

Fast 3D Mean Shift Filter for CT Images^{*}

Gustavo Fernández Domínguez, Horst Bischof, and Reinhard Beichel

Institute for Computer Graphics and Vision,
Graz University of Technology
Inffeldgasse 16/2, A-8010, Graz, Austria
{gustavo, bischof, beichel}@icg.tu-graz.ac.at

Abstract. In this paper we investigate the ability of the mean shift (*MS*) algorithm for denoising of 3D Computer Tomography (*CT*) data sets. The large size of the volume data sets makes it infeasible to apply a 3D version of the *MS* algorithm directly. Therefore, we introduce a variant of the *MS* algorithm using information propagation. We would like to make use of the 3D nature of the data with a considerably reduced running time of the algorithm. The proposed version is compared to a 2D implementation of the same algorithm applied slice by slice and other filter methods such as median filter and bilateral filtering. The advantages and disadvantages of each algorithm are shown on different *CT* data sets.

Keywords: Mean Shift Filter, Computer Tomography Imaging, Nonlinear Filtering.

1 Introduction

Image de-noising is an important task in image processing and pattern recognition. The goal is to reduce the noise present in the image, while preserving the important structures. For example in medical imaging, filtering techniques are often used to improve the sub-sequent segmentation [7]. Therefore, it is important that the filters preserve the (possibly small) structures which should be segmented.

Nonlinear filtering methods have gained increasing attention in the image processing community, due to their ability to preserve the edges while smoothing the noise. The well known *anisotropic diffusion* (*AD*) method developed by Perona and Malik in the early nineties [8] is a widely used iterative nonlinear filtering method. The recently developed *mean shift* (*MS*) algorithm by Comaniciu and Meer [3,4] is also a non-linear filter. This filter is strongly related to *anisotropic diffusion*, and the *bilateral filtering* (*BF*) method proposed by Tomasi and Manduci [9]. Barash and Comaniciu showed the relationship between *AD*, *BF*, and *MS* [1]. The goal of this paper is to investigate the applicability of the *MS* filter to 3D computer tomography data.

^{*} This work was supported by the Austrian Science Foundation (FWF) under project number P-14897.

The *MS* filter can be applied slice by slice in a straight-forward manner (hereafter called *2D* version). Since the data is *3D*, our goal is to make full use of this information in the filtering step. Due to the large size of the data a straight-forward generalization of the *MS* procedure to *3D* increases the run time of the filter considerably [3]. For this reason, we have developed a *3D* extension to speed-up the processing by taking the *3D* nature of the data into account.

The organization of this paper is as follows: In Section 2 we briefly outline the main ideas of the *MS* algorithm. In Section 3 we present our information propagation method. Section 4 compares the results of different filtering methods on *CT* volume data sets. In particular, we evaluate the smoothing and structure preserving properties of the filters. Discussion and future work conclude the paper.

2 Mean Shift

The original *MS* method was presented by Fukunaga and Hostetler [6] as a nonparametric method to estimate a probability density function (pdf) using the so-called Parzen window density estimator [5]. Cheng rediscovered the method and used it for cluster analysis [2].

Recently, Comaniciu et al. [3,4] developed the *MS* technique for the analysis of multi-modal feature spaces. Following a similar notation as in [4], *MS* technique can be stated as follows. Given n data points $\mathbf{x}_i, i = 1, \dots, n$ in the d -dimensional space \mathfrak{R}^d , the multivariate kernel density estimator with kernel $K_H(\mathbf{x})$ computed at the point \mathbf{x} is given by

$$f(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n K_H(\mathbf{x} - \mathbf{x}_i). \quad (1)$$

Comaniciu et al. [4] showed that for the mean shift vector $m_{h,G}(\mathbf{x})$ the following condition holds

$$m_{h,G}(\mathbf{x}) = \frac{h^2 c \nabla f_{h,K}(\mathbf{x})}{2 f_{h,G}(\mathbf{x})}, \quad (2)$$

where K and G are kernels with respective profiles k and g . h is the bandwidth of the kernel used, and c is the normalization constant. These profiles are related by the condition

$$g(\mathbf{x}) = -k'(\mathbf{x}), \quad (3)$$

where k' is the derivative of the profile k . Equation (2) indicates that the mean shift vector is aligned with the local gradient estimate, therefore it can be used to detect the local maxima of this distribution [3]. The main difference between *MS* and other nonlinear methods is the way how the information in the spatial and range domains are treated to obtain the filtered image. Basically, *MS* can be seen as an adaptive gradient ascendent method.

3 Fast 3D-Extension of MS

3D – CT data sets are commonly used in medical applications. Unfortunately, this kind of data can contain a considerable amount of noise. CT volume data sets can be generated by a single-slice scanner which provides consecutive 2D axial slices which form the 3D data. A 3D – MS algorithm is straight-forward to develop by simply adding one dimension to the vector \mathbf{x} in Equation (2). This means that we consider a 4D-space (3 spatial + 1 radiometric dimension) for the pdf estimation and apply the standard mean-shift procedure on this new pdf. The problem is that the execution time of the algorithm increases tremendously (e.g., for a typical CT data set with 73 slices this takes more than 6 hours on a standard PC). Therefore, we propose a new method that we call *fast 3D-MS*.

Our approach consists of applying the 2D version of the algorithm to one slice f^k to obtain the filtered slice g^k of the data set. For the next slice f^{k+1} , the filtered slice g^{k+1} is obtained using the procedure shown in Algorithm 1. The filtered image is obtained stacking the filtered slices g^k .

Algorithm 1 Fast 3D-MS algorithm.

Input: Original slices f^k, f^{k+1} ; filtered slice g^k ; bandwidths: h_s (spatial), h_r (range).
 Output: Filtered slice g^{k+1} .

```

1: for all pixels  $(x, y)$  of  $f^{k+1}$  do
2:    $t = f^{k+1}$ 
3:    $difference = |f_{x,y}^{k+1} - f_{x,y}^k|$ 
4:   if ( $difference < threshold$ ) then
5:      $t_{x,y} = g_{x,y}^k$ 
6:   end if
7:    $g_{x,y}^{k+1} = \text{meanShiftP}(t, x, y, h_s, h_r)$ 
8: end for

```

meanShiftP applies the mean shift at a single pixel (x, y) in slice t using spatial bandwidth h_s , and range bandwidth h_r . The *threshold* parameter is a prefixed value related to h_r .

In step 2 the image t is obtained by taking the slice f^{k+1} . Steps 3, 4 and 5 consider the information of the previous slice for the current one. For the pixel (x, y) , if the absolute value of the difference between the slices f^{k+1} and f^k is less than certain threshold (steps 3, and 4), then the pixel (x, y) of image t is replaced by the pixel (x, y) of the previous filtered slice g^k (step 5). Step 7 performs the MS at a single pixel (x, y) in slice t . To choose the critical threshold parameter, we can make the following consideration: The range bandwidth h_r of the MS algorithm measures the similarity of the gray values. If we use a small percentage of h_r (e.g. 15%, or 20%) as the threshold parameter, we have a high probability that the gray values come from the same mode of the distribution, therefore we can use the pdf of the previous slice. The basic idea of the algorithm is that when the difference of the pixels in neighbouring slices is small, we can

reuse the result of the already calculated slices. We have to rerun the algorithm only for those pixels where the difference in gray value is large. The procedure is simple and it allows us to achieve an acceptable running time, while some information along the vertical direction (z -axis) is used. The initial slice to start the algorithm is chosen in the middle of the data set.

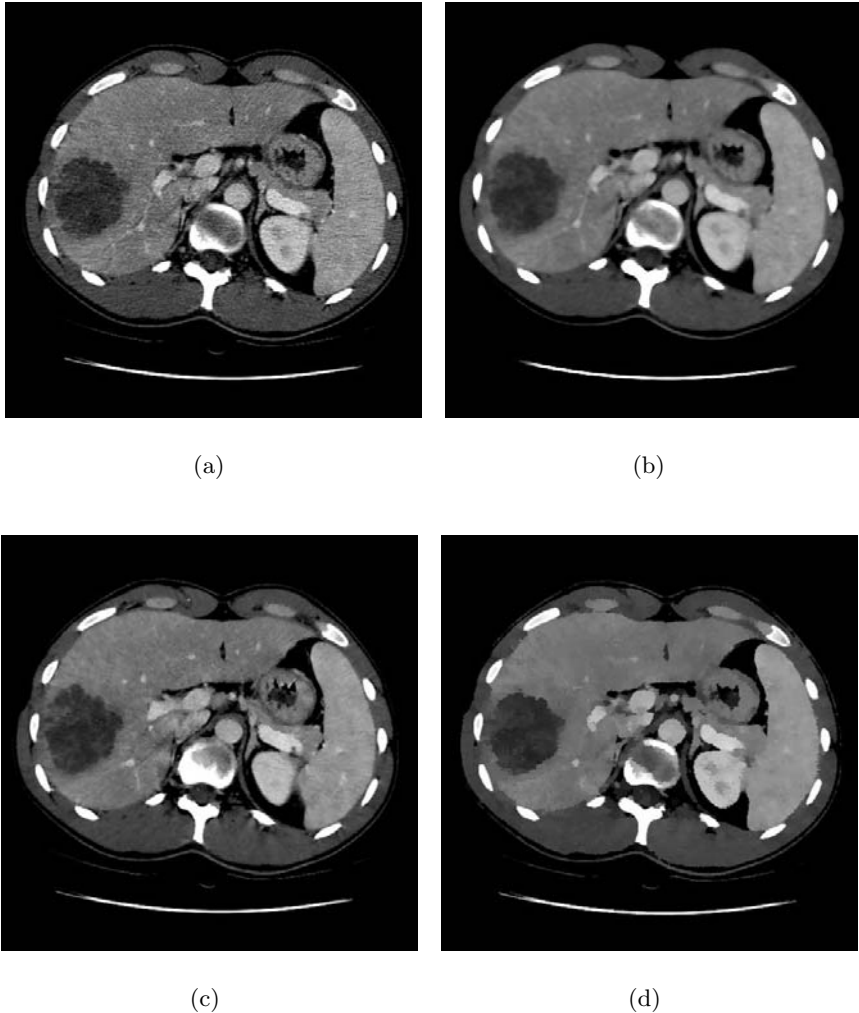


Fig. 1. Original data and filtered results of an axial *CT* image. a) Original slice, b) 3D median filter, c) 2D bilateral filter, d) fast 3D mean shift algorithm.

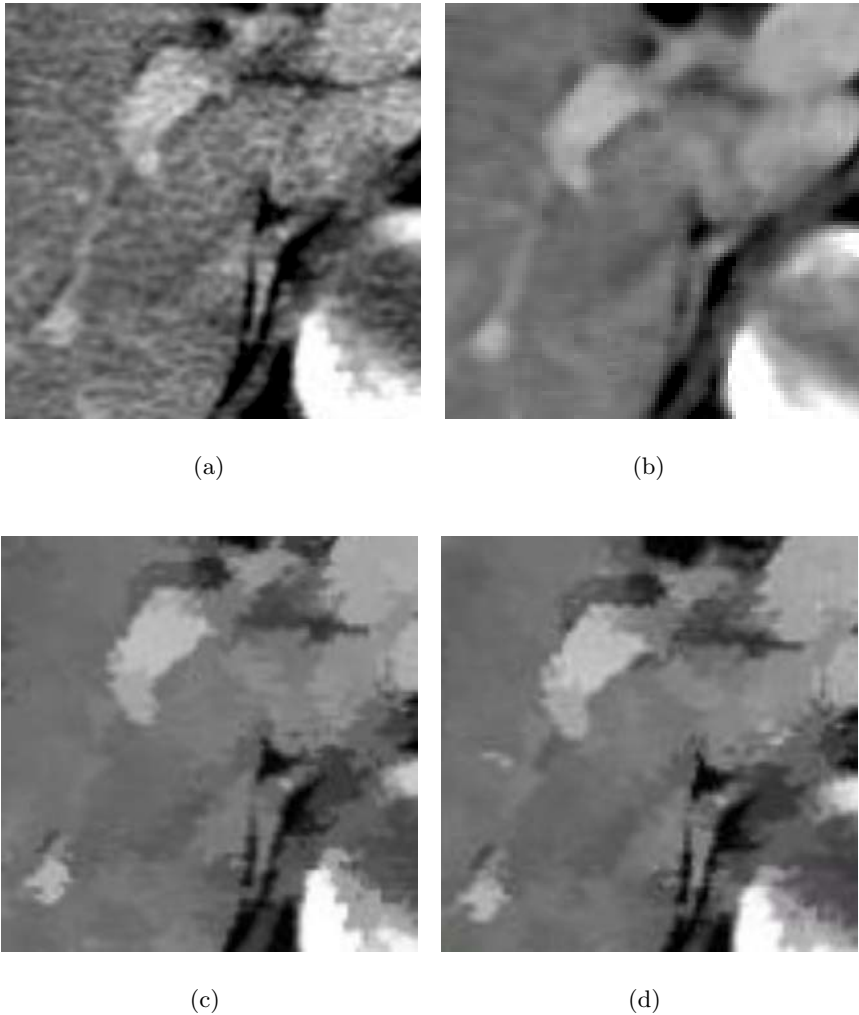


Fig. 2. Zoomed view of liver tissue and some vessels. a) Original slice, b) 3D median filter, c) 2D mean shift algorithm, d) fast 3D mean shift algorithm.

4 Results

The results presented here correspond to two data sets named *ds01*, and *ds02*, of a size equal to $512 \times 512 \times 103$ voxels, and $512 \times 512 \times 85$ voxels respectively. The gray value range of the data sets is 12 bit. We compare the obtained results against a 3D median filter (named *3D-MF*, size: $5 \times 5 \times 3$), a bilateral filter applied slice by slice (called *2D-BF* version, with parameters $h_{sx} = h_{sy} = 5.0$, and $h_r = 25.0$), and a *MS* version applied slice by slice (called *2D-MS*). The

Table 1. Standard Deviations σ_1 , and σ_2 and absolute value MVD between regions of medical structures in different data sets. Filter methods: 3D median filter (3D-MF), 2D bilateral filtering (2D-BF), 2D mean shift (2D-MS), fast 3D mean shift (Fast 3D-MS).

		Liver vessels			Liver tissue			Liver tumor		
Data set	Image	MVD	σ_1	σ_2	MVD	σ_1	σ_2	MVD	σ_1	σ_2
ds01	Original	104.84	21.07	23.07	23.61	17.98	17.27	21.98	13.35	19.24
	3D – MF	94.77	11.91	6.48	23.59	5.30	8.15	16.47	3.88	7.13
	2D – BF	103.64	13.49	13.85	23.36	9.34	4.79	18.94	5.36	11.17
	2D – MS	101.45	14.69	11.28	22.69	8.42	5.12	19.37	7.80	9.73
	Fast 3D – MS	101.75	7.58	3.22	21.51	4.58	5.08	17.64	6.37	6.52
ds02	Original	68.60	22.18	31.77	11.69	16.44	41.00	12.20	16.17	24.92
	3D – MF	55.32	15.66	21.54	11.06	5.22	29.67	11.68	6.68	18.44
	2D – BF	65.39	16.26	25.97	11.50	7.96	36.74	11.97	8.43	19.68
	2D – MS	65.02	20.04	25.64	11.43	7.47	36.11	11.75	9.00	20.20
	Fast 3D – MS	57.05	14.35	21.00	10.05	4.37	31.22	11.58	5.59	18.38

parameters used for the MS filters were $h_{sx} = h_{sy} = 5.0$ and $h_r = 25.0$. h_{sx} , and h_{sy} represent the spatial spread on each direction (x , y respectively), and h_r corresponds to the spread in the range domain. The threshold parameter of the Algorithm 1 was chosen as 20% of the h_r parameter. The parameter values of the filters were chosen in order to remove the noise and to preserve different objects such as vessels, organs or other structures of interest. Figure 1 shows original data and filtered results of an axial CT image. Figure 2 depicts a zoomed view of the original image and the filtered results.

In order to have a quantitative evaluation of the filters we manually segmented various regions of interest in several images and we use the evaluation criteria explained below. The chosen regions were part of three different structures we are interested in: vessels, liver tissue, and tumor zones. The purpose of the filter is to ease a latter segmentation step. Therefore, we would like to obtain the different tissues of interest as homogeneous as possible, while at the same time, they should preserve gray-value difference to the surrounding tissue. We used the following measures on the hand segmented regions: i) To measure difference between regions, we use $MVD(x) = |\mu_i - \mu_j|$, where μ_i is the mean gray-value of the region of interest and μ_j is the mean gray-value of the surrounding region; ii) in the case of uniform regions, we calculate the standard deviation σ inside the regions as a point spread indicator of the gray value variations. Both quantities should be analyzed together to have a correct idea of the filter quality. The aforementioned values were computed for each region. In case of vessels, we have selected four different regions, while for liver tissue and tumor zones five regions were chosen.

Table 1 shows these measures. σ_1 , and σ_2 are the standard deviations of the structure and its surrounding tissue. Table 2 shows the execution times of the algorithms. All the times are expressed in seconds.

Table 2. Execution times in seconds for different filter methods: 3D median filter (3D-MF), 2D bilateral filtering (2D-BF), 2D mean shift (2D-MS), fast 3D mean shift (Fast 3D-MS).

Data set	3D – MF	2D – BF	2D – MS	Fast 3D – MS
ds01	550	1823	1216	1069
ds02	437	1412	1131	1063

4.1 Analysis of Results

Analysis of Table 1 shows that: i) The standard deviation σ_1 in case of the fast 3D – MS algorithm is less than the σ_1 of the other filters in almost all the cases (except for the liver tumor of *ds01*); ii) there are no large differences between the *MVD* rates comparing the different filters, but considering the standard deviation we note that in the 3D case the σ value obtained is less than in other cases; iii) considering the surrounding tissue of the analyzed structures, the proposed 3D version gets the lowest σ_2 value in almost all the cases (except for the liver tissue for both data sets);

The aforementioned items give us the idea that the uniform regions are more smooth in case of *MS* than other filtering methods. Examining the images, it can be seen that the proposed version of the filter achieves good results. Noise was eliminated in liver tissue regions, while small structures, such as vessels, were kept. Also, the results of the fast 3D – MS are better than the 2D – BF method, and the 3D – MF. Note the difference comparing borders between different regions for the *MS* results against 3D – MF. In the former case, those borders were preserved, while in the latter case they seem to be blurred (see the zoomed Figure 2).

Table 2 shows that the new fast 3D – MS is faster than the other methods, except the 3D – MF. But since the quality of the results is better, the new algorithm is preferred.

5 Conclusions and Future Work

In this paper we have applied the mean shift filter on computer tomography volume data sets using a novel fast 3D version of the mean shift filter. The basic idea is to use a 2D – MS filter slice by slice but taking into account also pixels along the axial direction. Good results are obtained compared to a 2D version of the same method applied for each slice, a classical filter such as the median filter, and other nonlinear filter techniques such as bilateral filtering. The filtered images have good quality in terms of noise reduction and border preservation. The obtained results show the applicability of *MS* algorithm on 3D medical data. Considering the execution times, the proposed version performs similar to the 2D case.

Future work will be concentrated on analytical treatment of the approximation and improvement the robustness of the method.

Acknowledgements. The original *MS* software available at [10] was properly adapted for our purposes.

References

1. D. Barash and D. Comaniciu. A common viewpoint on broad kernel filtering and nonlinear diffusion. In *Proceedings of the 4th International Conference on Scale-Space Theories in Computer Vision, Scotland, U.K.*, to appear June 2003.
2. Y. Cheng. Mean shift, mode seeking, and clustering. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 17-8:790–799, 1995.
3. D. Comaniciu. *Nonparametric Robust Methods for Computer Vision*. PhD thesis, ECE Department, Rutgers University, July 2001.
4. D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Recognition and Machine Intelligence*, 24-5:603–619, 2002.
5. R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience, New York, 2nd Edit., 2000.
6. K. Fukunaga and L. D. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Trans. Information Theory*, 21:32–40, 1975.
7. K. Krissian, G. Malandain, N. Ayache, R. Vaillant, and Y. Troussel. Model-based multiscale detection of 3D vessels. In *Proceedings of Computer Vision and Pattern Recognition*, pages 722–727. IEEE, 1998.
8. P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12-7:629–639, 1990.
9. C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Proceedings of 6th International Conference on Computer Vision, New Delhi, India*, pages 839–846. IEEE, 1998.
10. www.caip.rutgers.edu/riul. www.caip.rutgers.edu/riul/research/code.html. 2001.