



**HAL**  
open science

## Image filtering using morphological amoebas

Romain Lerallut,, Etienne Decencière, Fernand Meyer

► **To cite this version:**

Romain Lerallut,, Etienne Decencière, Fernand Meyer. Image filtering using morphological amoebas. Image and Vision Computing, 2007, 25 (4), pp.395-404. 10.1016/j.imavis.2006.04.018 . hal-01431825

**HAL Id: hal-01431825**

**<https://minesparis-psl.hal.science/hal-01431825>**

Submitted on 25 Jan 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# ISMM05 Special Issue: Image filtering using morphological amoebas

Romain Lerallut<sup>\*</sup>, Étienne Decencière, Fernand Meyer

*Centre de Morphologie Mathématique, École des Mines de Paris  
35 rue Saint-Honoré, 77305 Fontainebleau, France*

---

## Abstract

This paper presents morphological operators with non-fixed shape kernels, or amoebas, which take into account the image contour variations to adapt their shape. Experiments on grayscale and color images demonstrate that these novel filters outperform classical morphological operations with a fixed, space-invariant structuring element for noise reduction applications. Proof-of-concepts tests are then performed on 3D images to show the high noise-reduction capacity of amoeba-based filters.

*Key words:* Anisotropic filters, noise reduction, morphological filters, color filters, 3D image processing

---

## 1 Introduction

Noise is possibly the most annoying problem in the field of image processing. There are two ways to work around it: either design particularly robust algorithms that can work in noisy environments, or try to eliminate the noise in a first step while losing as little relevant information as possible and consequently use a normally robust algorithm.

There are of course many algorithms that aim at reducing the amount of noise in images. Most are quite effective but also often remove thin elements such as canals or peninsulas. Even worse, they can displace the contours and thus create additional problems in a segmentation application.

---

<sup>\*</sup> Corresponding Author.

*Email address:* lerallut@cmm.ensmp.fr (Romain Lerallut).

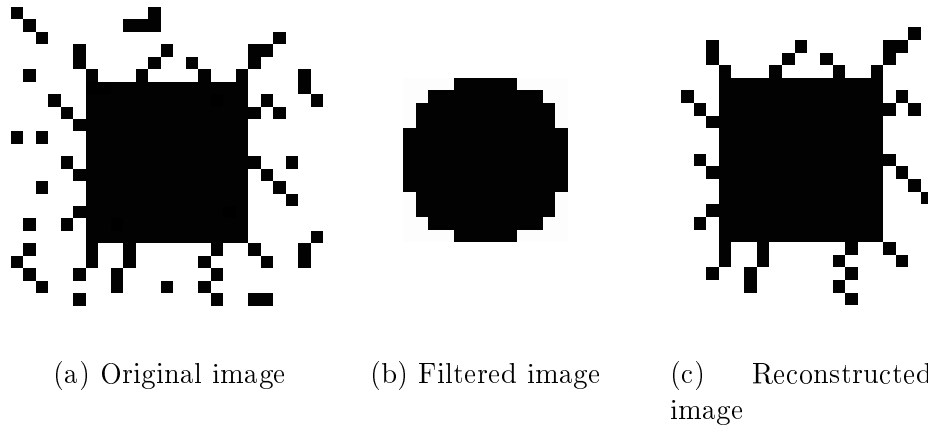


Fig. 1. Classic noise filtering (b) removes much contour information. Reconstruction (c) finds not only the contours, but also all the noise connected to the object.

In mathematical morphology we often couple one of these noise-reduction filters to a reconstruction filter that attempts to reconstruct only relevant information, such as contours, and not noise. However, a faithful reconstruction can be problematic when the contour itself is corrupted by noise. This can cause great problems in some applications which rely heavily on clean contour surfaces, such as 3D visualization, so a novel approach was proposed: morphological amoebas.



An amoeba (here *Amoeba proteus*) is a genus of protozoa that moves by projecting pseudopods and is a well-known representative unicellular organism. They are found in sluggish waters all over the world, both fresh and salt, as well as in soils and as parasites. They now begin a new life in the field of image processing.

## 2 Amoebas: dynamic structuring elements

### *Principle*

#### *Classic filter kernel*

Formally at least, classic filters work on a fixed-size sliding window, be they morphological operators (erosion, dilation) or convolution filters, such as the diffusion by a Gaussian. If the shape of that window does not adapt itself to

the content of the image (see figure 2), the results deteriorate. For instance, an isotropic Gaussian diffusion smooths the contours when its kernel steps over a strong gradient area.

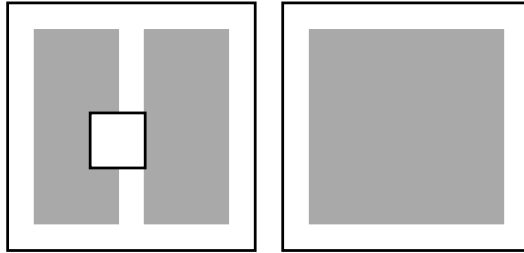


Fig. 2. Closing of an image by a large structuring element. The structuring element does not adapt its shape and merges two distinct objects.

### *Amoeba filter kernel*

Having made this observation, Perona and Malik [1] (and others after them) have developed anisotropic filters that inhibit diffusion through strong gradients. The first work on non-fixed shape structuring elements was done by Braga-Neto in [7] though it was restricted to alternate sequential filters of openings and closings. We were inspired by these examples to define morphological filters whose kernels adapt to the content of the image in order to keep a certain homogeneousness inside each structuring element (see figure 3) while at the same time keeping their size in check. Tomasi and Manduchi have described in [5] the idea of coupling performed a geometric distance between pixels and a distance between their values, which offers remarkable properties for our intended use.

The interest of this approach, compared to the analytical one is that it does not depart greatly from what we use in mathematical morphology, and therefore most of our algorithms can be made to use amoebas with little additional work. Most of the underlying theoretical groundwork for the morphological approach has been described by Jean Serra in his study [2] of structuring functions, although until now it has seen little practical use.

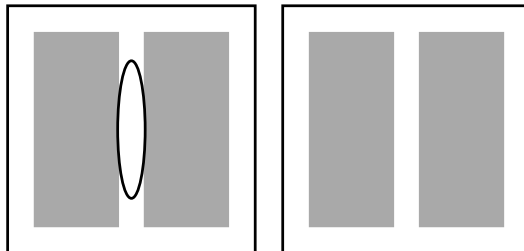


Fig. 3. Closing of an image by an amoeba. The amoeba does not cross the contour and as such preserves even the small canals.

The shape of the amoeba must be computed for each pixel around which it is

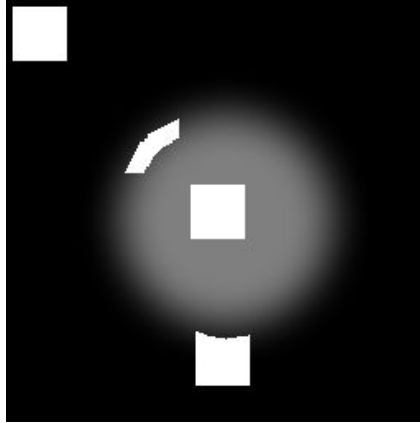


Fig. 4. Shape of an amoeba at various positions on an image.

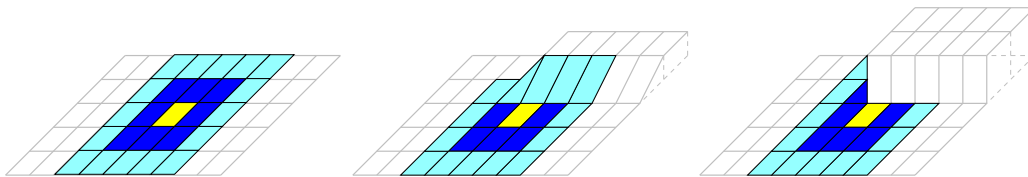


Fig. 5. Behavior of an amoeba on various relief types. Strong gradients should slow or even hamper the growth of the amoeba

centered. Figure 4 shows the shape of an amoeba depending on the position of its center. Note that in flat areas such as the center of the disc, or the background, the amoeba is maximally stretched, while it is reluctant to cross contour lines.

When an amoeba has been defined, most morphological operators and many other types of filters can be used on it: median, mean, rank filters, erosion, dilation, opening, closing, even more complex algorithms such as reconstruction filters, levelings, floodings, etc.

### *Construction*

#### *Amoeba distance*

In general, a filtering kernel of radius  $r$  is formally defined on a square (or a hexagon) of that radius, that is to say on the ball of radius  $r$  relative to the norm associated to the chosen connectivity. We will keep this definition changing only the norm, using one that takes into account the gradient of the image, so that we get the behavior described in figure 5.

**Definition 1** *Let  $d_{pixel}$  be a distance defined between the values of the image, for example a difference of gray-value, or a color distance.*

Let  $\sigma = (x = x_0, x_1, \dots, x_n = y)$  a path between points  $x$  and  $y$ . Let  $\lambda$  be a real positive number. The length of the path  $\sigma$  is defined as

$$L(\sigma) = \sum_{i=0}^{n-1} [1 + \lambda \cdot d_{\text{pixel}}(\text{Image}(x_i), \text{Image}(x_{i+1}))]$$

The “amoeba distance” with parameter  $\lambda$  is thus defined as:

$$\begin{cases} d_\lambda(x, x) = 0 \\ d_\lambda(x, y) = \min_\sigma L(\sigma), \text{ minimum taken on all paths between } x \text{ and } y \end{cases}$$

It is important to realize that  $d_{\text{pixel}}$  has no geometrical aspect, it is a distance computed only on the values of the pixels of the image. Furthermore, if  $n$  is the number of pixels of a path  $\sigma$ , then  $L(\sigma) \geq n$  (since  $\lambda \geq 0$ ), which bounds the maximal extension of the amoeba.

This distance also offers an interesting inclusion property:

**Property 1** *At a given radius  $r$  the family of the balls  $\mathcal{B}_{\lambda,r}$  relative to the distance  $d_\lambda$  is decreasing (for the inclusion),*

$$\begin{aligned} 0 \leq \lambda_1 \leq \lambda_2 &\Rightarrow \forall(x, y), d_{\lambda_1}(x, y) \leq d_{\lambda_2}(x, y) \\ &\Rightarrow \forall r \in \mathbf{R}^+, \mathcal{B}_{\lambda_1, r} \supset \mathcal{B}_{\lambda_2, r} \end{aligned}$$

Which may be useful when building hierarchies of filters, such as a family of alternate sequential filters with strong gradient-preserving properties.

*The pilot image*

We have found that the noise in the image can often distort the shape of the amoeba. For this reason, we often compute the *shape* of the amoeba on another image. Once the shape is computed, the values are sampled on the *original* image and processed by the filter (mean, median, max, min, ...). Usually, the other image is the result of a strong noise removal filtering of the original image that dampens the noise while preserving as much as possible the larger contours. A large Gaussian works fairly well, and can be applied very quickly with advanced algorithms, however we will see below that iterating amoeba filters yields even better results.

### 3 Amoebas in practice

#### *Adjunction*

Erosions and dilations can easily be defined on amoebas. However it is necessary to use *adjoint* erosions and dilations when using them to define openings and closings:

$$\begin{aligned}\delta(X) &= \bigcup_{x \in X} B_{\lambda,r}(x) \\ \epsilon(X) &= \{x/B_{\lambda,r}(x) \subset X\}\end{aligned}$$

These two operations are at the same time adjoint and relatively easy to compute, contrary to the symmetrical ones that use the transposition, which is not easy to compute for amoebas. See [2] for a discussion of the various forms of adjunction and transposition of structuring functions.

#### *Algorithms*

The algorithms used for the erosion and dilation are quite similar to those used with regular structuring elements, with the exception of the step of computing the shape of the amoeba.

The opening using these algorithms can be seen as the gray-level extension of the classic binary algorithm of first taking the centers of the circles that fit inside the shape (erosion), and then returning the union of all those circles (dilation). See [10] for a more detailed description of the algorithms used for adjoint erosion and dilation.

#### *Complexity*

The theoretical complexity of a simple amoeba-based filter (erosion, dilation, mean, median) can be asymptotically approximated by:

$$T(n, k, op) = O \left[ n * \left( op(k^d) + amoeba(k, d) \right) \right]$$

Where  $n$  is the number of pixels in the image,  $d$  is the dimensionality of the image (usually 2 or 3),  $k$  is the maximum radius of the amoeba,  $op(k^d)$  is the

cost of the operation and  $amoeba(k, d)$  is the cost of computing the shape of the amoeba for a given pixel.

The shape of the amoebas is computed by a common region-growing implementation using a priority queue. Depending on the priority queue used, the complexity of this operation is slightly more than  $O(k^d)$  (see [3] and [4] for advanced queueing data structures).

Therefore, for erosion, dilation or mean as operators, we have a complexity of a little more than  $O(n * k^d)$  which is the complexity of a filter on a fixed-shape kernel. It has indeed been verified in practice that, while being quite slower than with fixed-shape kernels (especially optimized ones), filters using amoebas tend to follow rather well the predicted complexity, and do not explode (tests have been performed on 3D images, size 512x512x100, with amoebas with sizes up to 21x21x21).

## 4 Results

### *Alternate sequential filters*

The images of figure 6 compare the differences between alternate sequential filters (ASF) built on classic fixed shape kernels and ASFs on amoebas in the filtering of the image of a retina. The filter should be able to reduce the amount of background noise while preserving the shape of the vessels.

### *Median and mean*

In the context of image enhancement, we have found that a simple mean or median coupled with an amoeba forms a very powerful noise-reduction filter.

The images in figure 7 show the median and the mean computed on amoebas compared to those built on regular square kernels. The pilot image that drives the shape of the amoeba is the result of a standard Gaussian filter of size 3 on the original image, and the distance  $d_{pixel}$  is the absolute difference of gray-levels.

For the filters using amoebas, the median filter preserves well the contour, but the mean filter gives a more “aesthetically pleasing” image. In either case, the results are clearly superior to filterings by fixed-shape kernels, as seen in the figure 7.



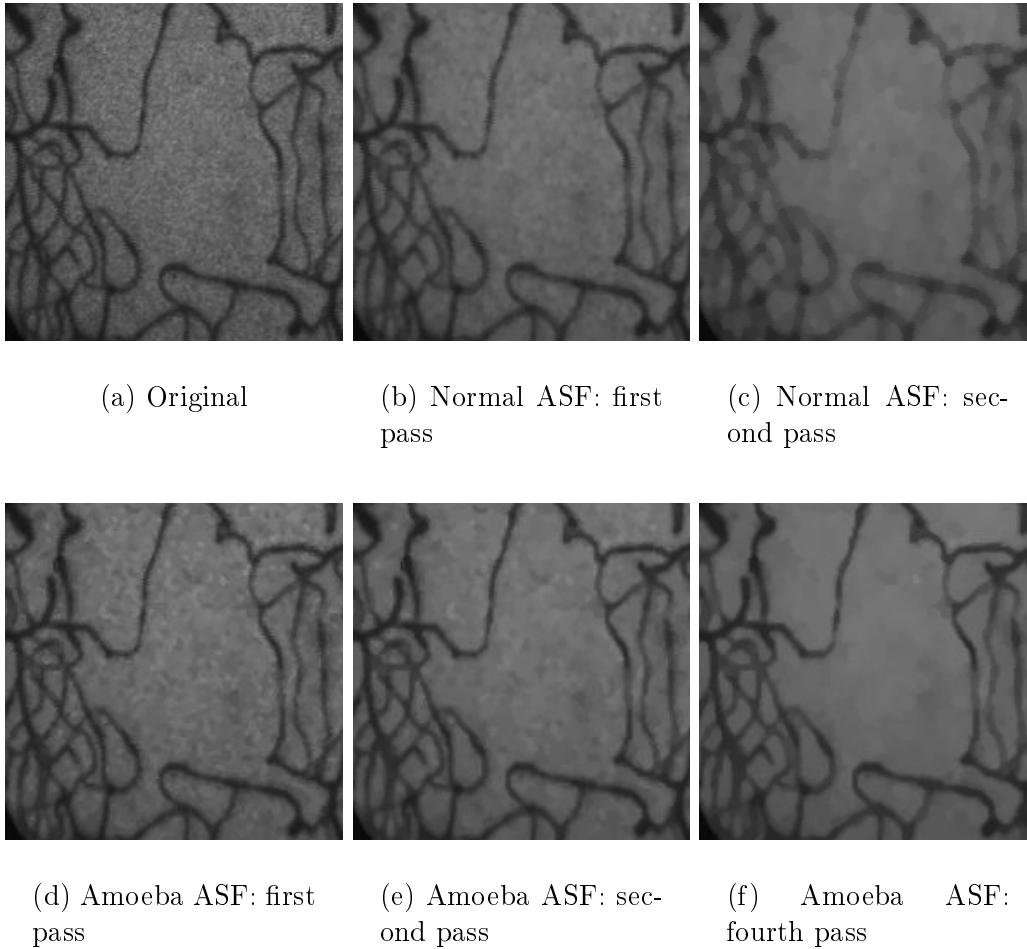


Fig. 6. Alternate sequential filters on classic kernels and on amoebas. The amoeba preserves extremely well the blood vessels while strongly flattening the other areas.

*Mean and median for color images*

In the case of color images, the mean is replaced by the mean on each color component of the RGB color space. For the “median”, the point closest to the barycenter is chosen. Other distances or colorspace can be used, depending on the application, the type of noise and the quality of the color information.

*Iteration*

The quality of the filtering strongly depends on the image that determines the shape of the amoeba. The previous examples have used the original image filtered by a Gaussian, but this does not always yield good results (also see [6]).



Fig. 7. Results of a “classic” median filtering and two amoeba-based filterings: a median and a mean on Edouard Manet’s painting “Le fifre”.

It is frequent indeed that a small detail of the image be excessively smoothed in the pilot image, and thus disappears completely in the result image. On the other hand, noisy pixels may be left untouched if the pilot image does not eliminate them. A possible solution is to improve the quality of the pilot image, so that it helps the amoeba in preserving these features. Such an image should be well-smoothed in flat regions, while preserving as well as possible the contour information. One good method to compute such an image would be of course to use an amoeba-based filter !

We will proceed in two steps: the first one follows the scheme described earlier, using the Gaussian-filtered original image as a pilot, with large amoebas, and outputs a well-smoothed image in flat areas while preserving as much as possible the most important contours. The second step takes the original image as input and the filtered image as a pilot, with smaller amoebas. These amoebas don’t need to be as large as the first ones, since their shapes will be computed on a very smooth image, and therefore they will preserve well the finer details.

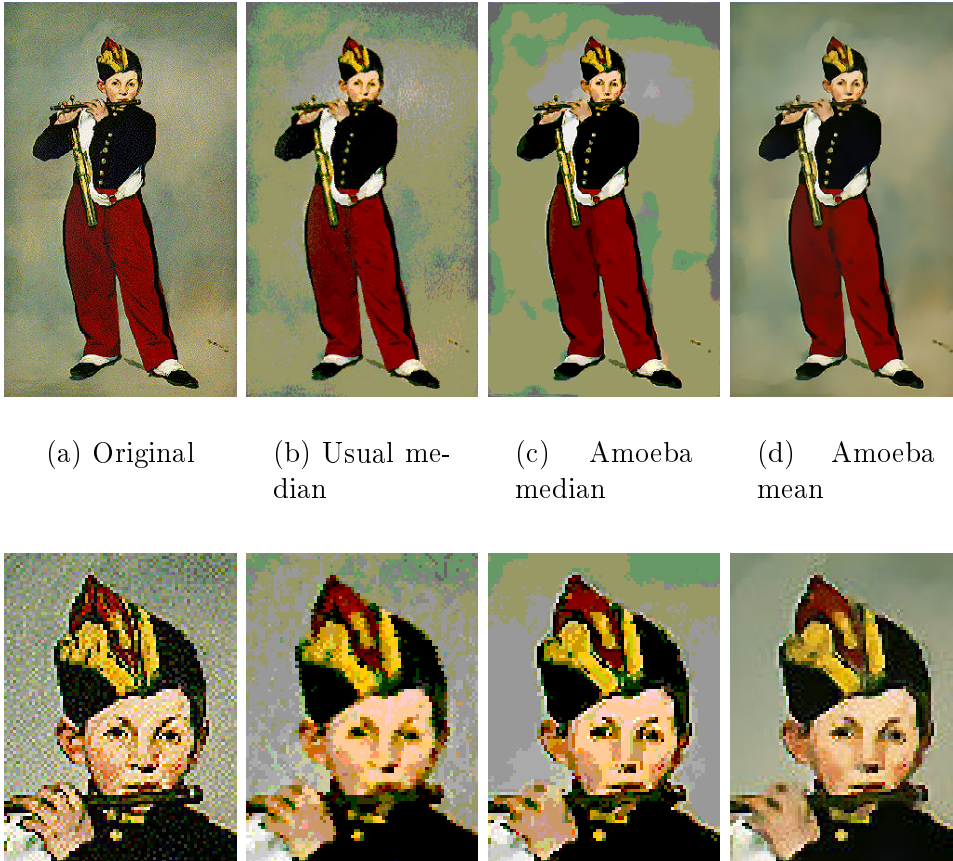


Fig. 8. Color images: results of a “classic” median filtering, and two amoeba-based filterings: a median and a mean. As a simple extension of the grayscale approach, each channel of the pilot image has been independently smoothed by a Gaussian of size 3.

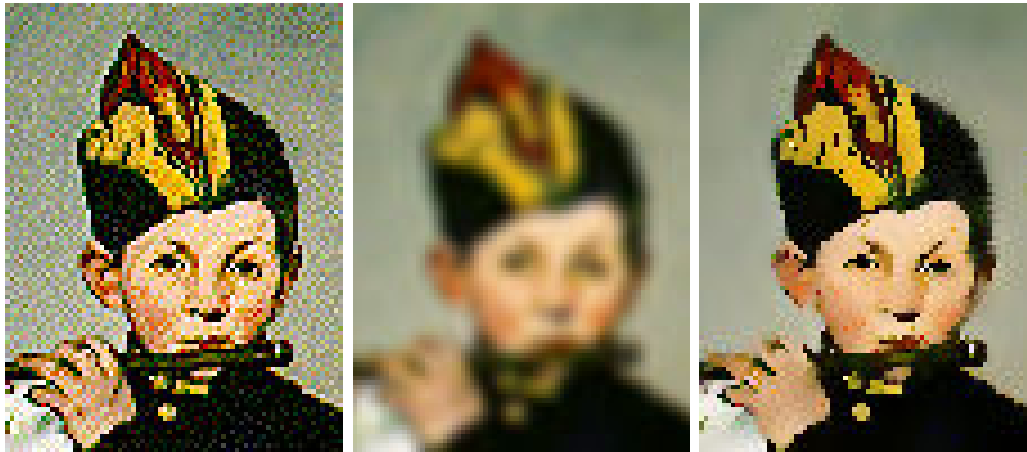
Although this refinement of the pilot image could be iterated, we have found in practice that once is enough to reduce the noise dramatically (see figure 9).

This method is also very useful for color images, since the amoeba-based pilot image provides better color coupling through the use of an appropriate color distance than simply merging the results of a Gaussian filtering of each channel independently.

## 5 Application to 3D images

### *3D images and contour noise*

While noise reduction is a nice bonus for 2D images, things are different with 3D medical images ([9]). The visualization algorithms often use local gradient



(a) Original

(b) Pilot image: Gaussian filter

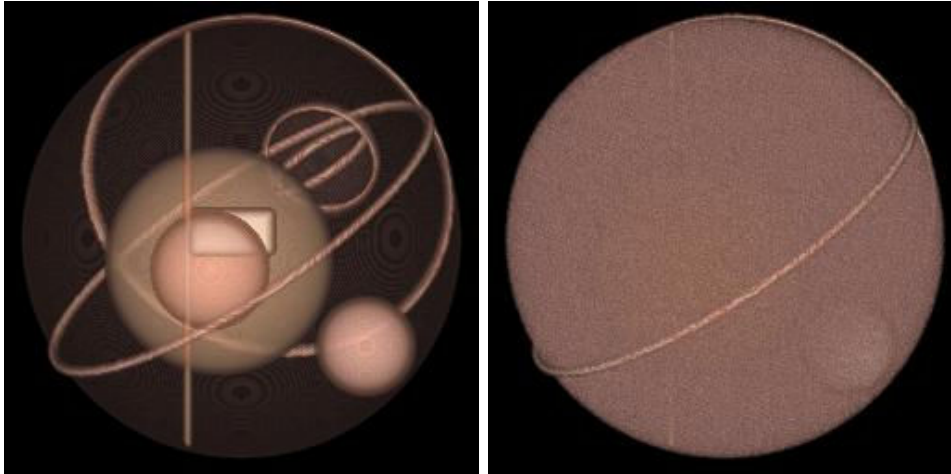
(c) Pilot image: amoeba mean filter



(d) Result image: amoeba mean with Gaussian pilot

(e) Result image: amoeba mean with amoeba pilot

Fig. 9. Comparison between two pilot images: a Gaussian one, and one based on a strong amoeba-based filtering. With the amoeba pilot image the hand is better preserved, and the eyebrows do not begin to merge with the eyes, contrary to the Gaussian-based pilot image. Having both less noise and stronger contours in the pilot image also enables the use of smaller values on the lambda parameter so that the amoeba will stretch more in the flatter zones, and thus have a stronger smoothing effect in those zones, while preserving the position and dynamics of the contours



(a) Ideal image

(b) Noisy image

Fig. 10. A synthetic 3D volume presenting many similarities with medical images of the cardiac regions. Left: original image. Right: image with addition of noise.

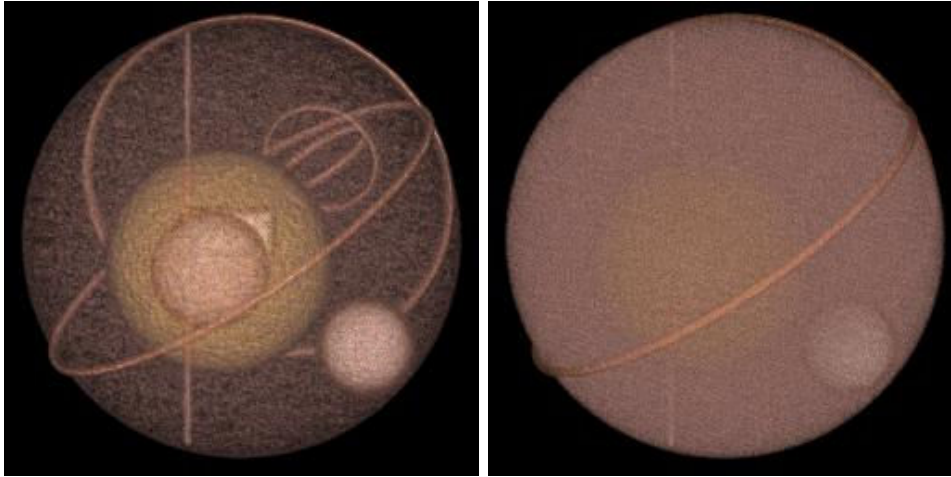
information for the computation of the shading of the voxels and many applications, especially in the medical field, use mostly the contour information to visualize the various objects and as such are very vulnerable to noise on the contours (see figure 10).

When displaying directly 3D data (as opposed to studying a stack of 2D images), it is essential that the user be able to see the objects they are interested in. This is why most modern renderers include a complex transparency and shading model that makes it possible to peek far inside the image to display the interesting objects. To make a quantitative analysis easier, a synthetic image was created that presents many similarities with 3D scanner images, especially images of the cardiac region: strong textures and thin vessels to preserve.

Figure 10 illustrates the problem due to strong noise in a 3D image and figure 11 shows that levelings are ineffective as they suppress the noise inside the objects but not on the contours. However, as with most morphological tools, amoebas are adimensional and can be used without modification on 3D images.

#### *Median filtering with amoebas*

One important aspect of the filtering of such medical images is that those images are monospectral: coloring and, ultimately, tissue identification is done using a look-up table. This means that a shifting of the values may have a



(a) Simple median, radius 2

(b) Median followed by by leveling

Fig. 11. Although the median filters much noise away, the reconstruction (needed to recover the position of the contours) reconstructs most of the noise on the borders of the object, removing most of the transparency.

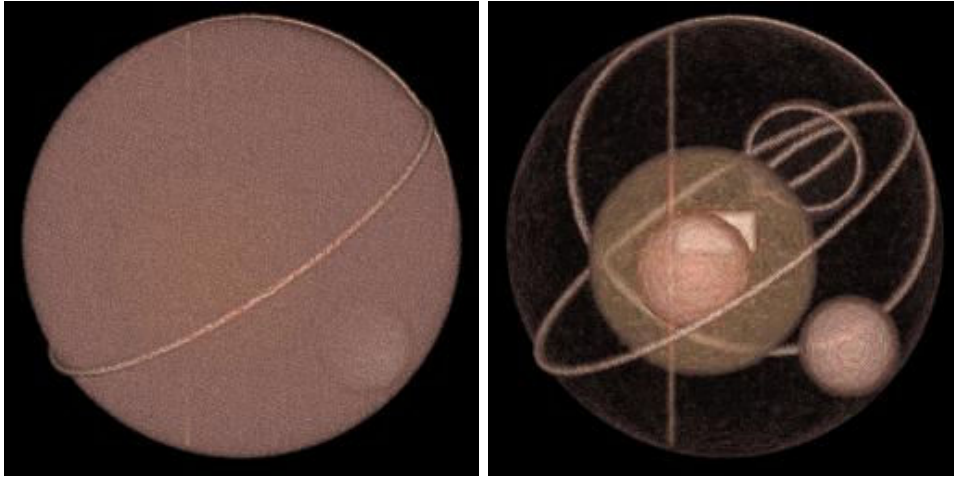
dramatic effect on the visualization and consequently on the interpretation of the images. This is why we have chosen to first test the median: with its property of returning only values existing in the image, the median lessens the risk of misinterpretation. However, traditional median filtering does not preserve well the contours, and may remove small details which may be crucial to a physician's analysis, hence the use of amoebas.

For each pixel the processing is done in two steps: first compute the shape of the amoeba centered on the pixel and then sample the values of the pixel inside the amoeba, feed them to the median operator and write the result at the center of the amoeba in the output image.

### *Results for 3D images*

Figure 12 shows the result of a median filter computed on amoebas. It is plain to see that most of the transparency effect has been preserved, which indicates that most of the noise has been filtered on the contours as well as inside the objects.

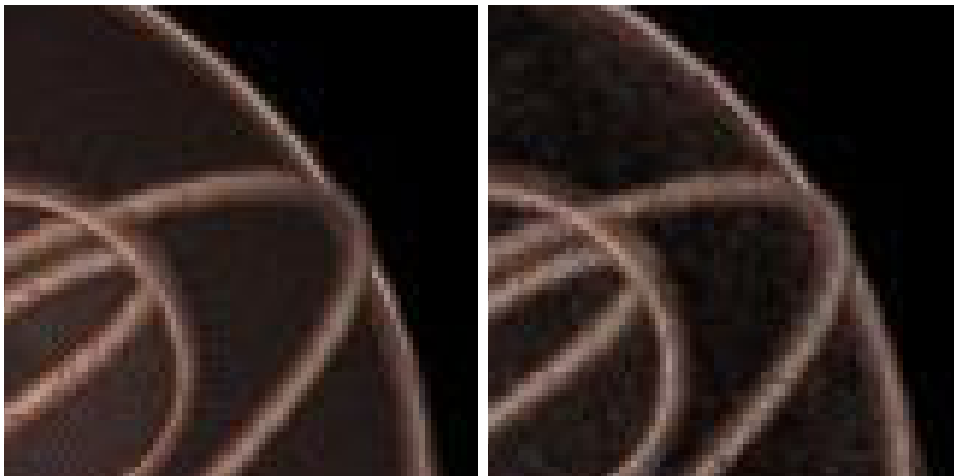
Figure 13 is a zoom of figure 12 centered on the thin structures. Once again we have good results in the preservation of those elongated structures. This is absolutely essential since this type of features include critical body parts such as blood vessels, nerves, etc.



(a) Noisy image

(b) Noisy image filtered by median computed on amoebas

Fig. 12. The median applied to the amoeba recovers most of the information very well, including thin details. The cleaner gradient on the contours results in a near-perfect transparency effect.



(a) Original image (zoomed)

(b) Amoeba median filter (zoomed)

Fig. 13. The amoeba-based median filter recovers very well the contours, as well as the transparency.

It is extremely hard to quantify *in a meaningful way* the results of such a filtering, especially since the usual signal-to-noise ratio does not express well the fact that we may tolerate small variations in many places but not a few strong variations in critical areas. A more meaningful measure for this problem was developed by Boehm in [9]. This measure is tied to a 3D volume renderer

(ray-caster) by comparing the visibility of each voxel in the filtered image with that of the corresponding voxel in the ideal image. When a voxel in the filtered image contributes in the same amount to the visualization as the corresponding voxel in the ideal image, then its associated quality measure is equal to one. The greater the difference between both contributions, the closer to zero the measure will be. Results are then averaged in the whole image as well as separately on each component.

Figure 14 shows a comparison between an amoeba-based median filter and a classic alternate sequential filter (ASF). The results show clearly that while the ASF is as good as the amoeba on the larger structures (spheres B2, B3 and B5), it fails completely to preserve the thin structures such as the rings. Furthermore, there is a clear shift of the values, especially on the outermost rings, which may cause diagnostic errors.

### *Simple optimizations*

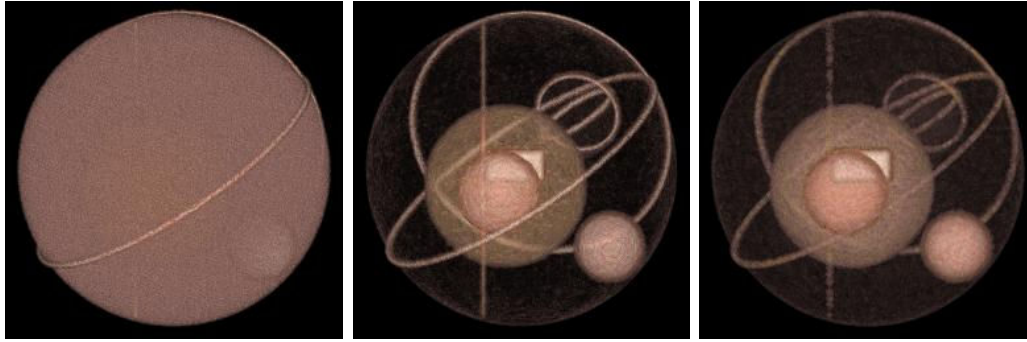
An important fact to take into account is that the radius parameter is like an amount of energy given to the amoeba. It can be used either to climb slopes (with a penalty given by the  $\lambda$  parameter) or it can be used to expand in flat areas. This amount of energy needs to be quite high so that the amoeba can jump over noisy pixels (though not too high so that it does not cross too much over strong gradient lines). However such a high energy means that in flat areas the amoeba will grow to a very large size, which means that not only will the shape be costly to compute but the resulting sample of pixel values will be quite large and so the filter operator will be accordingly long. A very simple yet dramatically effective optimization is thus to impose an upper bound on the size of the amoeba, the value of which depends on the type of noise and the characteristic size of the image elements. This can reduce the cost of computation by an order of magnitude without any detectable loss of effectiveness.

Another form of optimization is to compute the shape of the amoeba on a slightly filtered version of the original image, such as a Gaussian filtering. This will reduce small noise without moving too much the contours and enable the use of smaller amoebas.

## **6 Conclusion and future work**

We have presented here a new type of structuring element that can be used in many morphological algorithms. By taking advantage of outside information,

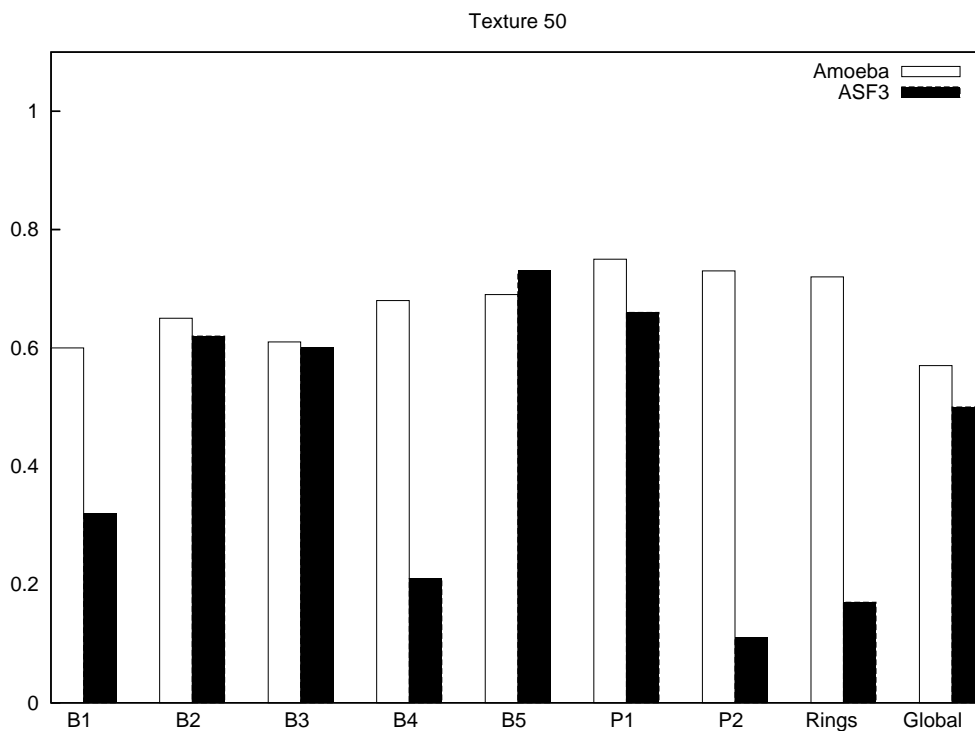




(a) Noisy image

(b) Amoeba-based median

(c) ASF size 3



(d) Measure of the results

Fig. 14. Comparison of the results of an amoeba-based median filter and an alternated sequential filter of size up to 3. The measure is performed on each component of the image (spheres B1-B5, parallelepipeds P1 and P2, rings, and total average).

filters built upon those structuring elements can be made more robust on noisy images and in general behave in a “more sensible” way than those based on fixed-shape structuring elements. In addition, morphological amoebas are very adaptable and can be used on color images as well as monospectral ones and, like most morphological tools, they can be used on images of any dimension (2D, 3D, ...). Depending on the application, alternate sequential filters are

very effective when looking for very flat zones, whereas median and mean filters output smoother images that may be more pleasing to the eye but could be harder to segment.

It is possible to use amoebas to create reconstruction filters and floodings that take advantage of the ability to parametrize the shape of the amoebas based on the image content. However, the behaviors of the amoebas are much more difficult to take into account when they are used in such complex algorithms. In particular, if connexity is important (such as in a reconstruction filter), then amoeba with a maximum radius of one pixel should be used.

The results show that simple extensions of the scalar algorithms to the RGB space already yield excellent results, especially when iterating. The use of more “perceptual” distances (HLS or LAB) would probably prevent some unwanted blending of features, although this is as yet conjectural and will be the basis of further work.

The filtering of 3D images by morphological amoebas, though still in its infancy, seems very promising. Expressing the coupling between image data and geometry through a kernel makes it possible to implement a much larger range of filters to an image than was possible before. Another area where improvement is to be expected is the computation of the shape of the amoeba. Not only should it be possible to use elaborate gradient estimation such as proposed in [6], but also providing more complex behaviors for the amoebas, such as an incompressible minimum element, to guarantee at least some diffusion, or on the contrary a minimum size requirement to prevent diffusion through small holes.

Finally, it is important to notice that this amoeba framework is general enough to accommodate other types of distances. The distance presented here couples geometry and grey levels (or color distances), but other similar schemes can be expressed in terms of amoebas. For instances, the approaches presented in [7] and [8], which offer very interesting results, can be implemented by amoebas with the appropriate distance, which thus inherit all the possibilities available to the ones described in this paper. These various approaches show the viability and the vitality of the amoeba framework, as well as its applicability to many fields of research.

## References

- [1] Perona, P. and Malik, J., “Scale-space and edge detection using anisotropic diffusion”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, July 1990

- [2] Serra, J. *et al*, "Mathematical morphology for boolean lattices", *Image analysis and mathematical morphology - Volume 2: Theoretical advances*, Chapter 2, pp 37-46, Academic Press, 1988
- [3] Cherkassky, B. V. and Goldberg, A V., "Heap-on-top priority queues", TR 96-042, NEC Research Institute, Princeton, NJ, 1996
- [4] Brodnik, A. *et al*, "Worst case constant time priority queue", Symposium on Discrete Algorithms, 2001, pp 523-528
- [5] Tomasi, C. and Manduchi, R., "Bilateral Filtering for Gray and Color Images", Proceedings of IEEE International Conference on Computer Vision, Bombay, India, 1998
- [6] Catté, F. *et al*, "Image selective smoothing and edge detection by nonlinear diffusion, SIAM J. Numerical Analysis, Vol. 29 No. 1, pp 182-193, February 1992
- [7] U.M. Braga-Neto, "Alternating Sequential Filters by Adaptive-Neighborhood Structuring Functions.", Proceedings of International Symposium on Mathematical Morphology ISMM'96, pp. 139-146, 1996.
- [8] Debayle, J. and Pinoli, J.-C., Multiscale image filtering and segmentation by means of adaptive neighborhood mathematical morphology, Proceedings of IEEE International Conference on Image Processing, Genoa, Italy, 2005
- [9] Boehm, M. "Contribution à l'amélioration du rendu volumique de données médicales 3D", PhD Thesis, Centre de Morphologie Mathématique, École des Mines de Paris, 2004
- [10] Lerallut, R. and Decencière, E. and Meyer, F., "Image processing using morphological amoebas", Proceedings of the 5th International Symposium on Mathematical Morphology, Kluwers Academic Publishers, 2005