCS. However, the initial error threshold is difficult to determine in advance in practical applications.

Figure 3 shows that the exponential-threshold POCS is superior to the traditional linear-threshold POCS on these two aspects: it is not sensitive to the initial error threshold ε_0 or to the factor α. In addition, the error curves stop decreasing within 50 iterations for most factors listed. According to Figs. 3, we select the initial error threshold as $\varepsilon_0=0\cdot5T$ and the factor as $\alpha=0\cdot7$ since they allow the exponential-threshold POCS to have a reasonable tradeoff between the convergence and the restoration accuracy. Figure 4 further shows that the exponential-threshold POCS has smaller residual error compared with the traditional linear-threshold POCS. Therefore, we only show results from the exponential-threshold POCS since it is superior to the traditional linear-threshold POCS in various important aspects, such as the computational cost, the robustness and the convergence.

Figure 5 shows six typical parameter groups suggested by Gao *et al.*.[16] The convergence of Gao *et al.*[16] is fast when the error threshold $\varepsilon_{min}=0\cdot00005$, but the residual error is very big; the residual error of Gao *et al.*[16] is small when the error threshold $\varepsilon_{min}=0\cdot005$, but the convergence is relatively slow. In addition, either big (e.g. 100 or 200) or small (i.e. 20) iteration number I does not show better accuracy than our scheme ($\varepsilon_0=0\cdot5T$ and $\alpha=0\cdot7$). Figure 5 shows that our scheme is superior to Gao *et al.*[16] since our scheme has a similar convergence but a smaller residual error.

However, both the traditional linear-threshold POCS and the exponential-threshold POCS are not able to obtain high resolution results, as shown in Fig. 1, even

6 Comparison of inpainting results between the CG and the hybrid method (POCS+CG) of the exponential-threshold POCS plus the CG: *a* the CG with 20 iterations; *b* the CG with 100 iterations; *c* the POCS+CG with 20 iterations of the exponential-threshold POCS and then 20 iterations of the CG; *d* the POCS+CG with 20 iterations of exponential-threshold POCS and then 30 iterations of the CG. For the exponential-threshold POCS, the initial threshold is $\varepsilon_0=0\cdot5T$ and the factor $\alpha=0\cdot7$

though we employ many more iterations (the results are not shown here because they are the same to Fig. 1d and f, respectively). This shortcoming of the POCS can also
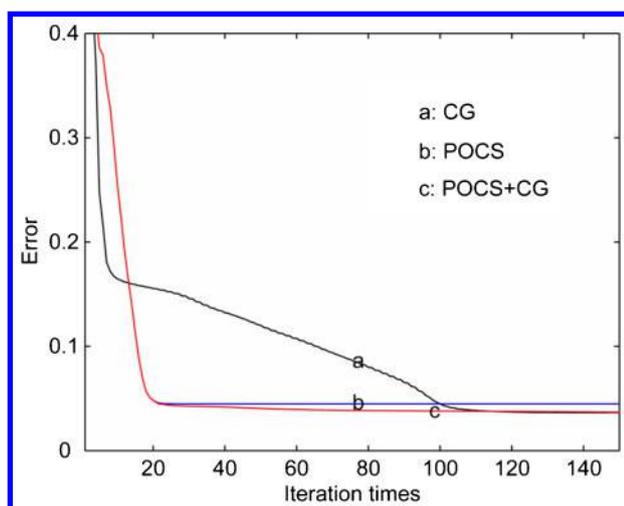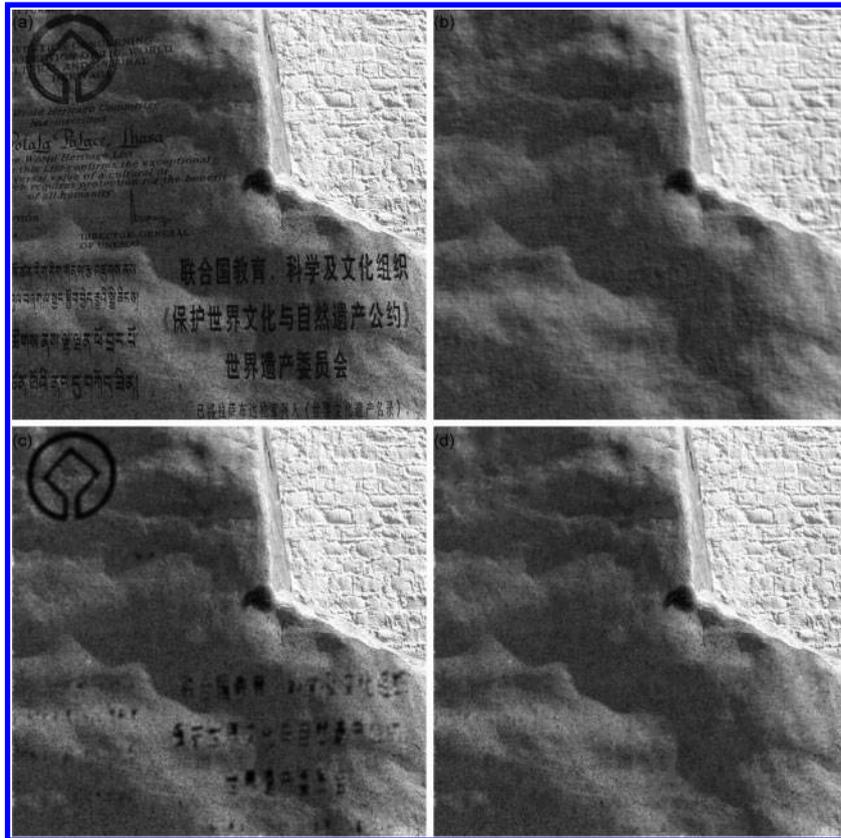

7 Comparison of error curves between the CG, the exponential-threshold POCS and the hybrid method (POCS+CG). The curve of the exponential-threshold POCS is from Fig. 3. The residual error of the hybrid method is only slightly smaller than that of the POCS; however, these 'small' improvements are mostly contributed by the improvements of the image details, which actually have small amplitude variations that have very important visual-effect consequences

be seen from the large residual errors shown in Figs. 2 and 3. Thus, we have to recover the image details using other methods (e.g. the CG) in order to overcome this shortcoming.

Figures 6a and b show the inpainting results of the CG at 20 and 100 iterations, respectively. Obviously, the CG can obtain a much higher resolution compared with the POCS. However, the big hole is not well recovered by the CG even at 100 iterations. Thus, the combination of the exponential-threshold POCS and the CG is necessary and would take advantage of the benefits of both methods. Figures 6c and d show the results obtained by the hybrid method. Clearly, only 20 iterations of the exponential-threshold POCS and 20 iterations of the CG obtain a much superior result than either method alone. For example, as shown in Fig. 7, we need more than 120 iterations to obtain a similar result if we only use the CG. Figure 7 also shows that the residual error of the hybrid method is only slightly smaller than that of the POCS, which is easy to be considered as an unapparent improvement; however, these 'small' improvements shown in error curve mostly contribute to the improvements of the image details, which actually have small amplitude variations but have very important visual effect implications.

Figure 8 shows another example for the application of image inpainting. We hope to remove the characters carved in the stone. We regard all kinds of characters (i.e. English, Tibetan and Chinese) as missing areas and mark them as invalid pixels. Figure 9 shows the local details
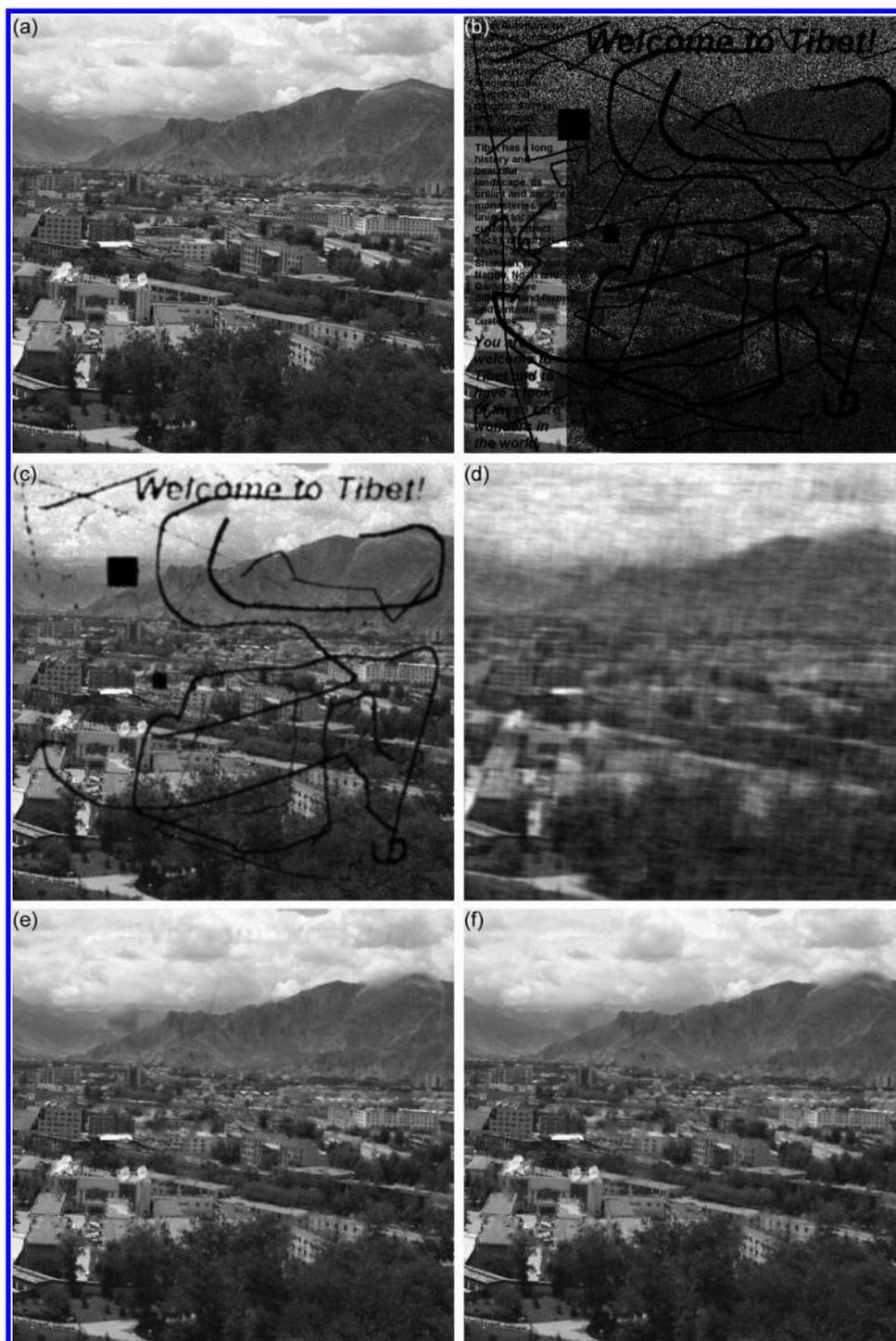
8 **Comparison of inpainting results between the CG, the exponential-threshold POCS and the hybrid method (POCS+CG): *a* original image; *b* the exponential-threshold POCS with 20 iterations of the CG; *c* the CG with 20 iterations; *d* the POCS+CG with 20 iterations for each part. For the exponential-threshold POCS, the initial threshold is $\varepsilon_0=0{\cdot}5T$ and the factor $\alpha=0{\cdot}7$**

within two typical areas. Again, the exponential-threshold POCS at 20 iterations can well recover the low frequency components except the details, and the CG at 20 iterations can well recover the details except the big holes. In contrast, the hybrid method can recover both the details and the low frequency components perfectly by combining these two methods.

Figure 10 shows another more complex example, which contains several possible missing features, such as characters, big holes, random dots and bold curves. Figures 10c and d are obtained by the CG with 20 iterations and the exponential-threshold POCS with 20 iterations, respectively. The exponential-threshold POCS shows to be powerful in recovering the background
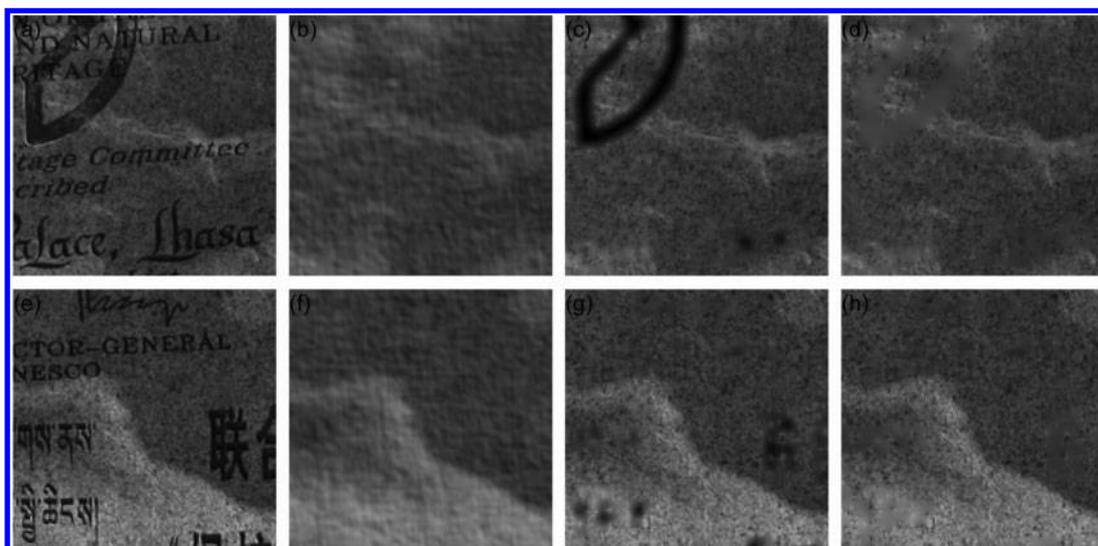

9 **Local details within two areas of Fig. 8. a–d and e–h correspond to Fig. 8a–d, respectively. The exponential-threshold POCS at 20 iterations can recover the low-frequency components except the details well, and the CG at 20 iterations can recover the details except the big holes well. In contrast, the hybrid method can recover both the details and the low-frequency components perfectly by combining these two methods**

10 Comparison of inpainting results between the CG, the exponential-threshold POCS and the hybrid method (POCS+CG): *a* **original image;** *b* **an image with randomly missing pixels, two big holes, bold lines and various-size characters;** *c* **the CG with 20 iterations;** *d* **the exponential-threshold POCS with 20 iterations;** *e* **the POCS+CG with 20 iterations of exponential-threshold POCS and then 20 iterations of the CG;** *f* **the POCS+CG with 20 iterations of exponential-threshold POCS and then 80 iterations of the CG. For the exponential-threshold POCS, the initial threshold is $\varepsilon_0 = 0.5T$ and the factor $\alpha = 0.7$**

within a limited number of iterations; and the CG shows to be strong in recovering the details if the big holes are ignored. As an optimal combination, the hybrid method is shown to be superior to either of these two methods. Of course, we can add more iterations of the CG for the hybrid method to refine the detailed structures, as shown in Figures 10e and f.

Comparing Figs. 6, 8 and 10, we find that the suggested parameters ($\varepsilon_0 = 0.5T$ and $\alpha = 0.7$) and the iteration times (20 times) are generally feasible for most cases, although these parameters are somewhat inefficient especially for the most severe case shown in Fig. 10. Nevertheless, only a small increase of the iteration times for the CG will significantly improve
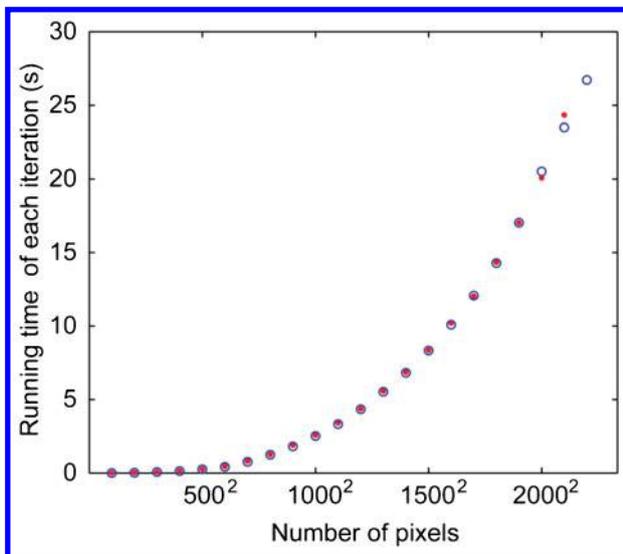
11 Running time of each iteration for various image sizes. The dots denote the exponential-threshold POCS and the circles denote the CG. For simplicity, we only show the image size of $(i \times 100)^2$, where $i = 1$, 2, 3, …. Owing to the memory limitation, the maximum image size is 2100×2100 for the exponential threshold POCS and is 2200×2200 for the CG, respectively. This figure shows that the computational cost of the exponential threshold POCS is quite similar to that of the CG

the inpainting results, without changing the other parameters, as shown in Figs. 6d and 10f.

We examine the running time via a single thread on a personal computer with a CPU Intel Core i5 M560 (2·67 GHz). We use the Fourier transforms functions fft2 and ifft2 included in Matlab 7. Figure 11 shows the time consumptions of the exponential-threshold POCS and the CG. Surprisingly, we see that they have almost the same computational efficiency since the time consumptions for any given image size are similar to each other. Therefore, it is easy to evaluate the hybrid method by purely counting the total iteration times.

Discussion

The computational efficiency and the resolution of the inpainting are greatly improved by means of the exponential-convergence POCS plus the CG. We can further employ the graphics processing units to accelerate the whole algorithm and can obtain a speedup ratio up to several tens of times.[35,36] As a powerful and computationally effective method, it is attractive for solving the inpainting problems for both large still images and long videos.[31,37]

We can further use the preconditioning technique to obtain a faster convergence for the CG. We only test a fixed factor α of the exponential threshold in the text. We also consider using a varying factor in order to obtain a much higher convergence and to save more computational cost for the POCS. If the contiguous missing areas are relatively large, we could use segmentation before inpainting.[38,39] Although the proposed method is very attractive both in computational cost and in reconstruction accuracy, it is not able to

recover the textured regions at this stage. Some recent approaches can be used[40] for this purpose.

Conclusion

We have proposed a hybrid approach for filling-in regions of missing pixels in still images. This approach combines the advantages of the POCS in reconstructing the low frequency components with the advantages of the CG in reconstructing the high frequency components. The convergence of the POCS is greatly improved by using an exponential error threshold compared with the traditional linear one. This exponential-threshold POCS allows us to iterate only 20 times to quickly recover the low frequency components. The resulting image is a fairly good initial input for the CG and allows the CG to iterate only twenty times to rapidly reduce those small residuals. A number of numerical results show that the proposed hybrid approach has a lower computational cost than the CG and has a higher resolution than the POCS.

Acknowledgements

References

1. Emile-Male, G. The Restorer's Handbook of Easel Painting, 1976 (Van Nostrand Reinhold, New York).
2. Ballester, C., Bertalmio, M., Caselles, V., Sapiro, G. and Verdera, J. Filling-in by joint interpolation of vector fields and gray levels. *IEEE Trans. Image Process.*, 2001, **10**, 1200–1211.
3. Hong, M. Iterative adaptive hybrid image restoration for fast convergence. *Opt. Eng.*, 2010, **49**, 080503.
4. Combettes, P. L. The foundations of set theoretic estimation. *Proc. IEEE*, 1993, **81**, 182–208.
5. Bauschke, H. H. and Borwein, J. M. On projection algorithms for solving convex feasibility problems. *SIAM Rev.*, 1996, **38**, 367–426.
6. Deutsch, F. Best Approximation in Inner Product Spaces, 2001 (Springer-Verlag, New York).
7. Theodoridis, S., Slavakis, K. and Yamada, I. Adaptive learning in a world of projections. *IEEE Signal Process. Mag.*, 2011, **28**, 97–123.
8. Youla, D. C. and Webb, H. Image restoration by the method of convex projections: part I — theory. *IEEE Trans. Med. Imaging*, 1982, **MI-I**, 81–94.
9. Sezan, M. I. and Stark, H. Image restoration by the method of convex projections: part 2 — applications and numerical results. *IEEE Trans. Med. Imaging*, 1982, **MI-1**, 95–101.
10. Stark, H. and Oskoui, P. High resolution image recovery from image-plane arrays, using convex projections. *J. Opt. Soc. Am. A*, 1989, **6**, 1715–1726.
11. Özkan, M. K., Tekalp, A. M. and Sezan, M. I. POCS-based restoration of space-varying blurred images. *IEEE Trans. Image Process.*, 1994, **3**, 450–454.
12. Dharanipragada, S. and Arun, K. S. A quadratically convergent algorithm for convex-set constrained signal recovery. *IEEE Trans. Signal Process.*, 1996, **44**, 248–266.
13. Crockett, J. S., Moon, T. K. and Gunther, J. H. Accelerating the convergence of POCS algorithms by exponential prediction, Proc. IEEE 11th Digital Signal Processing Workshop & IEEE Signal Processing Education Workshop, Taos Ski Valley, NM, USA, August 2004, IEEE, pp.173–177.

14. Ogawa, T. and Haseyama, M. Missing intensity interpolation using a kernel PCA-based POCS algorithm and its applications. *IEEE Trans. Image Process.*, 2011, **20**, 417–432.

15. Jung, C., Jiao, L., Liu, B., Qi, H. and Sun, T. Toward high-quality image communications: inverse problems in image processing. *Opt. Eng.*, 2012, **51**, 100901.

16. Gao, J. J., Chen, X. H., Li, J. H., Liu, G. C. and Ma, J. Irregular seismic data reconstruction based on exponential threshold model of POCS method. *Appl. Geophys.*, 2010, **7**, 229–238.

17. Azizi, N. and Zolfaghari, S. Adaptive temperature control for simulated annealing: a comparative study. *Comput. Oper. Res.*, 2004, **31**, 2439–2451.

18. Wang, S., Wang, S., Han, X. and Wang, H. Improved naive Bayesian classifier using simulated annealing algorithm. *Pattern Recogn. Lett.*, 2013, to be published.

19. Kirkpatrick, S., Gelatt, C. D. and Vecchi, M. P. Optimization by simulated annealing. *Science*, 1983, **220**, 671–680.

20. Gilboa, G., Darbon, J., Osher, S. and Chan, T. Nonlocal convex functionals for image regularization. *UCLA CAM Rep.*, 2006, Vol. 06–57.

21. Lou, Y., Zhang, X., Osher, S. and Bertozzi, A. Image recovery via nonlocal operators. *J. Sci. Comput.*, 2010, **42**, 185–197.

22. Peyré, G., Bougleux, S. and Cohen, L. P. Non-local regularization of inverse problems. *Inverse Problems Imaging*, 2011, **5**, 511–530.

23. Yang, Z. and Jacob, M. Nonlocal regularization of inverse problems: a unified variational framework. *IEEE Trans. Image Process.*, 2013, **22**, 3192–3203.

24. Lalush, D. S. and Tsui, B. M. W. A fast and stable maximum a posteriori conjugate gradient reconstruction algorithm. *Med. Phys.*, 1995, **22**, 1273–1284.

25. Fessler, J. A. and Booth, S. D. Conjugate-gradient preconditioning methods for shift-variant pet image reconstruction. *IEEE Trans. Image Process.*, 1999, **8**, 688–699.

26. Jiang, C., Li, H. and Rangaswamy, M. On the conjugate gradient matched filter. *IEEE Trans. Signal Process.*, 2012, **60**, 2660–2666.

27. Bregman, L.M. The method of successive projections for finding a common point of convex sets. *Sov. Math. Dokl.*, 1965, **6**, 688–692.

28. Gubin, L. G., Polyak, B. T. and Raik, E. V. The method of projections for finding the common point of convex sets. *USSR Comput. Math. Math. Phys.*, 1967, **7**, 1–24.

29. Aguena, M. L. S. and Mascarenhas, N. D. A. Multispectral image data fusion using POCS and super-resolution. *Comput. Vis. Image Understand.*, 2006, **102**, 178–187.

30. Jung, S. W., Kim, T. H. and Ko, S. J. A novel multiple image deblurring technique using fuzzy projection onto convex sets. *IEEE Signal Process. Lett.*, 2009, **16**, 192–195.

31. Patti, A. J., Sezan, M. I. and Tekalp, A. M. Superresolution video reconstruction with arbitrary sampling lattices and nonzero aperture time. *IEEE Trans. Image Process.*, 1997, **6**, 1064–1076.

32. Kwon, G. R., Kim, H. K., Kim, Y. and Ko, S. J. An efficient POCS-based post-processing technique using wavelet transform in HDTV. *IEEE Trans. Consum. Electron.*, 2005, **51**, 1283–1290.

33. Blatt, D. and Hero, A. O. Energy-based sensor network source localization via projection onto convex sets. *IEEE Trans. Signal Process.*, 2006, **54**, 3614–3619.

34. Shewchuk, J. R. 'An introduction to the conjugate gradient method without the agonizing pain', School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA, 1994.

35. Zhang, J. H., Wang, S. Q. and Yao, Z. X. Accelerating 3D Fourier migration with graphics processing units. *Geophysics*, 2009, **74**, WCA129–WCA139.

36. Wang, S. Q., Gao, X. and Yao, Z. Y. Accelerating POCS interpolation of 3D irregular seismic data with Graphics Processing Units. *Comput. Geosci.*, 2010, **36**, 1292–1300.

37. Ghoniem, M., Chahir, Y. and Elmoataz, A. Nonlocal video denoising, simplification and inpainting using discrete regularization on graphs. *Signal Process.*, 2010, **90**, 2445–2455.

38. Du, X., Cho, D. and Bui, T. D. Image segmentation and inpainting using hierarchical level set and texture mapping. *Signal Process.*, 2011, **91**, 852–863.

39. Bhavsar, A. V. and Rajagopalan, A. N. Range map super-resolution-inpainting, and reconstruction from sparse data. *Comput. Vis. Image Understand.*, 2012, **116**, 572–591.

40. Huan, X., Murali, B. and Ali, A. L. Image restoration based on the fast marching method and block based sampling. *Comput. Vis. Image Understand.*, 2010, **114**, 847–856.