Maximum Intensity Projection Weighted by Statistical Cues

Peter Mindek[*]
Supervised by: Ing. Peter Kapec[†]

Faculty of Informatics and Information Technologies
Slovak University of Technology in Bratislava
Bratislava / Slovakia

Abstract

Volumetric visualization of medical data is a specific task, as doctors and radiology technicians are not well trained in the field of computer graphics; therefore, algorithms for visualization of medical data must be as intuitive as possible, so that visualization tools employing them would be helpful in medical environment. Visualization of volumetric data acquired by medical imaging could not be effectively used without defining a proper transfer function, which transforms measured intensities to colours and opacity values. Many visualization methods use complex means for designing transfer functions, which could lead to decreased usability of said methods. We propose a modification of MIP, a common volumetric visualization method that uses a simple one-dimensional transfer function for classifying different materials. The goal of our method is to visualize individual tissues from the medical data, present them with minimal effort and enable users to observe areas of interest.

Keywords: Medical Data, Volumetric Visualization, Classification, Maximum Intensity Projection, Standard Deviation

1 Introduction

Volumetric visualization plays important role in medical imaging. Data acquired by CT (Computed Tomography) or MRI (Magnetic Resonance Imaging) scanners can be visualized using two-dimensional cross-sections and examined by moving these cross-sections through the data set. Another way is to visualize the data set as a whole by means of volumetric visualization. This approach has the advantage of displaying the data in their original context and it is better suited for some tasks.

Direct volume rendering (alpha blending accumulation of sampled values in front-to-back or back-to-front order) is a commonly used method for volumetric visualization. It is usually implemented using the volumteric ray casting algorithm. One of the disadvantages of this method is that it is rather slow (however, it can be accelerated by precalculation of gradients used for lighting) and it needs a proper transfer function, which is essential for the algorithm. It has to be properly designed so that it will correctly classify individual organs, tissues or materials represented by the volumetric data set. Designing such function is not a trivial task and it is often time consuming when done manually; consequently, it is not uncommon to use simpler methods of transforming intensities sampled from volume data along the viewing ray into pixel colour.

One possible solution is to visualize an iso-surface extracted from the visualized data set. [1] Major drawback of this rendering method is that only a small part of the data can be examined at once. Users do not see through the iso-surface, nor do they see anything above the rendered surface; therefore, this method is not suitable for visualizing complicated biological structures or tissues of human body, as they usually do not consist of voxels of the same intensity. However, it can be used to visualize bones in CT scans, or parts of objects scanned by industrial CT scanners.

In medicine, it is often necessary to display structures composed of materials with different intensity values. Where the direct volume rendering or the iso-surface extraction is not an option, simple visualization methods based on processing every sampled value along the rays can be used. A common way is to use some statistical property of the one-dimensional data sampled along the ray. This could be for instance standard deviation, mean value, or maximum value. Using the maximum value is a widespread method called Maximum intensity projection (MIP) [12]. These methods are non-photorealistic, as their goal is not to necessarily mimic real appearance of the visualized objects but rather provide adequate insight into the volumetric data.

Since these methods generally do not use any shading of sampled voxels, their advantage over direct volume rendering is in their speed. The speed enables these methods to be implemented in real time. This is important for usability of the rendering method, as dynamically changing view or other properties of the visualization is crucial for the medical imaging applications.

[*]mindek06@student.fiit.stuba.sk
[†]kapec@fiit.stuba.sk

[1]This could be done without ray casting as well; algorithms such as Marching cubes could be employed to extract the iso-surface and generate a polygon mesh that would be rendered using standard rendering capabilities.

Another major advantage of the MIP algorithm and similar methods is their simplicity. Very easy set-up of simple transfer functions or similar classification methods usually results in satisfactory images.

On the other hand, using only the statistical properties of the sampled data or an order independent-operators (such as a maximum operator used in the MIP rendering) has a drawback of discarding depth information of the visualized objects. Many different algorithms have been developed in order to maximize displayed information using these simple rendering methods [4, 11, 10, 1].

Additionally, single statistical property may not provide satisfactory representation of data of interest. For instance, the projection of maximum intensity is useful only when higher intensities are those of interest. This may be the case in the CT scans when the bones are to be visualized (bones have generally higher intensity than surrounding tissue), or in scans with application of contrast agents. However, areas of interest are often occluded by structures of higher intensity values in the MRI scans without contrast; therefore, MIP is not applicable in this case.

Sometimes the volumetric visualization algorithms use classification to enhance the final image [9]. The classification assigns colours, opacity values, or both (depends on the rendering method) to individual tissues or materials to help the user to easily distinguish them. One of the simplest and most intuitive methods of the classification is a one-dimensional transfer function. It is able to filter out some of the structures and reveal the areas of interest within the volumetric data set. However, blurryness and other factors cause this classification method to fail on some of the data sets when using MIP or similar visualization algorithms.

2 Related work

As one of the most commonly used volumetric visualization methods, the MIP rendering algorihtm (introduced in [12]) has been subject to many improvements. Its advantages make this method interesting, even though more complex visualization methods exist. Several enhancements that eliminate various disadvantages of MIP have been proposed.

Use of depth weighting has been proposed in [4]. Intensity values are weighted by a value dependant on a distance from the origin (position of virtual camera). This is essentially a fog effect applied to the MIP rendering. It results in images with distinguishable depth of individual visualized objects. However, in some cases it may partially occlude the areas of interest.

Another modification of the MIP method was proposed in [11]. Instead of using the global maximum of the values along the viewing ray, a first local maximum higher than a pre-selected threshold value is used. If no value is higher than the threshold, the global maximum is projected. An improvement of this method has been proposed in [10].

A method that combines advantages of MIP and direct volume rendering has been introduced in [1]. This method updates an opacity profile based on difference between the sampled value and a current maximum value.

Numerous volumetric visualization methods that provide additional information about visualized data have been introduced as well. The goal of these methods is to show as much information as possible without difficult transfer function design. The use of weighted distance transfrom has been proposed to enhance various features in rendered images of anatomical structures and to provide contextual information about selected body parts [5]. The gradient based rendering technique of objects boundaries was introduced in [2]. Additionally, visualization enhancement by level lines has been proposed in the paper. Probabilistic classification of different materials using several one-dimensional transfer functions has been proposed in [8].

Even though the classification methods using the one-dimensional transfer functions are widely used, volumetric rendering methods with multidimensional transfer functions are being examined as well. The problem of designing the multidimensional transfer functions has been also addressed [7]. Semi-automatic generation of transfer functions for direct volume rendering of boundaries between different materials within data set has been proposed in [6].

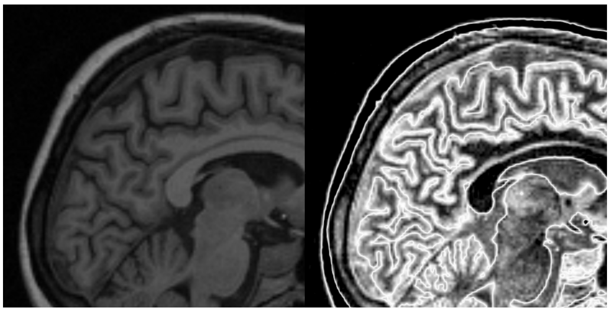
Statistical transfer function space has been introduced in [3] and addresses the problem of overlapping of intensity distributions of different materials that are to be classified. The method extracts statistical properties from the data and uses them to classify different materials. Described method uses adaptive growing approach to estimate statistical properties of each sample point. Estimation is based on neighbouring values. Extracted properties are also utilized to improve visual quality of volume shading by noise reduction.

3 Weighting by statistical cues

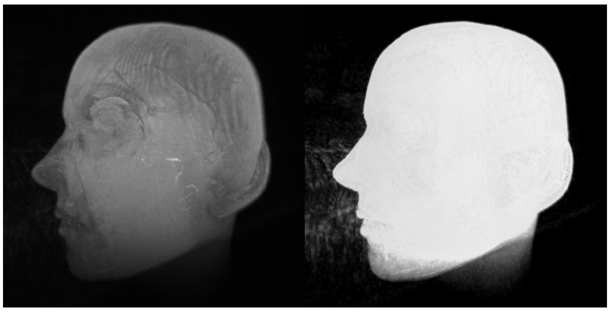
Our proposed approach is an enhancement of the standard MIP rendering with the distance weighting using the volumetric ray casting algorithm. The goal was to enhance the MIP rendering technique in such manner that it could be employed in rendering of CT or MRI scans with different areas of interest.

The algorithm uses a one-dimensional transfer function for material classification, which is designed by the user by placing and moving control points of a cubic spline. The transfer function assigns opacity values to individual voxels according to their intensity, which is used as an input for the transfer function. It could be used to suppress various ranges of intensities in the data set to reveal areas of interest.

Figure 1 shows the inability of a one-dimensional transfer function to classify a MRI scan in order to reveal the brain in the MIP rendered image. The goal was to remove the skull (assign it a zero opacity), which the one-dimensional transfer function failed to accomplish, as the voxels forming the outlines of the skull have the same intensity values as the brain tissue. Thin layers of voxels at the surface of the head after classification have the same or higher intensities than the brain tissue. MIP projects these voxels instead of the area of interest, which is the brain in this case.



(a) Cross-section without classification (b) Cross-section with classification



(c) MIP without classification (d) MIP with classification

Figure 1: Cross-section through an MRI scan of a human head without classification (1a) and with classification by a 1D transfer function (1b). Brain is occluded in the MIP rendering (1c), as well as in the MIP rendering with classification by the 1D transfer function (1d). In both renderings, higher intensity voxels of the skull occlude the brain.

Our algorithm tries to override this problem by using standard deviation of voxels along the projection ray to calculate the weight of the sampled values. These values are then treated as in the standard MIP rendering – the maximum of the weighted values is projected into the final image.

The standard deviation is calculated from n previously sampled values. These values form a window, whose size is constant during the rendering process. The size of the window should be as big as possible; however, voxel intensities should not be affected by voxels representing different materials. Bigger window size therefore requires use of a smaller step size for the ray traversing.

The algorithm uses parameter τ, which is adjustable along with the transfer function by the user. Weights of the sampled values are calculated by formula 1.

$$w_i = |2\sigma - \tau| \tag{1}$$

σ is the standard deviation of the window of i-th sample point, w_i is the weight. The sampled value is multiplied by the weight and finally, the maximum of the resulting weighted values along the ray is projected into the image.

The τ parameter can be adjusted by the user to further specify the area of interest. It serves as a parameter of a simple V-shaped transfer function for the standard deviation used to weight the sampled intensities from the data set. Appropriate value of the τ parameter can be found by trial and error by continually changing it while observing the final rendering.

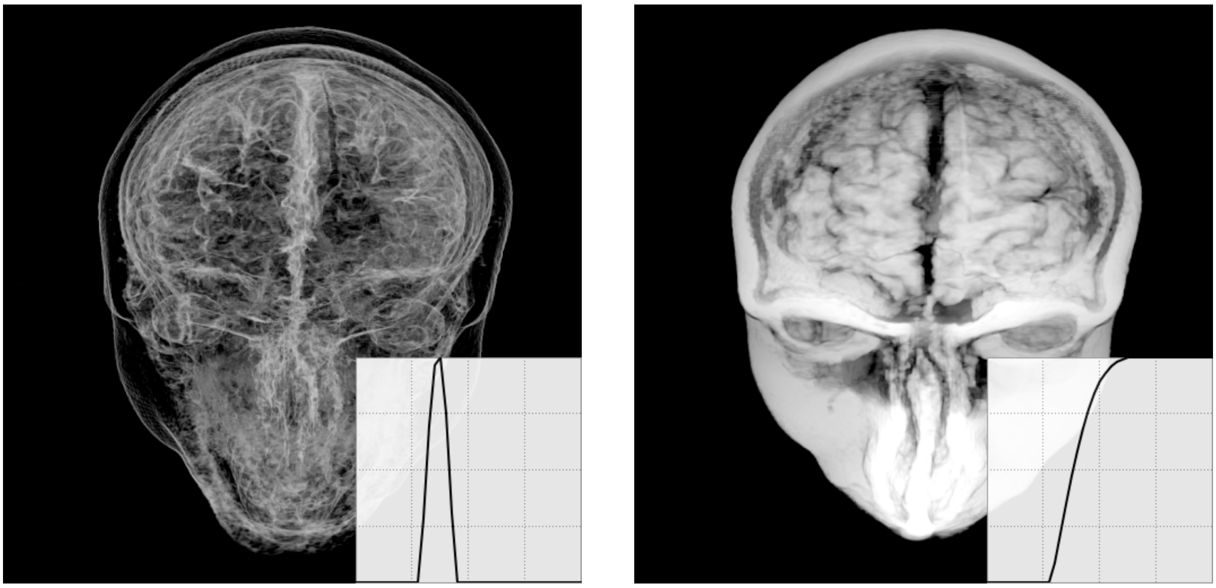
Our algorithm also uses the distance weighting immediately after sampling the value from the data set, as proposed in [4]. The depth of rendered objects is more perceptible this way. Figure 2 demonstrates the ability of our method to efficiently employ a 1D transfer function for material classification.

3.1 Implementation

The proposed method is based on the volumetric ray casting algorithm and processes one ray at the time. Colours of individual pixels in rendered image could be calculated independently; therefore, rendering could be parallelized. Implementation on state-of-the-art graphic hardware may take advantage of its massive parallelism and may result in interactive real time rendering.

We have implemented the method as a GLSL vertex/fragment shader pair using OpenGL library. The image is rendered as a single quadrilateral covering the whole screen while the vertex/fragment shader pair is in use. The vertex shader does not transform positions of rendered vertices with the model view matrix, but it uses this transformation to calculate the direction and position of the virtual camera. This design enables the use of the standard OpenGL matrix transformation commands to control the virtual camera of the shader.

The volume ray casting algorithm is implemented as a fragment shader described bellow. The fragment shader processes rays, as fragment colour is dependant only on the evaluation of its respective ray. Number of steps is taken in order to evaluate each ray. Every step consists of sampling two values along the ray. First value is sampled at the sampling position, which gradually move along the ray in constant intervals away from the camera position. Second value is sampled at the position n steps back. The steps, or the sampled values between these two positions are referred to as a window. The two sampled values (which are transformed by the transfer function, stored as a 1D texture) are used to calculate the standard deviation of the window. Standard deviation is calculated by following formula:



(a) Boundaries (b) Volumes

Figure 2: Comparison of a visualization of object boundaries and volumes by using our method with different transfer functions (x-axis of the transfer function represents sampled intensity, y-axis represents output opacity)
; $\tau = 0$ for both images.

$$\sigma = \sqrt{\frac{n(\sum_{i=1}^{n} x_i^2) - (\sum_{i=1}^{n} x_i)^2}{n(n-1)}} \tag{2}$$

$x_1 \ldots x_n$ are sampled intensities of the current window. The are calculated by trilinear interpolation of voxel values surrounding the sampling point.

To calculate the standard deviation in $O(1)$ time for the window of every sampling point, values of $\sum_{i=1}^{n} x_i$ and $\sum_{i=1}^{n} x_i^2$ are being accumulated as the shader marches along the ray. In every step, value sampled at the end of the window (farther from the camera) and its square are added to respective accumulators, while value at the begining of the window (closest to the camera) and its square are subtracted from the accumulators. This way, the formula 2 could be used to calculate the standard deviation for every window in constant time, as the accumulators would always contain the sum of the sampled values and the sum of their squares. Consequently, the window would not be centred at the current sampling point.

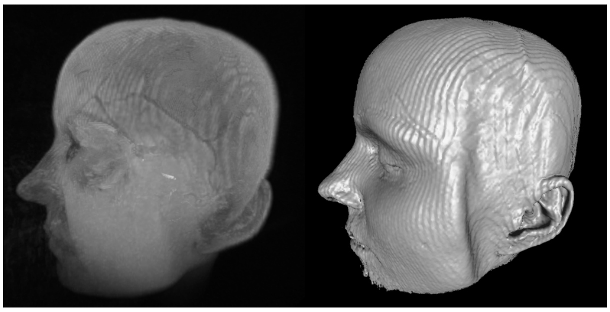
The volumetric data is enclosed in a bounding box in order to speed up the rendering. At the beginning of the evaluation of a ray, two intersections of the ray with the bounding box are calculated (in case the ray hits a vertex of the bounding box, these two intersections are equal). The sampling occurs only between these two positions in space. As the values are sampled from positions with uniform distances between each other, the number of steps for each ray vary. This helps to reduce the rendering time.

The volumetric data are stored as a 3D texture in the memory of the graphic card. This way, sub-voxel values (values from intermediate space between voxels) can be sampled from the data set. Using OpenGL commands, graphic card could be instructed to use the trilinear filtering for the sampling. The trilinear interpolation of values of neighbouring voxels significantly improves the rendering quality, even though it could introduce some minor artefacts to the visualized data. Listing 1 shows a fragment of the shader program implementing out method.

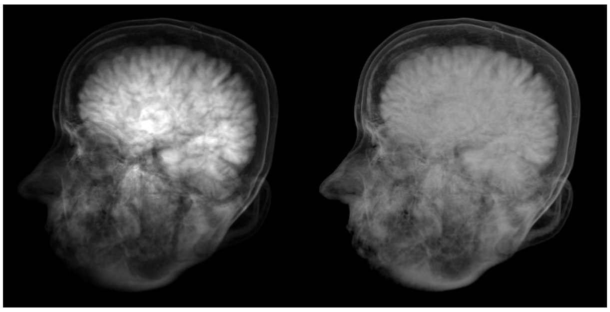
Listing 1: Part of the MIPWSC fragment shader; *p0* and *p1* are positions on the ray, *i0* and *i1* are step numbers on the begining and the end of the window, *t* is the τ parameter.

```
vox0 = dataRead(p0) * (1.0 - float(i0) /
    fogLen);
vox1 = dataRead(p1) * (1.0 - float(i1) /
    fogLen);

proj1 += vox0;
projSqr += vox0 * vox0;
proj1 -= vox1;
projSqr -= vox1 * vox1;
counter = float(i0 - i1);

s = pow((counter * projSqr - proj1 * proj1
    ) / (counter * (counter - 1.0)), 0.5);
vox0 = vox0 * abs(s * 2.0 - t);
if (vox0 > proj) {
    proj = vox0;
}
```

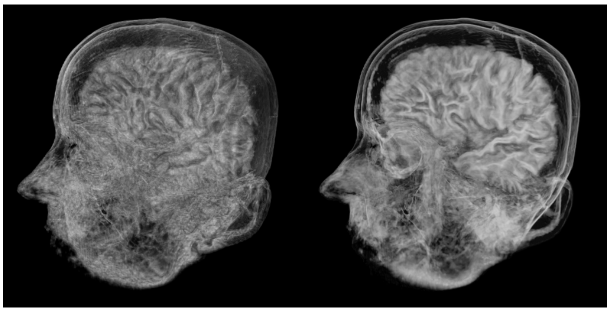
The listed fragment calculates the standard deviation of the window and weights the actual sampled value. The



(a) MIP (b) Iso-surface

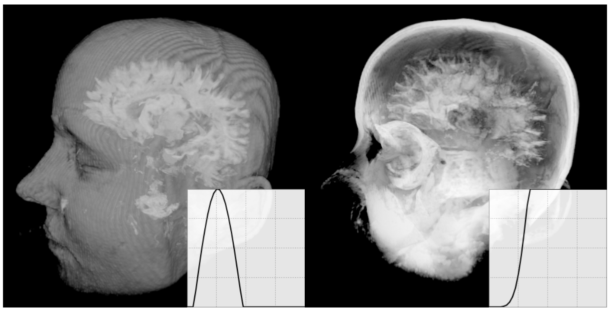


(c) AIP (d) SDP

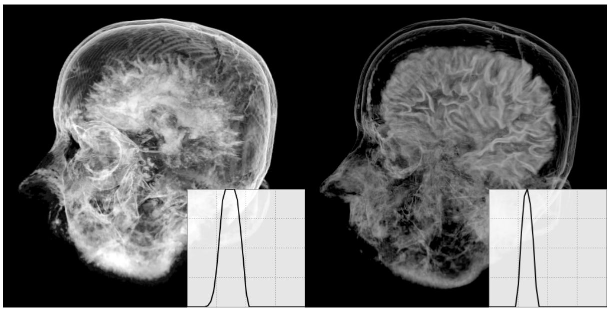


(e) DVR (f) MIPWSC

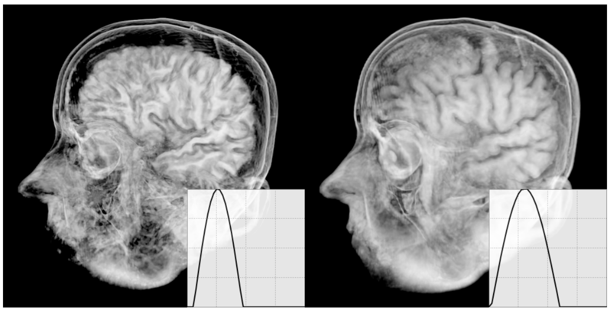
Figure 3: An MRI scan of a female head rendered with different volume visualization techniques using 1D transfer function for classification.



(a) $\tau = 0.6$ (b) $\tau = 0.3$



(c) $\tau = 0.3$ (d) $\tau = 0.2$



(e) $\tau = 0.2$ (f) $\tau = 0$

Figure 4: An MRI scan of a female head rendered using our method with different settings.

conditional branching at the end of the listing serves as a maximum operator for weighted values.

4 Results

As mentioned in Section 3, out method uses a one-dimensional transfer function for transforming intensities of voxels, window with adjustable length, and the τ parameter adjustable by users; consequently, great variety of result images could be achieved from a single data set. The transfer function and the τ parameter can be dynamically modified and are applied on the final rendering in real time. This enables users to find the ideal transfer function and the value of τ by trial and error.

Figure 3 compares several volumetric visualization methods with our proposed method. Drawbacks of individual rendering techniques are demonstrated: maximum intensity projection (3a) and iso-surface rendering (3b) are unable to efficiently use the transfer function classification to reveal the brain tissue. Average intensity projection (3c) and standard deviation projection (3d) can reveal brain occluded by skull, but the final images are blurry and do not show too much detail. Direct volume rendering (3e) is able to show the brain tissue in higher detail using a simple 1D transfer function, but the rendering quality is decreased by severe artefacts. Maximum intensity projection weighted by statistical cues (3f) is able to reveal brain in high detail; therefore, it is demonstrated that our method overrides some of the drawbacks of other commonly used visualization methods.

Figure 4 shows several images rendered using our visualization method as a demonstration of variability, which enables users to classify data in required fashion. One-dimensional transfer function and τ parameter are used to reveal different tissues.

Figures 5, 6 and 7 show MRI and CT scans of various human body parts rendered using the standard MIP method and our proposed method. One-dimensional transfer functions were used for the classification. The MIP algorithm failed to reveal areas of interest. Our algorithm was able to visualize the structural characteristics of individual objects in the scans.

The algorithm was tested on NVIDIA GeForce GTX 260. We have achieved interactive framerates (40 frames per second at 800x600 screen resolution) on the data sets containing 256^3 and 512^3 voxels. However, rendering speed depends on screen resolution.

5 Conclusions

We proposed a non-photorealistic method for visualization of volumetric data sets. Our approach is an improvement of the MIP rendering method with the goal of being simple, intuitive and fast, yet able to visualize different tissues or materials represented by the volumetric data. We have implemented the method in an interactive prototype application that uses real-time shader programs. We presented a comparison with commonly used methods of volumetric visualization. Results show that our method uses one-dimensional transfer function for classification more effectively and with better results than the other methods.

The algorithm could be improved in several ways. The length of the window used to calculate standard deviation could be adaptively changed during every step according to neighbouring voxels. This would make the algorithm more intuitive, although control over resulting renderings would be probably reduced in some cases.

Acknowledgements

This work was partially supported by the grant KEGA 244-022STU-4/2010: Support for Parallel and Distributed Computing Education. Volumetric data sets were obtained at `http://www9.informatik.uni-erlangen.de/External/vollib/`.

References

[1] Stefan Bruckner and Meister Eduard Gröller. Instant volume visualization using maximum intensity difference accumulation. *Computer Graphics Forum*, 28(3):775–782, June 2009.

[2] Balázs Csebfalvi, Lukas Mroz, Helwig Hauser, Andreas König, and Meister Eduard Gröller. Fast visualization of object contours by non-photorealistic volume rendering. Technical Report TR-186-2-01-09, Institute of Computer Graphics and Algorithms, Vienna University of Technology, Favoritenstrasse 9-11/186, A-1040 Vienna, Austria, April 2001. human contact: technical-report@cg.tuwien.ac.at.

[3] Martin Haidacher, Daniel Patel, Stefan Bruckner, Armin Kanitsar, and Meister Eduard Gröller. Volume visualization based on statistical transfer-function spaces. In *Proceedings of the IEEE Pacific Visualization 2010*, pages 17–24, March 2010.

[4] Wolfgang Heidrich, Michael McCool, and John Stevens. Interactive maximum projection volume rendering. In *Proceedings of the 6th conference on Visualization '95*, VIS '95, pages 11–18, Washington, DC, USA, 1995. IEEE Computer Society.

[5] Thomas Kerwin, Han-Wei Shen andf Brad Hittle, Don Stredney, and Gregory Wiet. Anatomical volume visualization with weighted distance fields. In *Eurographics Workshop on Visual Computing for Biology and Medicine*, 2010.

[6] Gordon Kindlmann and James W. Durkin. Semi-automatic generation of transfer functions for direct volume rendering. In *Proceedings of the 1998 IEEE symposium on Volume visualization*, VVS '98, pages 79–86, New York, NY, USA, 1998. ACM.

[7] Joe Kniss, Gordon Kindlmann, and Charles Hansen. Multidimensional transfer functions for interactive volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 8:270–285, July 2002.

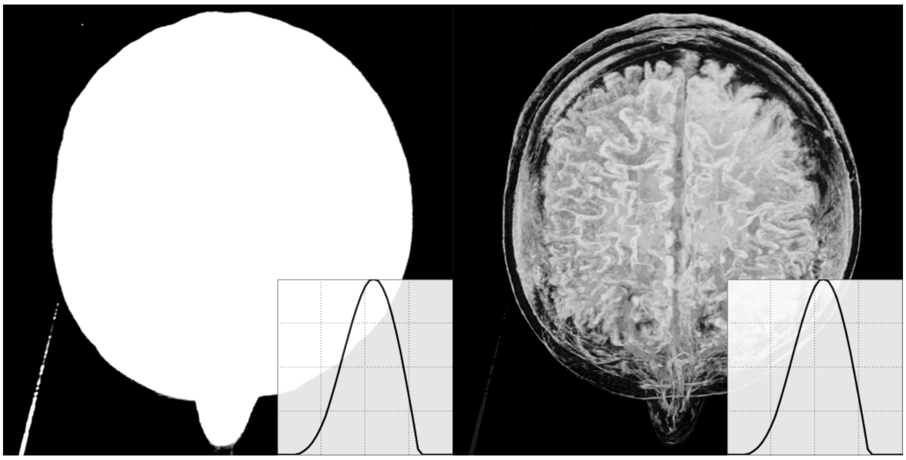
[8] Joe M. Kniss, Robert Van Uitert, Abraham Stephens, Guo-Shi Li, Tolga Tasdizen, and Charles Hansen. Statistically quantitative volume visualization. In *Proceedings of IEEE Visualization 2005*, pages 287–294, 2005.

[9] Marc Levoy. Display of surfaces from volume data. *IEEE Comput. Graph. Appl.*, 8:29–37, May 1988.

[10] Feng Ling and Ling Yang. Improved on maximum intensity projection. *Artificial Intelligence and Computational Intelligence, International Conference on*, 4:491–495, 2009.

[11] Yoshinobu Sato, Yoshinobu Sato Phd, Shin Nakajima, Nobuyuki Shiraga, Shinichi Tamura Phd, and Ron Kikinis Md. Local maximum intensity projection (lmip): A new rendering method for vascular visualization. *Journal of computer assisted tomography*, 22(6):912–917, 1998.

[12] Jerold W. Wallis, Tom R. Miller, Charles A. Lerner, and Eric C. Kleerup. Three-dimensional display in nuclear medicine. *IEEE Transactions on Medical Imaging*, 8(4):297–303, 1989.

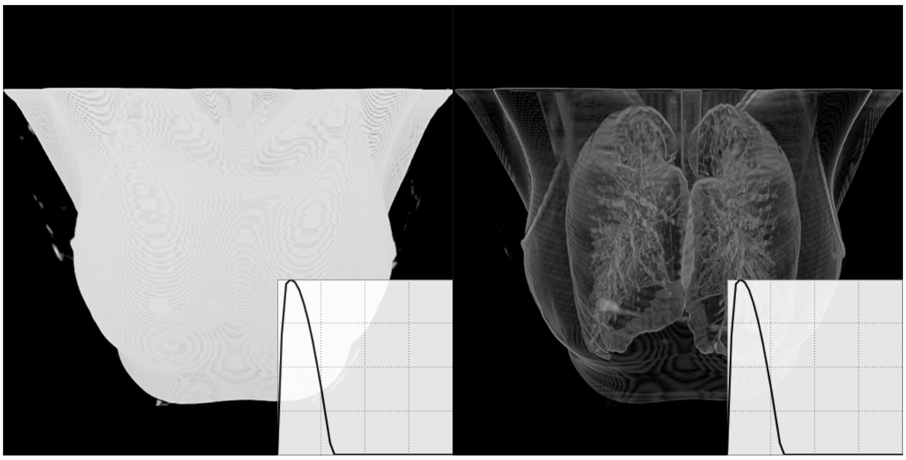


(a) MIP (b) MIPWSC, $\tau = 0.6$

Figure 5: A comparison of MIP and MIP weighted by statistical cues rendering of an MRI scan of a male head.

(a) MIP (b) MIPWSC, $\tau = 0.25$

Figure 6: A comparison of MIP and MIP weighted by statistical cues rendering of a CT scan of a knee.



(a) MIP (b) MIPWSC, $\tau = 0.34$

Figure 7: A comparison of MIP and MIP weighted by statistical cues rendering of a CT scan of a chest.