The Bitonic Filter: Linear Filtering in an Edge-preserving Morphological Framework

Graham Treece

***Abstract*—A new filter is presented which has better edge and detail preserving properties than a median, noise reduction capability similar to a Gaussian, and is applicable to many signal and noise types. It is built on a definition of signal as *bitonic*, i.e. containing only one local maxima or minima within the filter range. This definition is based on data ranking rather than value, hence the bitonic filter comprises a combination of non-linear morphological and linear operators. It has no data-level-sensitive parameters and can locally adapt to the signal and noise levels in an image, precisely preserving both smooth and discontinuous signals of any level when there is no noise, but also reducing noise in other areas without creating additional artefactual noise. Both the basis and the performance of the filter are examined in detail, and it is shown to be a significant improvement on the Gaussian and median. It is also compared over various noisy images to the image-guided filter, anisotropic diffusion, non-local means, the grain filter, and self-dual forms of levelling and rank filters. In terms of signal-to-noise, the bitonic filter outperforms all these except non-local means, and sometimes anisotropic diffusion. However it gives good visual results in all circumstances, with characteristics which make it appropriate particularly for signals or images with varying noise, or features at varying levels. The bitonic has very few parameters, does not require optimisation nor prior knowledge of noise levels, does not have any problems with stability, and is reasonably fast to implement. Despite its non-linearity, it hence represents a very practical operation with general applicability.**

***Index Terms*—Gaussian filter, median filter, morphology, noise reduction, edge preservation**

I. Introduction

THE removal of unwanted noise corrupting a digital signal is a very common operation. In many contexts the signal is not known *a priori*, except perhaps for some general expectations concerning its overall form, and can contain both smooth regions and discontinuities, or 'edges'. In this latter case, noise removal usually also leads to blurring of these edges, and many algorithms have been proposed which seek either to preserve them during noise removal, or restore them after blurring.

This paper addresses the preservation of signal discontinuities, in a way which still allows noise removal at the edge, rather than simply lessening (or disabling) the noise reduction where an edge is detected. Ideally this would still be the case at fairly high noise levels, where the discontinuity is less than the level variation due to noise: this is in contrast to techniques which rely on thresholds to distinguish between an edge to be preserved and noise to be reduced.

The difficulty of removing noise without corrupting the signal relates to the overlap in their respective definitions. Noise can usually be defined as a random component, whilst an unknown signal can be regarded as something which is either repetitive or smooth or simply 'not random'. In the absence of specific training, however, it is usually presumed that the noise has a greater, or higher, frequency content and hence 'smoothing', or the removal of these frequencies, is used as a synonym for 'noise removal'. If the actual signal is smooth (contains low frequencies) or repetitive (is relatively sparse in frequency) then some noise can be removed without damage to the signal. However, signal edges are often not repetitive, and contain very high frequency content, hence removing the noise leads to blurring, or other forms of corruption, of the edge. Edge-preservation during noise removal hence implies the removal of high frequency noise but not of high-frequency signal: however, at least in the frequency domain, these components are not distinct.

Here an alternative is considered, in which the signal is regarded as 'anything locally bitonic within a given range'. A bitonic sequence (defined in the context of sorting [1] as an extension of monotonic) is one which increases monotonically (or not at all) to a peak then decreases monotonically (or not at all), i.e. it has at most one local maxima. A signal which when cyclically shifted meets this definition is also bitonic. A slightly simpler definition is used here such that a signal is deemed locally bitonic if it has either only one local maxima, or only one local minima, or no maxima nor minima[1]. This concept of local bitonicity equally encompasses smooth signals and those containing edges, since only the data rank matters, not the level. Many real signals hence might be expected to exhibit local bitonicity over a reasonable range.

By this definition, noise is anything which is *not* bitonic over the given range (or equivalently anything which is only bitonic over a shorter range). Whilst the range does effectively impose a lower frequency limit on the noise, and an upper frequency limit on repetitive signals, it crucially does not impose an upper frequency limit on edges, so long as the overall shape is locally bitonic. The definition encompasses at least all noise types which are zero mean and uncorrelated over neighbouring samples (i.e. white noise), save for the lower frequency limit. Impulsive, or 'Salt & Pepper' noise is more ambiguous: an isolated impulse would be regarded as a signal; several impulses within the range as noise. The ability to preserve impulses whilst removing other types of noise is a considerable advantage in some scenarios; but in any case it will be demonstrated that it is possible to relax the bitonicity

G. Treece is with the Department of Engineering, University of Cambridge, Cambridge, UK, email: gmt11@eng.cam.ac.uk

Manuscript received ???; revised ???.

[1] This is a reasonable re-definition since a true cyclically bitonic signal will always be bitonic by this alternative definition over at least half of its length: hence the difference is purely one of the range of the bitonicity.

criteria slightly to allow for good rejection of impulsive noise as well.

In the context of filtering, interpreted here in the sense of replacing each value in the signal with some combination of the surrounding values within a specific range, a bitonic filter would hence be one which seeks to preserve any signal with bitonicity over the range of the filter, but reject anything else. Such a filter would naturally preserve both edges and smooth features in a signal.

II. The bitonic filter

A. *Definition*

Since bitonicity is concerned with the ordering, rather than the value, of the data, it is natural to turn to rank filters [2], also known as order-statistic filters. The median is the most well known example, where the data in a local window is ranked and the mid (50th centile) value is output. The median filter is commonly used to eliminate impulsive noise, whilst preserving edges well, at least if there is no more than one edge within the range of the filter. A rank filter is a generalisation of the median where any centile can form the output. Such filters can be considered as monotonic in that they preserve signals which are monotonically increasing or decreasing, and indeed this leads naturally to impulsive noise reduction, since impulses are bitonic rather than monotonic. For two-dimensional (2D) data, the shape of the window used to form the set of ranked data, in morphology known as the 'structuring element', has some impact on which features can be preserved. Here we use a circular disk for 2D image data to ensure isotropic behaviour.

Using a rank filter of 100th centile (or maximum, known as a dilation) and immediately following this with another of 0th centile (minimum, known as an erosion) results in a morphological closing operation, which preserves signals with a local maximum, whilst rejecting any signal with a local minimum. Reversing the order of these filters results in a morphological opening which has the opposite action. Such filters have many uses in processing the *shape* of data, particularly in granulometry [3]. Figure 1 shows some examples of opening and closing operations on one-dimensional (1D) signals. Here a robust opening operation is used, with a small centile c rather than the minimum, and $(100 - c)$ in place of the maximum. This is not the same as either the slightly better known rank-max opening [4] nor the soft-opening [5], but it has been previously suggested by Kass and Solomon [6]. The use of a small centile allows some control over impulsive noise rejection, since any impulse which takes up less of the filter range than c will be rejected. The robust opening $O_{w,c}(x)$ and closing $C_{w,c}(x)$ of a signal x can be defined as:

$$\begin{aligned} r_{w,c}(x) &= c^{\text{th}}\underset{i\in w}{\text{centile}}\left\{x_i\right\} && (1)\\ O_{w,c}(x) &= r_{w,100-c}(r_{w,c}(x)) && (2)\\ C_{w,c}(x) &= r_{w,c}(r_{w,100-c}(x)) && (3)\end{aligned}$$

where $r_{w,c}(x)$ is a rank filter, w is the filter window (or structuring element in 2D), $|w|$ is the window length (or number of elements in 2D) and c the chosen centile, which will generally be a fairly low percentage. The bitonicity can be seen in Fig. 1 (centre-left). However, robust opening and closing operations only do half of what is required, since they are not self-dual (symmetric in data value): they only preserve local minima or maxima respectively. In addition, it can be seen from Fig. 1 that the opening and closing operations do not preserve mean signal values in the case of a noisy signal, which would clearly be a vital property of a practical filter.

Fortunately both these drawbacks can be overcome by the same means. It is fairly clear, by comparison of the original signal with each of the opened and closed signals, which is the most appropriate output for each part of the signal. We can hence use such a comparison to weight a combination of the opening and closing operations. However, a weighting based on a point-wise comparison would simply return the original signal[2], so instead the differences between the original signal and each of the rank-filtered signals are smoothed with a linear filter. A Gaussian filter (i.e. a linear moving-window filter with Gaussian weights) is used for this purpose, since it is known to have good noise reduction properties. The filter length is determined experimentally to match the noise reduction from the rank filters, so that the standard deviation $\sigma = 0.33l$ where l is the window length in 1D, or the diameter of the structuring element in 2D. This smoothed error can be seen in the middle column of Fig. 1.

Defining the Gaussian linear filter as $G_\sigma(x)$, this is used to weight the results of the opening and closing operations as follows:

$$\begin{aligned} \epsilon_O(x) &= \left|G_\sigma\left(x - O_{w,c}(x)\right)\right| && (4)\\ \epsilon_C(x) &= \left|G_\sigma\left(C_{w,c}(x) - x\right)\right| && (5)\\ b_{w,c}(x) &= \frac{\epsilon_O(x)C_{w,c}(x) + \epsilon_C(x)O_{w,c}(x)}{\epsilon_O(x) + \epsilon_C(x)} && (6)\end{aligned}$$

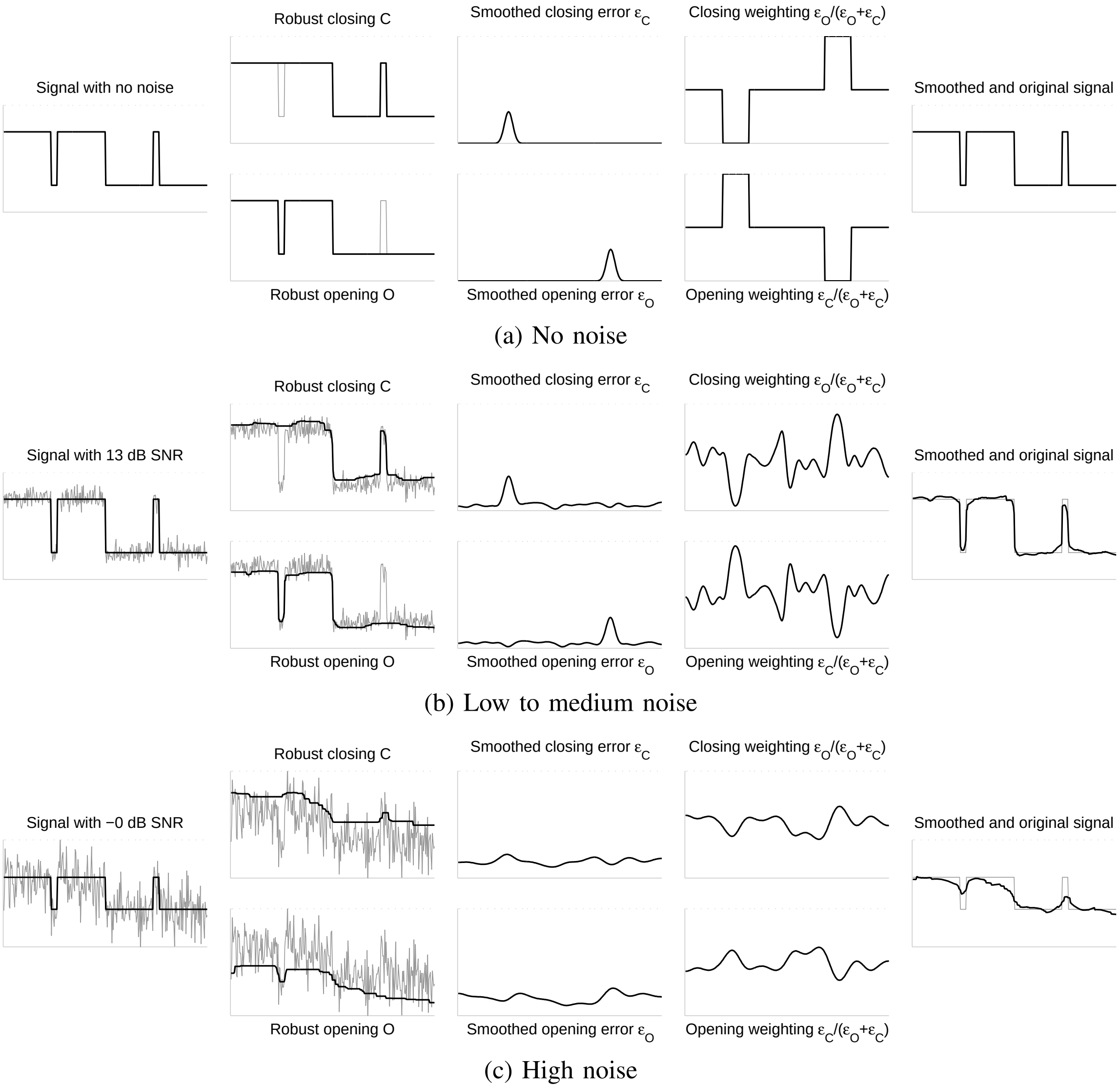
where $\epsilon_O(x)$ and $\epsilon_C(x)$ are smoothed opening and closing errors, and $b_{w,c}(x)$ is the output of the bitonic filter[3]: the errors, weights and filter output are shown in the middle and right-hand columns of Fig. 1. This seems to have the required properties, i.e. the preservation of any bitonic signal, and reduction of noise in all regions, including across edges: the opening and closing operations effectively remove bitonic signals, leaving the Gaussian to reduce any residual noise signal everywhere to the same extent.

B. *Analysis*

In fact, the noise reduction capabilities of the bitonic are very similar to the Gaussian. In high noise environments, the opening and closing operations will result in relatively constant signals (see Fig. 1(c)), such that $O_{w,c}(x) \approx k_O$ and $C_{w,c}(x) \approx$

[2] This can be seen by replacing $G_\sigma(x)$ with x in eq. (4) and following

[3] In the particular case that $\epsilon_O(x)$ and $\epsilon_C(x)$ are both zero at x, they are replaced by the arbitrary value 0.5: this would normally imply that both opening and closing returned the exact original signal and hence how they are weighted is inconsequential.



(a) No noise

(b) Low to medium noise

(c) High noise

Fig. 1. Stages of the bitonic filter. In each case the original signal (left) is both opened and closed (centre-left). The difference between the original and each of these signals is filtered using a Gaussian (centre) and the smoothed error applied as a weighting (centre-right) to the opened and closed signals to generate the final signal (right). The mathematical symbols relate to the equations in the main text. The morphological filters act to detect bitonicity in the data so that the Gaussian filter only eliminates the residual noise. This preserves edges exactly in no noise (a) and very well in low to medium noise (b). Even with noise as high as the signal level, some edge preservation is possible (c), whilst for very high noise the output tends to that of a Gaussian filter.

k_C, where k_O will generally be lower than the mean value, and k_C higher, in which case:

$$\begin{aligned}
\epsilon_O(x) &= G_\sigma(x) - k_O \\
\epsilon_C(x) &= k_C - G_\sigma(x) \\
b_{w,c}(x) &= \frac{(G_\sigma(x) - k_O)\, k_C + (k_C - G_\sigma(x))\, k_O}{k_C - k_O} \\
&= G_\sigma(x) \qquad (7)
\end{aligned}$$

i.e. in very high noise the bitonic filter reduces to a simple Gaussian smoothing of the signal. However, if there is any bitonic structure in the signal, this will be picked out and given appropriate weight according to how well it matches the original signal.

In fact eq. (7) demonstrates that, with increasing noise, the bitonic filter will tend towards the performance of whatever linear filter is used in its definition, eqs. (4) and (5), and hence this could be seen as a framework for combining any linear filter with rank filters in order to preserve signal edges.

The size of window $|w|$ controls the amount of smoothing of the filter in exactly the same way as would be expected for any moving-window linear filter. In contrast, the centile c determines the sensitivity to impulsive noise. $c = 0\%$ treats any isolated impulses as signal, $c \approx 20\%$ will reject impulses similarly to a median filter (but with significantly less non-linear distortion than the median), and $c = 50\%$ is the upper limit, since in this case $C_{w,c} \equiv O_{w,c}$. In nearly all scenarios, $c = 10\%$ provides a very good balance, and is used in all subsequent results except for the no-noise environment in Fig. 2. No setting of w nor c can cause instability, and they create useful output features across the entire range. Both Gaussian and rank filters are constrained to the convex hull of the input data, and hence the bitonic filter will never generate values outside of the input data range.

It should be noted that, whilst in practise (and as demonstrated by the results, particularly see Fig. 7) the bitonic filter certainly acts to preserve bitonic signals and remove anything else, it is not shown here that bitonicity over the filter length l is actually guaranteed by this filter. This could only be true if the centile c is set to zero, in which case the opening and closing operations are likely to be independently bitonic, though the weighted combination is more complex and may not be. In any case, better visual results are obtained by setting $c = 10\%$ and in this case whilst the filter may still generate a bitonic signal, it will certainly remove some components (i.e. isolated impulses) which are themselves bitonic. However, local bitonicity can still be advanced as the best summary of what the filter *seeks* to preserve in a signal.

C. Implementation

The bitonic filter can also be implemented reasonably efficiently. Since the first rank filters of each of eqs. (2) and (3) both act on the original signal, the ranking only needs to be performed once, following which both centiles can be selected. In addition, the Gaussian filters are generally faster than a rank filter and have little impact on the processing time. Hence the bitonic filter takes roughly 3 to 4 times as long as a single rank filter of the same window size. The efficient implementation of rank filters has been well studied, with histogram-based $\mathcal{O}(l)$ algorithms, as used here for image data, in common use. A constant-time implementation has also been proposed for the median filter [7] which is equally applicable to rank filters using other centiles.

III. Edge-preserving filters

Whilst, to the author's knowledge, there are no other filters designed for the specific purpose of preserving bitonic signals, there are various morphological filters which have some similarity.

Most of the variants of median filters, for instance adaptive median filters [8] focus on better preservation of signals in the context of reducing impulsive noise, often by some form of signal or impulse detection. The filter can then either be modified in extent, or some data weighted by repeating values in the ranked list: the rank filter equivalent of multiplying by a larger weight in linear filtering. Examples include the constant-time weighted median [9], rank-conditioned rank selection filtering [10], or permutation weighting. A comprehensive discussion of such filters has been presented in [11]. These filters vary in rank selection, but will return one of the input values: hence they are limited in their power to reduce non-impulsive noise, though more complex combinations of morphological operations such as discussed by [12] have been used to de-noise specific signals [13]. In particular, the OCCO filter [26] takes the average of an opening-then-closing and a closing-then-opening, which has the advantage of being self-dual (symmetric in data value).

Connected operators, like rank filters, are also sensitive to data ordering rather than value, however they divide an image into non-overlapping sets, and then operate on these sets: a very helpful recent review is in [14]. In this context *area* opening and closing remove features based on their area rather than their shape [18]. The sets can be defined inclusively, where the area of a component is considered to be the total of all included sets, and there exist very efficient algorithms for extracting such definitions, for instance the tree of shapes [17]. Self-dual filtering can be achieved by removing any sets which have areas below a threshold, either by 'pruning' the connected tree of sets and reconstructing the image, or by directly manipulating the original image as in the grain filter [15], similar to level sets [16].

Combinations of linear and morphological filters are somewhat less common. Possibly the earliest is the alpha-trimmed mean [21], which uses a rank filter to remove outliers before taking the mean (effectively a simple linear filter) of the remainder. Others simply represent linear combinations of the output of a rank filter and a linear filter, for instance the mean-median filter [22], or more general (and very similar) morphological/rank/linear (MRL) filter [23] and hybrid order statistic filter [24]. Such filters may require training to define the many possible rank and linear weights [25]. They are also limited by the use of a single rank filter which is fundamentally monotonic: hence edge-preservation is only possible for a single edge within the range of the rank filter, and also by limiting any subsequent smoothing from the associated linear filter.

An alternative combination of linear and rank filters is given by [6]. In order to improve morphological operations which can be implemented using histograms, they propose local smoothing (by a linear filter) of the histogram of the data values. Whilst this is rather different from the bitonic filter, setting $c = 20\%$ or higher can offer a similar improvement to the median as does histogram-smoothing on low-noise data.

It is also possible to combine self-dual levelling (effectively creating larger, more single-valued regions in the image) with linear filters [14], [19]. This can be implemented by reconstruction, i.e. iteratively dilating and eroding according to an independent mask: in this context the levelling re-introduces edge information from the image into the mask. Hence if the mask is a Gaussian filtered image as in [27] then the whole process can be seen as edge-enhanced de-noising.

There are numerous options for noise suppression without blurring edges which make no use of morphological filters, a theoretical comparison of many of which can be found in [28]. Three popular examples serve to provide a comparison with the bitonic: the image-guided filter [29], anisotropic diffusion [30], [31] and non-local means [28]. All of these can preserve edges, but do so by different mechanisms. The image-guided filter weights data according to difference in value as well as location: this is similar to the better known bilateral filter [32], but with improved performance at image edges. Anisotropic diffusion uses the local curvature to discourage averaging across steeper edges. Non-local means only averages data with that from similar surrounding distributions. Each of these has various derivatives but the fundamental form is used here, since many of the alternatives, for instance local-adaptation or iteration, could equally be applied to the bitonic filter.

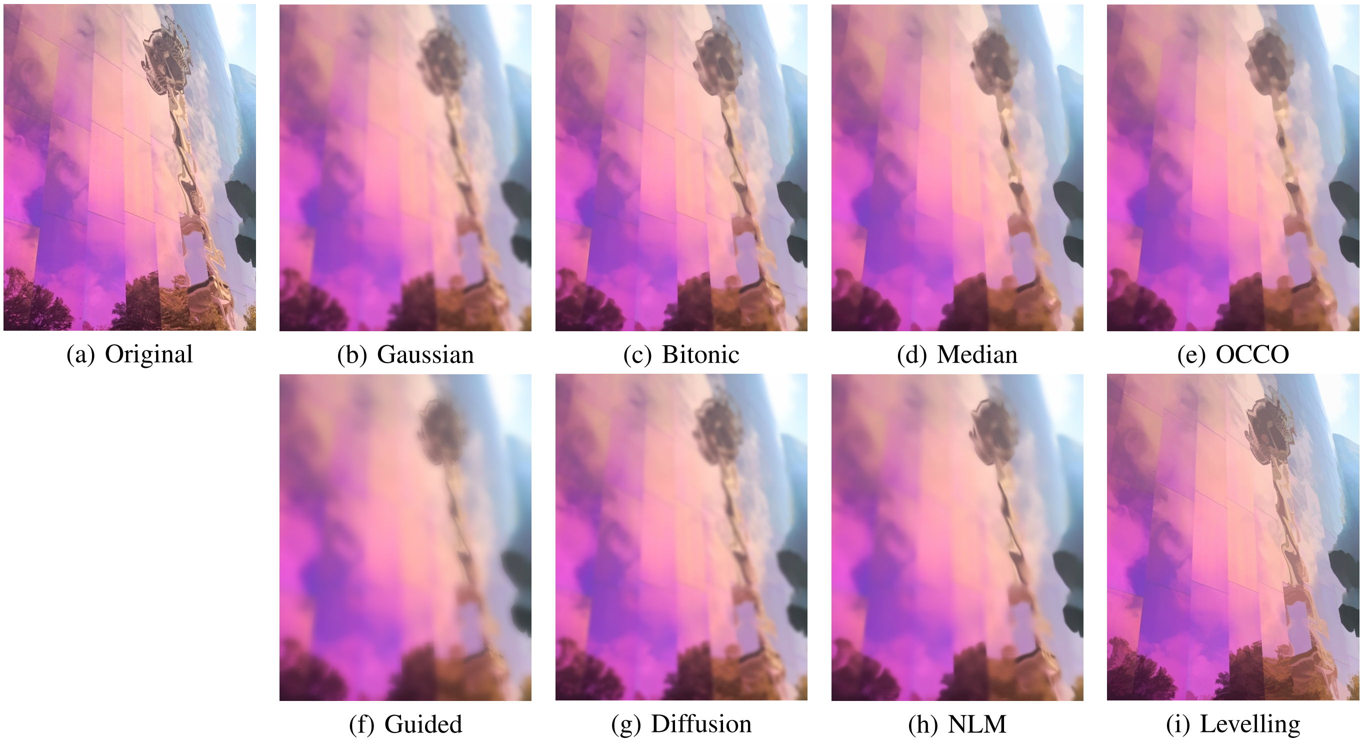

(a) Original (b) Gaussian (c) Bitonic (d) Median (e) OCCO
(f) Guided (g) Diffusion (h) NLM (i) Levelling

Fig. 2. Filtering results on ‘seattle’ image with no added noise. All outputs were generated with the same filter length l, and hence similar smoothing in areas regarded as noise at this scale, for instance the reflected trees at the base, and the top of the Seattle needle. The Gaussian (b) has uniform smoothing but no sharp edges. Techniques sensitive to data value, (f) image-guided filtering, (g) anisotropic diffusion and (h) non-local means, retain edges above a certain threshold but either remove or blur them elsewhere, for instance the panels at the top of the building. The median (d) has sharper edges but removes or distorts high level details, as does the OCCO filter (e), whilst Gaussian levelling (i) preserves all edges. The bitonic filter (c) smooths similarly to the Gaussian but with edges of all magnitudes preserved and much better handling of isolated detail than the median, though not as well as levelling.

IV. Results

Details of the tested filters are as follows, where the bitonic filter length l (window length in 1D or diameter of the structuring element in 2D) is chosen in each case to represent the parameter which most controls the extent of the filter:

Gaussian Linear filter with Gaussian weights, with the standard deviation σ set to $0.33l$, and sufficient filter length to cover up to $\pm 2\sigma$.

Median Median filter, in 1D with window length l, in 2D using an isotropic circular structuring element with diameter l.

Bitonic Bitonic filter as previously described, with length l, and c fixed at 10%, except for the no-noise case in Fig. 2 where it is set to 2%.

Guided Image-guided filter, implemented using the MATLAB[4] function imguidedfilter[5] [29], with the local neighbourhood size set to l, and the degree of smoothing set to four times the added noise variance in the image.

Diffusion Anisotropic diffusion, implemented for MATLAB[6] [30], with number of iterations set to l, the integration constant set to the standard deviation of the added noise, the gradient threshold set to twice the standard deviation of the added noise, and the wide-region conduction coefficient.

NLM Non-local means filter, implemented using a fast algorithm for MATLAB[7] [28], with the window and search length both set to l, and the filter parameter h set to the standard deviation of the added noise.

OCCO The average of a rank-based opening-closing, with a closing-opening [26], except that the centile-based operations $C_{w,c}, O_{w,c}$ were used with c fixed at 10% to improve performance. l defined the diameter of the structuring element as for the Bitonic filter.

Grain A self-dual grain filter based on area openings and closings, implemented similarly to [15], [16], with an inclusive definition of areas [33]. In 2D, the minimum area was given by $l \times l$.

Levelling A self-dual levelling based on reconstruction using a Gaussian mask [27], [19]. l controlled the standard deviation of the Gaussian, exactly as with the Gaussian filter above.

Where relevant, the data was symmetrically extended at the image edges, though very similar results are achieved by simply extending the values at each edge. Guided, Diffusion and NLM have additional parameters, and these were set pre-

[4] MATLAB R2014a, The MathWorks Inc., Natick, MA, 2000

[5] http://uk.mathworks.com/help/images/ref/imguidedfilter.html

[6] MATLAB file exchange: Anisotropic Diffusion (Perona & Malik) by Daniel Lopes,14 May 2007

[7] MATLAB file exchange: Fast Non-Local Means 1D, 2D Color and 3D by Dirk-Jan Kroon, 28 Apr 2010

suming knowledge of the added noise level, with l controlling overall extent, and the other parameters chosen for optimal signal to noise ratio (SNR) (and, in the case of Diffusion, reasonable stability) at the given noise level.

Whilst the grain filter is included in the summary data, images for this are not included, since both the performance and overall image features were very similar to the Gaussian-masked levelling.

A. *Noise-free image data*

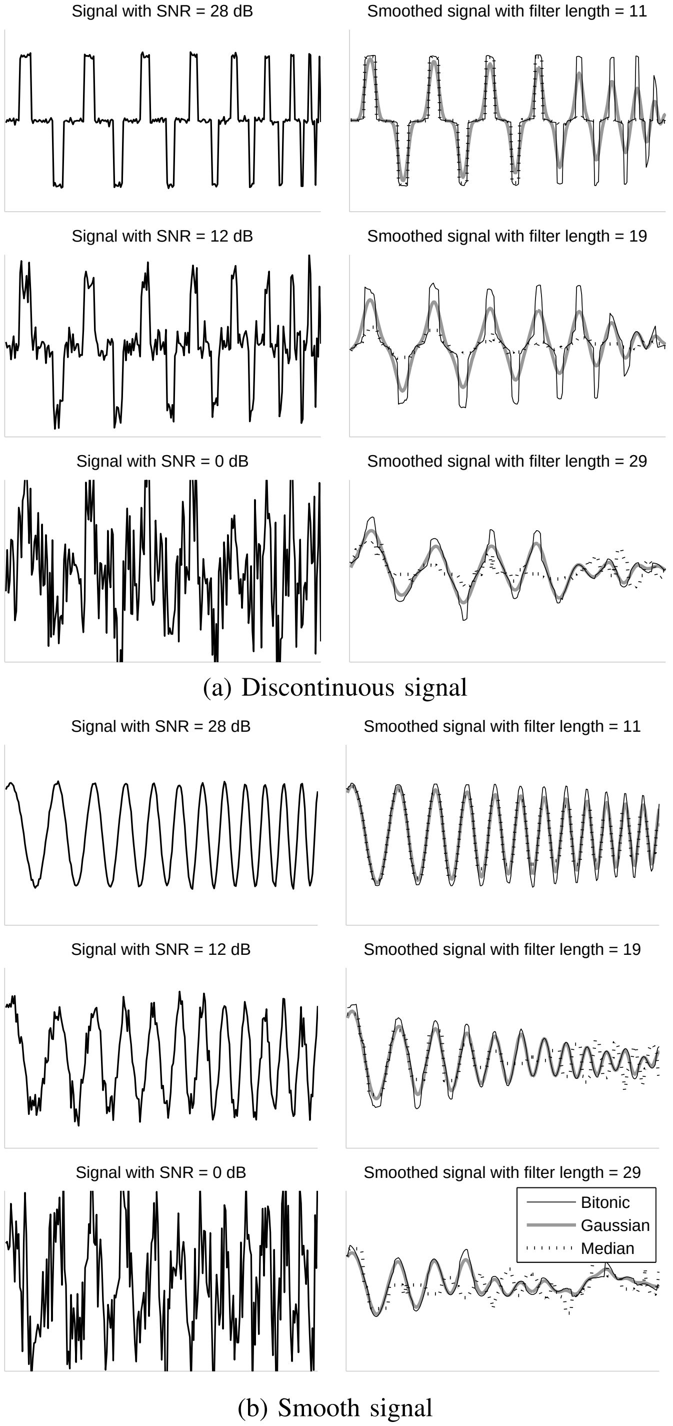
Figure 2 shows the different smoothing properties of all these filters, applied to an image with no added noise. In this case the edges have a variety of magnitudes and the non-rank filters are not able to smooth highly detailed regions of the image without also smoothing low-magnitude edges. For the image-guided filter, there is a specific data threshold below which data is more smoothed. With anisotropic diffusion, smoothing increases with the conduction coefficient, which has to be set above the noise level to guarantee stability. Smoothing in non-local means is controlled by the filter parameter h, which is usually set somewhat higher than the expected noise level in the image. As expected, the bitonic preserves more details than the median and OCCO, but smooths the residual data in a similar fashion to the Gaussian. Levelling preserves edges perfectly but is less effective at smoothing.

These results demonstrate the potential applications of the bitonic filter in the extraction of image edges or background layers, highlighting detail, or other artistic operations. However, the bitonic filter was motivated in terms of noise reduction, which is the focus for the remainder of this section.

B. *One-dimensional signals*

Figure 3 shows the performance of the Gaussian, median and bitonic filters on two test signals, one smooth and one with distinct edges. The median filter performs poorly on these signals, since they are only monotonic over a small scale: as soon as the filter length increases beyond this, the signal is in some cases completely removed, see the middle row of Fig. 3(a), and in other cases actually inverted, see the middle row of Fig. 3(b). The bitonic filter is much better at preserving the signal at all scales, whilst still reducing noise. This is obvious on the signal with distinct edges in Fig. 3(a) but it is still true to a lesser extent on the smooth signal of Fig. 3(b). For very low SNR (bottom rows of Fig. 3(a) and (b)) the bitonic filter starts to follow the Gaussian, though still with edge-preservation in some regions.

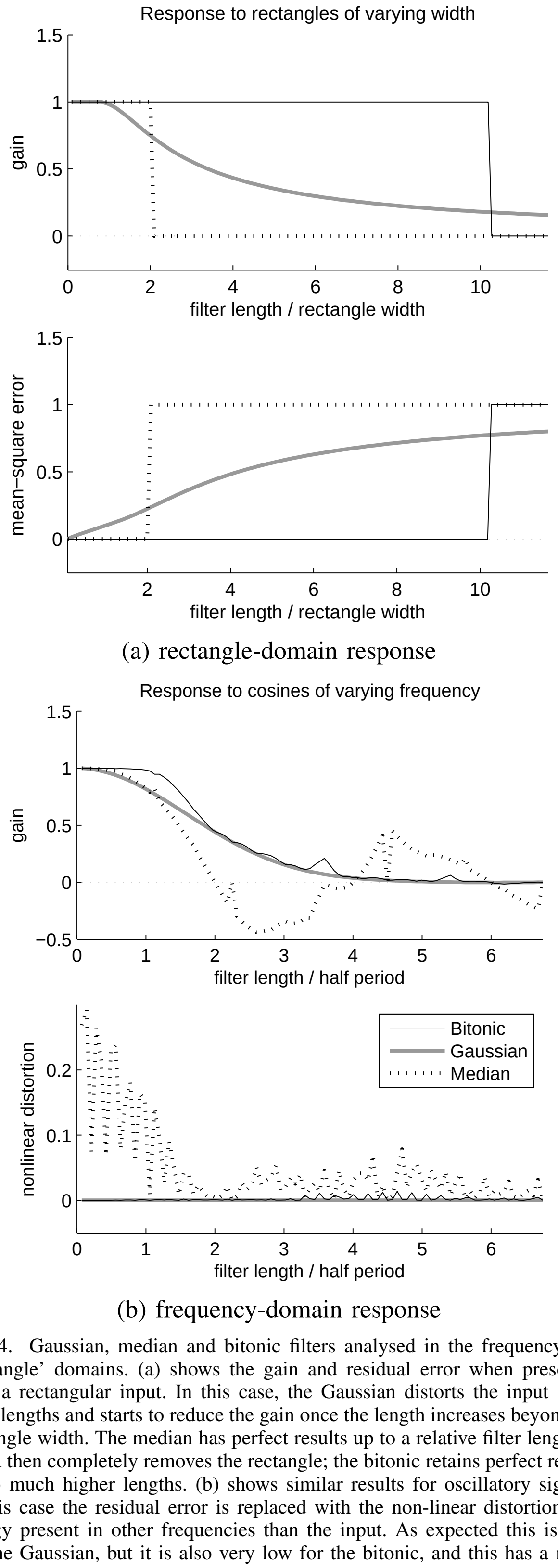
The performance in Fig. 3 can be explained by analysing the filters in terms of frequency and ‘rectangle’ response. Whilst frequency-domain analysis is natural for linear filters, which can be completely described by their transfer functions, it is not a complete description of non-linear filters. However, a similar analysis is possible by splitting the response of the filter to a sinusoidal input into two parts: the gain at the input frequency, and the non-linear distortion, i.e. the energy present at all other frequencies. Figure 4(b) shows the results of this analysis. As expected, the Gaussian filter also has a gain which is Gaussian over frequency, and zero distortion.



(a) Discontinuous signal

(b) Smooth signal

Fig. 3. The response of Gaussian, median and bitonic filters to signals with varying time scales. As expected, the bitonic filter has very good performance on the discontinuous signal in (a), with the Gaussian smoothing all edges, and the median completely removing many. The performance in (b) is perhaps more surprising, with the bitonic still outperforming the Gaussian even on this smooth signal, and the median actually reversing the signal in some cases.



(a) rectangle-domain response

(b) frequency-domain response

Fig. 4. Gaussian, median and bitonic filters analysed in the frequency and 'rectangle' domains. (a) shows the gain and residual error when presented with a rectangular input. In this case, the Gaussian distorts the input at all filter lengths and starts to reduce the gain once the length increases beyond the rectangle width. The median has perfect results up to a relative filter length of 2 and then completely removes the rectangle; the bitonic retains perfect results up to much higher lengths. (b) shows similar results for oscillatory signals. In this case the residual error is replaced with the non-linear distortion: the energy present in other frequencies than the input. As expected this is zero for the Gaussian, but it is also very low for the bitonic, and this has a much narrower gain transition band. The median introduces far more distortion, and the gain is negative at many frequencies.

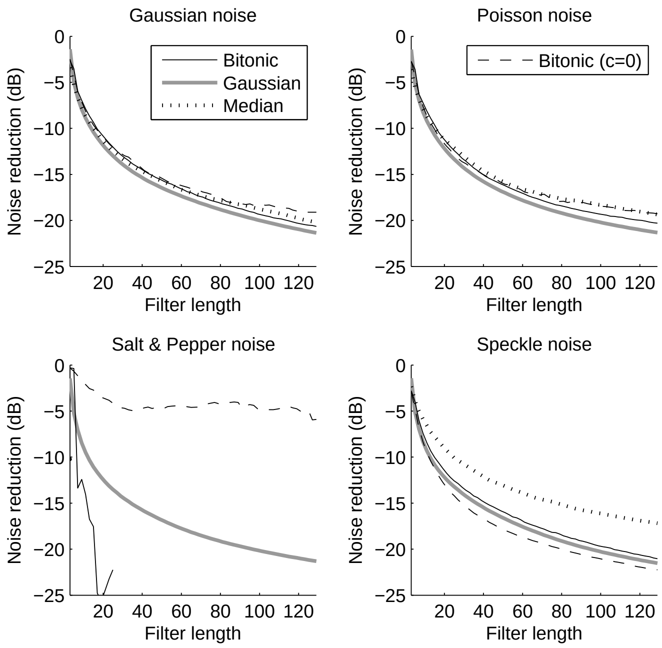


Fig. 5. Response of filters to various types of noise. The graphs show the noise reduction (residual compared to original noise level) for the average of 20 instantiations of each noise type. The bitonic filter closely follows the Gaussian in all cases. As expected, the median has very good response to 'Salt & Pepper' (impulsive) noise, with little shown since the noise is mostly completely eliminated, but is not quite as good for speckle noise. If the bitonic centile c is set to a lower value (e.g. $c = 0$, dashed line), the filter passes most impulsive signals while still rejecting other noise types: useful for retention of extremely small details in some noise environments.

However, the bitonic filter has a much better passband, with a gain of exactly unity until the filter length l is equal to half of the period. It also has a narrower transition band than the Gaussian, and introduces surprisingly little distortion, especially so for frequencies within the passband. This analysis also explains the poor performance of the median: the gain is in fact negative for some frequencies, and there is considerable distortion, particularly for low frequencies.

An equivalent analysis for signal transitions which are not repetitive is to consider the response to a rectangle of varying width, as demonstrated in Fig. 4(a). Here the equivalent to 'gain' is the change in height of the rectangular input, and the 'distortion' is replaced by the mean-square error of the output compared to the input. In this domain, the Gaussian has a much poorer response, with a unity gain only while the filter is shorter than the rectangle width, and a relatively large mean-square error in all cases. The median output is perfect until the filter length is twice the width of the rectangle, and after that the signal is completely removed. The bitonic has perfect preservation of the rectangle until the width is less than the chosen centile c in the filter — in this case set to 10%.

Whilst Fig.4 demonstrates that the bitonic preserves signals well, to be useful it also needs to reject noise well, and this is demonstrated in Fig. 5 for various noise types. The bitonic filter has very nearly the same noise rejection as the Gaussian. However, the response to impulsive noise can be modified by the centile c. Setting this to zero (as shown) preserves most impulses with little effect on other noise types. Setting it to

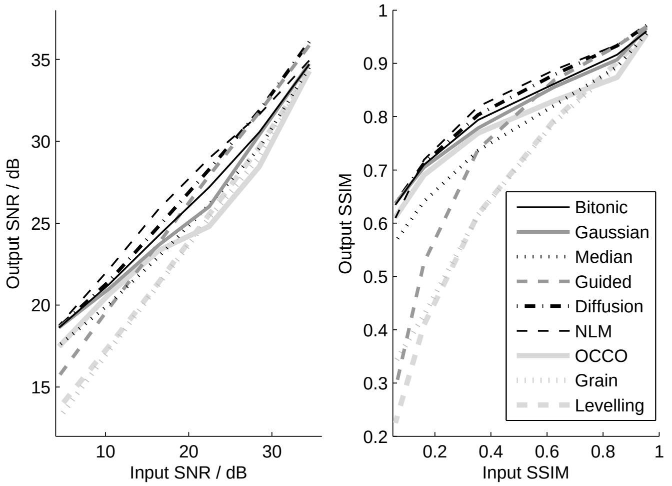


Fig. 6. Average results for the five images boat, peppers, house, fruits and tulips. Filter lengths were chosen to give the best result (independently for SNR and SSIM) on each image for each of six different levels of initial added Gaussian noise. The bitonic filter performs better than the Gaussian, median and other morphological filters, with clear improvements for moderate SNR levels, tending towards the Gaussian result for very low SNR. Image-guided performs well at high SNR but very poorly at low levels. Diffusion and NLM both perform well on this test, save at very low SSIM.

a larger value (e.g. 20%) gives a response to impulsive noise similar to the median.

C. Noisy image data

Various test images were used to demonstrate different aspects of filter performance, with Gaussian noise added to each. The ‘boat’, ‘peppers’, ‘house’, ‘fruits’ and ‘tulips’ images are each regions of public-domain test images[8], whereas ‘seattle’ is by the author (of the Seattle needle[9] reflected in the neighbouring EMP museum[10]) and ‘varying’ is computer-generated. The achievable SNR and Structural Similarity (SSIM, an attempt to measure something closer to visually perceived quality) [34] are given in Table I, along with the filter length l which gave the best overall SNR and SSIM, and an indication of processing time.

Figure 7 is an image containing noise which varies from left to right, and demonstrates the ability of the bitonic filter to automatically adjust to noise levels without the need to locally vary any parameters. In such scenarios it is hard to find an optimal set of parameters for the guided, diffusion and NLM filters, which simultaneously allow edge-preservation in the low noise regions whilst reducing noise (and maintaining stability) in the high noise regions. Inevitably either edge-preservation or noise rejection has to be sacrificed to some extent in each region.

Figures 8 and 9 are two different regions from the ‘boat’ image. The former has many strong features with short length scales, and NLM works well on such data, though it also starts to find some false structure within the noise, particularly evident in regions with little signal. The performance of the bitonic is better then diffusion, and considerably better than guided, since in the latter it is very difficult to set a sensible threshold for capturing edges but not noise. In this and all other examples, the bitonic seems to consistently achieve sharper edges than the Gaussian, without any loss of either the extent or quality of noise reduction. It also achieves better noise reduction then the median, particularly in terms of quality, as well as preserving smaller features. The ground in Fig. 9 is a large region with a fairly random texture and NLM removes much of this texture, whilst still performing well on the boat. OCCO is good at edge preservation but only at the cost of removing less noise, whereas the Gaussian levelling is not capable of removing noise close to edges.

Figures 10, 11 and 12 are further examples which highlight the behaviour in different scenarios: ‘peppers’ has relatively large smooth areas away from edges, ‘house’ has large areas of fairly constant value, and ‘fruits’ contains a particularly challenging mixture of different smoothness and scale. NLM is generally impressive except for the replacement of fine texture with fine artefacts; diffusion is also good but can be limited in edge sharpness in some places; image-guided and particularly levelling leave residual noise close to edges. OCCO generally performs better than the median, but with similar loss of small features. Figure 13 shows the ‘tulips’ image with less added noise, and hence the filter length is shorter. For this level of noise, the bitonic, diffusion and NLM all preserve the fine details (for instance the veins on the leaves) well, though the edges are slightly sharper on the bitonic and NLM images.

These images are at a specific noise level to show visual differences between the techniques, whereas Fig. 6 contains average results for all images but with different levels of added noise. It must be acknowledged that SNR and SSIM are not complete measures of image quality, and are not responsive to small but visually distracting artefacts. However, this indicates that the bitonic filter, alongside NLM and diffusion, performs well across all noise levels, in contrast to the morphological and image-guided filters whose performance deteriorates in high noise scenarios.

V. Discussion

In terms of SNR and SSIM, the bitonic filter is not the best performing filter of those tested here in all scenarios. However, it does seem to perform well in all tested cases, better than any of the other morphological filters, and has a different set of characteristics from the other linear filters which are better suited to some situations.

Most notably, where the noise is not uniformly distributed over an image, the bitonic is capable of very good noise reduction in some areas with potentially perfect preservation of signals, even those with edges, in others. This is a direct result of the definitions of signal and noise as bitonic over different scales. For low-noise images, the filter can be used to smooth over repetitive details (anything not bitonic over the filter length) whilst preserving individual fine details and any transitions very well, which is potentially of use in edge extraction, background extraction, feature enhancement or other artistic processing. The other linear filters are not built on

[8] e.g. https://homepages.cae.wisc.edu/~ece533/images/

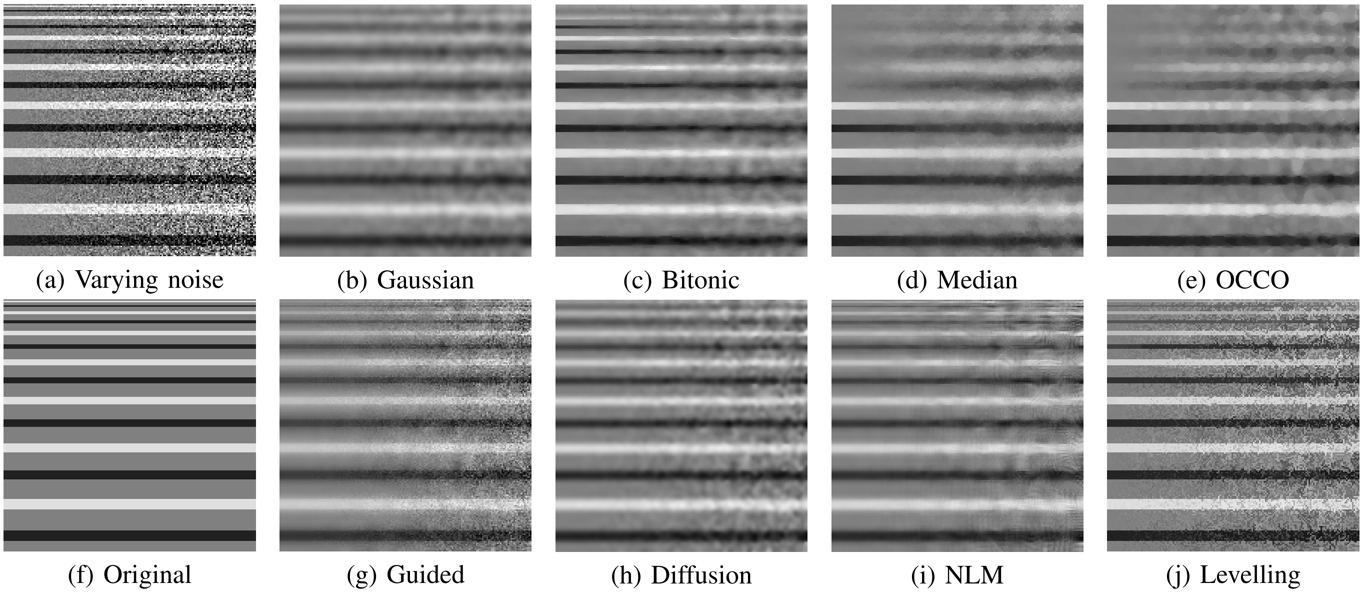
[9] www.spaceneedle.com

[10] www.empmuseum.org

TABLE I

SIGNAL TO NOISE RATIO (SNR) AND STRUCTURAL SIMILARITY (SSIM) FOR FULL VERSIONS OF ALL IMAGES. THE BEST THREE RESULTS ARE INDICATED BY THEIR RANKING, WITH ^{1} ALSO IN BOLD. †THE FILTER LENGTH l, IS AS DESCRIBED IN SECTION IV. *TIMES ARE FOR APPROXIMATE REFERENCE ONLY: THE CODE FOR GAUSSIAN, MEDIAN, BITONIC, NLM, OCCO, GRAIN AND LEVELLING WAS ALL REASONABLY WELL OPTIMISED AND SINGLE-THREADED, GUIDED WAS HIGHLY OPTIMISED AND MAY BE MULTI-THREADED, THE DIFFUSION CODE HAD LITTLE OPTIMISATION.

image	measure	Noise	Gaussian	Median	Bitonic	Guided	Diffusion	NLM	OCCO	Grain	Levelling
Varying	SNR (dB)	10.01	15.36	15.31	**17.24**^{1}	15.56	15.89	16.30^2	16.12^3	15.10	14.55
168×168	SSIM	0.546	0.683	0.705	**0.787**^{1}	0.708	0.720	0.747^3	0.762^2	0.696	0.664
Grey	$l^\dagger$	-	7	7	7	3	5	3	3	9	7
	Time (ms)*	-	1	2	11	11	30	40	47	120	404
Boat	SNR (dB)	10.54	19.59	18.58	19.99^3	19.20	20.36^2	**20.71**^{1}	19.54	17.06	17.17
512×512	SSIM	0.198	0.632^3	0.542	0.633^2	0.526	**0.635**^{1}	0.631	0.586	0.415	0.412
Grey	$l^\dagger$	-	9	9	9	5	5	9	3	21	17
	Time (ms)*	-	9	26	107	32	175	2,230	201	1,432	6,740
Peppers	SNR (dB)	9.95	19.68	19.10	20.64^2	19.55	20.54^3	**20.72**^{1}	19.60	16.66	16.85
512×512	SSIM	0.651	0.947^3	0.940	**0.953**^{1}	0.935	**0.953**^{1}	0.948^2	0.939	0.891	0.886
RGB	$l^\dagger$	-	11	11	11	5	9	7	5	21	17
	Time (ms)*	-	43	91	385	96	844	2,256	527	5,931	21,045
House	SNR (dB)	17.06	23.95	23.67	25.37^3	24.30	25.40^2	**27.28**^{1}	24.56	22.44	22.58
256×256	SSIM	0.335	0.779	0.739	0.780^3	0.737	0.801^2	**0.835**^{1}	0.771	0.615	0.627
Grey	$l^\dagger$	-	7	7	7	5	9	9	3	9	9
	Time (ms)*	-	2	6	27	12	112	556	67	156	1,269
Fruits	SNR (dB)	12.24	21.78	20.81	22.63^3	21.74	23.14^2	**25.30**^{1}	22.40	19.40	19.58
512×512	SSIM	0.492	0.901^3	0.882	0.907^2	0.861	0.907^2	**0.932**^{1}	0.881	0.789	0.801
RGB	$l^\dagger$	-	11	11	11	7	7	11	3	13	15
	Time (ms)*	-	43	91	385	128	585	12,109	466	3,637	21,192
Tulips	SNR (dB)	21.61	25.23	25.09	26.73^3	25.95	**27.39**^{1}	26.84^2	24.21	24.33	24.32
768×512	SSIM	0.857	0.960^2	0.951	0.959^3	0.955	**0.966**^{1}	0.956	0.954	0.921	0.927
RGB	$l^\dagger$	-	5	5	5	5	11	5	3	5	7
	Time (ms)*	-	45	91	407	125	1,499	1,023	710	1,100	7,055


(a) Varying noise (b) Gaussian (c) Bitonic (d) Median (e) OCCO

(f) Original (g) Guided (h) Diffusion (i) NLM (j) Levelling

Fig. 7. Filter performance on ‘varying’ image. These images have been optimised for best visual quality: see Table I for results optimised for SNR and SSIM.

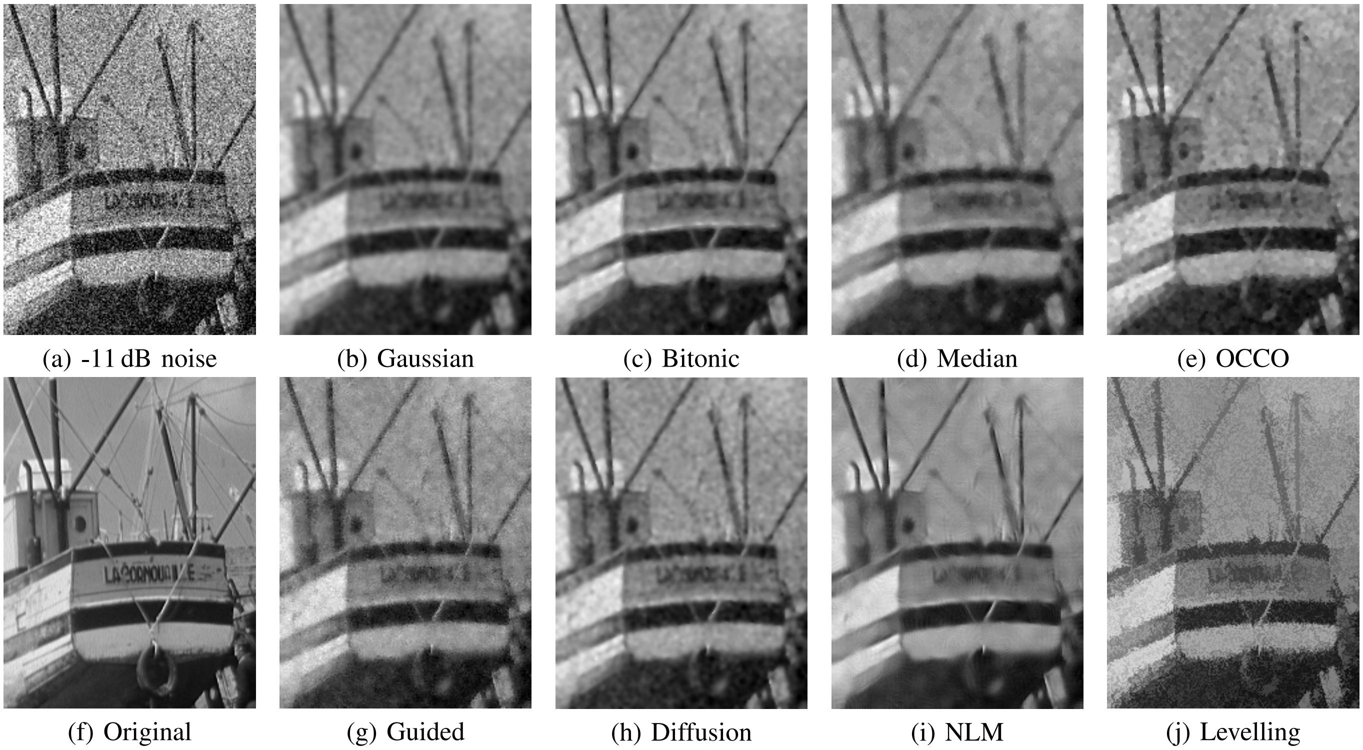


Fig. 8. Filter performance on a region of 'boat' image: see Table I for details.

a bitonic definition of signals, and are more naturally sensitive to data values, with the associated difficulty of preserving any edges in the data which are smaller than the level of the noise.

Unlike the image-guided filter, anisotropic diffusion or the non-local means filter, the bitonic filter also achieves this without the use of any parameter which would make it sensitive to a particular data threshold, nor any need for parameter optimisation to the level of noise in the image, nor any issues with stability nor the introduction of spurious features in high levels of noise. Indeed in nearly all scenarios, the only necessary adjustment is the filter length. It is hence easy to apply in practice since it adapts well to whatever it finds in the signal. It seems to present an improvement on both the Gaussian and median in all circumstances: sharper than the former, and more detail-preserving and with better noise qualities than the latter.

The bitonic filter can also be tuned (by lowering the centile c) to treat isolated impulses as signal, without much detrimental effect on the rejection of other non-impulsive noise.

In terms of processing speed, the bitonic filter is typically 3 to 4 times slower than the median, 10 times slower than the Gaussian. However, this is still competitive with other edge-preserving filters: faster than non-local means and diffusion, and not much slower than the less well-performing image-guided filter. The processing time for each data point as implemented scales linearly with the filter length, with constant time implementations also possible, hence it is perfectly applicable to longer filters and much larger images.

The main disadvantage of this filter, which only applies to signals of two dimensions or greater, is the slight sensitivity to the shape of the structuring element. The bitonic is in fact less sensitive in this respect than a median filter using the same element, however when using a circular disk, and for larger filter lengths than were used here, it will begin to round off square corners. If the geometry is of a known form, another more appropriate structuring element could be used, for instance a square for images with horizontal and vertical features.

VI. Conclusion

It has been demonstrated that a filter following a bitonic definition of signals can outperform the Gaussian and median filters and has different characteristics to other edge-preserving filters, which make it particularly suitable for situations of varying or unknown noise.

Since the bitonic filter is entirely local and not iterative, adaptations of other filters (e.g. iteration or local parameter variation) might also be appropriate adaptations of the bitonic. It could also provide a substitute for the Gaussian or median in any filter which makes use of them, including the non-local means filter. Alternatively, the bitonic filter could itself be built around a different linear filter than the Gaussian, in which case it should improve the edge-preserving capabilities of the underlying linear filter.

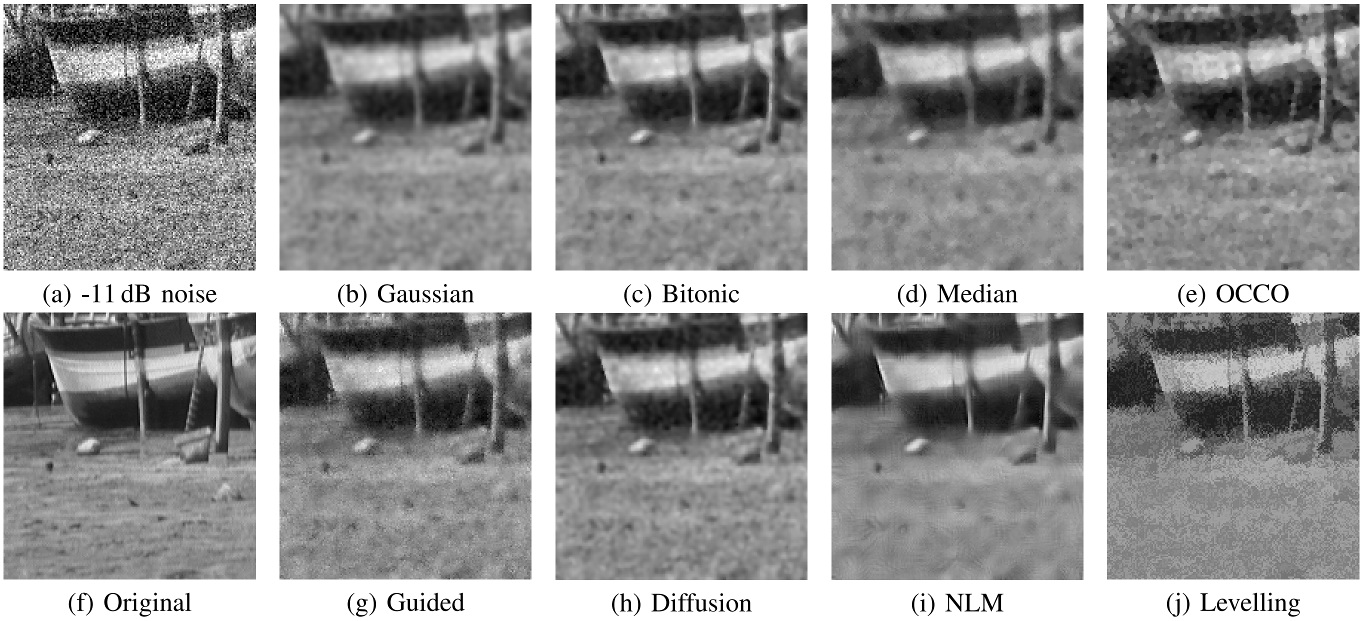
The bitonic filter has been implemented for MATLAB[11] and a 2D version for use on images is also available in free wxDicom[12] software.

References

[1] K. E. Batcher, "Sorting networks and their applications," in *ACM Proc.* ACM, Apr. 1968, pp. 307–314.

[11] MATLAB file exchange: Bitonic Filter by Graham Treece, 15 Dec 2015

[12] http://mi.eng.cam.ac.uk/Main/GMT_wxDicom


(a) -11 dB noise (b) Gaussian (c) Bitonic (d) Median (e) OCCO

(f) Original (g) Guided (h) Diffusion (i) NLM (j) Levelling

Fig. 9. Filter performance on a region of 'boat' image: see Table I for details.

[2] G. Heygster, "Rank filters in digital image processing," *Comput Graphics Image Process*, vol. 19, no. 2, pp. 148–164, 1982.

[3] J. Serra, "Morphological filtering: an overview," *Signal Process*, vol. 38, no. 1, pp. 3–11, 1994.

[4] C. Ronse, *Extraction of narrow peaks and ridges in images by combination of local low rank and max filters: implementation and applications to clinical angiography*. Working Document WD47, Philips Research Laboratories, Brussels, Belgium, 1988.

[5] P. Kuosmanen and J. Astola, "Soft morphological filtering," *J Math Imaging Vision*, vol. 5, no. 3, pp. 231–262, 1995.

[6] M. Kass and J. Solomon, "Smoothed local histogram filters," in *ACM Trans Graphics*, vol. 29, no. 4, 2010, p. Article 100.

[7] S. Perreault and P. Hébert, "Median filtering in constant time," *IEEE Trans Image Process*, vol. 16, no. 9, pp. 2389–2394, 2007.

[8] H. Hwang, R. Haddad *et al.*, "Adaptive median filters: new algorithms and results," *IEEE Trans Image Process*, vol. 4, no. 4, pp. 499–502, 1995.

[9] Z. Ma, K. He, Y. Wei, J. Sun, and E. Wu, "Constant time weighted median filtering for stereo matching and beyond," in *IEEE Int Conf Comput Vision (ICCV)*, 2013, pp. 49–56.

[10] R. C. Hardie and K. E. Barner, "Rank conditioned rank selection filters for signal restoration," *IEEE Trans Image Process*, vol. 3, no. 2, pp. 192–206, 1994.

[11] K. E. Barner and G. R. Arce, "Order-statistic filtering and smoothing of time-series: Part II," *Handbook of statistics*, vol. 17, pp. 555–602, 1998.

[12] P. Soille, "On morphological operators based on rank filters," *Pattern Recognition*, vol. 35, no. 2, pp. 527–535, 2002.

[13] S. Taouli and F. Bereksi-Reguig, "ECG signal denoising by morphological top-hat transform," *Global J Comput Science Technology*, vol. 13, no. 5, 2013.

[14] P. Salembier and M. H. Wilkinson, "Connected operators," *IEEE Signal Process Mag*, vol. 26, no. 6, pp. 136–157, 2009.

[15] V. Caselles and P. Monasse, "Grain filters," *J Math Imaging Vision*, vol. 17, no. 3, pp. 249–270, 2002.

[16] P. Monasse and F. Guichard, "Fast computation of a contrast-invariant image representation," *IEEE Trans Image Process*, vol. 9, pp. 860–872, 2000.

[17] E. Carlinet and T. Géraud, "MToS: A tree of shapes for multivariate images," *IEEE Trans Image Process*, vol. 24, no. 12, pp. 5330–5342, 2015.

[18] L. Vincent, "Morphological area openings and closings for grey-scale images," in *Shape in Picture*. Springer, 1994, pp. 197–208.

[19] J. Serra, C. Vachier, and F. Meyer, "Levelings," *Mathematical Morphology: From Theory to Applications*, pp. 199–228, 2013.

[20] S. Velasco-Forero, J. Angulo, and P. Soille, "Conditional toggle mappings: principles and applications," *J Math Imaging Vision*, vol. 48, no. 3, pp. 544–565, 2014.

[21] J. Bednar and T. L. Watt, "Alpha-trimmed means and their relationship to median filters," *IEEE Trans Acoustics, Speech Signal Process*, vol. 32, no. 1, pp. 145–153, 1984.

[22] A. B. Hamza and H. Krim, "Image denoising: A nonlinear robust statistical approach," *IEEE Trans Signal Process*, vol. 49, no. 12, pp. 3045–3054, 2001.

[23] L. F. Pessoa and P. Maragos, "MLR-filters: A general class of nonlinear systems and their optimal design for image processing," *IEEE Trans Image Process,*, vol. 7, no. 7, pp. 966–978, 1998.

[24] E. A. Thompson, R. C. Hardie, and K. E. Barner, "Hybrid order statistic filter and its application to image restoration," *Applied Optics*, vol. 40, no. 5, pp. 656–661, 2001.

[25] H. Hu and G. de Haan, "Classification-based hybrid filters for image processing," in *Electronic Imaging*. International Society for Optics and Photonics, 2006, pp. 607 711–607 711.

[26] E. Aptoula and S. Lefevre, "A comparative study on multivariate mathematical morphology," *Pattern Recognition*, vol. 40, no. 11, pp. 2914–2929, 2007.

[27] P. Maragos and G. Evangelopoulos, "Leveling cartoons, texture energy markers, and image decomposition," in *Proc Eighth Int Symposium on Mathematical Morphology*, ser. ISMM '07. Kluwer Academic Publishers, 2007, pp. 125–138.

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

[29] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Trans Pattern Anal Mach Intel*, vol. 35, no. 6, pp. 1397–1409, 2013.

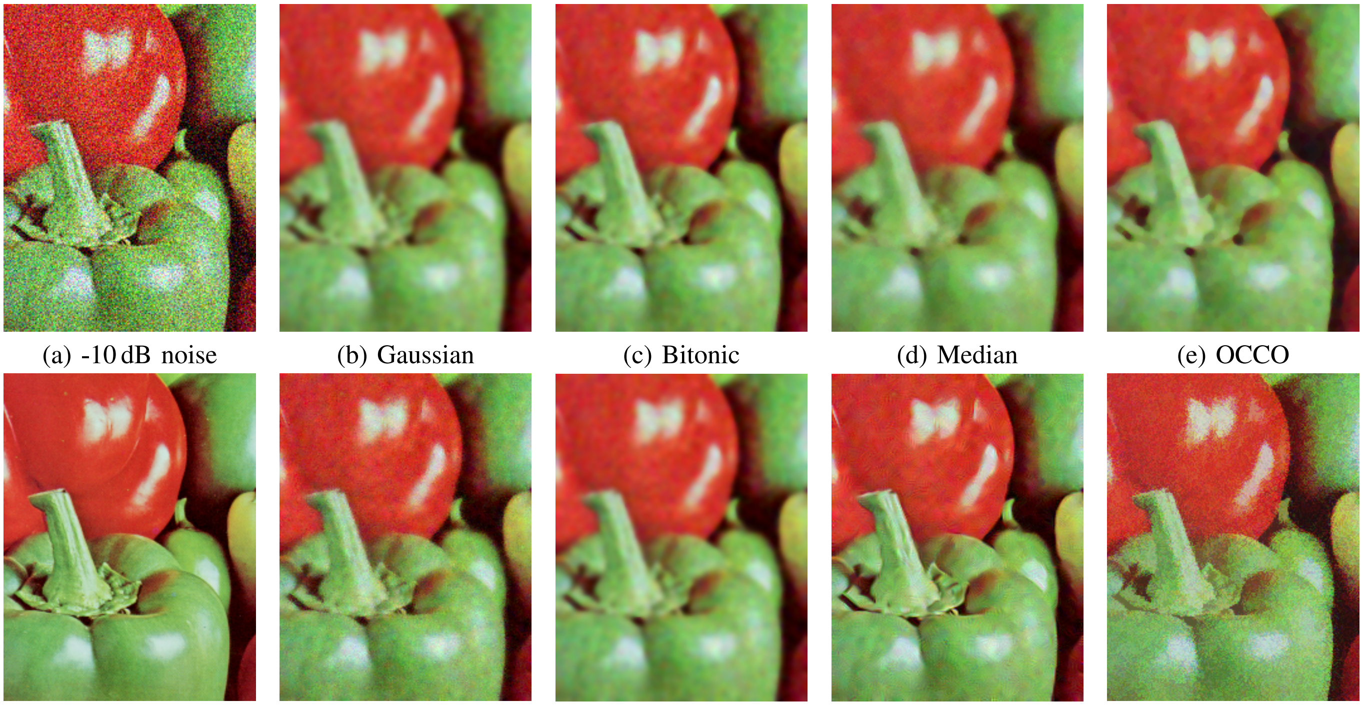
[30] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans Pattern Anal Mach Intel*, vol. 12, no. 7, pp. 629–639, 1990.

[31] J. Weickert, *Anisotropic diffusion in image processing*. Teubner Stuttgart, 1998, vol. 1.

[32] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Sixth International Conference on Computer Vision, 1998*. IEEE, 1998, pp. 839–846.

[33] N. Ray and S. T. Acton, "Inclusion filters: A class of self-dual connected operators," *IEEE Transactions on Image Processing*, vol. 14, no. 11, pp. 1736–1746, 2005.

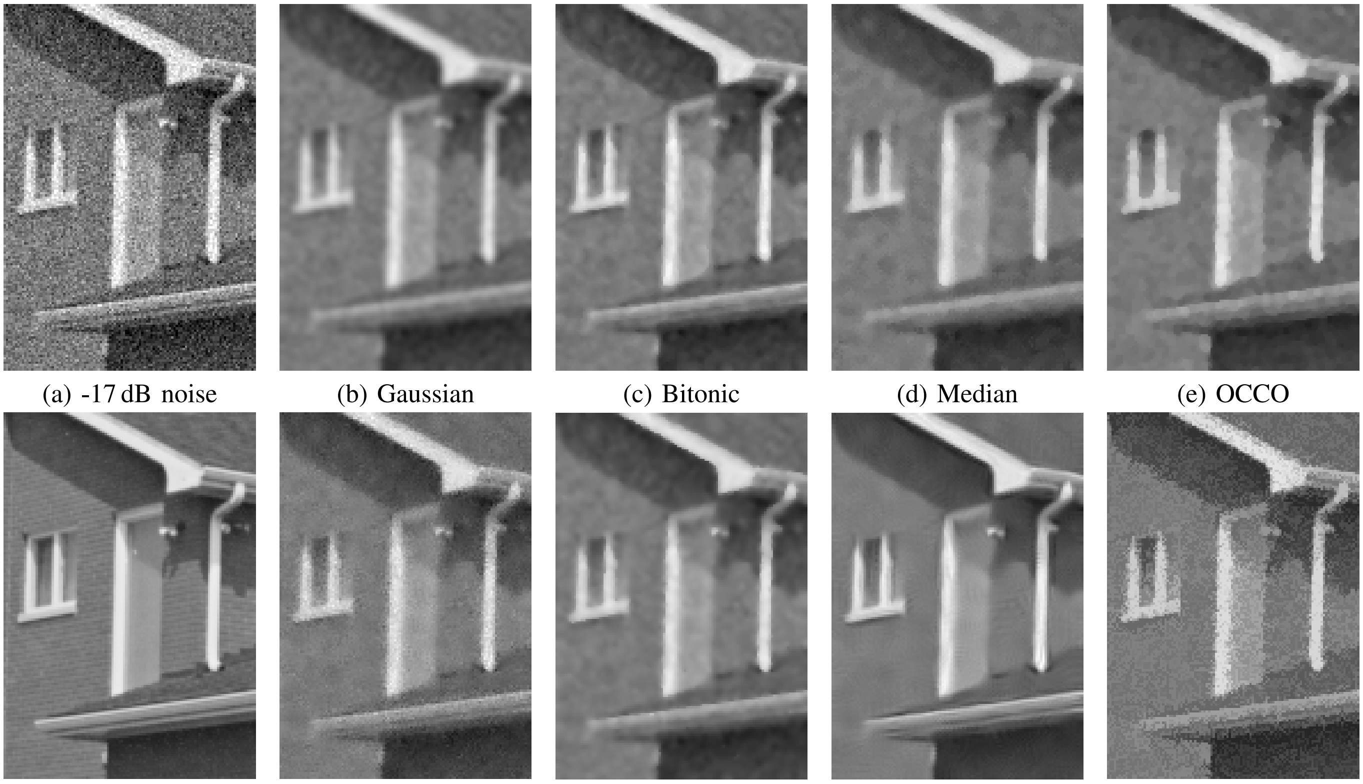
[34] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans Image Process*, vol. 13, no. 4, pp. 600–612, 2004.



(a) -10 dB noise (b) Gaussian (c) Bitonic (d) Median (e) OCCO

(f) Original (g) Guided (h) Diffusion (i) NLM (j) Levelling

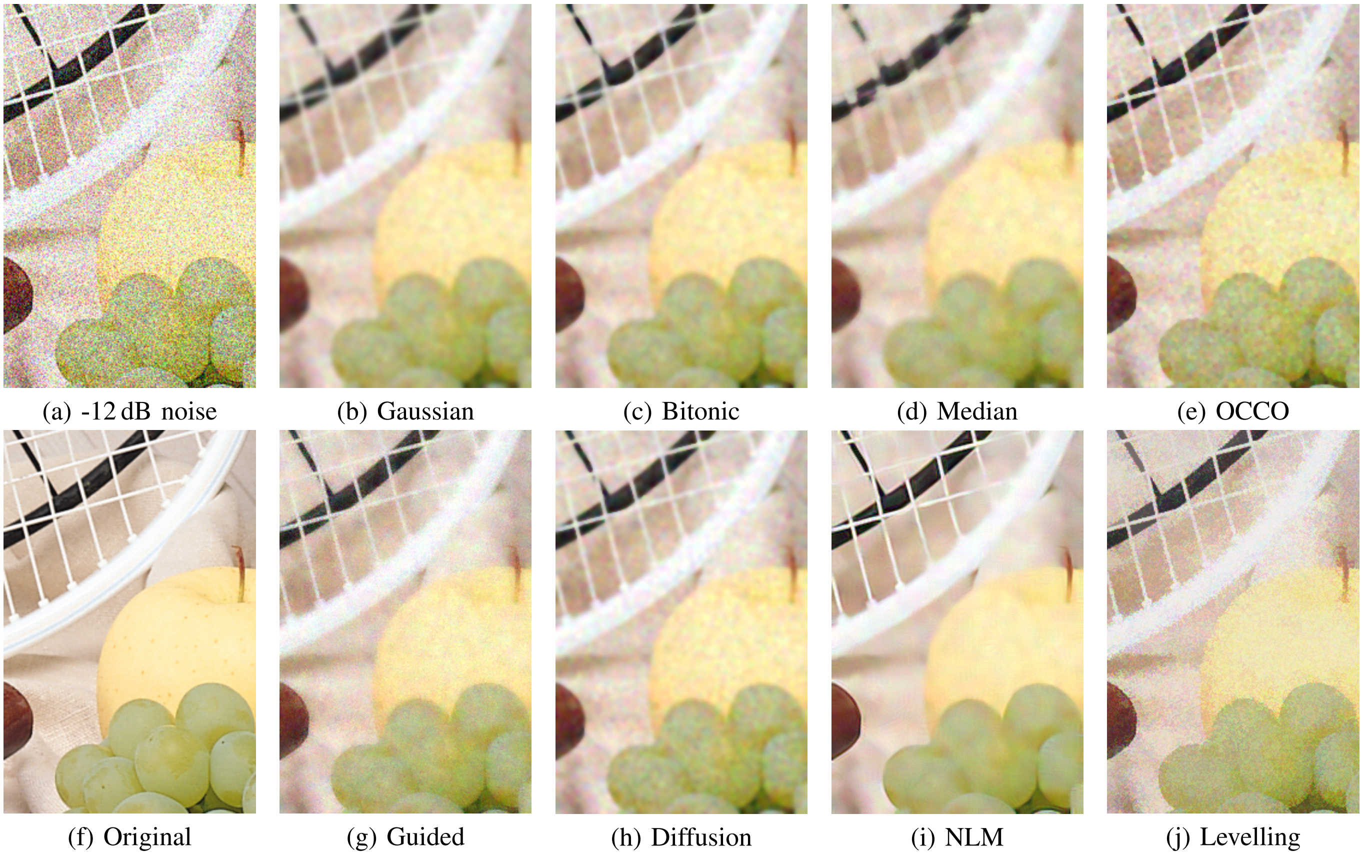
Fig. 10. Filter performance on a region of ‘peppers’ image: see Table I for details.



(a) -17 dB noise (b) Gaussian (c) Bitonic (d) Median (e) OCCO

(f) Original (g) Guided (h) Diffusion (i) NLM (j) Levelling

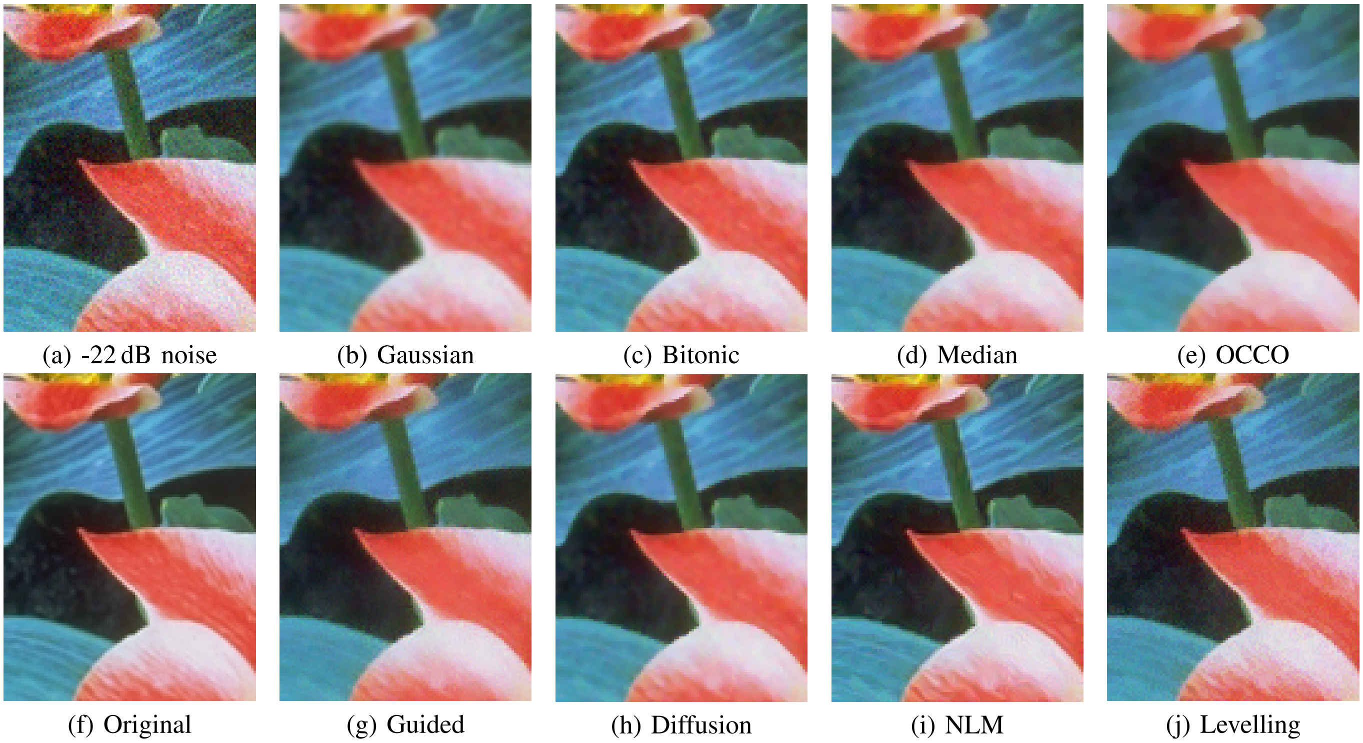
Fig. 11. Filter performance on a region of ‘house’ image: see Table I for details.



(a) -12 dB noise (b) Gaussian (c) Bitonic (d) Median (e) OCCO

(f) Original (g) Guided (h) Diffusion (i) NLM (j) Levelling

Fig. 12. Filter performance on a region of 'fruits' image: see Table I for details.



(a) -22 dB noise (b) Gaussian (c) Bitonic (d) Median (e) OCCO

(f) Original (g) Guided (h) Diffusion (i) NLM (j) Levelling

Fig. 13. Filter performance on a region of 'tulips' image with low noise: see Table I for details.

Graham Treece obtained an MA degree in engineering, with distinction, from the University of Cambridge, UK in 1994. Following a few years working as a software engineer in noise cancellation and radio communications, he returned to the University of Cambridge and completed his PhD in 3D ultrasound acquisition, visualisation and analysis in 2001. He continued to pursue ultrasound research at Cambridge during a five year research fellowship from the Royal Academy of Engineering and the Engineering and Physical Sciences Research Council, focusing on the measurement of attenuation and elastography. From 2008 to 2015 he was the inaugural Evelyn Trust lecturer in Engineering for Clinical Practice at the Department of Engineering in Cambridge, with research spanning most medical imaging modalities but particularly X-ray Computed Tomography for the detailed analysis and visualisation of bone diseases. He is currently a reader in information engineering at the University of Cambridge. He is the author of a variety of free software packages, including Stradwin for 3D ultrasound acquisition and medical data visualisation.