Locally optimized non-local means denoising for low-dose X-ray backscatter imagery

Brian H. TRACEY [a,1], Eric L. MILLER [a], Yue WU [a], Christopher ALVINO [b]
Markus SCHIEFELE [b] and Omar AL-KOFAHI [b]
[a] *Department of Electrical and Computer Engineering, Tufts University, Medford, MA, 02155, USA*
[b] *American Science and Engineering, Billerica, MA, USA, 01821*

Abstract. While recent years have seen considerable progress in image denoising, the leading techniques have been developed for digital photographs or other images that can have very different characteristics than those encountered in X-ray applications. In particular here we examine X-ray backscatter (XBS) images collected by airport security systems, where images are piecewise smooth and edge information is typically more correlated with objects while texture is dominated by statistical noise in the detected signal. In this paper, we show how multiple estimates for a denoised XBS image can be combined using a variational approach, giving a solution that enhances edge contrast by trading off gradient penalties against data fidelity terms. We demonstrate the approach by combining several estimates made using the non-local means (NLM) algorithm, a widely used patch-based denoising method. The resulting improvements hold the potential for improving automated analysis of low-SNR X-ray imagery and can be applied in other applications where edge information is of interest.

Keywords: X-ray backscatter, image denoising, non-local means

1. Introduction

Denoising is a fundamental problem in image processing that continues to attract significant attention. Much recent work exploits *patch-based* approaches, which compare small regions in the image to find those that have similar appearance, relying on the fact that natural images often contain repeated patterns. Notable patch-based approaches include non-local means denoising (NLM) [1,2] as well as the BM3D collaborative filtering approach [3]. Generally speaking, these algorithms compare a small patch surrounding the pixel being denoised to other patches in the image, and perform a weighted average based on a patch similarity measure. Many variations on the original NLM algorithm have been developed

[1] e-mail:btracey@eecs.tufts.edu

that offer increased computationally efficiency [4,5,6], patch comparison in lower-dimensional spaces [7,8], automatic parameter selection [9,10,11], and alternative noise or statistical models [12,13,14].

Most authors have focused on the general image denoising problem, where recovering both texture and edge content is important. However, in some applications, the edge content of the images is most important, either because the underlying images have little inherent texture or because this texture is dominated by noise. Our work was motivated by one such application, namely denoising of X-ray backscatter (XBS) images collecting during airport screening [15]. These systems use low X-ray dosage to minimize radiation exposure. The resulting images have low signal-to-noise ratio (SNR), meaning that texture tends to be obscured and therefore edge content provides important cues. These images also display slowly varying intensity, so straightforward Total Variation (TV) denoising [16], which assumes piecewise constant images, is not appropriate. It is important to consider how existing denoising approaches can be adapted to these types of imaging scenarios.

In previous work [15] we found that NLM outperformed standard methods for XBS image denoising. Despite this, there is a need to further improve denoising performance. First, NLM noise suppression in the vicinity of unusual features (so-called 'rare patches') is suboptimal [17]. Second, while NLM does an excellent job at preserving high-contrast edges, it can blur low-contrast edges, making detection of low-contrast objects more challenging.

We propose to address these problems by combining multiple denoising estimates (generated using different parameter settings) via solution of an optimization problem that encourages locally flat solutions through the use of TV-type gradient penalties. Because we solve a separate optimization problem for each pixel in the image, the denoised signal can adapt to recover piecewise smooth images. In combining different denoising estimates, our work is related to a recent series of papers [18,17] which has shown that noise halo effects (residual noise near high-contrast features) can be reduced by using weighted sums of multiple NLM outputs, which are generated using different patch shapes. Because the weights used to combine patch shapes vary on a pixel-by-pixel basis, this weighting in effect creates a customized patch shape for each pixel. This approach allows patches near unusual features (rare patches) to deform so as to encompass a more homogeneous region, making the patch less 'rare'. Several alternatives have been explored for combining multiple patch shapes; for example uniform averaging, or weights that minimize variance [18]. Recently, a patch weighting that minimizes Stein's Unbiased Risk Estimate (SURE) was proposed [17] that successfully reduced noise halo effects. Other recent work on locally adapting denoising include efforts related to NLM [19] and BM3D [20]. The important distinction between our work and these earlier contributions is that we include gradient penalties designed to enhance recovery of weak edges.

The contributions and outline of our work are as follows. First, in Section 2 we describe an optimization problem that includes both data fidelity and edge penalty terms in order to find the optimal weights for combining multiple denoising estimates. Also in Section 2, we present an alternate version of the problem by making use of inequality bounds and Iteratively Reweighted Least Squares (IRLS)

methods [21,22,23,24], giving an approach that is computationally manageable. In Section 3 we show this formalism leads to edge enhancement in denoised XBS images and quantify improvements in edge contrast. After briefly commenting on XBS dose estimation from imagery, we conclude and discuss possible future extensions in Section 4.

2. Methods

2.1. Calculation of individual denoising estimates

Denoising addresses the problem of recovering the true image values given a set of noisy observations. Assuming the image is unwrapped into vector form so pixels are labeled with a single subscript j, we have

$$v(j) = u(j) + n(j) \tag{1}$$

where $v(j)$ is the data, $u(j)$ is the noise-free image and $n(j)$ is additive noise. For a given pixel location, spatial domain methods reduce noise by averaging the data over neighboring pixels k using a kernel function:

$$\hat{u}(j) = \frac{\sum_{k \in N_{srch}(j)} K(j,k,\mathbf{v}) v(k)}{\sum_{k \in N_{srch}(j)} K(j,k,\mathbf{v})} \tag{2}$$

where $K(\cdot)$ denotes a kernel function, $N_{srch}(j)$ is a specified searching window centered at the jth pixel, and $\mathbf{v}$ is a vector containing all the pixel observations $v(j)$. For classical spatial domain methods, the kernel function used is data-independent, which has the advantage that filter coefficients can be precomputed and computation reduced. For example, a Gaussian smoothing kernel depends only on the spatial distance between the pixels being compared:

$$K_{\mathrm{GS}}(j,k,\mathbf{v}) = K_{\mathrm{GS}}(j,k) = \exp\left\{-\|l-k\|^2/h\right\} \tag{3}$$

where the parameter h controls the spatial width of the filter. In contrast, methods such as NLM use data-adaptive (and spatially varying) weights to achieve better performance. Each pixel location j is associated with a patch, typically a square region centered on the pixel. The NLM kernel calculates the weights based on measures of similarity between patches centered on points j and k. The most common similarity measure for NLM is the mean-squared patch difference [9], for which

$$\begin{aligned} K_{NLM}(j,k,\mathbf{v}) &= \exp\left(-\frac{\sum_{\delta \in \Delta}(v(j+\delta)-v(k+\delta))^2}{2L_\Delta \lambda^2}\right) \\ &\equiv \exp\left(-\frac{d^2(j,k)}{2L_\Delta \lambda^2}\right). \end{aligned} \tag{4}$$

(note [1] defined the denominator above as h^2). In Eq. (4), λ is a bandwidth parameter, while Δ represents a local patch of pixels surrounding j, containing L_Δ pixels; a patch of the same shape also surrounds k, and δ indicates the offset from each patch center. Although a variety of patch shapes are possible [1,17], square patches centered on the points of interest are most common. In (4) we have defined d^2 to denote the summed, squared point-by-point difference between pixels in the patches centered on j and k. If similar patches can be found throughout the image, then ideally the neighborhood N_{srch} is taken to be the entire image, so the averaging process is fully *non-local*. In practice, N_{srch} is usually limited to reduce computational load. For example, [2] chose 7×7 patches for Δ and a 21×21 pixel search region. The key NLM parameters are the patch size, specified as a half-width W (so $L_\Delta = (2W + 1)^2$), the size of N_{srch}, specified as a half-width M, and the bandwidth λ.

2.2. Convex combination of denoising estimates incorporating gradient penalties

We next consider methods for combining several estimates of a denoised image. Assume that we generate P different estimates of the denoised image, corresponding to different choices of denoising parameters (for example, different averaging weights in the case of moving average filters, or different NLM patch shapes or bandwidth parameters in the case of NLM denoising). For each output image we can calculate image gradients in x and y using finite differences of adjacent denoised pixel values. Thus we have available to us P different denoising estimates, denoted termed $\hat{u}_p(j)$, and the corresponding $x-$ and $y-$ gradients $(\nabla_x \hat{u}_p(j), \nabla_y \hat{u}_p(j))$. We seek a vector of pixel-dependent weights that combines these patch estimates to yield the estimate for pixel j:

$$\hat{u}(j) = \sum_{p=1}^{P} w_p(j) \hat{u}_p(j) \tag{5}$$

To determine the weights w_p, we will define a neighborhood N_j around j and seek to minimize the sum of a data fidelity term and a Total Variation penalty:

$$J(\mathbf{w}(j)) = \frac{1}{2} \sum_{l \in N_j} \left(\hat{u}(l) - v(l)\right)^2 + \lambda_1 \sum_{l \in N_j} |\nabla \hat{u}(l)|_1 \tag{6}$$

where $\mathbf{w}(j)$ is the vector of weights for pixel j, and $|\nabla \hat{u}|_1$ is the anisotropic Total Variation, namely

$$|\nabla \hat{u}(l)|_1 = | \nabla_x \hat{u}(l)| + |\nabla_y \hat{u}(l)|. \tag{7}$$

Minimization of the Total Variation (TV), or $l1$ norm of the gradient, has been widely adopted as an edge-preserving image reconstruction method since its introduction in the classic ROF paper [16]. Note that in the above, N_j is the set of pixels over which the optimization terms are calculated, and is not related to the NLM patch size.

Next, the expansion in Eq. (5) is substituted into Eq. (6). Here we use the fact that because the new $\hat{u}$ is a linear expansion of the component $\hat{u}_p$, the new gradients are linear expansions of the component gradients. Thus, we seek to find weights $w_p(j)$ that minimize the quantity

$$J(\mathbf{w}(j)) = \frac{1}{2}\sum_{l\in N_j}\left(\sum_{p=1}^{P} w_p(j)\hat{u}_p(l) - v(l)\right)^2 + \lambda_1 \sum_{l\in N_j}\left(|\sum_{p=1}^{P} w_p(j)\nabla_x \hat{u}_p(l)| + |\sum_{p=1}^{P} w_p(j)\nabla_y \hat{u}_p(l)|\right) \tag{8}$$

In addition, we require that the weight are all non-negative and sum to unity:

$$\begin{cases} \sum_p w_p = 1 \\ \forall p\,, w_p \succ 0 \end{cases} \tag{9}$$

We note that all patch weights explored in [17,18] also required nonnegative weights, as in Eq. (9). After finding the optimal weights for each pixel j, namely

$$\mathbf{w}^*(j) = \arg\min_{\mathbf{w}(j)} J(\mathbf{w}(j)) \tag{10}$$

subject to Eq. (9), our final denoised estimate is found by using the resulting $w_p^*(j)$ (individual elements of $\mathbf{w}^*(j)$) in Eq. (5).

Fig. 1 illustrates how combining denoising estimates found from various patch shapes can be beneficial. Subfigures a-c) show several NLM outputs corresponding to centered and offset patches. By shifting the patch relative to the pixel being denoised (shown as a circle) it is possible to generate output images that correctly estimate the homogeneous background on one side or the other of a high-contrast object, but not both. By combining the multiple outputs (using the solution approach proposed below) a final higher-quality image is generated.

2.3. Simplification using inequality bounds

Solving the optimization problem described above on a pixel-by-pixel basis is very computationally challenging. To reduce computation, we propose an approximation based on judicious use of inequality bounds. Our solution is based on the Iteratively Re-weighted Least Squares (IRLS) method [21,22]. Specifically, IRLS solves an $l1$ penalized type of problem iteratively where at each stage a set of normal-type equations must be solved. Within the context of the problem of interest here, we solve a two-stage problem: A) find the solution to a modified version of Eq. (8) where the TV penalty is replaced by a quadratic penalty, and B) modify the resulting normal equations according to the approach in [21,22] to obtain the IRLS iteration.

Temporarily moving to a quadratic penalty (step 'A') and adding a Lagrange multiplier λ_c to capture the constraint on the sum of the weights, Eq. (8) can be rewritten to give the quantity to be minimized:

$$J(\mathbf{w}(j),\lambda_c) = \lambda_c\left(\sum_p w_p(l) - 1\right) + \sum_{l\in N_j}\frac{1}{2}\left(\sum_p w_p(j)e_p(l)\right)^2$$
$$+ \lambda_1 \sum_{l\in N_j}\left((\sum_p w_p(j)\nabla_x\hat{u}_p(l))^2 + (\sum_p w_p(j)\nabla_y\hat{u}_p(l))^2\right) \quad (11)$$

subject to the constraints in Eq. (9). Here for convenience we have defined the *method noise* for each estimate as $e_p(l) = \hat{u}_p(l) - v(l)$, and have also written the overall method noise as the sum of the P individual method noise terms: $e(l) = \sum_p w_p(l)e_p(l)$ (this follows from the fact we constrain the weights to sum to 1). Now, we can employ an inequality relation to minimize an upper bound. For the first term, we have

$$\left(\sum_p w_p(j)e_p(l)\right)^2 \leq 2\sum_p w_p^2(j)e_p^2(l) \quad (12)$$

and similarly for the gradient penalty terms. The use of the upper bound leads to the simpler problem (absorbing a factor of 2 in to λ_1)

$$J(\mathbf{w}(j),\lambda_c) = \sum_{l\in N_j}\left[\sum_p w_p^2(j)e_p^2(l) + \lambda_1\sum_p w_p^2(j)\nabla\hat{u}_p^2(l)\right]$$
$$+ \lambda_c(\sum_p w_p(j) - 1) \quad (13)$$

again subject to Eq. (9); like Eq. (11), this minimizes a weighted sum of data fidelity and image gradient penalties. To simplify notation above, we have defined the squared magnitude of the gradient as

$$\nabla\hat{u}_p^2(l) = (\nabla_x\hat{u}_p(l))^2 + (\nabla_y\hat{u}_p(l))^2. \quad (14)$$

Taking derivatives with respect to each weight w_p and setting to zero, we find

$$\frac{\partial J(\mathbf{w}(j),\lambda_c)}{\partial w_p} = 2w_p(j)\left(\sum_{l\in N_j} e_p^2(l) + \lambda_1\nabla\hat{u}_p^2(l)\right) + \lambda_c = 0 \quad (15)$$

which can be solved to give w_p. The constant λ_c is then eliminated by summing all w_p's and using the knowledge that they sum to one. This gives the result for a quadratic penalty:

$$w_p(j) = \frac{1/\left(\sum_{l\in N_j} e_p^2(l) + \lambda_1\nabla\hat{u}_p^2(l)\right)}{\sum_k 1/\left(\sum_{l\in N_j} e_k^2(l) + \lambda_1\nabla\hat{u}_k^2(l)\right)} \quad (16)$$

Note that because all quantities are positive, the weights are all non-negative as desired.

Given this solution, the next step (step 'B') is to move back to imposing a TV-type penalty on the gradient. To do so, one approach is Iteratively Reweighted Least Squares (IRLS). Let t be the iteration number. Following [25] we introduce additional weighting terms $\gamma^{(t)}(l)$ into the optimization as follows.

$$w_p^{(t+1)}(j) = \frac{1/\left(\sum_{l\in N_j} e_p^2(l) + \lambda_1 \gamma^{(t)}(l) \nabla \hat{u}_p^2(l)\right)}{\sum_k 1/\left(\sum_{i\in N_j} e_k^2(l) + \lambda_1 \gamma^{(t)}(l) \nabla \hat{u}_k^2(l)\right)} \tag{17}$$

where the weighting factor is chosen to be

$$\gamma^{(t)}(l) = \frac{1}{\sqrt{\left(\nabla \hat{u}^{(t)}(l)\right)^2 + \epsilon}} \tag{18}$$

where ϵ is a small constant typically added to protect against division by zero [22]. Note that in Eq. (17), we iterate only on the weights; the underlying estimates $\hat{u}_p(l)$ are not iterated. The effect of the weighting factor is that as the solution converges, the gradient penalty converges to the desired L1 penalty.

In [17], the authors calculated patch weightings based on Stein' Unbiased Risk Estimate (SURE) and showed that it was helpful to smooth their SURE estimate prior to calculating weights. We also found it helpful to smooth the method noise estimates that are used in Eq. (17). We adapted the approach (and code) from [17] to estimate a filtered version of the method noise. In Eq. 18 of [17], the authors smooth their pixel-wise SURE estimate by using the divergence portion of the SURE, which is readily calculated for NLM. We apply exactly the same averaging to the method noise estimates, giving filtered estimates $\mathbf{e}_{p,filt}$ as follows:

$$e_{p,filt}(j) = \frac{1}{C(j)} \sum_k \mathbf{1}_{|\mathbf{DIV}_j - \mathbf{DIV}_\mathbf{k}| < \kappa} \mathbf{e}_p(k) \tag{19}$$

where $C(j)$ is a normalizing term, $\mathbf{DIV}_j$ is a vector containing the divergence portion of the SURE term at pixel j, κ is a threshold calculated from the data, and the notation $\mathbf{1}_{\_}$ indicates that all pixels where the divergence condition is satisfied are selected for averaging (full details are found in [17]).

Because our solution is locally optimized (solved pixel-by-pixel), we refer to it below as the 'Locally Edge-enhancing Optimization' (or 'LEO') result.

2.4. Simplification to quadratic penalty

For comparison purposes, we also calculate the solution if we impose a quadratic penalty on the gradient, rather than a TV-type penalty. The quadratic penalized solution is given by Eq. (16), and is simply the result found during the first iteration of IRLS. While the quadratic penalty will discourage roughness in the solution and can provide some benefit, it acts as a smoothing penalty [26] and tends to lead to solutions in which an edge is smeared across several pixels, rather than being captured by a sharper but more localized transition. We will refer to the quadratic penalty solution as 'Q-LEO' in results below.

3. Results

Here we present denoising results for a set of XBS images collected by American Science & Engineering during product development (as a note, images shown here are not from TSA scanners). The example images analyzed in this paper are shown in Fig. 2. While we explored a larger dataset during development, these images were selected as there are no privacy issues with them (i.e., all images are upper-body views of male subjects). Later figures will zoom in on specific features of interest within these images, but image metrics are reported for the full image.

Processing steps

Because the NLM weights in Eq. (4) are derived for Gaussian noise, we first apply the variance-stabilizing Anscombe transform to the X-ray data, following [27] and using code provided by the authors. After denoising the transformed images, we used an inverse transform to return to the original image domain, using the exact inverse found in [27]. This approach has been shown to perform as well as image denoising techniques that explicitly account for non-Gaussian statistics [27].

After applying the Anscombe transform, we compute a set of individual NLM estimates using several patch shapes. All of our results employ nine patch shapes per pixel, all of size 7×7, and search neighborhoods of 15×15 pixels. All are square patches, with the pixel being denoised located at different locations within the patch: the patch center (as in standard NLM), at the four corners, and at the midpoint of the 4 edges of the patches. Fig. 1 shows three of the patch shapes used. Because the true noise variance is not known in our data, we chose a noise estimate based on visual inspection of results, giving a value of $\sigma = .019$ (note, we normalized images to lie in the range [0, 1]). The NLM bandwidth parameter was then set to $\lambda = \sigma^2/2$, as suggested in [17].

In subsequent analysis, we confirmed this choice of noise variance by estimating it from each of the Anscombe-transformed images used in analysis, using the method described in [28] and implemented in [29]. The median estimated standard deviation was $\sigma = .0197$ and all values were contained in the range $[0.018 - 0.022]$. This confirms that our visual estimate was reasonable, but more importantly demonstrates that noise appears to be fairly stable across our dataset, and that the noise parameters can be directly estimated from the data.

To understand the effects of the edge penalty, we sweep the values of λ_1 over a range of 11 values (1e-5, 1e-4, 1e-3, 1e-2, 0.1, 0.2, 0.35, 0.5, 0.75, 1.0, 1.5). For each value, we compute both the L1-penalized result ('LEO') and the quadratic-penalized result ('Q-LEO').

We compare our methods (LEO and Q-LEO) against standard NLM. In addition, we compare our results against the exponentially weighted average (EWA) aggregation used in [17], and also more recently employed in [30]. In the EWA method, the individual denoising estimates are combined using Eq. (5), but with weighting factors found as

$$w^*_{p,EWA}(j) = \frac{\exp(-\mathrm{SURE}_p(j)/T)}{\sum_q \exp(-\mathrm{SURE}_q(j)/T)} \tag{20}$$

with T, the temperature value, set to $0.4\sigma^2$, and $\text{SURE}_p(j)$ being an estimate of Stein's Unbiased Risk Estimate for denoising estimate p and pixel j. Because we used a different set of denoised images as input to the EWA aggregation, our EWA results differ from those presented in [17] (even we use the authors' EWA code). However, by using the same inputs to EWA and our method, we are able to directly compare the two aggregation methods. The EWA was employed in [17] to reduce 'rare patch' effects by combining multiple NLM estimates generated by different patches shapes, under the assumption that at least some of those patch shapes will be relatively homogeneous. Unlike LEO, the EWA approach in [17] does not directly address the question of preserving low-contrast edges. Finally, we also compare against an implementation [31,32] of Total Variation, using a Split Bregman approach [32,33].

Because we do not have reference (noise-free) images for our XBS data, we are not able to compute commonly used metrics such as peak signal-to-noise ratio (PSNR) or structural similarity index (SSIM) [34]. However, we are able to measure edge contrast using the edge-based contrast metric (EBCM) from [35], which captures image contrast by measuring the intensity of edge pixels in local image regions.

Example images and results

Fig. 3 shows a first example of denoising an XBS image, zooming in on an organic package (light area) that the subject is wearing beneath clothing, near the metallic belt-buckle (dark area). Standard NLM (b) usefully denoises the image as compared to the raw data (a), but slightly blurs the package outline. In addition, greater residual noise can be seen near the belt buckle, as the buckle feature has few matching regions in the image and these pixels belong to 'rare patches'. Subfigures (c)-(e) show LEO results with increasing values of the gradient penalty weight; note the increased sharpness of the vertical edges of the package as compared to standard NLM. Finally, subfigure (f) shows that the Q-LEO result is much smoother, even which high values of the gradient penalty are used, and edges remain blurred. In (c)-(f), the optimization problem at each pixel is solved without averaging across neighboring pixels (in other words, the neighborhood N_j is a 1×1 region).

Because Total Variation is designed to be edge-preserving, it is interesting to compare how a straightforward TV denoising performs for our image set. Fig. 4 shows the result of a Split Bregman implementation of TV denoising [31] for several values of the TV regularization parameter. When the TV penalty is small (subfigure a)), significant noise remains; as the penalty is increased, the noise is smoothed but edge features in the image are also smoothed, most notably the dark belt buckle feature in the lower middle of the image. We varied the TV penalty from values that visually gave under-smoothed images up to those giving oversmoothed images (ranging from 0.01 - 0.1, in 10 steps). The best-appearing image for this subject is in subfigure b), but one can observe that the belt buckle is already smeared in this result.

Fig. 3 showed results for LEO as solved locally on a 1-pixel neighborhood (i.e., N_j). This can sometimes lead to slight roughness or pixelation in the outline

of the object. To address this, the LEO result can be optimized over a small local neighborhood, for example 3×3 neighborhoods. Fig. 5 compares 1×1 and 3×3 neighborhoods N_j for several XBS examples. As can be seen, the edges become more continuous, at the cost of a slight decrease in the contrast of the edges.

Several further examples of results obtained using LEO are shown in Figs. 6-8. In each plot, subfigure (a) shows the raw data, subfigure (b) shows standard NLM output, (c) shows the EWA result, and (d) shows the LEO result for $\lambda_1 = 0.2$, which provided visually good results across the set of images.

In Fig. 6, the subject has an organic object below the arm. While visible on both standard NLM and EWA results, the outlines of this object are somewhat blurred as compared to the raw data. The LEO result shows a visually sharper edge to the object, particularly along the lower edge.

A similar result is shown in Fig. 7; note here the increased contrast along the two vertical edges.

Fig. 8 shows a challenging case in which the contrast between the organic material and the body surface is quite small and is clearly blurred in the standard NLM result. Here, the proposed LEO approach helps to clarify the object outline. Edge enhancement in cases such as this holds potential for improving performance of edge-detection-based image analysis algorithms.

Fig. 9 shows the effect of changing the LEO gradient penalty weight on the edge contrast metric, for subject 25. At small values of λ_1, the penalty has negligible effect and LEO and Q-LEO results are equivalent; note however that the averaging of various shapes to minimize data mismatch does provide a benefit in the EBCM metric. As λ_1 is increased, the edge contrast in the L1-penalized LEO result increases. On the other hand, when the penalty is quadratic, smooth solutions are encouraged and the EBCM metric drops as λ_1 is increased. These results are typical of those seen across the range of images, though the trends are not as clear for some subjects. However, the trend to increased edge contrast for LEO as λ_1 increases, with less contrast for Q-LEO, matches the trends seen in the other plots.

Table 1 shows values for the contrast metric across all images. The proposed LEO method solved on a per-pixel basis (1×1) produces higher-contrast images than either standard NLM, the quadratic gradient, or EWA for 7 of the 9 images, with Q-LEO doing slightly better on one of the remaining images. Subject 39 is unusual in that there are several metal objects that have large halo effects in standard NLM, so the EBCM metric may be over-optimistic for standard NLM in this case as it will give an increased score to bright pixels that may be caused by noise. Here we are selecting out the LEO and Q-LEO values for $\lambda_1 = 0.2$, which does not always produce the highest EBCM but was subjectively judged to give good visual quality. For TV, the results plotted are the maximum across all TV regularization values tested. Although this selection approach gives TV a potentially unfair advantage in the comparison, all NLM variants except EWA achieve higher EBCM scores than TV denoising.

All results presented above are for square patches with varying offsets, which allows us to exploit fast integral image calculation methods [5]. However, the wide variety of edge geometries means that there should be benefit to considering more general patch shapes, and computationally efficient methods to computing results

for the arbitrary patch shapes have been proposed [17]. As an illustration that the LEO method can be applied to arbitrary patch shapes, we used code provided by the authors of [17] to compute NLM-denoised images for a series of 15 circular and semi-circular patch shapes with different diameters and orientations (see demo example from [36]). We then combined these patches using the LEO framework. An example result is shown in Fig. 10, which compares a denoised result using a circular patch with the LEO combination. Subfigure a) shows the result for a circular patch (mid-sized circle from [36]); some residual noise is observed near the dark metal objects due to rare patch effects, which is removed in the LEO result with $\lambda_1 = 0.2$ (subfigure b). While a more complete study of the benefit of various patch shapes is a topic for future work, this example suggests that the proposed LEO method can provide benefits for more general patch shapes.

Computational load

For a 256x256 pixel image, finding the weights with our triangle inequality / IRLS approach required 49 sec, vs. 0.4 sec for EWA and 0.1 sec for the L2-normed weight penalty. All run times were less than the 116 sec needed to compute the 10 NLM estimates for different patch shapes. For 260x480 pixel images (those shown in Fig. 2), run times essentially doubled, with 246 sec needed to generate the 10 NLM estimates, and 99 sec needed for the triangle inequality / IRLS approach. Testing was done under Matlab 2012b on a 2GHz processor running Linux.

Likely effect on edge detection

Edge detection is a key low-level element of machine vision system designed to analyze images, particularly images such as XBS where edge information plays a key role in analysis. Edge detection algorithms can be expected to perform better for images where gradients show a clearer edge. Thus, in Fig. 11 the absolute value of image gradients (calculated using Matlab 'gradient' command) are shown for two images, both before and after denoising. As the upper subfigures show, denoising of some sort is needed as otherwise the high-frequency noise in the data obscures the true edge map. Improvements are seen when using NLM denoising (subfigures (c) - (d)), and use of the LEO approach ((e)-(f)) noticeably boosts the ability to identify edge pixels. Thus even without generating system-level results, we can see that the LEO method can be expected to boost system performance.

XBS dose estimation from imagery

The radiation dose experienced by the traveling public due to XBS scanning has been a topic of public debate. In [37], Rez *et al.* measured the noise in publicly available XBS images and, assuming the noise is described by Poisson statistics, estimated the X-ray dose needed to generate the observed noise. This led to estimates higher than those reported by manufacturers or the government, though other investigators have estimated lower dosages [38]. In the context of the work presented here, it is important to note that estimating X-ray dose from the denoised images presented above would lead to a significant over-estimate; while the denoised images have noticeably lower noise than the raw data, this is clearly

due to post-processing effects. In general, it is important to establish that no post-processing has been applied to XBS images before attempting to estimate dose from image noise.

4. Discussion and Conclusions

Here we have described a method for combining multiple denoising estimates in which we seek to minimize image gradients while simultaneously minimizing data mismatch. This denoising method is specifically developed for applications such as the low-dose X-ray backscatter (XBS) application discussed above, where edge information is of interest. We have formulated the problem such that relatively fast computationally solutions are feasible for local (pixel-wise) solutions of the optimization problem. The examples above demonstrate that local optimization gives the user a means to trade off edge enhancement vs. other measures of image quality, enhancing low-contrast features while maintaining the noise halo reductions seen in other multi-patch methods [17].

We envision several useful paths forward for future research. In our implementation, we mainly examined combinations of NLM estimates made using offset rectangular patch shapes, although we demonstrate that the proposed method can be used to combine results for arbitrary patch shapes [17]. More fully exploring the use of more general patch shapes, in combination with the local optimization methods proposed here, is a promising path for further improving denoising performance. As a second research direction, we note that while the results shown above were all for combinations of multiple non-local means denoising estimates, the basic approach can be applied to combine nearly any type of denoising estimate. The computational load of the proposed technique is mainly incurred due to the need to estimate multiple NLM outputs. For real-time applications, an intriguing idea is to combine multiple moving average filter estimates. Initial experiments suggest that this approach can provide high-quality denoising results at a greatly reduced cost, and this concept will be explored in a future publication.

Acknowledgment

This work was supported by the U.S. Department of Homeland Security under Award 2008-ST-061-ED0001, by American Science and Engineering (AS&E), and by BBN under MADCAT funding. The authors are grateful for the insightful and detailed comments of the reviewers, which greatly strengthened the paper.

References

[1] A. Buades, B. Coll, and J. M. Morel. A review of image denoising algorithms, with a new one. In *Simul*, pages 490–530, 2005.

[2] A. Buades, B. Coll, and J. M. Morel. A Non-Local Algorithm for Image Denoising. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2 of *CVPR '05*, pages 60–65, Washington, DC, USA, June 2005. IEEE.

[3] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *Image Processing, IEEE Transactions on*, 16(8):2080–2095, 2007.

[4] M. Mahmoudi and G. Sapiro. Fast image and video denoising via nonlocal means of similar neighborhoods. *IEEE Signal Processing Letters*, 12(12):839–842, December 2005.

[5] J. Darbon, A. Cunha, T. F. Chan, S. Osher, and G. J. Jensen. Fast nonlocal filtering applied to electron cryomicroscopy. In *Biomedical Imaging: From Nano to Macro, 2008. ISBI 2008. 5th IEEE International Symposium on*, pages 1331–1334. IEEE, May 2008.

[6] Jin Wang, Yanwen Guo, Yiting Ying, Yanli Liu, and Qunsheng Peng. Fast Non-Local Algorithm for Image Denoising. In *2006 International Conference on Image Processing*, pages 1429–1432. IEEE, October 2006.

[7] Tolga Tasdizen. Principal neighborhood dictionaries for nonlocal means image denoising. *Trans. Img. Proc.*, 18(12):2649–2660, December 2009.

[8] N. Azzabou, N. Paragios, and F. Guichard. Image Denoising Based on Adapted Dictionary Computation. In *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, volume 3, pages III – 109–III – 112. IEEE, 2007.

[9] D. Van De Ville and M. Kocher. SURE-Based Non-Local Means. *Signal Processing Letters, IEEE*, 16(11):973–976, November 2009.

[10] Dimitri Van De Ville and M. Kocher. Nonlocal Means With Dimensionality Reduction and SURE-Based Parameter Selection. *IEEE Transactions on Image Processing*, 20(9):2683–2690, September 2011.

[11] Yue Wu, B. Tracey, P. Natarajan, and J. P. Noonan. JamesStein Type Center Pixel Weights for Non-Local Means Image Denoising. *Signal Processing Letters, IEEE*, 20(4):411–414, April 2013.

[12] Yue Wu, B. Tracey, P. Natarajan, and J. P. Noonan. Probablistic Non-Local Means. *(accepted) Signal Processing Letters, IEEE*, 2013.

[13] Charles-Alban Deledalle, Florence Tupin, and Loic Denis. Poisson NL means: Unsupervised non local means for Poisson noise. In *Image Processing (ICIP), 2010 17th IEEE International Conference on*, pages 801–804. IEEE, September 2010.

[14] Charles-Alban Deledalle, Loïc Denis, and Florence Tupin. How to Compare Noisy Patches? Patch Similarity Beyond Gaussian Noise. *International Journal of Computer Vision*, 99(1):86–102, March 2012.

[15] B. Tracey, E. Miller, M. Schiefele, C. Alvino, and O. Al-Kofahi. Denoising approaches for X-ray personnel screening systems. In *2012 International Conference on Technology for Homeland Security*. IEEE, November 2012.

[16] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, November 1992.

[17] Charles-Alban Deledalle, Vincent Duval, and Joseph Salmon. Non-local Methods with Shape-Adaptive Patches (NLM-SAP). *Journal of Mathematical Imaging and Vision*, pages 1–18, May 2011.

[18] J. Salmon and Y. Strozecki. From patches to pixels in Non-Local methods: Weighted-average reprojection. In *Image Processing (ICIP), 2010 17th IEEE International Conference on*, pages 1929–1932. IEEE, 2010.

[19] Arian Maleki, Manjari Narayan, and Richard G. Baraniuk. Anisotropic nonlocal means denoising. *Applied and Computational Harmonic Analysis*, December 2012.

[20] Kostadin Dabov, Ro Foi, Vladimir Katkovnik, and Karen Egiazarian. A Non-Local and Shape-Adaptive Transform-Domain Collaborative Filtering. In *in Proc. 2008 Int. Workshop on Local and Non-Local Approximation in Image Processing, LNLA 2008*, 2008.

[21] Michael Elad. *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. Springer, 1st edition. edition, August 2010.

[22] *Iteratively reweighted algorithms for compressive sensing*, 2008.

[23] Ingrid Daubechies, Ronald DeVore, Massimo Fornasier, and C. Sinan Güntürk. Iteratively reweighted least squares minimization for sparse recovery. *Comm. Pure Appl. Math.*, 63(1):1–38, 2010.

[24] I. F. Gorodnitsky and B. D. Rao. Sparse signal reconstruction from limited data using FOCUSS: a re-weighted minimum norm algorithm. *Signal Processing, IEEE Transactions*

on, 45(3):600–616, 1997.

[25] Antonin Chambolle and Pierre-Louis Lions. Image recovery via total variation minimization and related problems. *Numerische Mathematik*, 76(2):167–188, April 1997.

[26] S. N. Wood. Modelling and smoothing parameter estimation with multiple quadratic penalties. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 62(2):413–428, January 2000.

[27] M. Makitalo and A. Foi. Optimal Inversion of the Anscombe Transformation in Low-Count Poisson Image Denoising. *Image Processing, IEEE Transactions on*, 20(1):99–109, January 2011.

[28] J. Immerkr. Fast Noise Variance Estimation. *Computer Vision and Image Understanding*, 64(2):300–302, 1996.

[29] T. Birdal. *Fast Noise Estimation in Images*. MATLAB Central File Exchange, May 2012.

[30] Hoosein Talebi, Xiang Zhu, and Peyman Milanfar. How to saif-ly boost denoising performance. *Image Processing, IEEE Transactions on*, 22(4):1470–1485, 2013.

[31] Benjamin Tremoulheac. Split bregman method for total variation denoising (matlab code), April 2012.

[32] Tom Goldstein and Stanley Osher. The Split Bregman Method for L1-Regularized Problems. *SIAM J. Img. Sci.*, 2(2):323–343, April 2009.

[33] Charles A. Micchelli, Lixin Shen, and Yuesheng Xu. Proximity algorithms for image models: denoising. *Inverse Problems*, 27(4):045009+, April 2011.

[34] Zhou Wang, Alan Bovik, Hamid Sheikh, and Eero. Simoncelli. Contrast enhancement technique based on local detection of edges. *Image Processing, IEEE Transactions on*, 13(4):600–612, April 2004.

[35] A. Beghdadi and A. Le Negrate. Contrast enhancement technique based on local detection of edges. *Computer Vision, Graphics, and Image Processing*, 46(2):162–174, May 1989.

[36] J. Salmon. *NLM Shape-Adapative Patches (NLM-SAP)*. www.josephsalmon.eu, 2011.

[37] P. Rez, R.L. Metzger, and K.L. Mossman. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *Radiat Prot Dosimetry*, 145(1):75–81, 2011.

[38] Michael E. Hoppe and Taly G. Schmidt. Estimation of organ and effective dose due to Compton backscatter security scans. *Medical Physics*, 39(6):3396–3403, May 2012.

Table 1. EBCM edge contrast metrics for *locally* optimized XBS images The approach with highest EBCM is shown in bold.

ID	Standard	TV	EWA	Q-LEO 1×1	Q-LEO 3×3	LEO 1×1	LEO 3×3
6	10.52	8.56	3.01	11.06	10.79	**11.08**	10.84
8	10.55	8.58	2.93	10.56	10.48	**10.71**	10.33
25	11.44	8.45	2.98	11.55	11.43	**11.71**	11.44
30	11.52	8.44	3.24	11.51	11.36	**11.64**	11.22
39	**12.97**	8.56	2.98	12.08	11.76	10.76	10.43
46	11.19	9.32	2.60	11.42	11.26	**11.56**	11.22
63	11.71	9.53	3.21	12.01	11.99	**12.19**	11.90
69	12.14	9.72	3.47	**12.34**	12.18	12.03	11.89
75	10.43	8.17	3.37	10.27	10.24	**10.58**	10.18

List of Figure captions

Figure 1: Several NLM denoisings of cameraman image, with associated offset patches (square shows patch shape, while dot shows pixel being denoised). Subfigure a) uses a centered patch. Subfigures b) and c) use offset patches as shown, causing noise halos to be concentrated to left or right of the cameraman's head. Subfigure d) shows a pixel-by-pixel optimized combination of patches as described in the text.
Figure 2: XBS images used in testing, prior to denoising.
Figure 3: Results of local optimization on Subject 30 image. a) Noisy image, b) Standard NLM, c) LEO, 1×1, $\lambda_1 = 1e-5$, d) LEO, 1×1, $\lambda_1 = 0.2$, e) LEO, 1×1, $\lambda_1 = 1.5$, f) Q-LEO, 1×1, $\lambda_1 = 1.5$. For the LEO result, package edges become sharper as regularization increases; the Q-LEO result is much smoother.
Figure 4: Total Variation (TV) denoising for Subject 30 image, for different TV regularization parameters: a) $\lambda_{TV} = 0.02$, b) $\lambda_{TV} = 0.05$, c) $\lambda_{TV} = 0.08$, d) $\lambda_{TV} = 0.1$. Note that for higher values, the dark belt-buckle feature is eroded.
Figure 5: Difference between 1x1 and 3x3 local LEO solutions. a) 1x1, Subject 20, b) 1x1, Subject 46, c) 3x3, Subject 20, d) 3x3, Subject 46.
Figure 6: Results of local optimization XBS image, Subject 25. a) Noisy image, b) standard NLM result, c) EWA combination, d) LEO local combination. Note the improvement in edge contrast.
Figure 7: Results of local optimization XBS image, Subject 46. a) Noisy image, b) standard NLM result, c) EWA combination, d) LEO local combination. Note the improvement in edge contrast.
Figure 8: Results of local optimization XBS image, Subject 75. a) Noisy image, b) standard NLM result, c) EWA combination, d) LEO local combination. Note the improvement in edge contrast.
Figure 9: Results of varying λ_1 for Subject 25 image. As regularization parameter increase, edges become sharper as measured by the EBCM metric when gradient is penalized according to the L1 measure; when an L2 (quadratic) penalty is applied, image becomes smoother and EBCM decreases.
Figure 10: Results for non-square patch shapes for Subject 8 image. a) 'standard' NLM result but using circular patch shape, b) LEO local combination of 15 circular- and semi-circular patches. Note the smoother result near the belt buckle (dark object at lower left).
Figure 11: Image gradients for two XBS images. a) and b), gradient of raw images for subjects 30 and 75, respectively. c) and d), gradient of standard NLM outputs; e) and f), gradients of proposed LEO method.

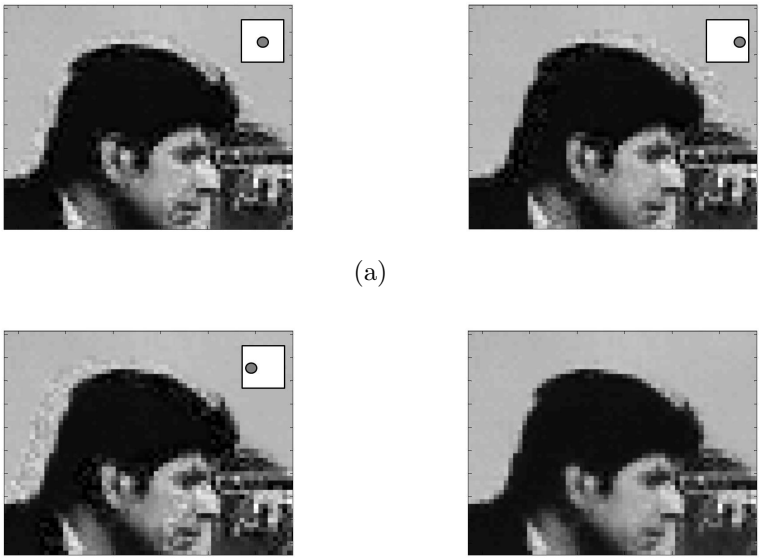

Figure 1. Several NLM denoisings of cameraman image, with associated offset patches (square shows patch shape, while dot shows pixel being denoised). Subfigure a) uses a centered patch. Subfigures b) and c) use offset patches as shown, causing noise halos to be concentrated to left or right of the cameraman's head. Subfigure d) shows a pixel-by-pixel optimized combination of patches as proposed below.

Figure 2. XBS images used in testing, prior to denoising.

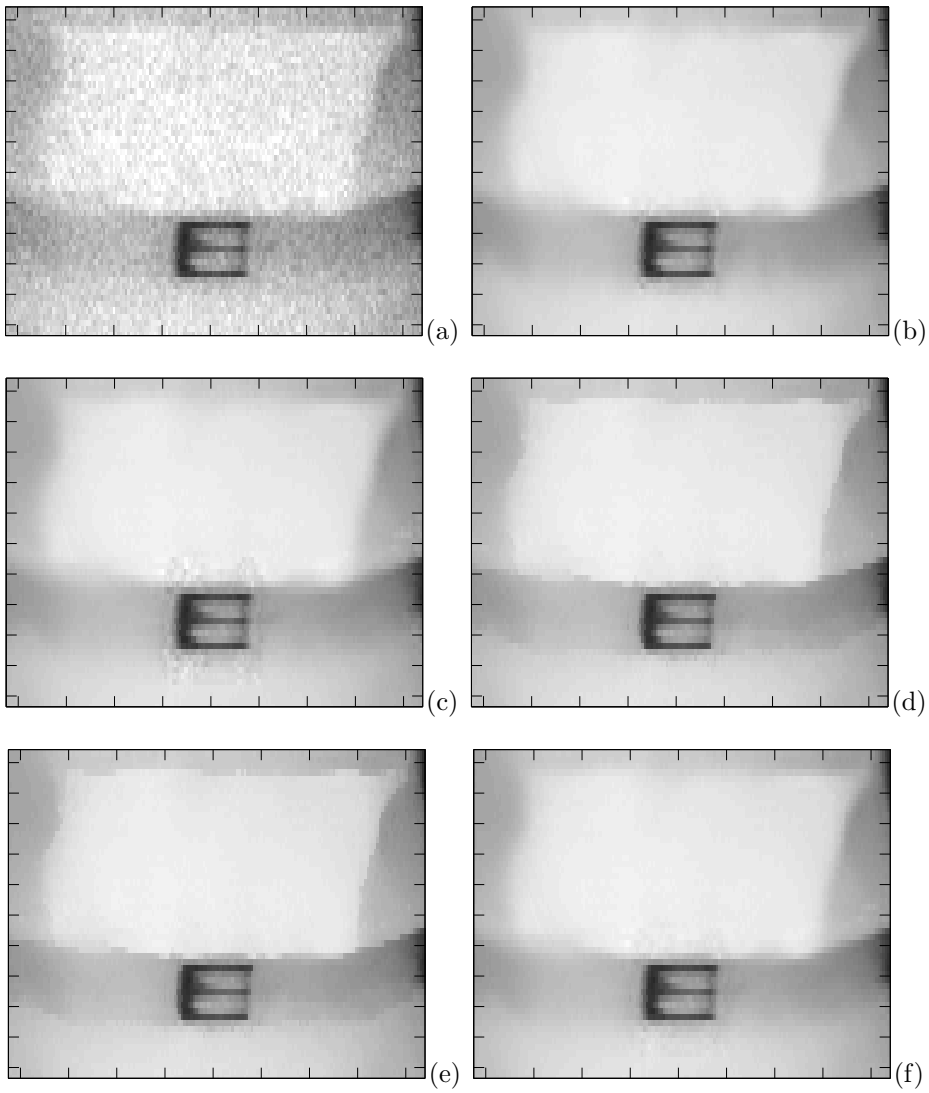

Figure 3. Results of local optimization on Subject 30 image. a) Noisy image, b) Standard NLM, c) LEO, 1×1, $\lambda_1 = 1e - 5$, d) LEO, 1×1, $\lambda_1 = 0.2$, e) LEO, 1×1, $\lambda_1 = 1.5$, f) Q-LEO, 1×1, $\lambda_1 = 1.5$. For the LEO result, package edges become sharper as regularization increases; the Q-LEO result is much smoother.

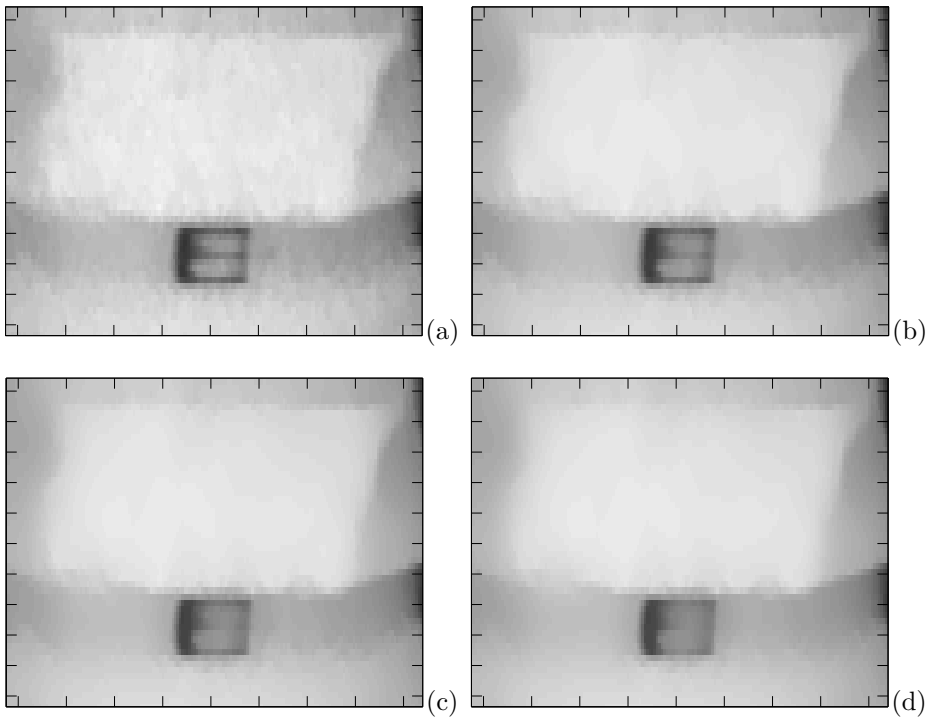

Figure 4. Total Variation (TV) denoising for Subject 30 image, for different TV regularization parameters: a) $\lambda_{TV} = 0.02$, b) $\lambda_{TV} = 0.05$, c) $\lambda_{TV} = 0.08$, d) $\lambda_{TV} = 0.1$. Note that for higher values, the dark belt-buckle feature is eroded.

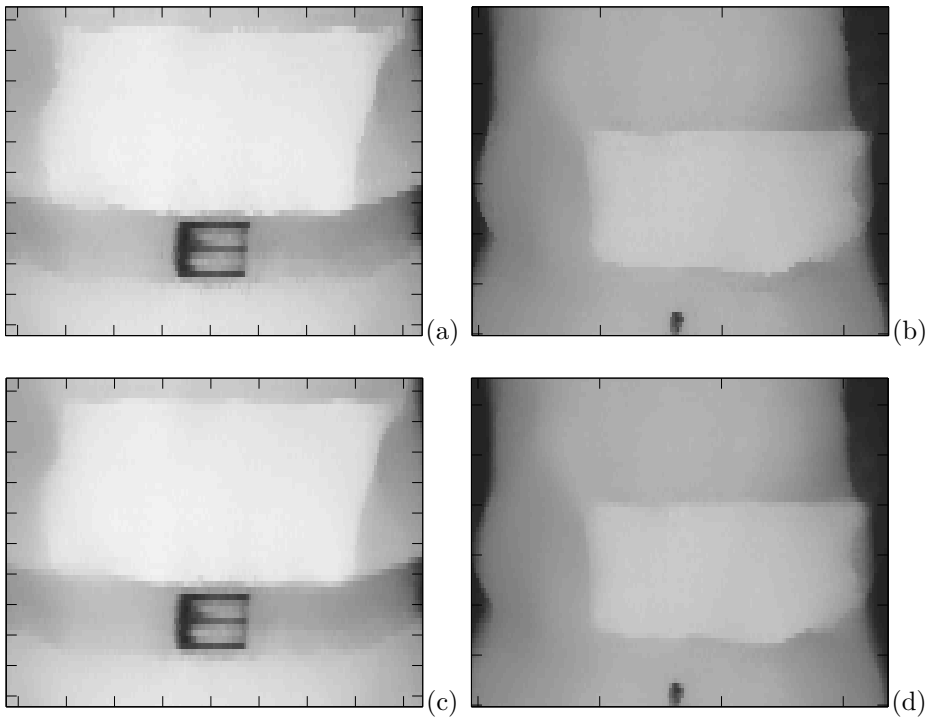

Figure 5. Difference between 1x1 and 3x3 local LEO solutions. a) 1x1, Subject 20, b) 1x1, Subject 46, c) 3x3, Subject 20, d) 3x3, Subject 46.

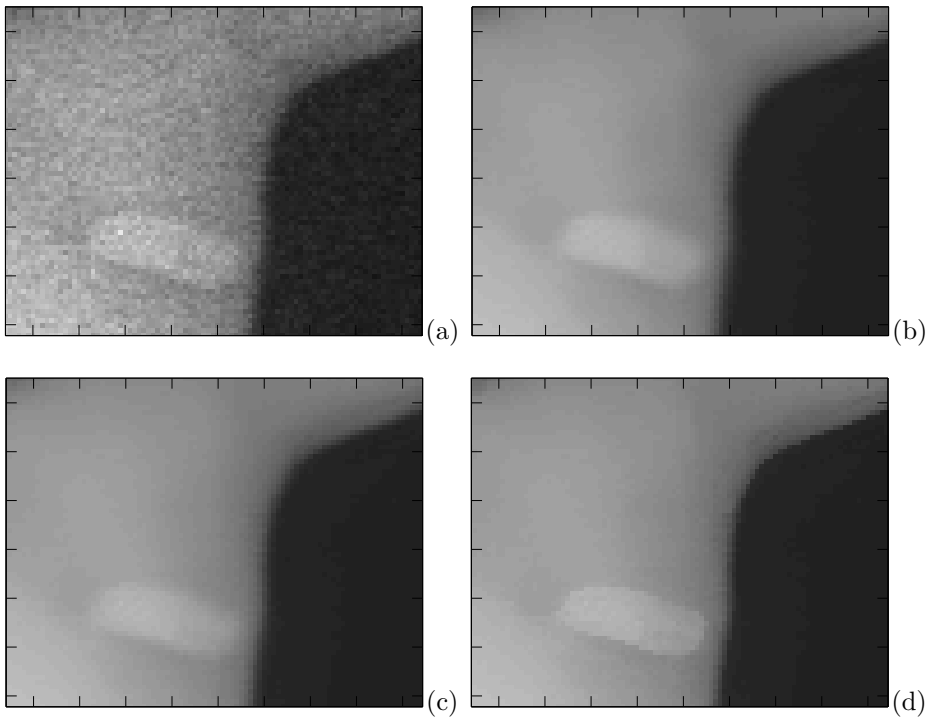

Figure 6. Results of local optimization XBS image, Subject 25. a) Noisy image, b) standard NLM result, c) EWA combination, d) LEO local combination. Note the improvement in edge contrast.

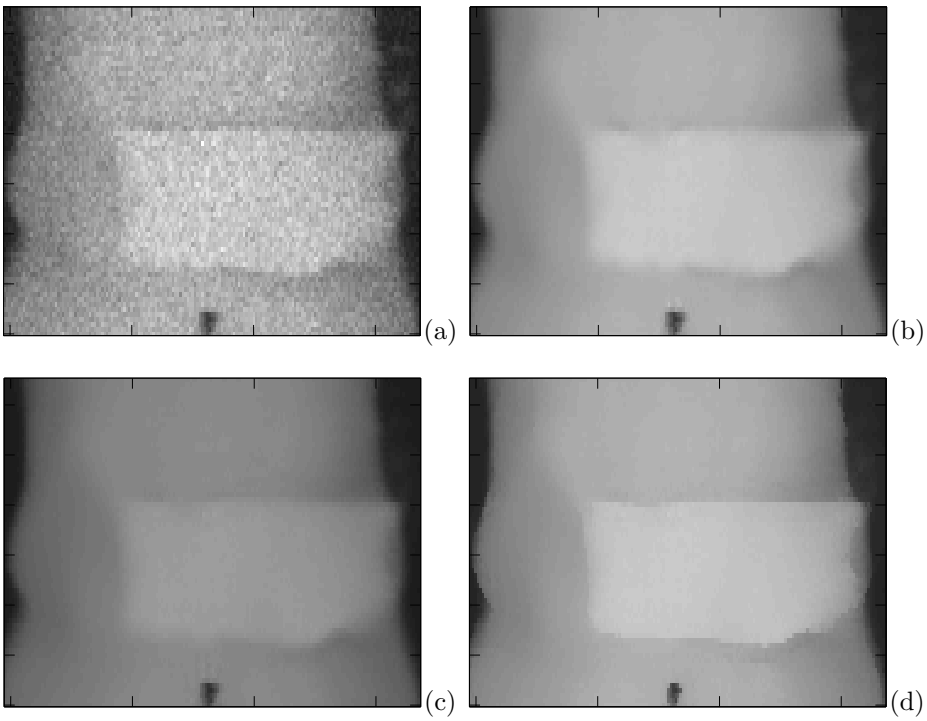

Figure 7. Results of local optimization XBS image, Subject 46. a) Noisy image, b) standard NLM result, c) EWA combination, d) LEO local combination. Note the improvement in edge contrast.

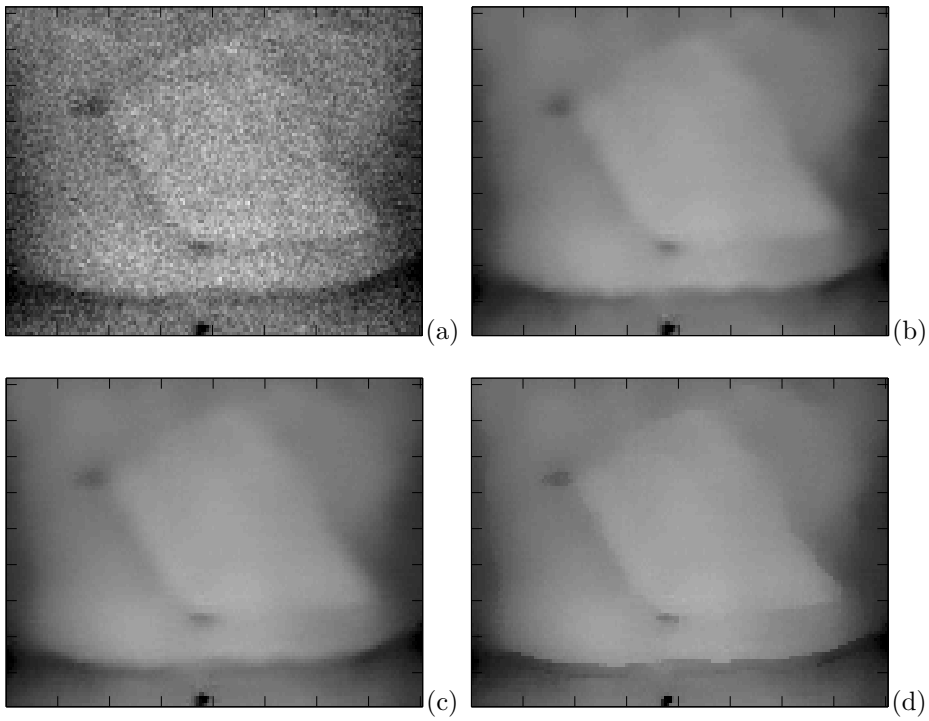

Figure 8. Results of local optimization XBS image, Subject 75. a) Noisy image, b) standard NLM result, c) EWA combination, d) LEO local combination. Note the improvement in edge contrast.

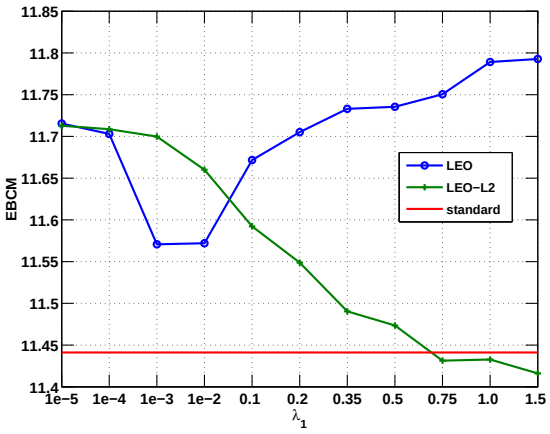

Figure 9. Results of varying λ_1 for Subject 25 image. As regularization parameter increase, edges become sharper as measured by the EBCM metric when gradient is penalized according to the L1 measure; when an L2 (quadratic) penalty is applied, image becomes smoother and EBCM decreases.

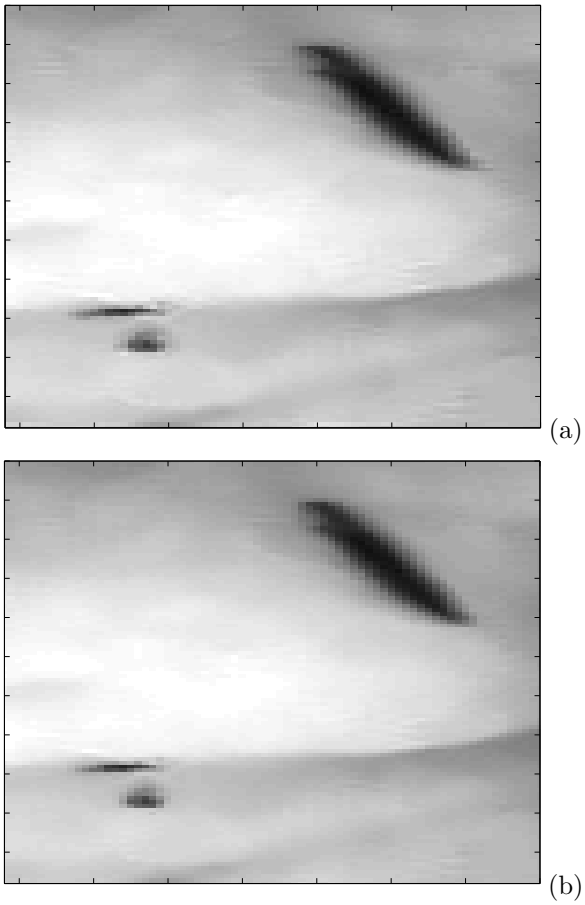

Figure 10. Results for non-square patch shapes for Subject 8 image. a) 'standard' NLM result but using circular patch shape, b) LEO local combination of 15 circular- and semi-circular patches. Note the smoother result near the belt buckle (dark object at lower left).

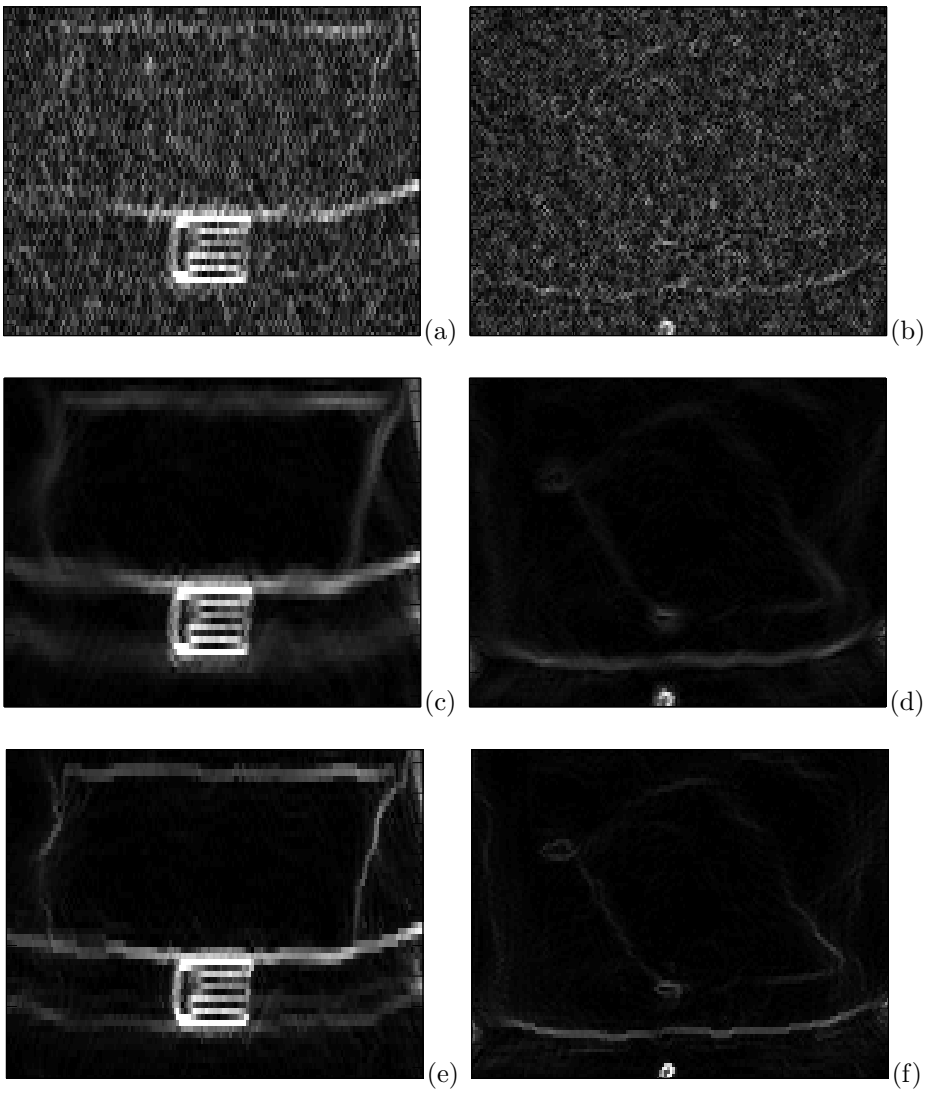

Figure 11. Image gradients for two XBS images. a) and b), gradient of raw images for subjects 30 and 75, respectively. c) and d), gradient of standard NLM outputs; e) and f), gradients of proposed LEO method.