Non-Photorealistic Volume Visualization

Caroline Langer*
Martin Artner†

Seminar of Computer Science
Institute of Computer Graphics and Algorithms
Vienna University of Technology / Austria

Abstract

This paper is an introduction to non-photorealistic volume rendering and shall give an overview of the terms used and the main concepts. Recently the focus of research has concentrated on photorealistic rendering of 3D volume datasets. To overcome the constraints of photorealistic visualization new methods were found using non-photorealistic rendering techniques. We present a comprehensible introduction into non-photorealistic volume visualization, its objectives and four different approaches. First we introduce to the non-photorealistic volume rendering pipeline and show at which levels of the rendering process non-photorealistic effects can be achieved. The second one deals with volume illustrations, its methods and two possible implementations. Stipple drawings extend the idea of volume illustrations and focus on quick previews in a good quality. The last method described adapts pen-and-ink-rendering to volumes.

Keywords: non-photorealistic rendering, volume rendering, scientific visualization

1 Introduction

In the last decade scientific visualisation has become very popular. Many concepts and algorithms were published to achieve accurate and informative pictures in a short time or even in real-time. Focus was on exploring the data and the possibility to adapt to the viewers need. Especially direct volume rendering has proven to be very useful and versatile. In the past few years improvements were achieved to enhance the expressiveness of volume visualization by using non-photorealistic rendering methods. Non-photorealistic rendering has adressed 2D (paint)- and 3D (surface rendering)- systems but has ignored the visualization of volume rendering although hand-drawn volume illustrations can , e. g., already be found in early medical books.

We started our research with the paper which received most attention at the Visualisation 2002 conference [6] recognizing that the term ”non-photorealistic volume rendering” is used with different meanings and concentration on diverse aspects. In this paper we want to clarify the terminology of non-photorealistic volume rendering by giving a brief introduction into non-photorealistic rendering and volume visualization. After that we explain the aims of non-photorealistic volume visualization with focus on non-photorealistic volume rendering.

Then we will have a look at the volume rendering pipeline as described in [15] and explain at what stages mostly artistic effects can be achieved.

We continue with the presentation of some methods and approaches starting with a paper of the year 1999 [10], focusing on the perhaps most cited paper in this field [13] and including [1] and the siggraph courses [4, 8], which all give most attention to feature enhancement and depth and orientation cueing. Two different methods to implement these styles proposed by [13] and [7] will be sketched.

Finally we will present latest achievements referring to [6] which present a new system to interactively preview large, complex volume datasets in a concise, meaningful and illustrative manner.

Pen-and-ink Illustrations play a decisive role in non-photorealistic rendering. We sum up the adaptation of this technique to volume visualization described in [14].

We do not want to explain neither implementation details nor mathematical formulas but rather resume the work that has been done in this area in the past few years to reflect on different approaches and ideas, and describe the effects that can be accomplished when applying non-photorealistic methods to volume rendering.

2 Topic-Specific Terms

2.1 Non-Photorealistic Rendering

The traditional way of computer graphics is the process of converting a virtual scene into a photorealistic picture. But sometimes the effect of being non-photorealistic is better suited ,e. g., for cartoons, artistic expression, or technical illustrations. Non-photorealistic rendering is the means of creating imagery that does not aspire to realism [2]. Illustrations are often capable of conveying the information better as they can focus the viewers attention on rele-

*e9825633@student.tuwien.ac.at
†e9825089@student.tuwien.ac.at

vant features, omit extraneous details, clarify and simplify shapes and expose parts that are hidden [16, 12].

Typical fields where non-photorealistic rendering is applied, are pen-and-ink illustrations, painterly rendering and technical illustrations.

Pen-and-Ink Illustrations : The main elements of pen-and-ink illustrations are *Strokes* (curved lines of varying thickness and density), *Textures* (character conveyed by a collection of strokes, indication enables a selective addition of detail), *Tone* (perceived grey level across image or segment) and *Outline* (boundary lines that disambiguate structure).

Painterly rendering : The goal of painterly rendering is to simulate different styles as impressionism, expressionism or pointilism etc. A Brush Stroke is defined through its line centre, orientation, thickness and length. There are various algorithms for placing them and determining other parameters such as colour, curvature and boundaries.

Technical Illustrations : In technical illustrations different levels of abstraction are used. Extraneous details are attenuated or eliminated and important properties are accented. Lines - mostly in black- disambiguate the outlines. Different line thickness has a different meaning. Diffuse shading with an added hue shift or added outlines or metal shadings but also colour temperature (cool to warm shading) are often utilized in technical illustrations.

2.2 Volume Visualization

Visualization of scientific computations is a rapidly growing application of computer graphics. It serves to explore, analyse and present huge datasets. The data sets are recorded by medical applications, e. g., computed tomography (CT), magnetic resonance imaging (MRI), ultrasound or applications in engineering and science as computational fluid dynamic (CFD), aerodynamic simulations, meteorology, astrophysics, etc.

Volume visualization aims at gaining insight into the data showing as much as possible and still keeping track of data. To find areas of interest applications to change view and settings interactively are designed. In the literature two basic approaches can be found: *volume rendering* and *surface rendering* [3].

When applying **surface rendering**, regions of same level generate a 3D model of surfaces, which are converted to triangles. In the rendering pipeline only primitives have to be drawn. This method is very useful when data is loose or disguised, the objects are opaque or the isosurface is of interest.

On the other hand **volume rendering** is good for complex objects or when isosurfaces are unnatural. The volume is rendered directly so the data is seen as given, not only isosurfaces are rendered. Each voxel is converted to a RGBA-value which enables semintransparency. This transformation is performed with help of transfer functions.

In this paper we are focusing on recent developments in volume rendering.

2.3 Why Non-Photorealistic Volume Visualization

Strictly speaking, traditional volume rendering is not necessarily a form of photorealistic rendering since it often involves the visualization of values like temperature, pressure or density that are not directly visible in reality [8]. Furthermore it is often impossible to find one transfer function which generates photorealistic pictures for a class of data as the same values may have distinct meanings in different datasets and the spreading of values varies among different datasets.

Photorealistic rendering of scientific data can sometimes imply a precision that is not quite there whereas non-photorealistic rendering assumes that rendering is only an approximation of reality [10]. This may be a desirable effect to influence the observer.

The main problem of volume visualization as presented in 2.2 is, that the user has to know about the specific data in advance to gain insight. It usually is very difficult to find a good transfer function. Furthermore the expedience of the visualization may depend on the viewing direction or the light source. Especially the opacity is difficult to adjust and may produce pictures of distinct quality. So one aim of non-photorealistic rendering is to find a method to create a transfer function automatically that fits the data to give a general survey or show remarkable regions.

In 2.1 we have introduced the usefulness of illustrations especially in technical illustrations as drawings clarify the structural or conceptual information by minimizing secondary details. Also in medical books human-drawn illustrations are used rather than photographs as they communicate the information more comprehensively. This may be initiated by adapting non-photorealistic methods (such as tone shading, silhouette enhancement, or even pen-and-ink rendering) for surfaces to volumes to emphasize most important or significant features. By highlighting important details rendering time may be reduced significantly [6].

By varying and manipulating the volume visualization methods also artistique impressions can be achieved. Some sample pictures can be seen in Section 3.

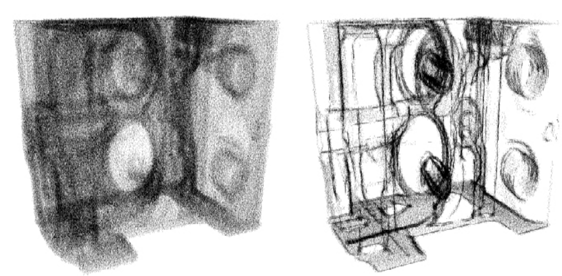


Figure 1: Motorblock with default volume rendering and boundary and silhouette enhancements

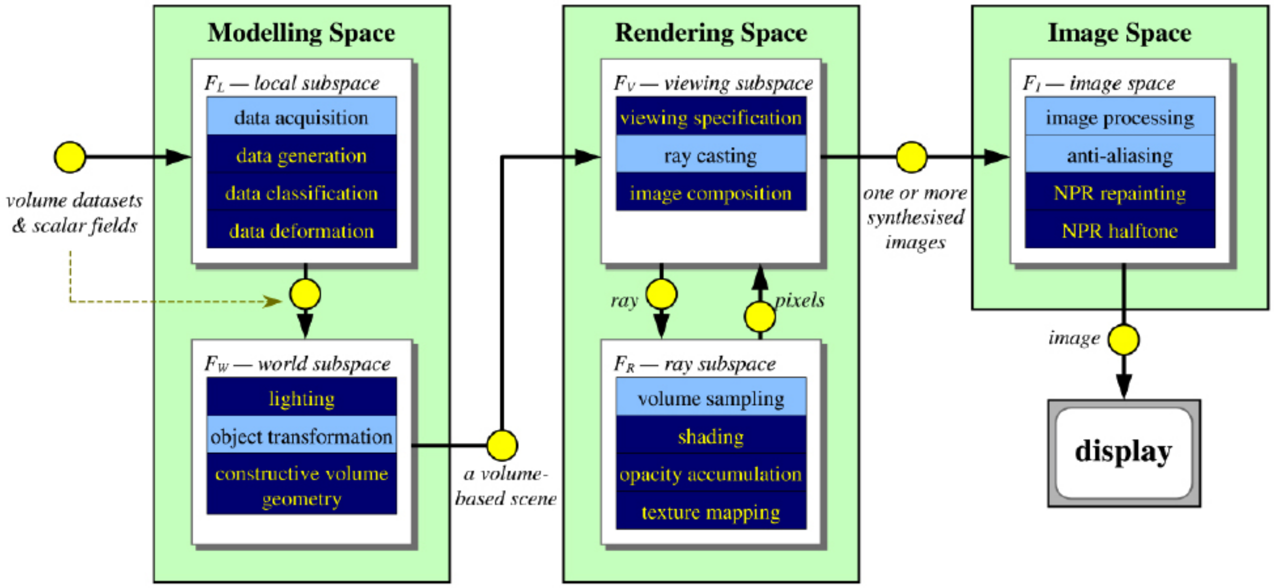


Figure 2: A volume based graphics pipeline.

3 Where to apply NPR effects

Through the past years a lot of work in Non-Photorealistic Rendering has been published. We will now discuss in more detail where in the rendering pipeline different effects can be created.

In Section 3.1 a volume-based graphics pipeline presented in [13] will be introduced, afterwards in Section 3.2 and Section 3.3 the stages which can be influenced during the rendering process are described in more detail.

3.1 A Volume-Based graphics Pipeline

The general structure of a volume-based graphics pipeline, as shown in Figur 2 highlights operational components that incorporate possible starting points for non-photorealistic effects with dark blue boxes. The three main stages namely *modelling space*, *rendering space*, and *image space* make the generation of different classes of effects possible. The amount of information over the rendering process available in each stage and the operations performed therein influences strongly the outcoming results.

3.1.1 Modelling Space

The most commonly-used representation of volume data consists of a regular grid of voxels, each associated with a value. The values at points in between of the voxels are obtained by tri-linear interpolation of the neighbourhood voxels. This homogenous data organisations allows simple data structures and efficient implementations of algorithms, unfortunately it has a high memory consumption.

3.1.2 Rendering Space

The concept of rendering space as used here has two aspects of direct volume rendering, the viewing subspace F_V and the ray subspace F_R. The viewing subspace can contain one or more image buffers which can be used for some interesting effects. From the eyepoint, rays are cast into the scene, where F_R and F_V operate in a master-slave fashion. In comparison to the techniques used in modelling space, effects applied here do not change the scene permanently. The results can be repeated by using the same rendering technique again, but the only remains are in the produced output images. The great amount of information available in this stage leads to a great number of effects which can be produced. One point which should be noted is that many effects are implemented in F_R and therefore they are view-dependent. Which means that additional care must be taken in generation of coherent frames for animation.

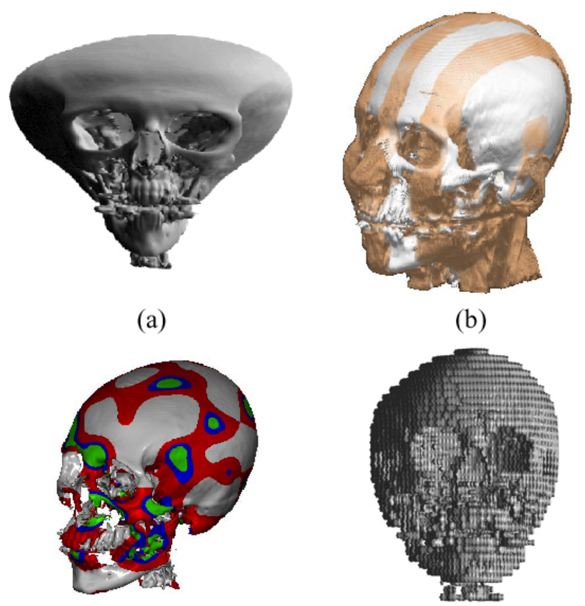
3.1.3 Image Space

In image space all commonly known 2D techniques to manipulate images are settled. This is, to date, the most extensively exploited domain for generating NPR effects. In some applications only the 2D image is available and the algorithms have to deal with this limited view to the former 3D space.

In the presented pipeline partial 3D information associated with the original 3D object can be passed to image space. Often the generation of this additional data is a necessity in rendering space, and so consumes little additional extra computing time. As only 2D Buffers are used the amount of memory needed is negligible compared with the storage space for the volume dataset. Because of the additional information this is also know as 2^{+}D rendering.

3.2 Artistic Modelling

In volume graphics are two general approaches for the introduction of expressive effects into the model, by manipulating existing features and by introducing NPR features directly.



(a) (b) (c) (d)

Figure 3: Examples of artistic manipulation.

A very powerful branch of these effects can be produced by changing the threshold value of iso-surfaces and manipulating the normal estimation during rendering. In Figure 3 (a) a distortion filter was used to alter the volume. In Figure 3 (b) the opacity of the skin iso-surface has altered. Nice looking effects can be gained by defining stroke volumes and introducing them into the volume object like in Figure 3 (d) spherical strokes have been applied to a skull volume. The inherent presence of 3D opacity and color information allows to model NPR effects, which have been introduced as textures in surface-based pipelines, more accurate. Figure 3 (c) demonstrates the usage of such a 3D texture.

3.3 Artistic Rendering

There are two subsections in direct volume rendering, namely *ray casting* and *forward projection*. The first is classified into *direct volume rendering* which treats the volume as an amorphous object, and *direct surface rendering* which is interested in rendering one or more iso-surfaces in the volume. The rendering stage in the presented pipeline is based on ray casting and supports both, the direct volume rendering and direct surface rendering.

In the rendering space we have the richest concentration of information that can be used to generate a big amount of expressive NPR effects. In this section we will take a closer look to the stages in the rendering process where NPR-effects can be introduced.

3.3.1 Viewing System

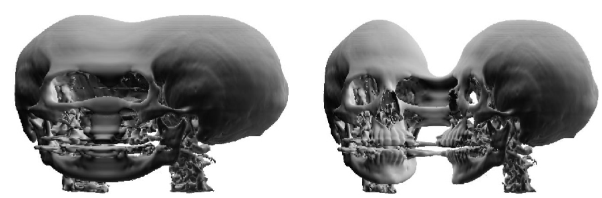


Figure 4: 'Meeting of minds'.

In traditional computer graphics the viewing system is designed to simulate a conventional camera. Any manipulation of this system will lead to anomalies in the resulting image. By careful control of these manipulation it can lead to pleasing expressive effects.

In Figure 4 the normally flat image plane has been transformed into a curved plane, and with it the direction of each ray fired into the scene. This gives the impression of two skull merging together where there is in fact only one in the volume representation. In the image on the right a tighter curve is used than in that on the left.

3.3.2 Normals

In ray casting the rays are sampled in regular intervals searching for an intersection with a specific iso-surface. When the iso-surface is hit by the ray, a normal is estimated, often by using the central difference method. Manipulation of this gradient data has been used in surface based graphics for bump mapping, which is perhaps the most well known example of such effects.

In Figure 5(a) the normals, after being rotated to the ray direction, are drawn as strokes on to the image. A coarsely drawn pencil sketch effect is the result. Like in an photorealistic illumination model, light sources can influence the size and density of NPR strokes as visible in Figure 5(b).

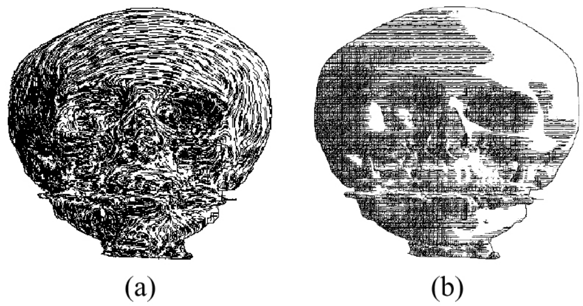


Figure 5: Painting with normals.

3.3.3 Color and Opacity

The modification of opacity and color which is often defined over transfer functions, enables rendering in context and focus. The interesting part of the volume has zero opacity and therefore is fully visible, but the surrounding volume of less interest is still visible to give an idea of the relationships.

3.3.4 Geometry

The detailed knowledge of position, size and shape of objects in rendering space can be used to add effects to a scene.

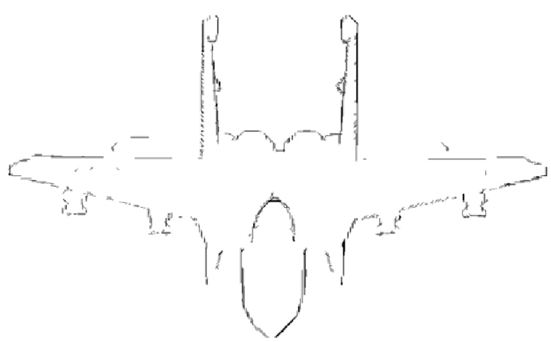


Figure 6: Using distance Information.

For example from the 3D scene the distances between the objects and the viewpoint can be extracted. This depth information can be used to decide which pixel is part of which object, especially when overlapping objects have the same color or are similarly shadowed. In Figure 6 the depth information was used to render an outline image of a plane. It is furthermore possible to use the length of viewing rays to calculate distance attenuation.

3.3.5 Texture Mapping

This technique is often used in both photorealistic and NPR graphics, which applies surface details to objects by using 2D or 3D textures during the rendering process.

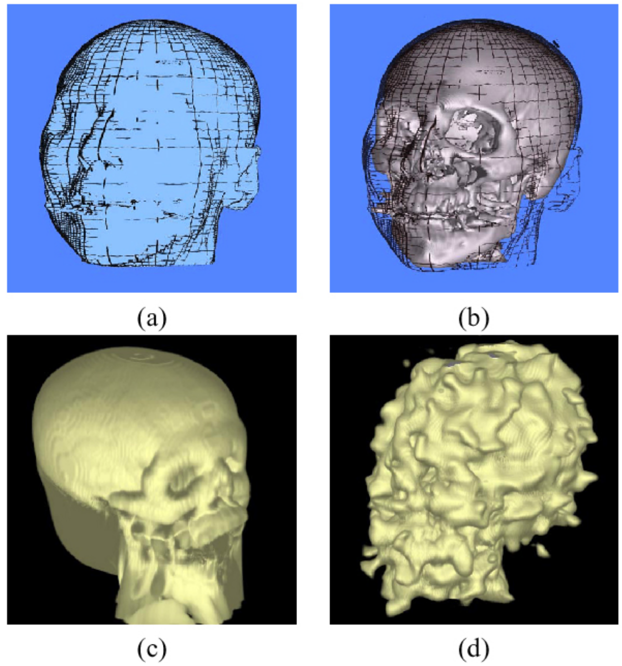


Figure 7: Texture mapping.

Figure 7 (a) shows an image with a mathematically defined texture, we will discuss this algorithm in more Detail in Section 6. It is also possible to combine photo-realistic and NPR techniques as presented in Figure 7 (b).

3.3.6 Hypertextures

In difference to traditional textures which exist only on the surface of objects, hypertextures fill some of the space around the object.

In the image in Figure 7 (c) a melting hypertexture is applied to the skull dataset. The effect of a low frequency noise applied across the whole skull surface can be seen in Figure 7 (d).

4 Volume Illustration

4.1 Adapting NPR Techniques to Volume Rendering

Many ideas in this field were first contributed by [13]. The focus lies on strengthening the appearance with volume illustrations. The pictures were taken from [13, 1] and [9] where the concepts introduced in the papers [13] and [1] were implemented.

4.1.1 Feature Enhancement

Surface Enhancement/ Boundary Enhancement

One of the main advantages of volume rendering com-

Figure 8: original gaseous rendering compared to rendering with boundary enhancement compared to rendering with boundary and silhouette enhancement, taken from [13]

pared to surface rendering is the ability of visualizing continuous regions of semi-transparent material. However enhancing surfaces can still be very useful in clarifying some structures in the volume. By enhancing the surfaces and decreasing the opacity of regions of same level behind, more and deeper lying features can be made visible as the ray travels further into the volume. Even thin areas of the same value are more visible. The surfaces of a volume are drawn more opaque (see Figure 9).

One extended approach is that the user can determine which gradients to use. By considering small gradients or only allowing large ones the level of enhancement may be adapted.

Drawing dark lines around the contours of an object is also very popular. These lines can also be used to help indicate spatial relationship between objects especially when these objects are semitransparent and overlapping (see also Section 3.3.4).

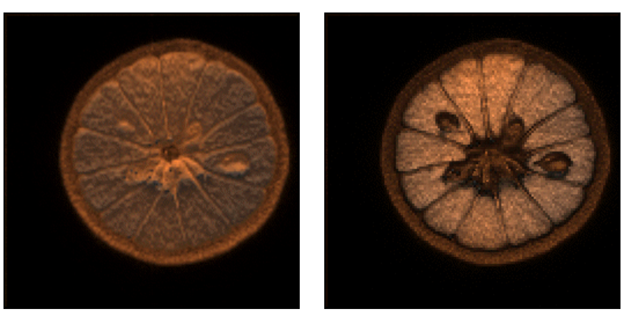


Figure 9: Orange with default volume rendering compared to boundary enhancement, taken from [9]

Silhouette Enhancement

One of the most basic image enhancement is using silhouettes. Silhouette lines are particulary important in the recognition of surface shape. Enhancing these outlines increases the perception of volumetric features. By increasing the opacity of volume samples where the gradient is perpendicular to the view direction emphasizes the silhouettes.

Silhouette edge rendering can be particulary useful in volume rendering applications since transfer functions are often set such that objects are semi-transparent, sometimes making spatial relationship difficult to determine [8]. In Figure 8 the effect of the described methods can be followed.

Oriented Fading

This technique emphasizes the orientation of the volume objects again (see Figure 10). However this time structures oriented towards the viewer are desaturated and those away from the view are darkened and saturated. These effect can be simulated by allowing the volumetric gradient orientation to the viewer to modify the colour, saturation, value and transparency of the given volume sample.

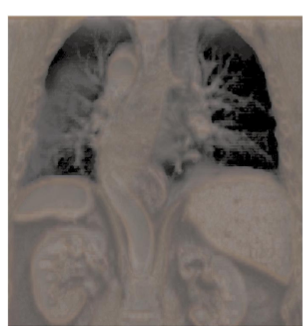


Figure 10: Fading, taken from [13]

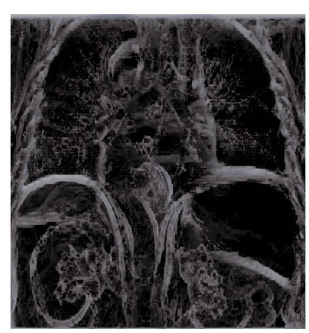


Figure 11: Volumetric Sketches, taken from [13]

Volumetric Sketchlines

Decreasing the opacity of volume samples oriented toward the viewer emphasizes the contours of the volume. In the extreme cases, only the sketches can be seen. Figure 11 is a significant picture where non-volumetric features are faded out.

Visualization of Volume Contours

[1] focuses on visualizing volume contours similar to the sketch lines technique described above in Volumetric Sketchlines. The great achievement is that the contours of many features can be seen at the same time (as of the skin, the bones and the tissue in Figure 12).

The gradient magnitude and the angle between view vector and gradient determine whether a voxel is part of a contour. The two components are multiplied together to give the final intensity at that voxel location. Then the maximum intensity projection is used for compositing so that all contours are visualized regardless of opacity.

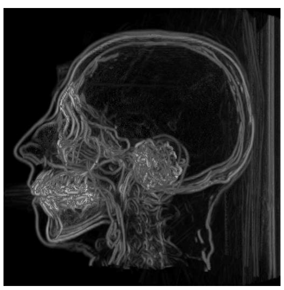


Figure 12: Volumetric contour of a skull, taken from [1]

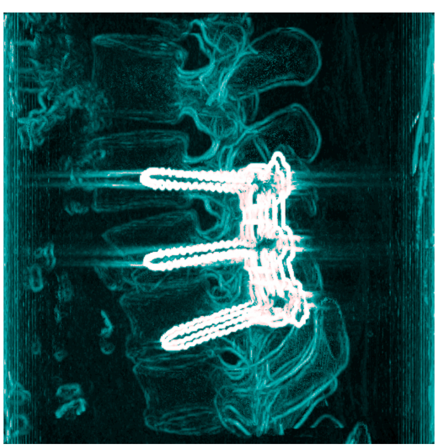


Figure 13: Volumetric contour of a human vertebrae with screws inside, taken from [1]

It would have taken quite a long time to find an appropriate transfer function to detect the screws in Figure 13 if traditional direct volume rendering was applied. Furthermore any additional lines would be more confusing than helpful.

4.1.2 Depth and Orientation Cues

Toneshading

Colour can be manipulated based on illumination to improve depth perception. In real life objects are illuminated by the sun and get a warm colour touch on the surface facing the sun and a cool touch on the opposite. This suggestive illumination by a warm light source is also used by painters. This effect can be accomplished by modulating each voxel colour with a lighting texture that manipulates colour temperature. Yellow is applied on the warm side whereas blue is used to simulate cold colours in Figure 14. In the middle the colour values are interpolated.

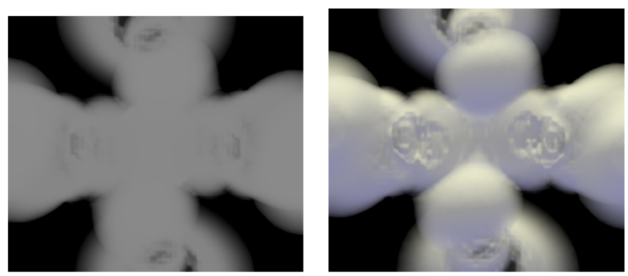


Figure 14: no toneshading in comparison to toneshading with yellow and blue of iron protein, taken from [13]

Distance Color Blending

Depth cueing is important in favour of we can easily identify the front and the back of displayed objects for a particular viewing direction. Usually the color or the intensities are varied according to their distance from the viewer. Part of objects closest to the viewing position are displayed with the highest intensity whereas the colour of objects farther away is dimmed. Coloring objects blue which are in the back and yellow those in the foreground is one possibility used in [13]. In Figure 15 just the background is colored blue.

Feature Halos

Illustrators sometimes use null halos around foreground features to reinforce the perception of depth relationships within a scene. The effect is to leave the areas just outside surfaces empty, even if an accurate depiction would show a background object in that place. Halos are created primarily in planes orthogonal to the view vector by making regions just outside features (in the plane orthogonal to the view vector) darker and more opaque, obscuring background elements which would otherwise be visible (Figure 16).

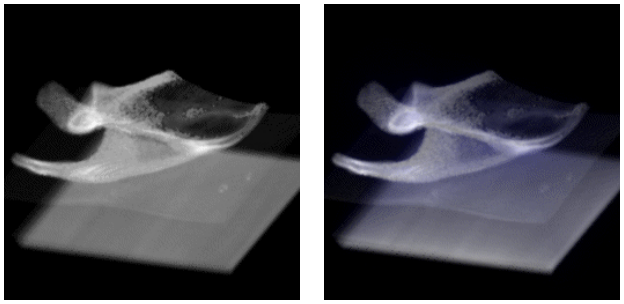


Figure 15: distance color blending (blue in the back) of a human scapula, taken from [9]

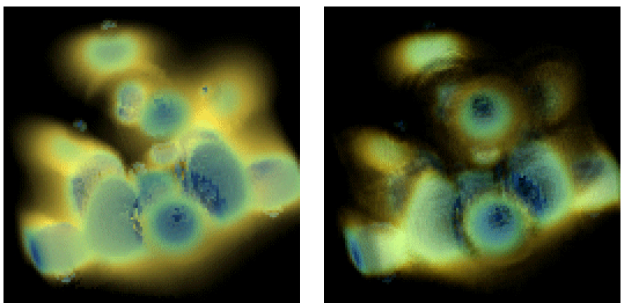


Figure 16: iron protein molecule (neghip dataset) without and with feature halos, taken from [9]

4.1.3 Regional Enhancement

Especially in technical ilustrations the level of detail is varied in one picture. Corresponding to volume rendering the user may interactively choose some regions of greater or just different enhancement.

4.2 Creation of an Illustrative Transfer Function

As discussed in 2.3 transfer functions used in traditional volume rendering have many drawbacks. They require in-depth knowledge of the data and need to be adjusted for each data set. One goal may be to find a suitable transfer function when opening the data file which also considers non-photorealistic effects.
By considering a combination of voxel values and gradients, surfaces, boundaries and silhouettes can be taken into account. However the adaptation of opacity values to enhance these features would be specified not only by voxel value and gradient but also by voxel location and view vector which makes a total of ten parameters. To implement multivariate colour transfer functions to accomplish oriented fading, distance colour blending, and tone shading as discussed in 4.1, voxel value, initial voxel colour, voxel location, gradient, view vector and light position and intensity have to be taken into account which makes a total of 17 parameters in addition. The transfer function would finally be very complex to compute but work is already done by breaking this huge function into smaller design problems.
Considering feature halos would even more complicate the problem as the value, location and gradient of each voxel in the neighbourhood would have to be taken into account.

4.3 Architectures for Expressive Volume Rendering

4.3.1 Ray-Based Solution in Software

In the traditional volume rendering pipeline which is sketched in Figure 17, the volume enhancement relies on functions of the volume sample values. The classification of the volume data is independent of the shading.
In dependence upon this pipeline, [13] developed an adapted volume rendering pipeline for volume illustration (Figure 18). The volume illustration techniques (including volume sample location and value, local volumetric properties, such as gradient and minimal change direction, view direction and light information) are fully incorporated into the volume rendering process.

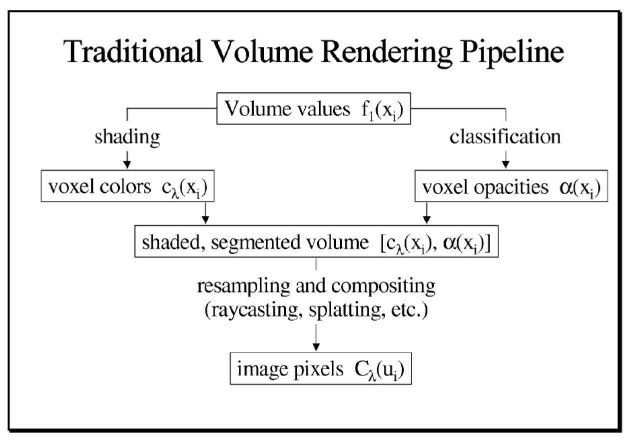


Figure 17: Traditional Volume Rendering Pipeline, taken from [13]

4.3.2 Hardware-accelareted Parallel Rendering Using Textures

In [7] a hardware-based volume renderer method is proposed which makes extensive use of the multi texturing capabilities of modern graphics cards. Multi-texturing allows several textures to be combined on a single polygon during the rendering process. Several seperate volumetric textures that store scalar data value, gradient magnitude and gradient direction combined with color palettes enable many different non-photorealitic techniques in hardware. Through the manipulation of the pallette over time,

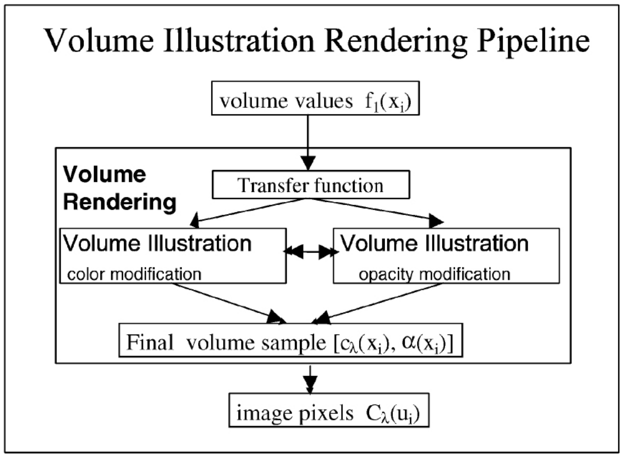


Figure 18: Volume Illustration rendering pipeline, taken from [13]

the textures can be varied based on the viewing parameters without changing the data in the textures themselves.

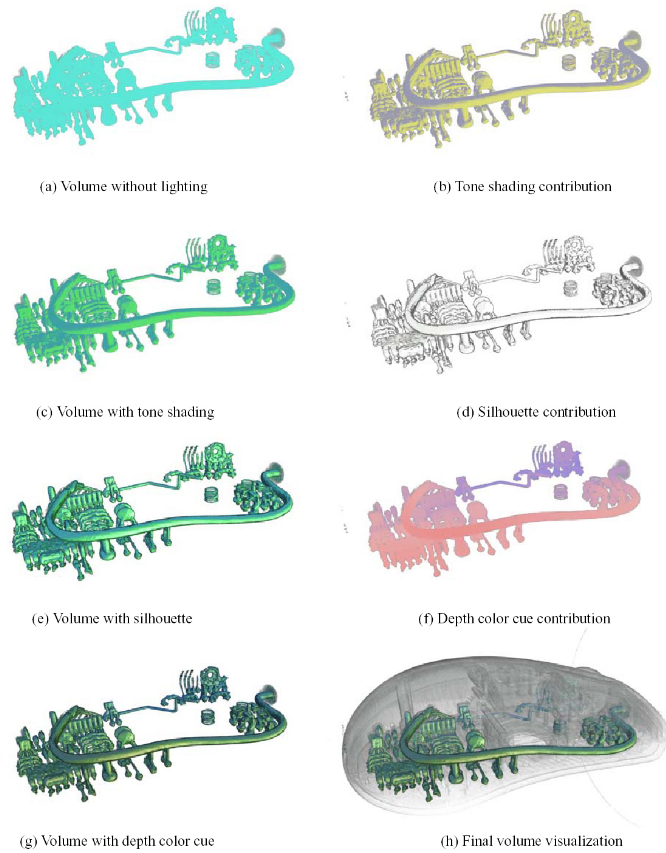
As sketched in Figure 19 four texture units are used for each rendering pass whereas the first pass renders the tone shaded volume while the second pass contains silhouette and specular contributions. The result is shown in Figure 20.

		Pass I	Pass II
Texture Unit 1	Texture	Scalar data values	Scalar data values
	Palette	Color & opacity maps	Opacity map
Texture Unit 2	Texture	Gradient directions	Gradient directions
	Palette	Tone shading	Silhouettes & specular
Texture Unit 3	Texture	Gradient magnitude	Gradient magnitude
	Palette	Surface enhancement	Surface enhancement
Texture Unit 4	Texture	Distance modulation	Distance modulation

Figure 19: Rendering requires two passes each with 4 texture units, taken from [7]

5 3D Stippling Techniques

The pleasing results, as shown in Figure 21, coupled with the high speed of hardware point rendering makes volume stippling an effective approach for illustrative rendering of volume data. An interactive stipple renderer as proposed in [6] consists of two main components: a pre-processor



(a) Volume without lighting
(b) Tone shading contribution
(c) Volume with tone shading
(d) Silhouette contribution
(e) Volume with silhouette
(f) Depth color cue contribution
(g) Volume with depth color cue
(h) Final volume visualization

Figure 20: A complete example of combined techniques using the mouse dataset, taken from [7]

and an interactive point renderer with feature enhancement.

5.1 Pre-processing

Before interactive rendering can begin, an appropriate number of stipple points for each volume are generated, based on volume characteristics, including gradient properties and resolution requirements. Also a number of calculations that do not depend on viewport or enhancement parameters are precalculated. These are volume gradient direction and magnitude, the initial estimation of stipple density from volume resolution, and the generation of an initial point distribution.

5.1.1 Gradient Processing

The magnitude and direction of Gradients are essential in feature enhancement (as described in Section 4.1.1 and 5.2), especially when rendering CT data. Often the widely known central difference gradient method is used, but this can create problems especially with Noisy volume data. Also, the first and second derivate discontinuity in voxel gradients can effect the accuracy of feature enhancement.

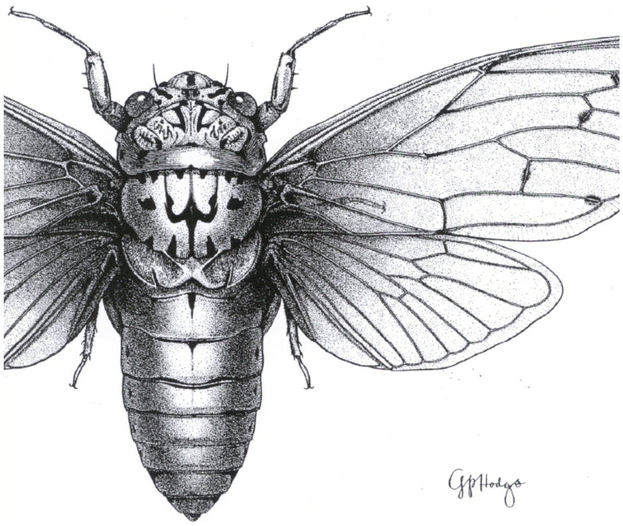


Figure 21: Cicadidae stipple drawing by Gerald P. Hodge [5].

However, an improved method can be found in [11].

5.1.2 Initial Resolution Adjustment

With increasing volume size, the screen projection of each voxel is reduced, so the volume has to be normalised and the maximum number of points which can be assigned to a voxel is calculated. This reduces redundant point generation and increases the performance of the system.

5.1.3 Initial Point Generation

The point distribution often is calculated every frame, this is very time-consuming and leads to problems with frame-to-frame coherency. To avoid this problems a maximum number of stipples are generated and stored in a preprocessing step. Any processing that is subsequently performed simply adjusts either the number of stipples that are drawn, within each voxels or their respective size.

5.2 Feature Enhancements

To simulate traditional stipple illustrations several feature enhancement techniques, as introduced in Section 4.1.1, have to be used. They are based on specific characteristics of a particular voxel, as for example: whether it is part of a boundary or silhouette, its spatial position in relation to both the entire volume and the entire scene, and its level of illumination due to a light source.

5.2.1 Boundaries and Silhouettes

By making the stipple placement denser in voxels of high gradient, boundary features are selectively enhanced. In manual stipple drawings, the highest concentration of stipples is usually in areas oriented orthogonally to the view plane, forming the silhouette edge. By using boundary and silhouette enhancement we can effectively render the outline of the features in the volume.

5.2.2 Resolution

The number of stipples used to represent a given feature varies with the viewed resolution. By using a resolution factor, stipple points can be prevented from being too dense or sparse. This factor influences the number of points in each voxel and produces the effect that features becomes larger and cleaner when the volume moves closer to the observer. Eliminating unnecessary rendering of distant points helps increasing the rendering performance.

5.2.3 Distance

Depth perception is enhanced by using the position of a voxel in the volume to modify both, the point count and the size of the points which are representing it. More distant parts of the volume contain fewer and smaller points.

5.2.4 Interior

Rendering of points is transparent by definition, background objects are showing through foreground object. Exaggerating this effect, by explicit interior enhancement, allows us to observe more detail inside the volume. To get an unhindered view to inside features, the point count of the outer volume elements should be smaller than that of the interior.

5.2.5 Lighting

Achieving good looking lighting effects in stipple rendering is not an easy task. Especially in noisy volumes where gradient information do not describe the shading of some structures correctly. Often structures overlap in the volume and so it can be difficult to identify to which structure a point belongs in a complex scene. Also capturing the inner and outer surface at the same time, while their gradient directions are opposite, is not easy to handle. These problems can significantly reduce the quality of lighting effects.

5.2.6 Silhouette Curves

Searching for potential silhouette curves in the neighbourhood of each voxel could easily lead to a performance bottleneck. To avoid the search in a 3x3x3 subspace around each voxel, an alternative technique using the Laplacian of Gaussian operator proposed in [6] could be used. Silhouette curves can improve the image quality significantly.

6 Pen-and-Ink Rendering

Pen-and-Ink techniques can be classified into three main categories according to the specification of pen-and-ink strokes and the use of information associated with the original 3D objects. The least categorie is 2D drawing, where 2D strokes are generated based only on the 2D information available in image space. Because this is not a real volume visualisation technique our focus will be on the other two categories.

6.1 3D Pen-and-Ink Rendering

In this category 3D strokes are generated in the object space and are projected onto an image plane by a rendering algorithm.

The approach developed by Treavett and Chen [14] is based on NPR textures defined in 3D. As a few examples in Figure 22 shows these textures can differ in density, line width and length, noise level, etc.

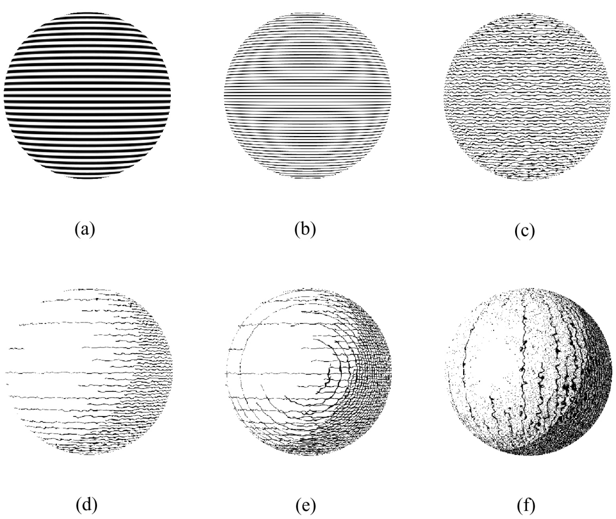


Figure 22: Pen-and-ink techniques of increasing complexity: (a) straight 'pen lines', (b) varying thickness, (c) noise added to lines, (d) introducing lighting effects, (e) a second set of lines, (f) advanced techniques.

The first step in this algorithm is to define a 3D texture. In a simple case it can consists of black horizontal rings defined in cylindrical manner, that simulates the pen strokes during the rendering process. Some hints for the creation of the texture have to be mentioned: Too hard transitions in the texture like in Figure 22 (a) can lead to aliasing effects, this is avoided by adding grey scale to reduce the intensity gradually along the edges. Denser textures could cause unnatural effects, as in Figure 22 (b). Adding noise can be a solution to get results like Figure 22 (c), there it is recommendable to use pre-computed noise lattices instead of a run-time random number generator, to maintain coherence in animations. Adding noise in addition lowers the degree of artificial perfection which is rare in real life hand drawings.

A wide variety of textures can be defined, and they can be combined easily together like in Figure 22 (e) and Figure 22 (f).

In the second step standard iso-surface rendering is accomplished, where the NPR texture is treated in the same way as a PR texture. When a point p on a specific iso-surface is detected, the rendering process first determines the light intensity l at p by using a traditional illumination model. With the texture value at p the final pixel color can be calculated. The result of the rendering process leads to images compareable with Figure 22 (d) and Figure 23. Both volumes of Figure 23 were lit by a point light source in the front.

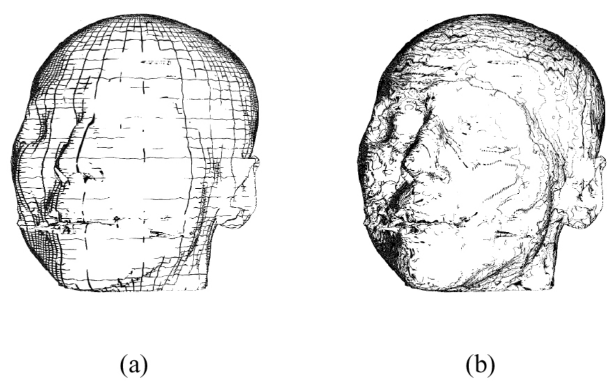


Figure 23: Pen-and-ink style rendering of a CT dataset.

6.2 2^{+}D Pen-and-Ink Rendering

In this category 2D strokes are generated based on 3D or partial 3D information associated with original 3D objects.

In the first phase of rendering, in the object space, all necessary 3D information, relevant to what is to be synthesised in the image plane, is passed in image buffers to the image space. This can be shading information, distance from the image plane, normal vectors for the corresponding point on the visible surface, the curvature of the surface in the horizontal and vertical direction, etc. The choice of information to pass on very much depends on the NPR effects one wishes to achieve, but obviously the more information, the more flexibility and control is available in image space.

In the second phase, in the image space, the renderer applies different pen-and-ink filters to synthesise an NPR image. For example an outline filter which determines the pixel values according to distance information of the related image buffer, as shown in Figure 24 (a). Or a more complicated filter that generates sketch drawings inside the boundary, as shown in Figure 24 (b). A simple filter combines the two images of Figure 24 (a) and Figure 24 (b), into a final image Figure 24 (c).

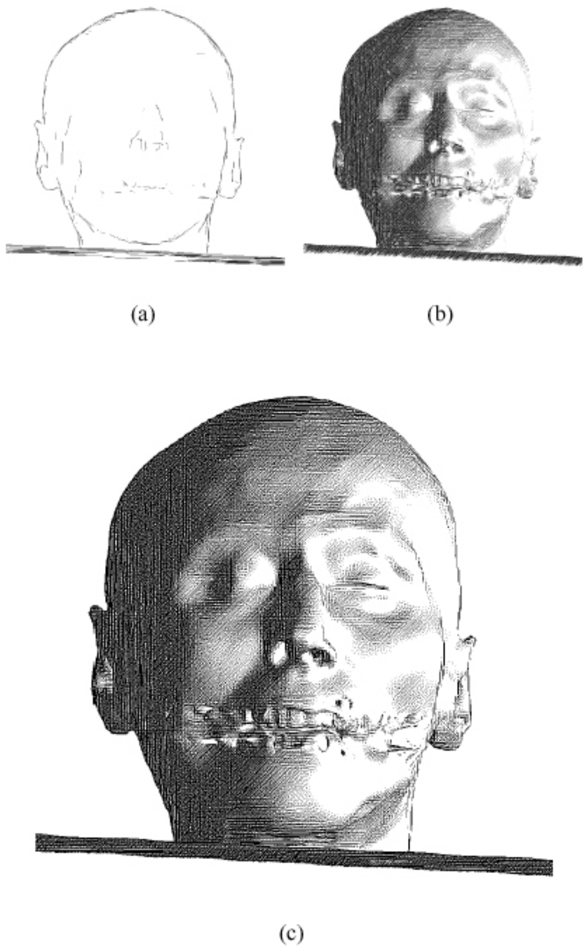


Figure 24: The application of 2^{+}D pen-and-ink rendering to a CT dataset taken from [14]: (a) the outline, (b) the interior shading and (c) the final image.

7 Conclusion

Non-photorealistic volume visualization is an emerging domain. It is far-reaching and it is quite difficult to classify as research is done in such diverse fields. The permanent further development of computer and graphics card will offer new perspectives in future especially as far as interactivity and hardware rendering is concerned. New developments in the near future will be very exciting not only with reference to new implementations but also regarding new ideas for further enhancements or more illustrative views. Up to now the results are quite impressive and expressive and often communicate details and information better than traditional volume rendering.

References

[1] Balázs Csébfalvi, Lukas Mroz, Helwig Hauser, Andreas König, and Eduard Gröller. Fast Visualization of Object Contours by Non-Photorealistic Volume Rendering. In *Proceedings of EuroGraphics 2001 (Manchester, UK, September 2001)*, pages 452–460, Sep 2001.

[2] Stuart Green, David Salesin, Simon Schofield, Aaron Hertzmann, Peter Litwinowicz, Amy Gooch, Cassidy Curtis, and Bruce Gooch. *Non-Photorealistic Rendering*. SIGGRAPH 99 Course Notes, 1999.

[3] Helwig Hauser. Visualisierung. Slides for the Lecture Visualisierung, Technical University of Vienna.

[4] Christopher G. Healey, Victoria Interrante, davidkremers, David H. Laidlaw, and Penny Rheingans. Nonphotorealistic rendering in scientific visualization. In *Course Notes of SIGGRAPH 2001*, volume Course 32, pages 12–20, August 2001.

[5] E. Hodges(editor). *The Guild Handbook of Scientific Illustration*. John Wiley and Sons, 1989.

[6] Aidong Lu, Christopher J. Morris, David S. Ebert, Penny Rheingans, and Charles Hansen. Nonphotorealistic volume rendering using stippling techniques. In *Proceedings of the conference on Visualization '02*, pages 211–218. IEEE Press, August 2002.

[7] Eric B. Lum and Kwan-Liu Ma. Hardware-accelerated parallel non-photorealistic volume rendering. In *Proceedings of the International Symposium on Non-Photorealistic Rendering and Animation 2002*, June 2002.

[8] Kwan-Liu Ma, Aaron Hertzmann, Victoria Interrante, and Eric B. Lum. Recent advances in nonphotorealistic rendering for art and visualization. In *SIGGRAPH 2002 Course Notes. Course 23.*, August 2002.

[9] Joanne McKinley. Non-photorealistic rendering techniques applied to volumes. CS 788H S01 Project, University of Waterloo, Computer Graphics Lab, http://www.cgl.uwaterloo.ca/˜jlmckinl/cs788/project/, 2002.

[10] M.Sharps, K.Harrington, and E.C.Wang. Volume visualization illustration - nonphotorealistic rendering. University of Maryland, Baltimore County; Graphics, Animation, Visualization Lab; http://www.cs.umbc.edu/˜msharp1/npr.html, December 1999.

[11] L. Neumann, B. Csébfalvi, A. König, and E. Gröller. Gradient estimation in volume data using 4D linear regression. In M. Gross and F. R. A. Hopgood, editors, *Computer Graphics Forum (Eurographics 2000)*, volume 19(3), pages 351–358, 2000.

[12] Frank Pfenning. Non-photorealistic rendering. Slides for the Lecture Computergraphic 1, Carnegie Mellon University.

[13] Penny Rheingans and David Ebert. Volume illustration: Nonphotorealistic rendering of volume models. *IEEE Transactions on Visualization and Computer Graphics*, 7(3):253–264, October 2000.

[14] S. M. F. Treavett and M. Chen. Pen-and-ink rendering in volume visualisation. In T. Ertl, B. Hamann, and A. Varshney, editors, *Proceedings Visualization 2000*, pages 203–210. IEEE Computer Society Technical Committee on Computer Graphics, 2000.

[15] S. M. F. Treavett, M. Chen, R. Satherley, and M. W. Jones. Volumes of expression - artistic modelling and rendering of volume datasets. In *Computer Graphics International 2001 (CGI'01)*, pages 99–106. IEEE Computer Society, July 2001.

[16] Anna Vilanova. Non-photorealistic rendering. Reader for the Lecture Computergraphic 2, TU Vienna.