Computational Aesthetics in Graphics, Visualization, and Imaging (2010)
O. Deussen and P. Jepp (Editors)

Interactive volume illustration using intensity filtering

Marc Ruiz, Imma Boada, Miquel Feixas, and Mateu Sbert

Graphics and Imaging Laboratory, Universitat de Girona, Spain

Abstract

We propose a simple and interactive technique for volume illustration by using the difference between the original intensity values and a low-pass filtered copy. This difference, known as unsharped mask, provides us with a spatial importance map that captures salient and separability information about regions in the volume. We integrate this map in the visualization pipeline and use it to modulate the color and the opacity assigned by the transfer function to produce different illustrative effects. We also apply stipple rendering modulating the density of the dots with the spatial importance map. The core of our approach is the computation of a 3D Gaussian filter, which is equivalent to three consecutive 1D filters. This separability feature allows us to obtain interactive rates with a CUDA implementation. We show results of our approach for different data sets.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Transfer functions—Volume Rendering

1. Introduction

A main step in direct volume rendering is the definition of the transfer function. This assigns optical properties such as color and opacity to the original values of the data to visualize the internal parts of the volume. In the case of one-dimensional transfer functions, there is a direct mapping between optical values and voxel scalar values. On the other hand, multidimensional transfer functions take more information into account, such as first and second derivatives [KKH02]. This additional gradient information allows a better separation between materials and thus better visualizations. The need for a good detection of boundaries becomes crucial for the quality of the rendering. Knowledge of boundaries facilitates the understanding of the volume data set by focusing on the most pertinent subset of data, i.e., which can be considered as the most salient parts of data [IK01].

Although it has already been used by different authors [KV06], the concept of saliency in a volume model, different to an image or a 3D polygonal scene, is not yet a well defined concept. Saliency should facilitate learning by focusing on the most pertinent subset of available sensory data. On the other hand, saliency of a voxel (or region) should arise from contrast between this voxel (or region) and its neighborhood. Thus, it should include recognition of the boundaries of the inner parts of the volume data, as well as finding local deviations. When examining a 3D polygonal model, we just have one isosurface, we do not have to single it out, we just want to single out the particular oddities or irregularities of this surface. In the case of volume data, we first have to single out the components, i.e., the boundaries that make up its structure. On the other hand, the observation of the volume data model comes always via a transfer function, thus modulating an original, neutral transfer function with this extended saliency will allow us to learn about the data. A quantity very simple to compute that fills the above requirements of generalized saliency is an unsharped mask. This is the difference between low-pass filtered data and the original one.

Different feature enhancement strategies based on unsharp masking have been proposed. Cignoni et al. [CST05] performed unsharp masking to the normal field over the 3D surfaces, to enhance the perception of discontinuous geometric features. Luft et al. [LCD06] enhanced depth perception by unsharp masking the depth buffer. The difference between the filtered image and the original one was called spatial importance map. Ritschel et al. [RSI*08] coherently enhanced the scene by unsharp masking the outgoing radiance field over the mesh surface. Tao el al. [TLB*09] use the difference between the radiance volume and the smooth radiance volume to enhance local contrast of features. In this paper, we propose a similar approach but instead of using

DOI: 10.2312/COMPAESTH/COMPAESTH10/051-058

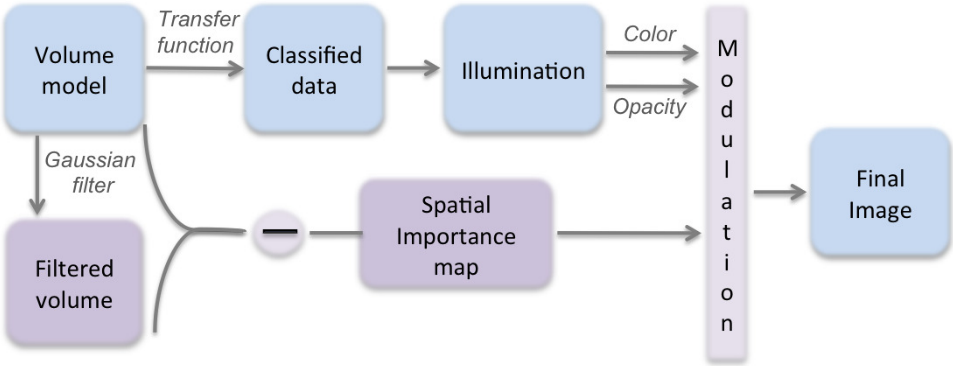


Figure 1: *Main steps of the proposed approach.*

radiance we propose to operate with the original intensity values, having lower memory requirements. We use the spatial importance map to obtain both an enhanced visualization of the data, by modulating color and opacity, and illustrative effects such as stippling.

The structure of the paper is as follows. In Section 2, we review related work. In Section 3, we describe the proposed approach. Then, in Section 4 we show the different effects that can be obtained in the proposed approach. Finally, conclusions are presented in Section 5.

2. Related Work

Direct volume rendering techniques allow to explore the structures embedded in the volume data set by varying the opacities and colors assigned to them. For the rendering to be effective it is required a transfer function that assigns colors and opacities to the different materials. Although a large number of strategies have been proposed for automatic and semi-automatic generation of transfer functions, the definition of a proper transfer function is still challeging. The main limitation is the classification process required to define the density intervals corresponding to the structures, i.e., the boundaries that separate the different materials.

Levoy proposed the use of the gradient magnitude to identify surfaces in volume data [Lev88]. Kindlmann and Durkin used the first and second derivatives along the gradient direction to calculate a boundary emphasis to be included in the opacity transfer function [KD98]. In addition to the design of the opacity transfer function, general multi-dimensional transfer functions were studied to better convey the boundaries and features in volume data [KWTM03, KKH02, KPI*03, LM04]. These methods create two-dimensional histograms where each entry represents the number of voxels at a given feature space pair at which the user in a trial-and-error manner assigns color and opacity until the desired visualization is obtained. To avoid this trial-and-error process, Maciejewski et al. proposed the addition of non-parametric clustering within the transfer function feature space in order to extract patterns and guide transfer function generation [MWCE09]. Recently, Correa et al. [CM09] presented a method for classifying volume data based on the ambient occlusion of voxels. They detected occlusion patterns that reveal the spatial structure of materials or features of a volume and represented them in an occlusion spectrum. This occlusion spectrum leads to better two-dimensional transfer functions that can help classify complex data sets in terms of the spatial relationships among features.

Illustrative techniques are suitable for emphasizing certain features or properties while omitting or greatly simplifying other, less important details [RE01]. The most popular styles, such as stippling, hatching and silhouettes, are from the pen-and-ink family [CMH*01, LME*02]. To incorporate illustrative effects in a volume renderer, Kindlmann et al. [KWTM03] utilized curvature-based transfer functions. Hauser et al. [HMBG01] proposed the two-level volume rendering concepts which allows focus+context visualization of volume data. Different rendering styles, such as direct volume rendering and maximum intensity projection, are used to emphasize objects of interest while still displaying the remaining data as context. Viola et al. [VKG04] introduced importance-driven volume rendering, where features within the volumetric data are classified according to object importance. Bruckner et al. [BGKG06] presented context-preserving volume rendering, where the opacity of a sample is modulated by a function of shading intensity, gradient magnitude, distance to the eye point, and previously accumulated opacity.

Different computational models have been proposed to interpret the selective visual attention. The biologically-inspired model of bottom-up attention of Itti et al. [IK01] permits us to understand our ability to interpret complex scenes in real time. The selection of a subset of available sensory information is controlled by a *saliency map* which is a topographic representation of the instantaneous saliency of the visual scene and shows what humans find interesting in visual scenes.

Inspired in Itti's work, Lee et al. [LVJ05] introduced the concept of mesh saliency, a measure of regional importance for 3D meshes, computed using a center-surround mechanism that is able to identify regions that are different from their surrounding context. Feixas et al. [FSG09] defined a view-based saliency of a polygon as the average information-theoretic dissimilarity between this polygon and its neighbors. In the volume rendering field, Kim et al. [KV06] presented a visual-saliency-based operator to enhance human perception of the volume data by guiding viewer's attention to selected regions. In different works on saliency, it has been shown that attention is attracted by changes in luminance, color, curvature, texture, shape, etc. [TIR05]. That is, salient features are generally determined from the local differential structure of images and different operators such as color or luminance gradient have been used [vdWGB06]. In González et al. [GSF08], from an information theory perspective, ambient occlusion has been defined as the occlusion information associated with each polygon of the model. In Ruiz et al. [RBV*08] voxel saliency is defined as the magnitude of the gradient of obscurances estimated by using the 4D linear regression method. Considering that obscurance represents occlusion information associated with a voxel, its variation with respect to its surround can indeed be perceptually salient, i.e., it can be considered as a salient feature of the volume.

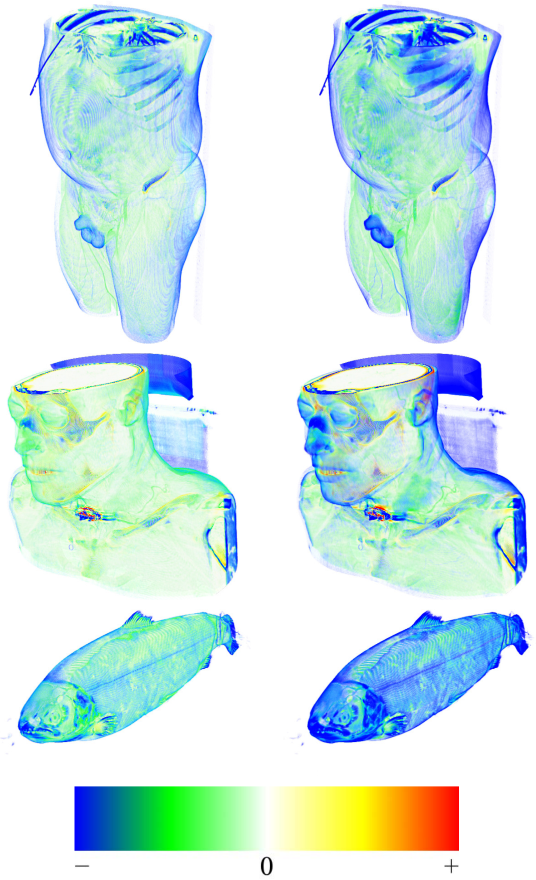


Figure 2: *From top to bottom, spatial importance maps obtained for CT body (256 × 256 × 415), CT head (512 × 512 × 297), and salmon (336 × 173 × 511). The images of the first and second columns have been obtained with a radius of 5 and 10 for the Gaussian filter, respectively.*

3. Proposed Approach

Detection of the structures is an important step towards the interpretation of volume data. In general, each structure is represented by an interval of densities. The identification of these intervals can be obtained from the identification of the boundaries that separate from each other. Spatial importance maps can be used to identify these boundaries. The idea of our method is to exploit this fact to provide a simple strategy for exploring volume data models and also obtain different illustrative effects that enhance the perceptual quality and interpretation of the images taking only the original intensity values.

The main steps that compose the proposed approach are represented in Figure 1. First, in a preprocessing step, we compute the spatial importance map by filtering the input model and subtracting the obtained result from the original model. Then, the spatial importance map is integrated in the visualization pipeline and used to modulate the color and/or the opacities returned from the illumination process in order to produce different illustrative effects. A detailed description of these steps and main implementation details are given below.

3.1. Spatial Importance Map

The first step of our approach consists in the computation of the spatial importance map, that can be considered the core of the method. This map is the extension to 3D of the spatial importance function ΔD proposed by Luft [LCD06]. In contrast to Tao et al. [TLB*09], who used radiance, we use intensity values. The advantages of our approach are that it does not depend on the transfer function and lighting, and thus can be done as a preprocessing step, and that it requires less memory.

Given a volume model $V : \mathbb{N}^3 \rightarrow \mathbb{Z}$ where $\mathbb{Z}$ represents the scalar values of the voxels, we compute the spatial importance map ΔD by

$$\Delta D = G * V - V, \tag{1}$$

where $G * V$ is a Gaussian blur of the volume. A Gaussian

Volume	R	1D filter (ms)	Subtract (ms)	Total (ms)
CT body	1	20.85	18.03	335.05
	5	52.96		438.13
	10	93.59		560.23
CT head	1	57.90	51.90	928.41
	5	155.61		1221.18
	10	272.77		1570.10
Salmon	1	22.66	19.66	376.37
	5	59.03		490.04
	10	104.93		617.24

Table 1: *Times in milliseconds (ms) to compute the spatial importance map for the CT body (256 × 256 × 415), the CT head (512 × 512 × 297), and the salmon (336 × 173 × 511) with different radii (R) of the Gaussian kernel. The 1D filter column reports the time to do a 1D convolution with the kernel (this process has to be done once for each axis). The Subtract column reports the time required to do the subtraction. The Total column includes the time to do the 3 convolutions, the subtraction, and all the other additional operations like memory allocations and transfers.*

blur is the convolution of an image with the Gaussian function

$$G(x) = \frac{1}{\sqrt{2}} e^{-\frac{x^2}{2\ ^2}}, \tag{2}$$

where x is the distance from the origin and is the standard deviation of the Gaussian distribution. The above function is for the 1D case, for 2D it is the product of two Gaussians and in 3D the product of three Gaussians, one per direction. The effect of a Gaussian blur is the reduction of the high-frequency elements of the image, so it is a low-pass filter.

In Figure 2, we illustrate the spatial importance maps obtained for different models, a CT body (256 × 256 × 415), a CT head (512 × 512 × 297), and a salmon (336 × 173 × 511) and different radii of the Gaussian kernel. The maps are colored using the thermal scale represented at the bottom, where warm colors correspond to positive importance values and cool colors to negative ones. Observe that the voxels that are near density boundaries have values different from zero, because in the blurred volume the boundaries are smoothed and have values lower than the originals at one side, and higher at the other side. So, positive difference values indicate voxels near a boundary with a higher density material since this makes that those voxels have higher intensity in the filtered volume. In a similar way, negative difference values correspond to voxels near a boundary with a lower density material because in the filtered volume their intensities are lowered. Note that both positive and negative values are important.

3.2. Implementation

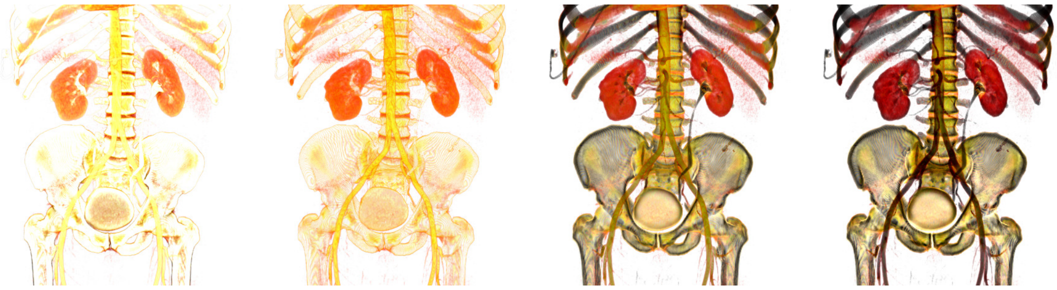
The spatial importance map is the result of subtracting the original model from the filtered model. Therefore, the main step is the computation of the filtered volume model. To implement the Gaussian blur we take advantage of the separability property of the 3D Gaussian function. That is, the 3D Gaussian function can be separated into the product of three 1D functions. From a practical point of view, this means that convolving the volume with a 3D Gaussian function is equivalent to convolving the volume with a 1D Gaussian function along one axis, then convolving the result again with a 1D Gaussian function along another axis, and finally convolving this last result with a 1D Gaussian function along the other axis. This results in a computationally cheaper implementation.

Moreover, we have to take into account another feature of the Gaussian function in the implementation. The Gaussian function extends infinitely at both sides, so we would have to take into account the full image to compute the filtered version of each voxel. In practice, though, values further than 3 are small enough to be negligible, so we can clamp the function at that distance. In our implementation, we have only one parameter regulable by the user, which is the radius of the Gaussian kernel measured in pixels (see Figure 2). Then, we define to be one third of the radius, so that all the values inside the radius are significant. The values in the kernel are sampled from the Gaussian function at the center of each voxel, and then we normalize them so that they sum 1.

The method has been implemented with CUDA to achieve real-time performance. We take advantage of the fact that both the filtering and the difference are parallelizable for each voxel. The steps of the CUDA implementation are the following:

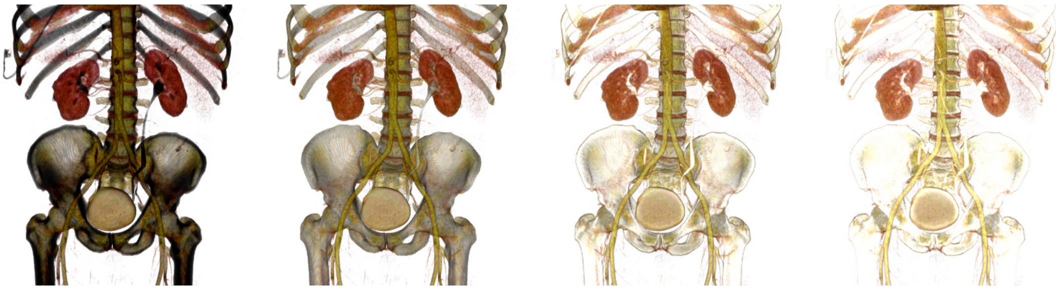
1. Prepare volume data converting it to floating point values.
2. Copy volume data to a 3D CUDA array and bind a 3D texture to it. This texture returns the real values (not scaled), and is accessed with non-normalized texture coordinates with clamp address mode and with nearest neighbour interpolation, so that it works like an array.
3. Allocate space in global memory to store the result of the filtering.
4. Compute the Gaussian kernel according to the radius and copy it to global memory.
5. Filter the array data in the X axis.
6. Copy the result of the previous filtering to the volume 3D array so it is the source in the next step.
7. Filter the array data in the Y axis.
8. Copy the result of the previous filtering to the volume 3D array so it is the source in the next step.
9. Filter the array data in the Z axis.
10. Copy the original volume data again to the 3D array.
11. Subtract the original volume from the filtered volume (the last result).
12. Copy the final result to host memory.
13. Clean up all allocated memory in the graphics card.

In Table 1, we collect the time required to compute the



(a) $= -10$ (b) $= 0$ (original) (c) $= 10$ (d) $= 20$

Figure 3: *Color modulation driven by spatial importance map of the CT body with ambient lighting.*



(a) $= -5$ (b) $= 0$ (original) (c) $= 5$ (d) $= 10$

Figure 4: *Color modulation driven by absolute spatial importance map of the CT body with local lighting.*

spatial importance map for the CT body, the CT head, and the salmon considering different radii of the Gaussian kernel. The *1D filter* column reports the time to do a 1D convolution with the kernel (this process has to be done once for each axis). The *Subtract* column reports the time required to do the subtraction. The *Total* column includes the time to do the 3 convolutions, the subtraction, and all the other additional operations like memory allocations and transfers previously described.

3.3. Modulation and Illustration

As it is illustrated in Figure 1, once the spatial importance map has been obtained it is integrated at the end of the visualization pipeline to obtain different effects. The modulation process considers the map and also the colors and/or opacities assigned by the illumination module to the volume. The set of effects that can be obtained depends on the considered parameters. We can use the raw importance values or convert them to absolute values. In addition, we can consider applying the modulation to the color assigned to the voxel or to the opacity. Moreover, we can use it to regulate the density of the dots in a stipple rendering. In the next section, we give a detailed description of the most representative effects.

4. Applications

To describe the different effects that can be obtained with the proposed approach, we grouped them into three different sections. First, we consider the modulation of the illumination model (effects on the colors), then the modulation of the transfer function (effects on the opacities), and at last the modulation of dot density in stippling. In all the following examples, the radius of the Gaussian filter is 10.

4.1. Color modulation

As we have shown in Section 3.1, the spatial importance map contains information about the boundaries distributed in positive and negative values, which can be interpreted as both sides of boundaries. To emphasize them, importance values can be used to modulate the color and contrast of the volume dataset. We propose different strategies that differ in how we consider the values of the map for the modulation. Being I the color of a voxel, the first strategy for adding the spatial importance value ΔD is by considering the raw values. In this case, we apply

$$I' = I + \Delta D \cdot \quad , \tag{3}$$

where is a factor used to modulate the effect.

The second effect we obtain is by taking absolute importance values instead of the raw ones. In this case, we obtain

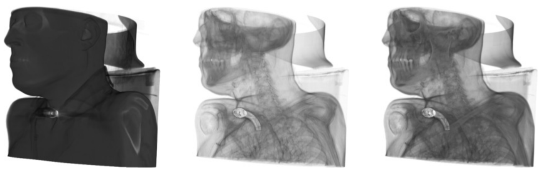
(a) original (b) $k_l = 0.5$ (c) $k_l = 1$ (d) $k_l = 2$ (e) $k_l = 5$

Figure 5: *Different opacity modulation effects with different parameters applied to the CT head model. (a) Original model, and (b-e) with $t_l = 1$, $t_h = 1$, $k_h = 1$ and modifying the k_l parameter.*

the final color with

$$I' = I + |\Delta D| \cdot \quad . \tag{4}$$

All these effects can be applied both to the ambient lighting or the local illumination. In Figure 3, we show the effects obtained using original importance values and ambient lighting on the CT body. From left to right, the values are -10, 0, 10, and 20. In Figure 4, we show the effects on the same model with local lighting and absolute importance values. From left to right, the values are -5, 0, 5, and 10. Observe that with raw values and a positive we obtain a darker image, due to the fact that most values in the map are negative; on the other hand, a negative produces a brighter image. With absolute values, the darkening and brightening effects are reversed because all the values are positive.



(a) original (b) $k_l = 1$ (c) $k_l = 2$

Figure 6: *Different opacity modulation effects using a default transfer function applied to the CT head model. (a) Original model, and (b-c) with $t_l = 1$, $t_h = 1$, $k_h = 1$ and modifying the k_l parameter.*

4.2. Opacity Modulation

Now we describe how to modulate the opacity. In this case, the idea is to use the information of the spatial importance map to increase or decrease the opacity in order to emphasize the most salient parts.

Being $A(z)$ the opacity of the voxel z, we compute the new opacity $A'(z)$ by

$$A'(z) = \begin{cases} A(z)k_l\overline{|\Delta D(z)|}, & \text{if } \overline{|\Delta D(z)|} < t_l, \\ A(z)k_h\overline{|\Delta D(z)|}, & \text{if } \overline{|\Delta D(z)|} > t_h, \\ A(z), & \text{otherwise,} \end{cases} \tag{5}$$

where t_l and t_h are the low and high thresholds respectively, k_l and k_h are factors to regulate the effect of the modulation, and $\overline{|\Delta D(z)|}$ is the absolute spatial importance value of the voxel z normalized in the range $[0,1]$. We use absolute values because we are interested in detecting the boundaries, and both positive and negative values give this information.

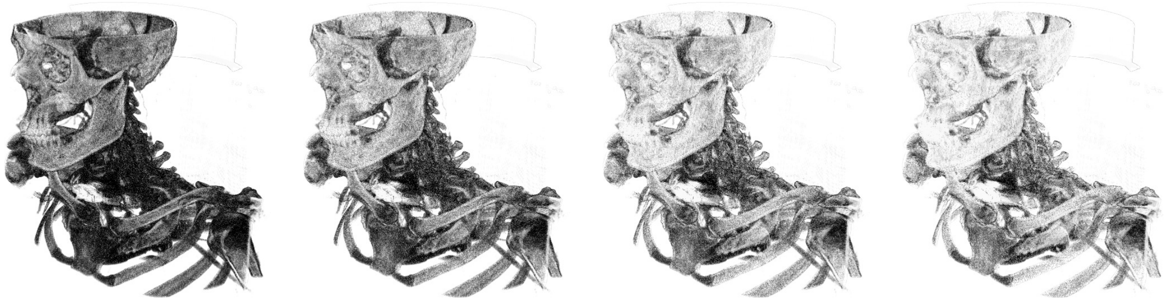
In Figure 5, we illustrate the opacity modulation effects applied to the CT head. Column (a) corresponds to the original model without any modulation. Columns (b-e) are obtained with $t_l = 1$, $t_h = 1$, $k_h = 1$ and k_l set to 0.5, 1, 2, 5. From Equation 5 we can see that k_h has no influence on the rendering since $t_h = 1$. Furthermore, since $t_l = 1$, nearly all voxels are multiplied by the spatial importance and the weighting factor k_l. As k_l increases, more detailed information about inner structures is captured.

Figure 6 shows that the same previous effects can be applied to the original model considering a default transfer function that is obtained by linearly mapping intensities to opacities and gray values. Note that the proposed approach can be used for first explorations of a volume dataset since no *a priori* knowledge is required.

4.3. Stippling

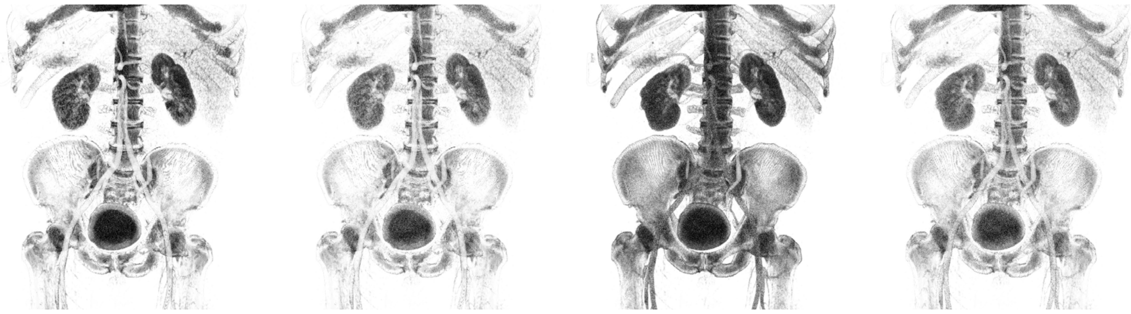
Stippling is an illustration technique in which the image is drawn using dots. This technique has been simulated algorithmically by several authors. Deussen et al. [DHvOS00] applied half-toning techniques to arrive at an initial stipple distribution and then interactively applied relaxation based on centroidal Voronoi diagrams. Secord [Sec02] used a fast probabilistic method in which stipples are automatically packed more densely in dark regions and more sparsely in lighter regions. Schlechtweg et al. [SGS05] created a multi-agent system to position the stipples. Sousa et al. [SFWS03] approximate stippling by using short, serrated ink strokes modeled directly over the mesh's edge.

In our approach, we propose to regulate the density of the dots according to the normalized absolute spatial importance map $\overline{|\Delta D|}$. We have two user-defined parameters: a threshold t above which everything is white, and a factor f that regulates the density scale. In addition, there is a random value $r(s)$ generated for each sample s in the ray casting. The color



(a) $t = 0.1,\ f = 5$ (b) $t = 0.1,\ f = 10$ (c) $t = 0.1,\ f = 15$ (d) $t = 0.1,\ f = 20$

Figure 7: *Stipple rendering of the CT head modulated by the spatial importance map with different parameters.*



(a) $t = 0.03,\ f = 10$ (b) $t = 0.03,\ f = 20$ (c) $t = 0.1,\ f = 10$ (d) $t = 0.1,\ f = 20$

Figure 8: *Stipple rendering of the CT body modulated by the spatial importance map with different parameters.*

of each sample $C(s)$ is defined as

$$\begin{cases} \text{white}, & \text{if } \overline{|\Delta D(s)|} \geq t \text{ or } f \cdot \overline{|\Delta D(s)|} \geq r(s), \\ \text{black}, & \text{if } \overline{|\Delta D(s)|} < t \text{ and } f \cdot \overline{|\Delta D(s)|} < r(s). \end{cases} \quad (6)$$

In Figures 7 and 8, this stippling effect has been used to render the CT head and the CT body, respectively, considering different parameters. This effect can also be applied in combination with the opacity modulation, as shown in Figure 9.

5. Conclusions

We have presented a new approach for obtaining illustrative volume renderings. The method computes a spatial importance map that captures information about the most salient parts of the model. This map is integrated in the visualization pipeline allowing to modulate color and opacity values. Such modulations have been used to obtain differenct effects that enhance volume data interpretation giving visual clues about structures contained in the volume. In addition, we have used the spatial importance map to modulate the density of the dots in a stipple rendering.

In our future work, we will explore the use of other low-pass filters, such as the trilateral filter, and evaluate how they behave compared to the Gaussian filter.

References

[BGKG06] BRUCKNER S., GRIMM S., KANITSAR A., GRÖLLER M. E.: Illustrative context-preserving exploration of volume data. *IEEE Transactions on Visualization and Computer Graphics 12*, 6 (2006), 1559–1569.

[CM09] CORREA C., MA K.-L.: The occlusion spectrum for volume classification and visualization. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009), 1465–1472.

[CMH*01] CSÉBFALVI B., MROZ L., HAUSER H., KÖNIG A., GRÖLLER E.: Fast visualization of object contours by non-photorealistic volume rendering. *Computer Graphics Forum 20*, 3 (2001), 452–460.

[CST05] CIGNONI P., SCOPIGNO R., TARINI M.: A simple normal enhancement technique for interactive non-photorealistic renderings. *Computer & Graphics 29*, 1 (2005), 125–133.

[DHvOS00] DEUSSEN O., HILLER S., VAN OVERVELD C., STROTHOTTE T.: Floating points: A method for computing stipple drawings. *Computer Graphics Forum 19* (2000), 40–51.

[FSG09] FEIXAS M., SBERT M., GONZÁLEZ F.: A unified information-theoretic framework for viewpoint selection and mesh saliency. *ACM Transactions on Applied Perception 6*, 1 (2009), 1–23.

[GSF08] GONZÁLEZ F., SBERT M., FEIXAS M.: Viewpoint-based ambient occlusion. *IEEE Computer Graphics and Applications 28*, 2 (2008), 44–51.

[HMBG01] HAUSER H., MROZ L., BISCHI G.-I., GRÖLLER M. E.: Two-level volume rendering. *IEEE Transactions on Visualization and Computer Graphics 7*, 3 (2001), 242–252.

Figure 9: *The CT body rendered combining stippling ($t = 1$, $f = 1$) and opacity modulation ($t_l = 1, t_h = 1, k_l = 1, k_h = 1$) effects.*

[IK01] ITTI L., KOCH C.: Computational modelling of visual attention. *Nature Reviews Neuroscience 2*, 3 (2001), 194–203.

[KD98] KINDLMANN G., DURKIN J. W.: Semi-automatic generation of transfer functions for direct volume rendering. In *IEEE Symposium on Volume Visualization* (1998), pp. 79–86.

[KKH02] KNISS J., KINDLMANN G., HANSEN C.: Multidimensional transfer functions for interactive volume rendering. *IEEE Transactions on Visualization and Computer Graphics 8*, 3 (2002), 270–285.

[KPI*03] KNISS J., PREMOZE S., IKITS M., LEFOHN A., HANSEN C., PRAUN E.: Gaussian transfer functions for multi-field volume visualization. In *Proceedings of the IEEE Visualization 2003 (VIS'03)* (2003), p. 65.

[KV06] KIM Y., VARSHNEY A.: Saliency-guided enhancement for volume visualization. *IEEE Transactions on Visualization and Computer Graphics 12*, 5 (2006), 925–932.

[KWTM03] KINDLMANN G., WHITAKER R., TASDIZEN T., MÖLLER T.: Curvature-based transfer functions for direct volume rendering: Methods and applications. In *Proceedings of the IEEE Visualization 2003 (VIS'03)* (2003), pp. 513–520.

[LCD06] LUFT T., COLDITZ C., DEUSSEN O.: Image enhancement by unsharp masking the depth buffer. *ACM Transactions on Graphics 25*, 3 (2006), 1206–1213.

[Lev88] LEVOY M.: Display of surfaces from volume data. *IEEE Computer Graphics and Applications 8*, 3 (1988), 29–37.

[LM04] LUM E. B., MA K.-L.: Lighting transfer functions using gradient aligned sampling. In *Proceedings of the IEEE Visualization 2004 (VIS'04)* (2004), pp. 289–296.

[LME*02] LU A., MORRIS C. J., EBERT D. S., RHEINGANS P., HANSEN C.: Non-photorealistic volume rendering using stippling techniques. In *Proceedings of the IEEE Visualization 2002 (VIS'02)* (2002), pp. 211–218.

[LVJ05] LEE C. H., VARSHNEY A., JACOBS D. W.: Mesh saliency. *ACM Transactions on Graphics 24*, 3 (2005), 659–666.

[MWCE09] MACIEJEWSKI R., WOO I., CHEN W., EBERT D.: Structuring feature space: A non-parametric method for volumetric transfer function generation. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009), 1473–1480.

[RBV*08] RUIZ M., BOADA I., VIOLA I., BRUCKNER S., FEIXAS M., SBERT M.: Obscurance-based volume rendering framework. In *Proceedings of IEEE/EG International Symposium on Volume and Point-Based Graphics* (2008), pp. 113–120.

[RE01] RHEINGANS P., EBERT D.: Volume illustration: Non-photorealistic rendering of volume models. *IEEE Transactions on Visualization and Computer Graphics 7*, 3 (2001), 253–264.

[RSI*08] RITSCHEL T., SMITH K., IHRKE M., GROSCH T., MYSZKOWSKI K., SEIDEL H.-P.: 3d unsharp masking for scene coherent enhancement. In *ACM SIGGRAPH 2008 papers* (2008), pp. 1–8.

[Sec02] SECORD A.: Weighted voronoi stippling. In *Proceedings of the 2nd international symposium on Non-photorealistic animation and rendering (NPAR'02)* (2002), pp. 37–43.

[SFWS03] SOUSA M. C., FOSTER K., WYVILL B., SAMAVATI F.: Precise ink drawing of 3d models. *Computer Graphics Forum 22* (2003), 369–379.

[SGS05] SCHLECHTWEG S., GERMER T., STROTHOTTE T.: Renderbots – multi-agent systems for direct image generation. *Computer Graphics Forum 24* (2005), 137–148.

[TIR05] TSOTSOS J. K., ITTI L., REES G.: A brief and selective history of attention. In *Neurobiology of Attention*, Itti L., Rees G., Tsotsos J. K., (Eds.). Elsevier, San Diego, CA, 2005.

[TLB*09] TAO Y., LIN H., BAO H., DONG F., CLAPWORTHY G.: Feature enhancement by volumetric unsharp masking. *The Visual Computer 25*, 5-7 (2009), 581–588.

[vdWGB06] VAN DE WEIJER J., GEVERS T., BAGDANOV A. D.: Boosting color saliency in image feature detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence 28*, 1 (2006), 150–156.

[VKG04] VIOLA I., KANITSAR A., GRÖLLER M. E.: Importance-driven volume rendering. In *Proceedings of the IEEE Visualization 2004 (VIS'04)* (2004), pp. 139–146.