Alternating guided image filtering

Alexander Toet [Corresp. 1]

[1] TNO, Soesterberg, The Netherlands

Corresponding Author: Alexander Toet
Email address: lextoet@gmail.com

Edge preserving filters aim to simplify the representation of images (e.g., by reducing noise or eliminating irrelevant detail) while preserving their most significant edges. These filters are typically nonlinear and locally smooth the image structure while minimizing both blurring and over-sharpening of visually important edges. Here we present the Alternating Guided Filter (AGF) that achieves edge preserving smoothing by combining two recently introduced filters: the Rolling Guided Filter (RGF) and the Smooth and iteratively Restore Filter (SiR). We show that the integration of RGF and SiR in an alternating iterative framework results in a new smoothing operator that preserves significant image edges while effectively eliminating small scale details. The AGF combines the large scale edge and local intensity preserving properties of the RGF with the edge restoring properties of the SiR while eliminating the drawbacks of both previous methods (i.e., edge curvature smoothing by RGF and local intensity reduction and restoration of small scale details near large scale edges by SiR). The AGF is simple to implement and efficient, and produces high-quality results. We demonstrate the effectiveness of AGF on a variety of images, and provide a public code to facilitate future studies.

Alternating Guided Image Filtering

Alexander Toet

TNO
Kampweg 5, 3769DE, Soesterberg, The Netherlands
E lex.toet@tno.nl

2   **1 Introduction**

4   Natural scenes contain meaningful visual elements at different spatial scales. Small
5   elements typically represent texture, small objects, and noise, while large scale structures
6   generally represent object or region boundaries, spatial color transitions, or homogeneous
7   regions. Spatial filtering is a common operation in image processing and computer vision that
8   is typically used to reduce noise or eliminate small spurious details (e.g., texture) and
9   enhance image contrast. In spatial filtering the value of the filtered image at a given location
10   is a function of the original pixel values in a small neighborhood of the same location. In
11   linear filtering the value of an output pixel is a linear combination (e.g., a weighted average)
12   of the values of the pixels in the input pixel's neighborhood. When the image intensity varies
13   smoothly over space, nearby pixels are likely to have similar values (i.e., they will be
14   correlated). In contrast, the noise that corrupts these pixel values will be less correlated than
15   the signal values, so that averaging reduces noise while preserving the mean signal value.
16   However, the assumption of smooth variation evidently does not hold near edges. Thus,
17   although linear filtering can effectively reduce image noise and eliminate unwanted details
18   smaller than the filter kernel size, it degrades the articulation of (blurs) the remaining edges,
19   lines and other details that are important for the interpretation of the image. Therefore, edge
20   preserving filters have been developed that reduce small (relative to the filter kernel size)
21   scale image variations (noise or texture) while preserving larger scale discontinuities (edges).
22
23   Some well-known non-linear edge-preserving smoothing filters are for instance anisotropic
24   diffusion (Perona & Malik, 1990), robust smoothing (Black et al., 1998) and the bilateral
25   filter (Tomasi & Manduchi, 1998). However, anisotropic diffusion tends to oversharpen
26   edges (i.e., it produces halos) and is computationally expensive, which makes it less suitable
27   for application, for instance, in multiresolution schemes (Farbman et al., 2008). The non-
28   linear bilateral filter (BLF) assigns each pixel a weighted mean of its neighbors, with the
29   weights decreasing both with spatial distance and with difference in value (Tomasi &
30   Manduchi, 1998). While the BLF is quite effective at smoothing small intensity changes
31   while preserving strong edges and has efficient implementations, it also tends to blur across
32   edges at larger spatial scales, thereby limiting its value for application in multiscale image
33   decomposition schemes (Farbman et al., 2008). In addition, the BLF has the undesirable
34   property that it can reverse the intensity gradient near sharp edges (the weighted average
35   becomes unstable when a pixel has only few similar pixels in its neighborhood: He, Sun &
36   Tang, 2013). In the joint (or cross) bilateral filter (JBLF) a second or guidance image serves
37   to steer the edge stopping range filter thus preventing over- or under- blur near edges
38   (Petschnigg et al., 2004). The recently introduced Guided Filter (GF: He, Sun & Tang, 2013)
39   is a computationally efficient, edge-preserving translation-variant operator based on a local
40   linear model which avoids the drawbacks of bilateral filtering and other previous approaches.
41   When the input image also serves as the guidance image, the GF behaves like the edge
42   preserving BLF. However, similar to the BLF, the GF also tends to produce halos near
43   significant edges.
44
45   Two recently presented iterative guided filtering frameworks have the ability to perform edge
46   preserving filtering without the introduction of halos. Zhang et al. (2014) showed that the
47   application of the JBLF in an iterative Rolling Guided Filter (RGF) framework results in size
48   selective filtering of small scale details combined with the recovery of larger scale edges.
49   However, the RGF also has a drawback: it tends to smooth the curvature of large scale edges.

50   Kniefacz and Kropatsch (2015) recently introduced a similar framework called the Smooth
51   and iteratively Restore (SiR) filter. SiR restores large scale edges while preserving their
52   curvature. However, SiR also has two drawbacks: it reduces the local image intensity and
53   restores small scale details in the neighborhood of large scale edges.
54
55   In this paper we propose an Alternating Guided Filter (AGF) scheme that integrates the RGF
56   and SiR in a single framework, resulting in a new image smoothing operator that preserves
57   significant edges while effectively eliminating small scale details. AGF combines the edge
58   preserving properties of both the RGF (i.e., the elimination of small details in combination
59   with local intensity preservation) and the SiR (i.e., the articulated restoration of large scale
60   edges) while it does not suffer from their respective drawbacks (the curvature smoothing of
61   large scale edges by RGF and local intensity reduction in combination with the restoration of
62   small scale details near large scale edges by SiR).
63
64   The rest of this paper is organized as follows. Section 2 briefly discusses some related work
65   on edge preserving filtering and introduces the RGF and SiR filter techniques on which the
66   proposed AGF scheme is based. Section 3 presents the proposed Alternating Guided Filter
67   scheme. Section 4 presents the results of the application of the AGF filter to natural and
68   synthetic images, compares the performance of this new framework with the RGF and SiR
69   filter schemes, and provides some runtime estimates. Finally, in Section 5 the conclusions of
70   this study are presented.
71
72
73
74

75   **2 Related work**
76
77   In this section we briefly review the edge preserving bilateral and joint bilateral filters and
78   show how they are related to the guided filter. We also discuss two recently introduced
79   frameworks that iteratively apply joint bilateral filters to achieve selective guided filtering of
80   small scale details in combination with large scale edge restoration.
81
82

83   *2.1 Bilateral Filter*
84
85   The bilateral filter is a non-linear filter that computes the output at each pixel as a Gaussian
86   weighted average of their spatial and intensity distances. It prevents blurring across edges by
87   assigning larger weights to pixels that are spatially close and have similar intensity values
88   (Tomasi & Manduchi, 1998). It uses a combination of (typically Gaussian) spatial and a
89   range (intensity) filter kernels that perform a blurring in the spatial domain weighted by the
90   local variation in the intensity domain. It combines a classic low-pass filter with an edge-
91   stopping function that attenuates the filter kernel weights at locations where the intensity
92   difference between pixels is large. Bilateral filtering was developed as a fast alternative to the
93   computationally expensive technique of anisotropic diffusion, which uses gradients of the
94   filtering images itself to guide a diffusion process, thus avoiding edge blurring (Perona &
95   Malik, 1990). More formally, at a given image location (pixel) i, the filtered output O_i is
96   given by:

97
$$O_i = \frac{1}{K_i} \sum_{j \in \Omega} I_j \, f(\|i - j\|) \, g(\|I_i - I_j\|) \tag{1}$$

98   where
99   f is the *spatial* filter kernel (e.g., a Gaussian centered at i),
100   g is the *range* or intensity (edge-stopping) filter kernel (centered at the image value at i),
101   Ω is the spatial support of the kernel,
102   K_i is a normalizing factor (the sum of the $f \cdot g$ filter weights) .

103

104   The bilateral filter is controlled by only two parameters: the extent of respectively the spatial
105   kernel and the range kernel. Intensity edges are preserved since the bilateral filter decreases
106   not only with the spatial distance but also with the intensity distance. Though the filter has
107   efficient implementations (Paris & Durand, 2006) and effectively reduces noise while
108   preserving edges in many situations, it has the undesirable property that it can reverse the
109   intensity gradient near sharp edges (the weighted average becomes unstable when a pixel has
110   only few similar pixels in its neighborhood (He, Sun & Tang, 2013)).

111

112   In the joint (or cross) bilateral filter (JBLF) the range filter g is applied to a second or
113   guidance image G (Petschnigg et al., 2004):

114
$$O_i = \frac{1}{K_i} \sum_{j \in \Omega} I_j \cdot f(\|i - j\|) \cdot g(\|G_i - G_j\|) \tag{2}$$

115   The JBLF can prevent over- or under- blur near edges by using a related image G to guide
116   the edge stopping behavior of the range filter. That is, the JBLF smooths the image I while
117   preserving edges that are also represented in the image G . The JBLF is particularly favored
118   when the edges in the image that is to be filtered are unreliable (e.g., due to noise or
119   distortions) and when a companion image with well-defined edges is available (e.g., in the
120   case of flash /no-flash image pairs).

121   *2.2 Guided Filter*

122

123   A Guided Filter (GF: He, Sun & Tang, 2013) is a translation-variant filter based on a local
124   linear model. Guided image filtering involves an input image I , a guidance image G (which
125   may be identical to the input image), and an output image O . The two filtering conditions are
126   (*i*) that the local filter output is a linear transform of the guidance image G and (*ii*) as similar
127   as possible to the input image I . The first condition implies that

128
$$O_i = a_k G_i + b_k \qquad \forall i \in \omega_k \tag{3}$$

129   where ω_k is a square window of size $(2r+1) \times (2r+1)$. The local linear model ensures that the
130   output image O has an edge only at locations where the guidance image G has one, because
131    $\nabla O = a \nabla G$. The linear coefficients a_k and b_k are constant in ω_k . They can be estimated by
132   minimizing the squared difference between the output image O and the input image I (the
133   second filtering condition) in the window ω_k , i.e. by minimizing the cost function E :

134
$$E(a_k, b_k) = \sum_{i \in \omega_k} \left((a_k G_i + b_k - I_i)^2 + \varepsilon \, a_k^2 \right) \tag{4}$$

135   where ε is a regularization parameter penalizing large a_k . The coefficients a_k and b_k can
136   directly be solved by linear regression (He, Sun & Tang, 2013):

137
$$a_k = \frac{\frac{1}{|\omega|}\sum_{i\in\omega_k} G_i I_i - \bar{G}_k \bar{I}_k}{\sigma_k^2 + \varepsilon} \tag{5}$$

138

139
$$b_k = \bar{I}_k - a_k \bar{G}_k \tag{6}$$

140 where

141     $|\omega|$ is the number of pixels in ω_k,

142     $\bar{I}_k$ and $\bar{G}_k$ represent the means of respectively I and G over ω_k, and

143   σ_k^2 is the variance of I over ω_k.

144 Since pixel i is contained in several different (overlapping) windows ω_k, the value of O_i in

145 Equation (3) depends on the window over which it is calculated. This can be accounted for by

146 averaging over all possible values of O_i:

147
$$O_i = \frac{1}{|\omega|}\sum_{k|i\in\omega_k}\left(a_k G_k + b_k\right) \tag{7}$$

148 Since $\sum_{k|i\in\omega_k} a_k = \sum_{k\in\omega_i} a_k$ due to the symmetry of the box window Equation (7) can be written as

149
$$O_i = \bar{a}_i G_i + \bar{b}_i \tag{8}$$

150 where $\bar{a}_i = \frac{1}{|\omega|}\sum_{k\in\omega_i} a_k$ and $\bar{b}_i = \frac{1}{|\omega|}\sum_{k\in\omega_i} b_k$ are the average coefficients of all windows

151 overlapping i. Although the linear coefficients $(\bar{a}_i, \bar{b}_i)$ vary spatially, their gradients will be

152 smaller than those of G near strong edges (since they are the output of a mean filter). As a

153 result we have $\nabla O \approx \bar{a}\,\nabla G$, meaning that abrupt intensity changes in the guiding image G

154 are still largely preserved in the output image O.

155

156 Equations (5), (6) and (8) define the GF. When the input image also serves as the guidance

157 image, the GF behaves like the edge preserving bilateral filter, with the parameters ε and the

158 window size r having the same effects as respectively the range and the spatial variances of

159 the bilateral filter. Equation (8) can be rewritten as

160
$$O_i = \sum_j W_{ij}(G) I_j \tag{9}$$

161 with the weighting kernel W_{ij} depending only on the guidance image G:

162
$$W_{ij} = \frac{1}{|\omega|^2}\sum_{k:(i,j)\in\omega_k}\left(1 + \frac{(G_i - \bar{G}_k)(G_j - \bar{G}_k)}{\sigma_k^2 + \varepsilon}\right) \tag{10}$$

163 Since $\sum_j W_{ij}(G) = 1$ this kernel is already normalized.

164

165 The GF is a computationally efficient, edge-preserving operator which avoids the gradient

166 reversal artefacts (i.e., over sharpening of edges) of the BLF. However, just like the BLF, GF

167 has the limitation that it tends to produce halos (unwanted blurring) near large scale edges

168 (He, Sun & Tang, 2013).

169

170 *2.3 Rolling Guidance Filter*

171

172 Zhang et al. (2014) showed that the application of the joint bilateral filter (Equation 2) in an

173 iterative framework results in effective size selective filtering of small scale details combined

174 with the recovery of larger scale edges. In their Rolling Guidance Filter (RGF) framework the

175   result G^{t+1} of the t-th iteration is obtained from the joint bilateral filtering of the input image I
176   using the result G^t of the previous iteration step as the guidance image:

177
$$G_i^{t+1} = \frac{1}{K_i} \sum_{j \in \Omega} I_j \cdot f(\| i - j \|) \cdot g(\| G_i^t - G_j^t \|) \tag{11}$$

178   In the RGF scheme details smaller than the Gaussian kernel of the bilateral filter are initially
179   removed while the edges of the remaining details are restored by iteratively updating the
180   guidance image. At the start of the iteration process the term $\| G_i^t - G_j^t \|$ is almost zero,
181   making the range filter g inoperative, so that the joint bilateral filter effectively behaves like
182   a Gaussian filter. Details removed by this filter cannot be recovered later in the process. After
183   each iteration step the influence of the range filter gradually increases due to its increasing
184   weights. Hence, guided by the input image I, the RGF scheme gradually restores the edges
185   that remain after the initial Gaussian filtering. Note that the initial guidance image G^1 can
186   simply be a constant (e.g., zero) valued image since it updates to the Gaussian filtered input
187   image in the first iteration step. The pseudocode for the RGF scheme is presented in Table 1.
188
189   The RGF scheme is not restricted to the JBF (Zhang et al., 2014). Any average-based joint
190   edge aware averaging filter (e.g., the Guided Filter) can be applied in the RGF framework. In
191   practice the RGF converges rapidly after only a few (3-5) iterations steps. The RGF scheme
192   is simple to implement and can be efficiently computed.
193
194   Although the RGF enables effective and efficient size selective filtering, it has as drawback
195   that the remaining large scale edges are smoothly curved.
196

197   *2.4 Smooth and iteratively Restore Filter*
198
199   Kniefacz and Kropatsch (2015) recently introduced the Smooth and iteratively Restore (SiR)
200   filter. Similar to RGF, SiR initially removes small scale details (e.g. through Gaussian
201   filtering) and uses an edge-aware filter to iteratively restore larger scale edges. Whereas RGF
202   iteratively filters the (fixed) input image while iteratively restoring the initially blurred
203   guidance image, SiR does the opposite: it gradually restores the initially blurred image using
204   the original input image as (fixed) guidance. Thereto, small details are initially removed
205   through blurring with a Gaussian kernel:

206
$$G_i^0 = \frac{1}{K} \sum_{j \in \Omega} I_j \cdot f(\| i - j \|) \tag{12}$$

207   Then, the edges of the remaining details are iteratively restored by repeatedly computing an
208   updated image G^{t+1} by joint bilateral filtering of the blurred image G^t using the input image
209   I as the guidance image:

210
$$G_i^{t+1} = \frac{1}{K_i} \sum_{j \in \Omega} G_j^t \cdot f(\| i - j \|) \cdot g(\| I_i - I_j \|) \tag{13}$$

211   Similar to RGF, SiR converges rapidly after only a few (3-5) iterations steps. Compared to
212   RGF, SiR has the advantage that it produces edges with articulated curvature, since it
213   attempts to restore them as similar as possible to the edges in the original image. In contrast,
214   the curvature of edges produced by RGF is smoothed (less articulated).
215
216   SiR also has two drawbacks (Kniefacz & Kropatsch, 2015). First, the overall intensity of the
217   result is less than the overall intensity of the original image, since SiR restores edges starting
218   from the initially blurred input image. RGF does not have this problem because it operates on

219   the original image itself. Second, SiR tends to restore small scale details that are close to
220   large scale edges. This is a result of a spill-over effect (blurring) of the large scale edges
221   which enables the JBF (Equation 13) to restore small scale details guided by the original
222   input image. As will be shown in Section 4.1 this restoration of small details near larger
223   edges by SiR can effectively be suppressed by applying a median filter at the end of each
224   iteration step with a kernel that is smaller than the Gaussian kernel of the bilateral spatial
225   filter (e.g., 3×3). At each iteration step the median filer "cleans" the image by removing
226   small details that are recovered by the bilateral filter. In the rest of this paper we will refer to
227   this modification of the original SiR algorithm as SiRmed (Smooth and iteratively Restore
228   including median filtering). The pseudocode for the SiRmed algorithm is presented in Table
229   2. The pseudocode for the SiR algorithm is obtained by omitting Step 5 (the median filter).
230

3 Alternating Guided Filter

232
233   In this section we propose a new filter scheme that effectively integrates the RGF and SiR
234   frameworks in such a way that it retains the desired effects of both earlier schemes (i.e., the
235   elimination of small details and restoration of large scale edges) while eliminating their
236   respective drawbacks (i.e., the smoothing of the curvature of large scale edges by RGF and
237   the loss of local image intensity by SiR). Each iteration in the AGF framework consists of
238   three consecutive steps. First, joint bilateral filtering is applied to the input image I, using the
239   result G^t of the previous iteration step as the guidance image (Equation (11). Second, joint
240   bilateral filtering is applied to the result from the first step, using the original image I as the
241   guidance image (Equation 13).Third, a median filter with a small kernel size (e.g., 3×3) is
242   applied to the result of the second step. The alternating use of the original image and the
243   filtered result as either input or guidance images guarantees that both the overall image
244   intensity (positive effect of the RGF scheme) and the curvature of the large scale edges
245   (positive effect of the SiR scheme) are preserved, while the integration of the median filter
246   prevents the reintroduction of filtered small scale details near large scale edges (a negative
247   side effect of SiR). Hence, the AGF scheme combines the positive effects of both the RGF
248   (intensity preservation) and SiR (edge curvature preservation) while eliminating their
249   negative side effects (edge curvature smoothing by the RGF scheme and contrast reduction
250   plus the reintroduction of small details near larger edges by the SiR scheme). The pseudo
251   code for the AGF scheme is presented in Table 3.

4 Experimental results

253
254   In this section we present the results of some experiments that were performed to study the
255   performance of the proposed AGF framework. We tested AGF by applying it to a wide range
256   of different natural images (for an overview of all results see the Supplemental Information
257   with this paper). Also, we compare the performance of AGF with that of RGF and SiRmed.
258
259   The images used in this study have a width of 638 pixels, and a height that varies between
260   337 and 894 pixels. We selected a set of test images with widely varying features and
261   statistics, such as landscapes and aerial images, portraits and mosaics, embroidery and
262   patchwork. The full set of test images with the results of the different filter schemes are
263   provided as Supplemental Information.
264
265   In this study, color images are processed by applying the different algorithms to each channel
266   in RGB color space independently. This was done to enable straightforward comparison with

267 the results from previous studies (Kniefacz & Kropatsch, 2015; Zhang et al., 2014). Note that
268 in practical applications, filtering should preferably be performed in the CIE-Lab color space,
269 so that only perceptually similar colors are averaged and only perceptually significant edges
270 are preserved.
271
272 As noted before, a bilateral filter (see Equation 1) is controlled by two parameters: the size of
273 the spatial filter kernel ($\sigma_{spatial}$) and that of the range filter kernel (σ_{range}). Except when stated
274 otherwise, we used the same constant values for the spatial and range kernels in the bilateral
275 filters that are included in all three frameworks investigated in this study (i.e., RGF, SiRmed
276 and AGF): $\sigma_{spatial} = 5$ and $\sigma_{range} = 0.05$. These values were empirically determined and
277 resulted in an effective performance for all three frameworks on the entire set of selected test
278 images.
279

280 *4.1 Relative performance of SiR and SiRmed*
281
282 Figures 1 and 2 illustrate the effect of including a median filter with a kernel size of 3×3 at
283 the end of each iteration step in the SiR framework (see Table 2). Figure 1 shows the original
284 input image (a) with the result of respectively SiR (Figure 1d) and SiRmed (Figure 1e) after 5
285 iterations. In addition, this figure also shows the 1D image intensity distribution along a
286 horizontal cross section of the input image (yellow line in Figure 1a) after each of the first 3
287 iterations of both SiR (Figure 1c) and SiRmed (Figure 1d). The lowest curves represent the
288 original intensity distribution (Figure 1a). Comparison of Figure 1d and Figure 1e shows that
289 SiRmed effectively removes small details all over the image, while SiR reintroduces small
290 details near large scale edges (notice for instance the light poles all over the stadium, that are
291 restored in Figure 1d but are nicely removed in Figure 1e). A comparison of Figure 1b and
292 Figure 1c shows that SiR indeed restores high frequency details near large edge
293 discontinuities, while SiRmed restores these large scale edges without the reintroduction of
294 small scale details. This effect is also clearly demonstrated in Figure 2, where SiR fails to
295 remove the small scale details all around the outlines of the tentacles and the fishes, while
296 SiRmed effectively filters small elements all around these larger objects while both
297 preserving their outlines and smoothing their interior.

298

299 *4.2 Relative performance of RGF, SiRmed and AGF*
300
301 Figure 3 shows the results of RGF, SiRmed and AGF filtering after the first 3 iteration steps.
302 Notice that all three filter schemes iteratively restore large scale image edges proceeding
303 from an initial low resolution version of the input image. In this process RGF gradually
304 smoothes the curvature of the large scale edges, while SiRmed smoothes the overall image
305 intensity resulting in a global contrast reduction. In contrast, AGF restores the large scale
306 edges without smoothing their curvature and with preservation of local image contrast.
307 Figures 4b-d illustrate these effects by showing 1D horizontal cross sections. The contrast
308 reducing effect of SiR is clearly shown in Figure 4c as an overall reduction of the height of
309 the large scale peaks. Comparison of Figure 4e and Figure 4g illustrates that RGF smoothes
310 the curvature of large scale edges while AGF preserves edge curvature.

311 Figure 5 shows the details that are removed by filtering with respectively RGF (Figure 5b),
312 SiRmed (Figure 5c) and AGF (Figure 5d) after 5 iterations, together with the original input

313   image (Figure 5a). These images are obtained by subtracting the original image from the
314   filtered ones and computing the mean of the absolute value across the three color channels.
315   This example illustrates that each of the three filters removes small scale details while
316   preserving the larger scale edges to a certain extent. Note that AGF removes small image
317   details all over the image support, while the original large scale edges (e.g., the outlines of the
318   temple) remain relatively unaffected. In contrast, both RGF and SiRmed significantly alter
319   the articulation of the original large scale edges. RGF smoothes the local edge curvature
320   while SiRmed affects the local mean image intensity (resulting in local contrast reduction).
321   As a result, large scale image contours (e.g., the outlines of the temple) are clearly visible in
322   Figure 5b and Figure 5c, and much less pronounced in Figure 5d.
323
324   Figure 6 further illustrates the differential performance of all four filters investigated here
325   (AGF, RGF, SiR, and SiRmed) on an artificial image with noise. Figure 6a shows some
326   simple geometric shapes (cross, triangle, square and wheel) with different sizes on a step-
327   edge background with 30% additional Gaussian noise. Figures 6b-e show the results of
328   respectively AGF, RGF, SiR and SiRmed filtering of Figure 6a after five iteration steps. This
329   example shows that SiR restores noise near the edges of larger details and along the vertical
330   step edge in the background, while AGF, RGF and SiRmed effectively reduce noise all over
331   the image plane. Also, RGF (Figure 6c) smoothes edge curvature, in contrast to AGF, SiR
332   and SiRmed which retain the original edge curvature. This can for instance be seen in Figure
333   6c where the sharp edges of the crosses, the corners of the triangles and squares, and the
334   spokes of the wheels are all rounded after filtering. This example also illustrates that both SiR
335   (Figure 6d) and SiRmed (Figure 6e) effectively reduce image contrast due to the intensity
336   smoothing that is inherent in this method. AGF does not introduce any halos. In contrast,
337   RGF produces high intensity halos, and SiR and SiRmed both produce halos with a large
338   spatial extent (see Figure 6c near the crosses and wheels).

4.3 Effects of different parameter settings

341   The results of RGF, SiRmed and AGF for different values of the parameters σ_s and σ_r are
342   shown in Figures 7-9. In these figures rows correspond to different amounts of domain
343   filtering (i.e., different values of σ_s) and columns to different amounts of range filtering (i.e.,
344   different values of σ_r). These figures show that large spatial kernels and large range kernels
345   both result in blurred image representations. Large spatial kernels cause averaging over large
346   image areas, while large range kernels cause averaging over a wider range of image values.
347
348   In RGF (Figure 7) the smoothing of large scale edge curvature increases with increasing σ_s.
349   This effect can for instance clearly be seen in the first column of Figure 7 (for $\sigma_r = 0.05$)
350   where the curves in the roof of the temple just above the pillars are still visible for $\sigma_s = 3$,
351   reduced for $\sigma_s = 6$ and removed for $\sigma_s = 9$.
352
353   In SiRmed (Figure 8) the reduction of local image contrast increases with increasing σ_s. This
354   effect can for instance clearly be seen in the first column of Figure 8 (for $\sigma_r = 0.05$) where the
355   contrast in the background vegetation around the temple is still visible for $\sigma_s = 3$, reduced for
356     $\sigma_s = 6$ and almost removed for $\sigma_s = 9$.
357
358   Figure 9 shows the effects of different parameter settings on the performance of the proposed
359   AGF framework. In contrast to RGF, AGF preserves large scale edge curvature, as can be

360   seen in the first column of Figure 9 (for $\sigma_r = 0.05$). Notice that the curves in the roof of the
361   temple just above the pillars remain unaffected, even for the largest size of the spatial kernel
362   ($\sigma_s = 9$). In contrast to SiRmed, AGF also preserves local image intensity (contrast).
363

364   *4.4 Runtime evaluation*
365

366   In this study we used a Matlab implementation of the RGF written by Zhang et al. (2014) that
367   is freely available from the authors (at
368   http://www.cs.cuhk.edu.hk/~leojia/projects/rollguidance). The JBF, which is part of this
369   code, is also used here to implement both the SiR filter (see Section 2.1) and the AGF filter
370   (see Section 3). We made no effort to optimize the code of the algorithms. We conducted a
371   runtime test on a Dell Latitude laptop with an Intel i5 2 GHz CPU and 8 GB memory. The
372   algorithms were implemented in Matlab 2016a. Only a single thread was used without
373   involving any SIMD instructions. For this test we used a set of 22 natural RGB images (also
374   provided in the Supplemental Information with this paper). The spatial and range kernels
375   used in the bilateral filters for this test were fixed at respectively $\sigma_{spatial} = 5$ and $\sigma_{range} = 0.05$,
376   and the number of iterations was set to 5. The mean runtimes for AGF, RGF, SiR and
377   SiRmed were respectively 3.80±0.27, 1.64±0.12, 1.80±0.14 and 2.17±0.16 seconds.
378   This shows that the mean runtime of AGF is about equal to the sum of the runtimes of RGF
379   and SiRmed. This result is as expected, since the steps involved in AGF are a combination of
380   the steps involved in both RGF and SiRmed.
381
382

383   **5 Conclusions**
384

385   In this paper we presented the new edge-preserving Alternating Guided Filter (AGF)
386   smoothing filter that eliminates small scale image details while preserving both the
387   articulation and curvature of large scale edges and local mean image intensity. The AGF
388   framework integrates the recently introduced RGF and (a slightly adapted version of) SiR
389   filters in an alternating iterative scheme. AGF combines the large scale edge and local
390   intensity preserving image smoothing properties of the RGF with the large scale edge
391   restoring properties of the SiR. However, it does not suffer from the drawbacks of both of its
392   components: the curvature smoothing of large scale edges by RGF and the local intensity
393   reduction and restoration of small scale details near large scale edges by SiR. The AGF is
394   simple to implement and efficient. Application to a wide range of images has shown that
395   AGF consistently produces high-quality results. Possible applications areas for AGF are for
396   instance detail enhancement, denoising, edge extraction, JPEG artifact removal, multi-scale
397   structure decomposition, saliency detection etc. (see e.g. He, Sun & Tang, 2013; Kniefacz &
398   Kropatsch, 2015; Zhang et al., 2014).
399
400

401

402

403

References

405

406

Black, M.J., Sapiro, G., Marimont, D.H. & Heeger, D. 1998. Robust anisotropic diffusion. *IEEE Transactions on Image Processing, 7(3),* 421-432. DOI 10.1109/83.661192.

Farbman, Z., Fattal, R., Lischinski, D. & Szeliski, R. 2008. Edge-preserving decompositions for multi-scale tone and detail manipulation. *ACM Transactions on Graphics, 27(3 - Article No. 67),* 1. DOI 10.1145/1360612.1360666.

He, K., Sun, J. & Tang, X. 2013. Guided image filtering. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35(6),* 1397-1409. DOI 10.1109/TPAMI.2012.213.

Kniefacz, P. & Kropatsch, W. 2015. Smooth and iteratively Restore: A simple and fast edge-preserving smoothing model. In S. Hegenbart, R. Kwitt & A. Uhl (Eds.), *Proceedings of The 39th Annual Workshop of the Austrian Association for Pattern Recognition (OAGM 2015),* eprint arXiv:1505.06702 (pp. 1-9).

Paris, S. & Durand, F. 2006. A fast approximation of the bilateral filter using a signal processing approach. In A. Leonardis, H. Bischof & A. Pinz (Eds.), *Proceedings of the 9th European Conference on Computer Vision (ECCV 2006),* Part IV (pp. 568-580). Berlin, Heidelberg: Springer Berlin Heidelberg. DOI 10.1007/11744085_44.

Perona, P. & Malik, J. 1990. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 12(7),* 629-639. DOI 10.1109/34.56205.

Petschnigg, G., Agrawala, M., Hoppe, H., Szeliski, R., Cohen, M. & Toyama, K. 2004. Digital photography with flash and no-flash image pairs. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2004),* (pp. 664-672). New York, USA: ACM Press. DOI 10.1145/1015706.1015777.

Tomasi, C. & Manduchi, R. 1998. Bilateral filtering for gray and color images. *Proceedings of the IEEE Sixth International Conference on Computer Vision,* (pp. 839-846). Piscataway, NJ, USA: IEEE. DOI 10.1109/ICCV.1998.710815.

Zhang, Q., Shen, X., Xu, L. & Jia, J. 2014. Rolling guidance filter. In D. Fleet, T. Pajdla, B. Schiele & T. Tuytelaars (Eds.), *Computer Vision − ECCV 2014: 13th European Conference,* Part III (pp. 815-830). Cham, Switzerland: Springer International Publishing. DOI 10.1007/978-3-319-10578-9_53.

436

437

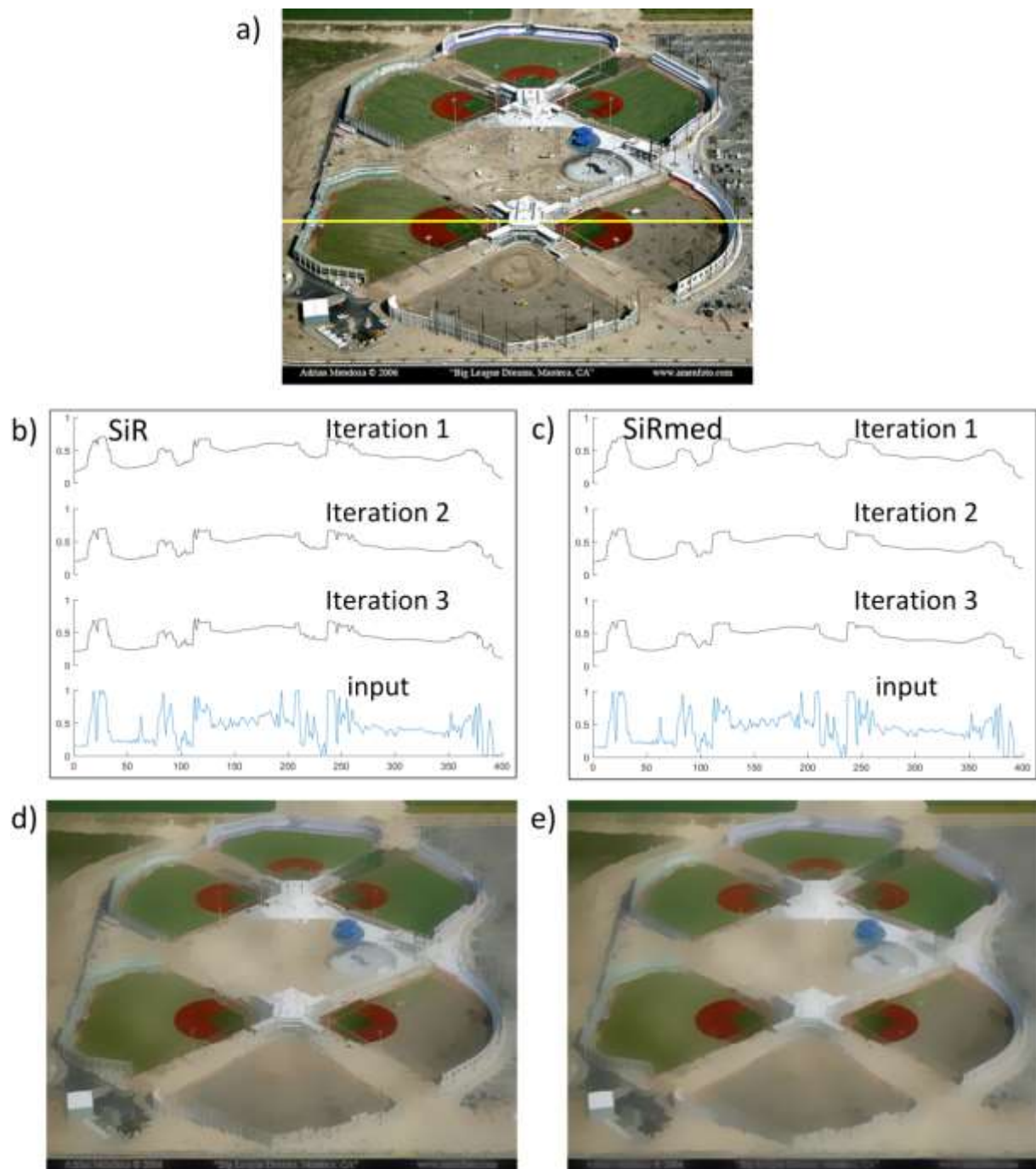
438

Figures

Figure 1. **Effects of SiR and SiRmed filtering on image intensity**.

(a) Input image (Photo credit: Adrian Mendoza). (b,c) 1D Image intensity distribution along a horizontal cross section of (a) (yellow line) after the first 3 iterations of respectively SiR and SiRmed. The lowest curves show the original intensity distribution in (a). (d,e) Results after 5 iterations of respectively SiR and SiRmed. The spatial and range kernels used in the bilateral filters for this example were respectively $\sigma_{spatial} = 5$ and $\sigma_{range} = 0.05$.

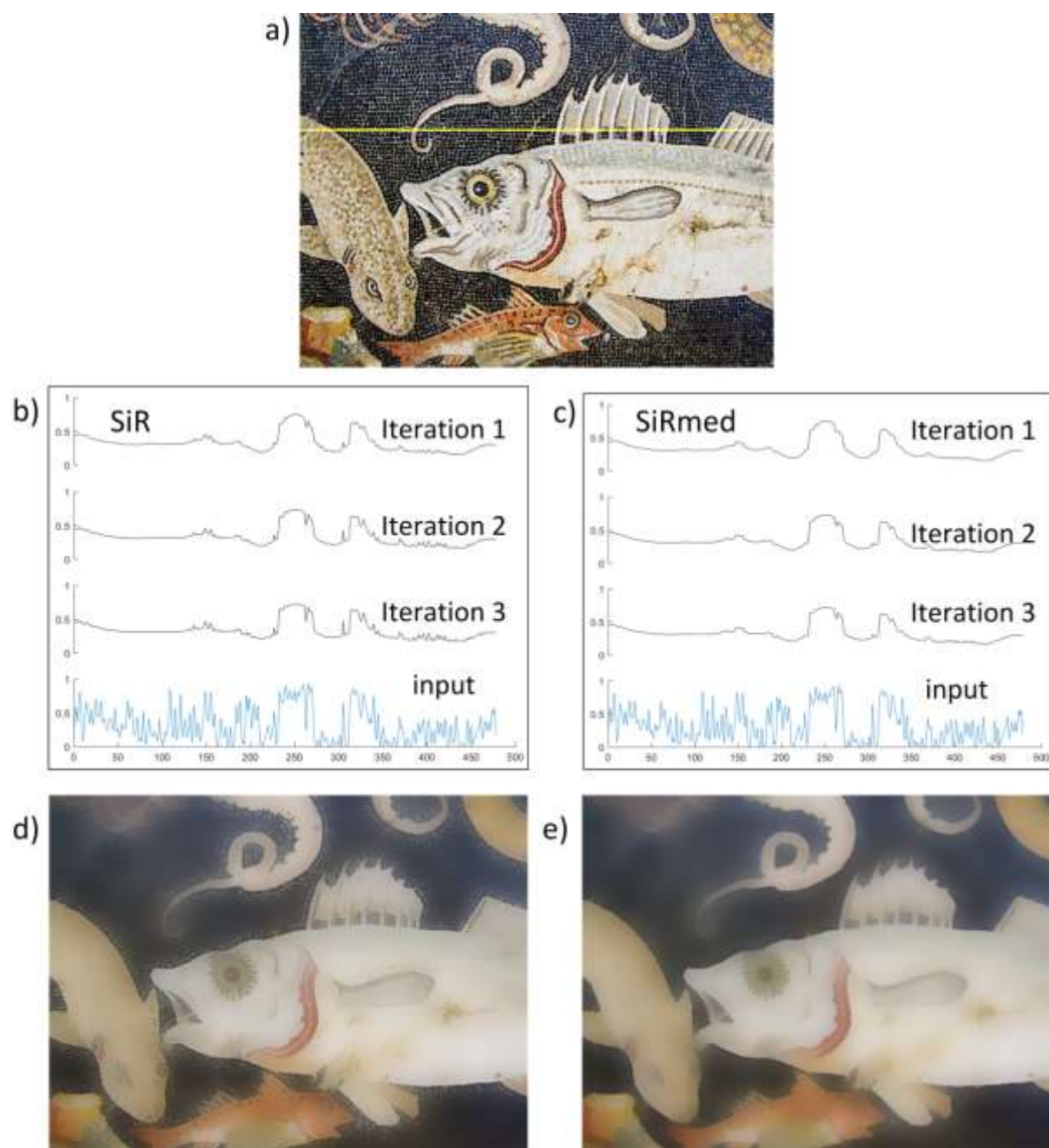



Figure 2. **Effects of SiR and SiRmed filtering on image intensity**

As Figure 1.

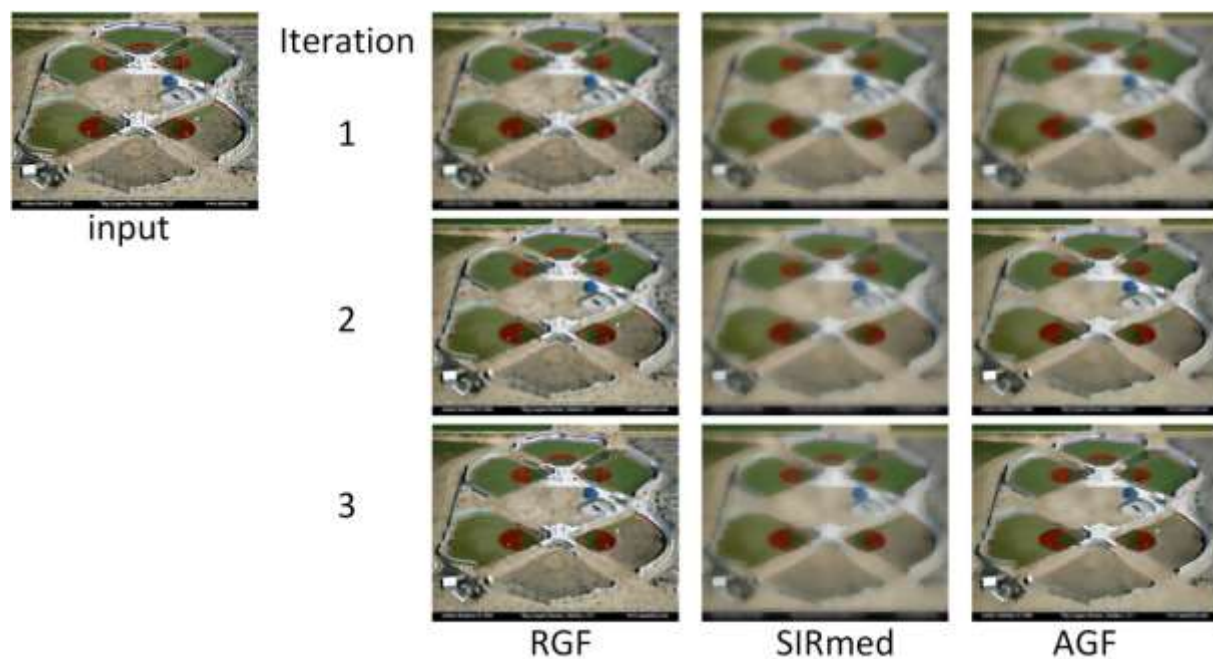



Figure 3. **Results of RGF, SiRmed and AGF filtering**.

The input image is shown on the upper left (Photo credit: Adrian Mendoza). Results are shown for the first 3 iteration steps. The spatial and range kernels used in the bilateral filters for this example were respectively $\sigma_{spatial} = 5$ and $\sigma_{range} = 0.05$.

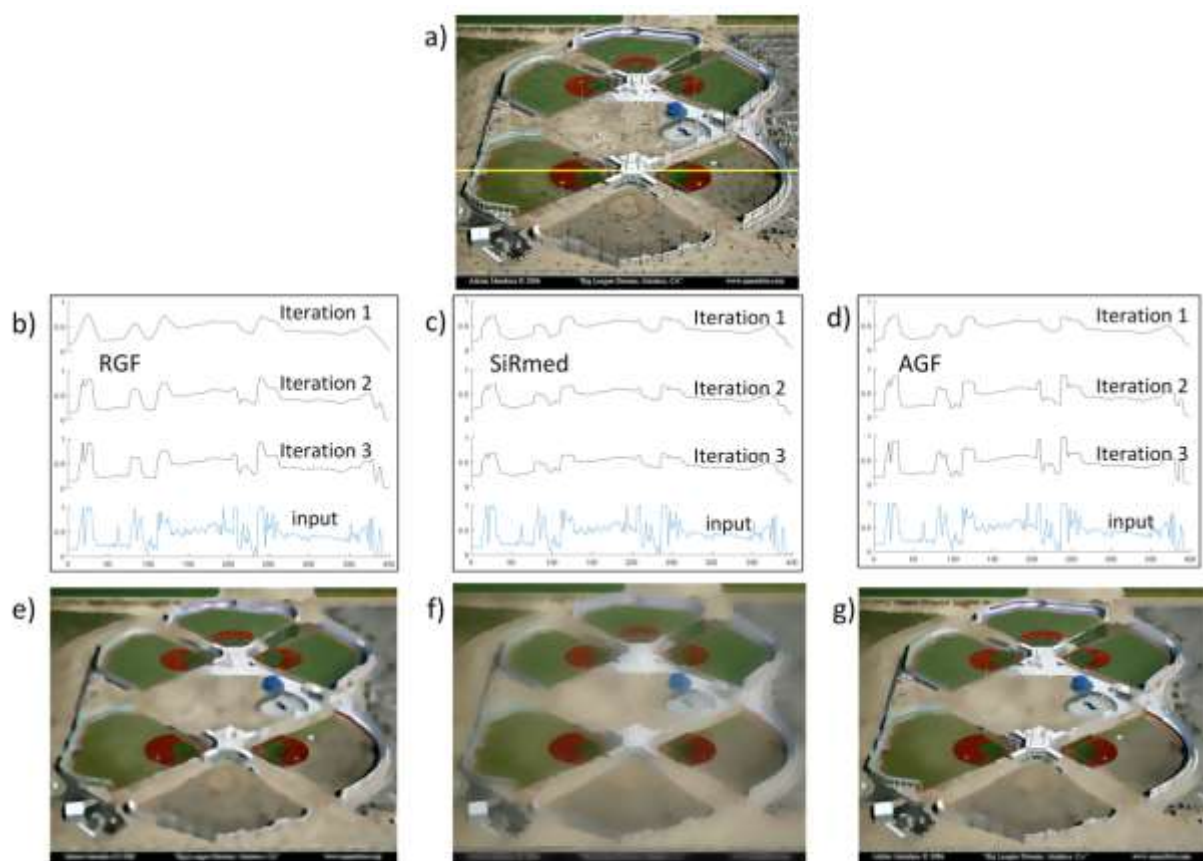



Figure 4. **Comparison of RGF, SiRmed and AGF filtering**.

(a) Input image (Photo credit: Adrian Mendoza). (b-d) 1D Image intensity distribution along a horizontal cross section (yellow line in (a)) after 3 successive iterations. (e-g) Results after 5 iteration steps. The spatial and range kernels used in the bilateral filters for this example were respectively $\sigma_{spatial} = 5$ and $\sigma_{range} = 0.05$.

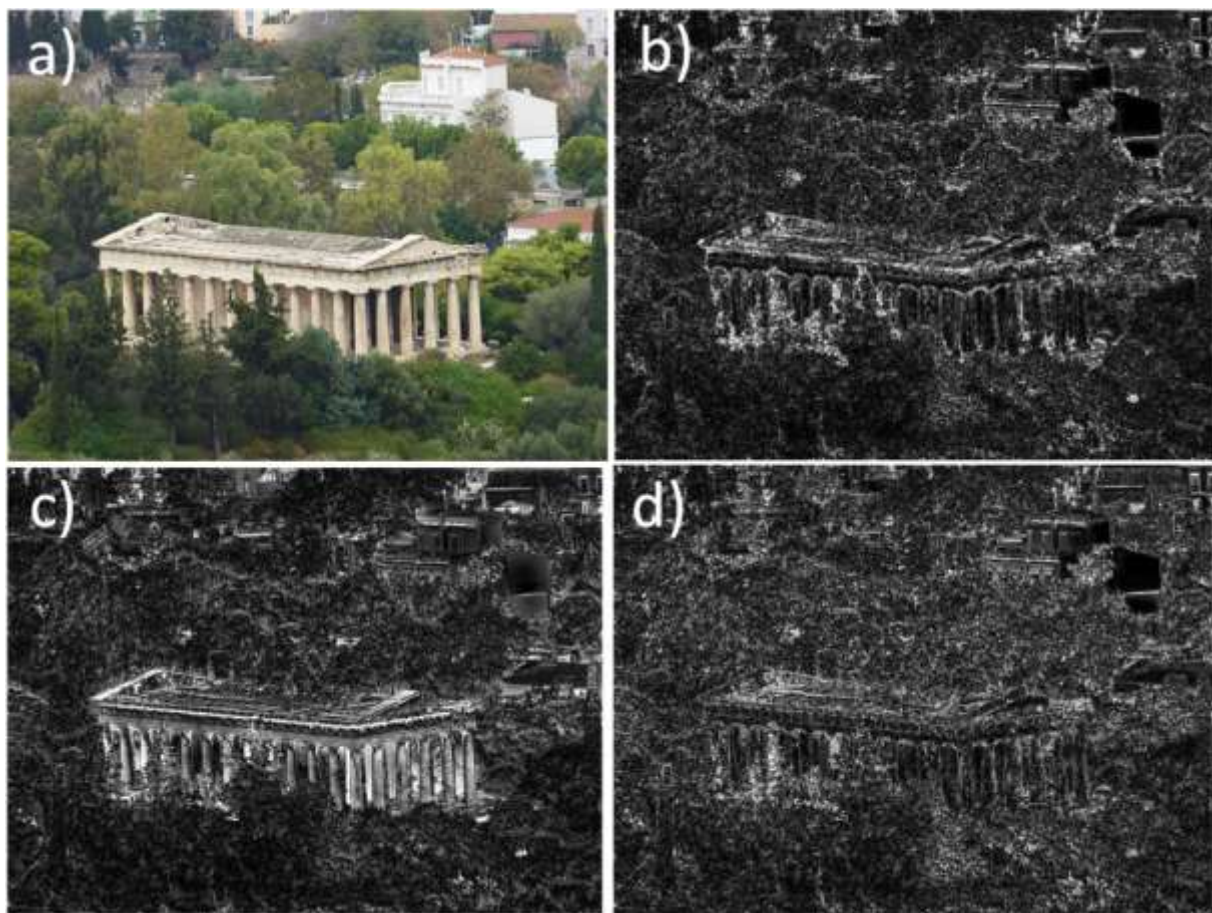



Figure 5. **The elimination of small scale details by RGF, SiRmed and AGF filtering.**

Absolute difference between input image (a) and the results of respectively RGF (b), SiRmed (c) and AGF (d) after 5 iterations. The images have been enhanced for visual display. The spatial and range kernels used in the bilateral filters for this example were respectively $\sigma_{spatial}$ = 5 and σ_{range} = 0.05.

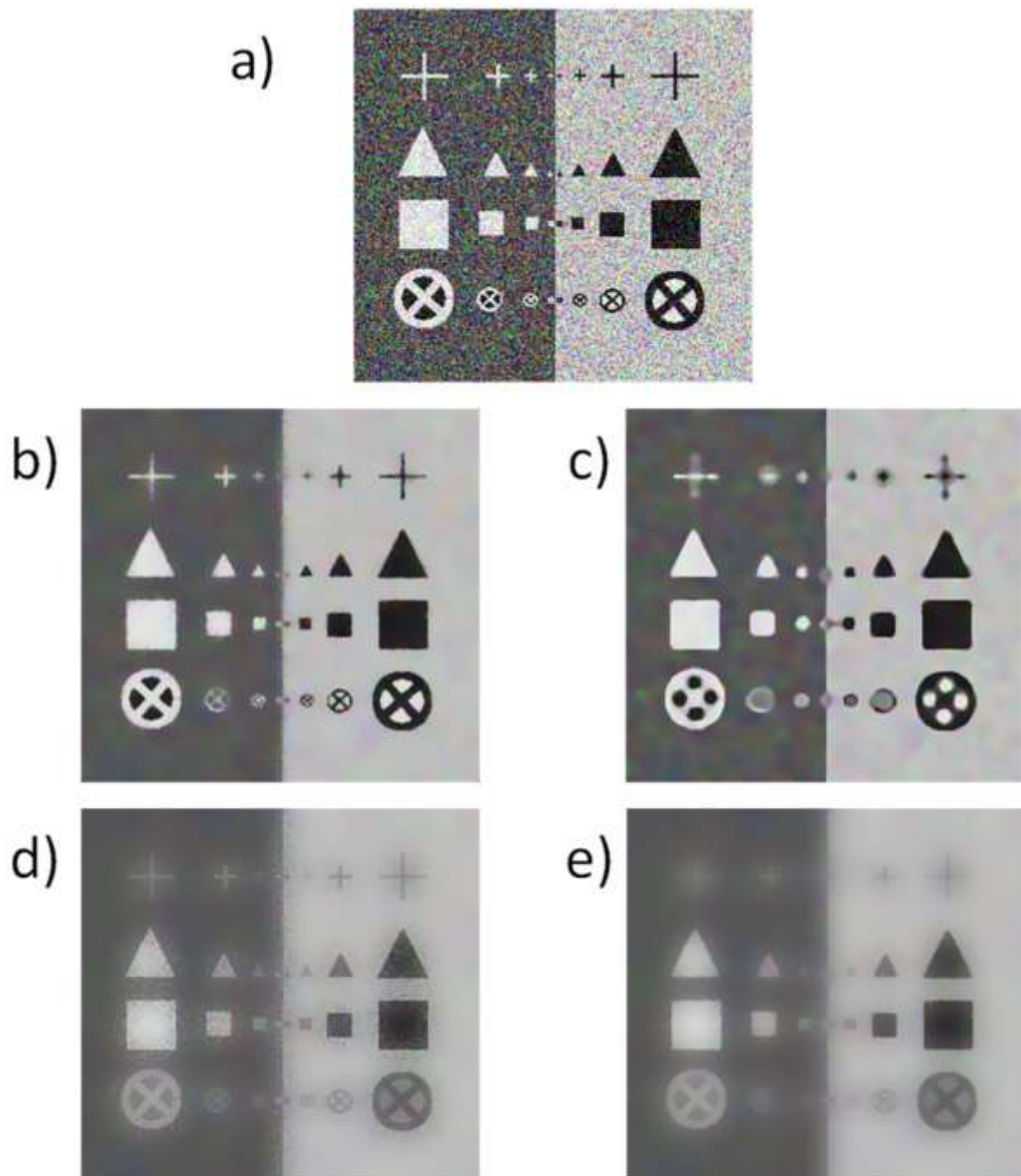



Figure 6. **Comparison of AGF, RGF, SiR and SiRmed filtering**.

(a) Artificial input image with 30% Gaussian noise added. Results of (b) AGF, (c) RGF, (d) SiR and (e) SiRmed filtering after 5 iteration steps. The spatial and range kernels used in the bilateral filters for this example were respectively $\sigma_{spatial} = 5$ and $\sigma_{range} = 0.05$.

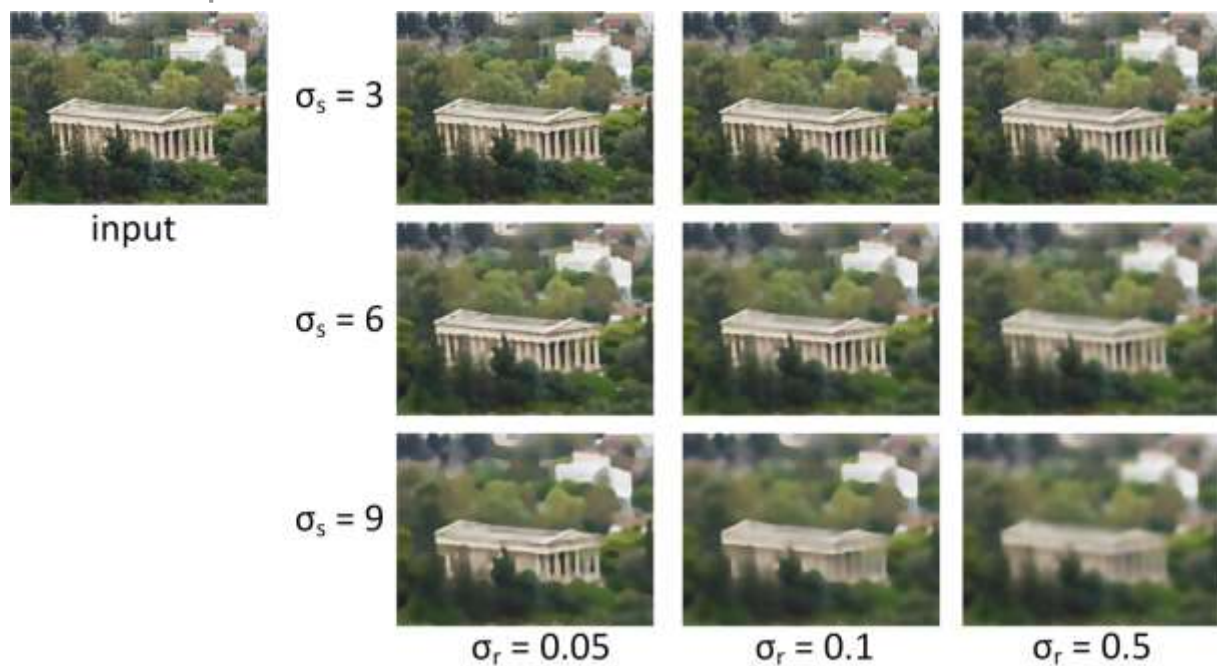



Figure 7. **The effects of different parameter settings on RGF filtering.**

Results of RGF filtering of the input image on the upper left for different values of the variances of the spatial ($\sigma_s = 3, 6$ and 9) and range ($\sigma_r = 0.05, 0.1$ and 0.5) filters.

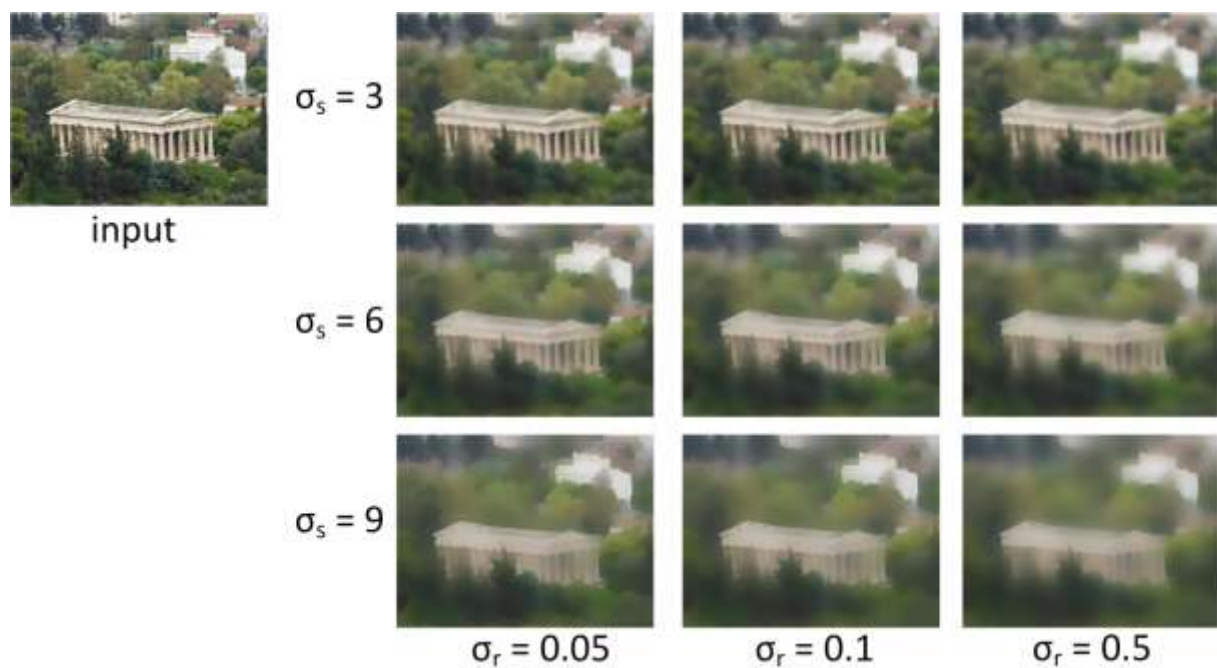



Figure 8. The effects of different parameter settings on SiRmed filtering.

Results of SiRmed filtering of the input image on the upper left for different values of the variances of the spatial ($\sigma_s = 3, 6$ and 9) and range ($\sigma_r = 0.05, 0.1$ and 0.5) filters.

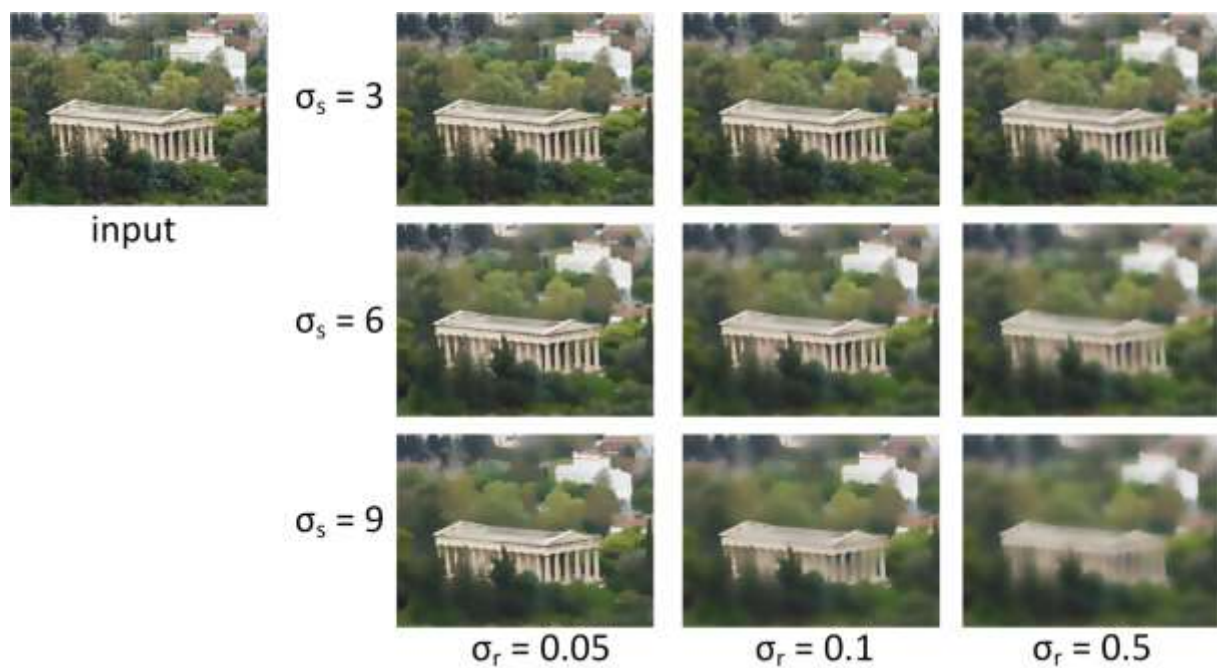



Figure 9. **The effects of different parameter settings on AGF filtering.**

Results of AGF filtering of the input image on the upper left for different values of the variances of the spatial ($\sigma_s = 3, 6$ and 9) and range ($\sigma_r = 0.05, 0.1$ and 0.5) filters.

Tables

Table 1. Rolling Guidance Filter (RGF) algorithm (from (Zhang et al., 2014)).

Rolling Guidance Filter (RGF) algorithm	
Input:	input image I , σ_{spatial} , σ_{range} , N^{iter}
Output:	output image O
Procedure:	1: Initialize G^0 as a constant image
	2: **for** t:=1 **to** N^{iter} **do**
	3: $G^t \leftarrow \text{JointBilateralFilter}(I, G^{t-1}, \sigma_{spatial}, \sigma_{range})$ {Input: I ; Guidance: G^{t-1}}
	4: **end for**
	5: $O \leftarrow G^{N^{iter}}$

Table 2. Smooth and iteratively Restore with median filtering (SiRmed) algorithm (adapted from (Kniefacz & Kropatsch, 2015)).

Smooth and iteratively Restore (SiR) algorithm	
Input:	input image I , σ_{spatial} , σ_{range} , N^{iter}
Output:	output image O
Procedure:	1: Initialize G^0 as a Gaussian filtered version of the input image I
	2: **for** t:=1 **to** N^{iter} **do**
	3: $G^t \leftarrow \text{JointBilateralFilter}(G^{t-1}, I, \sigma_{spatial}, \sigma_{range})$ {Input: G^{t-1} ; Guidance: I}
	4: $G^t \leftarrow \text{MedianFilter}(G^t)$
	5: **end for**
	6: $O \leftarrow G^{N^{iter}}$

Table 3. Alternating Guided Filter (AGF) algorithm.

Alternating Guided Filter (AGF) algorithm	
Input:	input image I , σ_{spatial} , σ_{range} , N^{iter}
Output:	output image O
Procedure:	1: Initialize G^0 as a constant image
	2: **for** t:=1 **to** N^{iter} **do**
	3: $G^t \leftarrow \text{JointBilateralFilter}(I, G^{t-1}, \sigma_{spatial}, \sigma_{range})$ {Input: I ; Guidance: G^{t-1}}
	4: $G^t \leftarrow \text{JointBilateralFilter}(G^t, I, \sigma_{spatial}, \sigma_{range})$ {Input: G^t ; Guidance: I}
	5: $G^t \leftarrow \text{MedianFilter}(G^t)$
	6: **end for**
	7: $O \leftarrow G^{N^{iter}}$