JOURNAL OF APPLIED
COMPUTER SCIENCE
Vol. 21 No. 1 (2013), pp. 107-117

Interactive Visualization of Magnetic Fields

Piotr Napieralski[1], Krzysztof Guzek[1]

[1]*Institute of Information Technology,*
Lodz University of Technology
Address: ul. Wólczańska 215,
90-924 Lodz, Poland
piotr.napieralski@p.lodz.pl

Abstract. *In this paper, we present an interactive method for visualizing three-dimensional nonlinear, unsteady vector fields. Visualization of three-dimensional electromagnetic fields is a difficult issue because the user has to be able to assess the distribution of the vector in the context of complex geometry. The authors propose a novel scheme to visualize the flow of vectors.*
Keywords: *vector fields, 3D visualisation, interactive computer visualisation.*

1. Introduction

Commonly used methods for visualizing vector fields [1, 2] are divided into two groups depending on whether the visualization shows the local or the global characteristics. The standard approach distinguishes between three main groups of methods for visualization:

- Methods for direct visualization;
- Methods based on texture;
- Methods based on geometry;

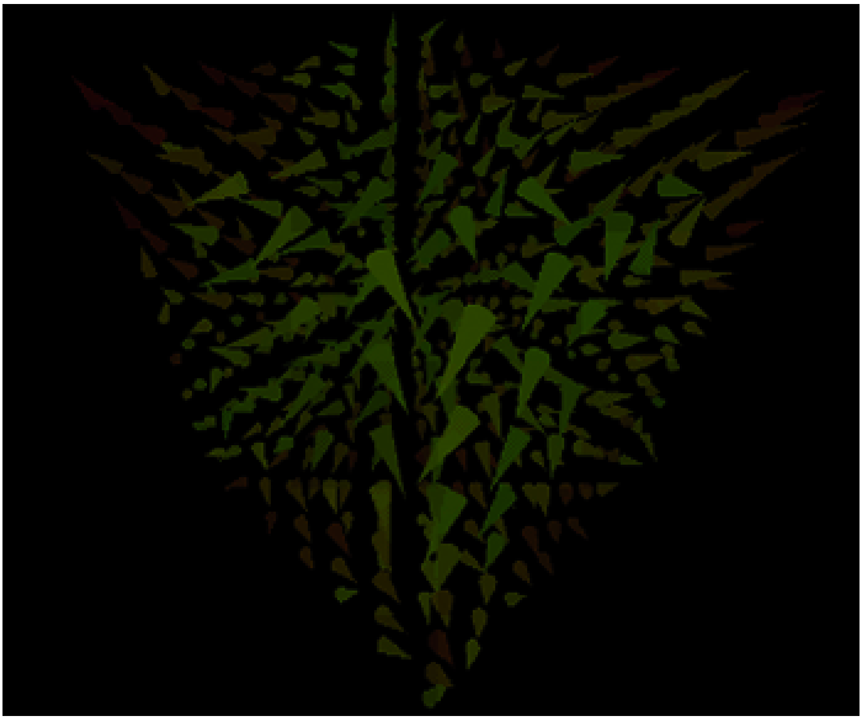

Figure 1. Direct visualization of a vector field using the author's program

Direct visualization methods indicate the local characteristics of a vector field, while methods based on textures and geometry mainly visualize global characteristics. Each of these groups has the features that predispose them for different applications. An easy way to visualize the vector field is the use of symbols or basic graphic shapes (primitives). Graphic primitives are placed in space, showing the status of a vector field V at a particular point. This approach requires a knowledge of the state of the field V at point P, hence primitives are usually drawn only in the fixed sampling points of the vector fields. Samples are typically placed on a uniform grid. This method is mostly used for external surfaces, or for sections. Visualization of this type often leads to ambiguous results, especially for complex geometries or large amounts of data (Fig. 1).

A popular method for visualization based on texture is the Direct Image Synthesis. It consists in the direct modification of the output image, taking into account

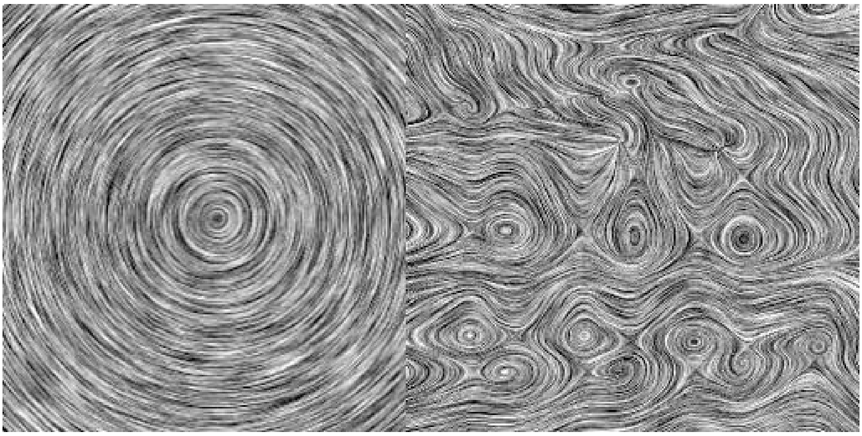

Figure 2. Circular and turbulent fluid dynamics vector fields imaged using Line Integral Convolution[3]

the flow of a vector field. This method is based on the modification of the original image by convolution operator with an appropriately modified kernel (the core of the transformation). Convolution operator assigns to each cell of an output image a value equal to the weighted sum of a set of cells of the input image. Selection of cells of the input image and their weights depends on the kernel transformation. Convolution makes certain features of the input image becomes more visible than others. The method modifies the base image in such a way that the pixels along the current lines are blurred together (color is averaged) /citeCabral. In this way, they become identifiable as a part of the same current line. At the same time, this process reduces the similarity of pixels belonging to different lines [1]. Extension of this method, for use in 3D fields, in the simplest variant requires a change in method of creating a grid of field values, in order to select surface on which we want to visualize the flow.

These surfaces are then mapped on the rectangular texture of appropriate dimensions. Then, similarly to the 2D version, the input image is modified based on the value of the visualized surface. As a result the color texture is obtained. Finally, the selected area is displayed in 3D space using the resulting color texture. The level of detail with expertly selected parameters and the usefulness to the visualization of global characteristics of a field are undoubted advantages of this method. However, compared to other methods used to visualize three-dimensional fields, this one has a fairly significant drawback. Namely, one can only visualize only the selected surfaces in space, and there is no information about the field outside the surface. This defect limits the spatial application of this method. Motion trajectories of particles moving in a filed overlap with the current lines. Visualization of the field using current lines is fairly intuitive. In an area of the specific field, a set number of particles is distributed. Then the movement of particles is simulated according to the flow of the field. Motion path of each particle gives the current lines of the field. Current lines are suitable to show the flow of the field and critical points (sources, vents). Different criteria for the cessation of growth of a particular lines of the field are used. Also, whether the ready line is suitable to show, or should be ignored, is determined by the general and specific criteria.There are also a variety of algorithms for line growth. Current lines do not show certain characteristics of the field, which may be considered to be important in certain applications, i.a. the rotation of movement and the measured values of the convergence/divergence. These characteristics can be represented using techniques that in some way use the stream lines. One of of these techniques is the method of current ribbon. This method instead of initiating a single current line at the starting point, creates two lines in the slightly separated points. Subsequently, the movement of particles and the growth of lines is performed along the field in the same way as in the basic stream line method. Ribbons are displayed using two created current lines, connected with polygons. The manner and degree in which the lines rotate relatively to each other visualize the flow. The distance between the two lines gives information about the convergence or divergence of the field. If the ribbon along its whole length is the same width, then the stream lines are neither converging nor diverging. This technique allows you to show two characteristics invisible when using a single line of the stream. It should be noted that this feature also give insight into the local characteristics of the field. At the current surface any number of current lines are initiated along any open or closed curve. Current surfaces are therefore a natural extension of the current ribbon techniques. Visualization of the current surface is similar to the visualization of ribbons - component lines are connected with polygons. Advantages of this method, compared to the technique of single stream lines are the same as in the case of ribbon stream - the width of the surface visualizes the extent to which the flow is divergent, and its twist shows rotation of the field.

A special case of the current surface are the current tube. They arise when the component current lines are initialized along a closed curve.

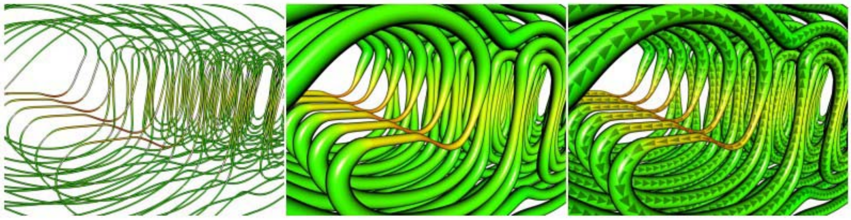

Figure 3. Stream lines, current tubes and current tubes with applied texture informing about direction of the flow [4]

2. Visualization of the field in contemporary software

All presented methods for visualizing vector fields in 3D space are commonly used in contemporary computer software. These methods seem to be insufficient for the visualization of certain phenomena and special cases Both methods of cones, and tubes do not allow users to trace the flow of the vector field [5, 6]. There are some aspects of the visualization method that are particularly important to provide complete information about the field. It is important to effectively use the display space - each pixel of the output image should carry maximum information, furthermore the method should provide information in an intuitive way that allows easy analysis of the investigated phenomenon. The most common need is to render a number of sections [7, 8] to find out where the stream turns. Figure 1 shows an attempt to visualize the vector stream enforced by the clamp perpendicular to the tested conductor. An example problem is to determine how close from the clamp the flow is perpendicular to the section of the conductor. Answer to this type of question often occurs during the design and analysis of phenomena occurring in the devices. Simulations showing the results of this situation were carried out using two most popular programs for the calculation of electromagnetic fields: Opera (Fig. 4) and ANSYS (Fig. 5).

Both application uses a simple method based on textures [10]. Field values are presented on the surface geometry, preventing the analysis of phenomena in 3D space. Described above visualizations do not give a clear answer to the question at which point and how the current flow begins to turn. Our solution allows the assessment of the distribution of the vector in the context of complex geometry. We assume that having knowledge about the values of the vector field in space, you can use abstract particles, whose movement is the list of position as a function

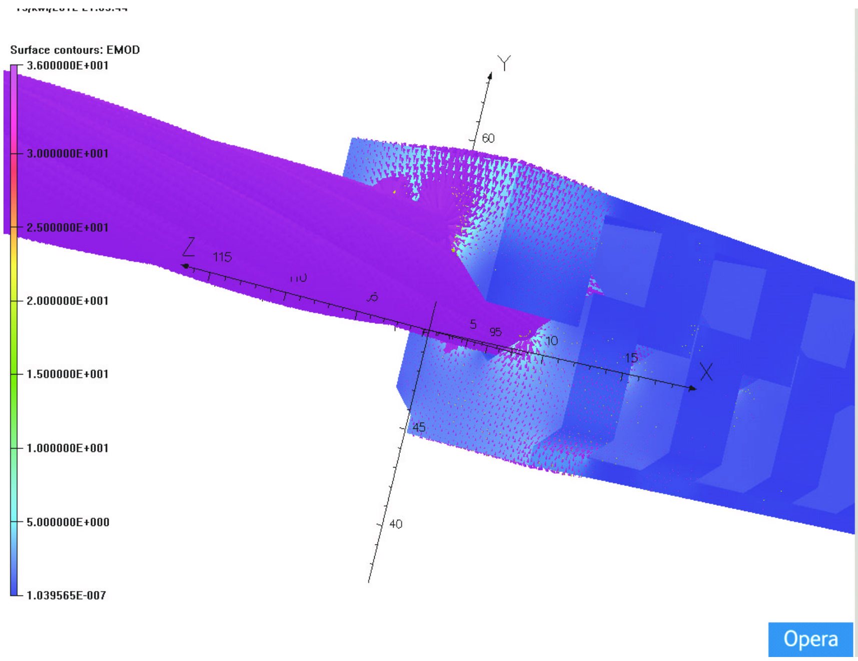

Figure 4. Visualization of the current flow with Opera

of time. The resulting trajectories of the particles in the field, assuming stationarity of the field, are equivalent to the current line. Thus, the movement of the particles can be used to visualize the fields in the context of complex geometry, giving possibility to present the spatial distribution of the field and facilitating analysis of the studied phenomena.

3. Description of the method

We propose a novel scheme to interactively visualize magnetic fields. Our method depends on the simulation of the movement of particles in a magnetic field, based on the properties of the field and the use of particle systems common in computer graphics. The results proved useful in the analysis of multiple cases of visualizing time-varying vector data fields. Particle systems are a technique that allows to create interesting visual effects without much effort [9]. Thanks to this technique many effects can be realized [9,10]. Particle systems have been studied

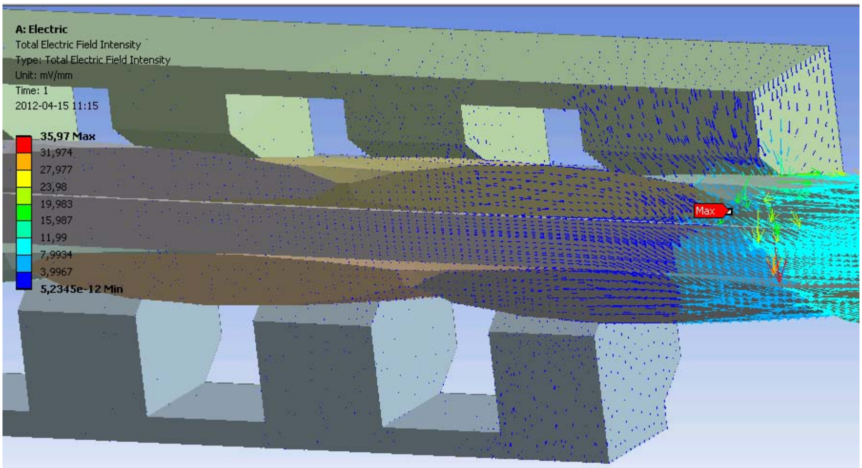

Figure 5. Visualization of the current flow with Ansys Software

in computer graphics for various applications. These techniques provide more insight on field structure with greater usability. A natural approach seems to be to use a model based on geometry. We create a 3D point in R3 space based on the computed data stored in vectors (the vector contains information about the position in space and the value at that point). Input data can come from different programs, our solution is focused only on the visualization and the later exploration of data. Specific importance was placed on the visualization of the field in a way that allows for easy orientation in the geometry of the test device. Moving particles (animation) give an indication of the direction and magnitude of the velocity in the space. In the proposed solution illumination and shading effects were added for the geometry and particles, providing the viewer with a better 3D impression (Fig 6.)

Particle motion can be effectively visualized using animation. Consecutive particle positions are calculated on the basis of constant time steps, and are displayed in a new position (Figure 4). The movement of each particle is a list of positions in the function of time. Therefore for each particle we acquire its trajectory. For this task we create an abstract interface for the class responsible for calculating the movement of the object in the field depending on the position and time step. In our solution the user has the possibility to explore the distribution of particle fields in real time.

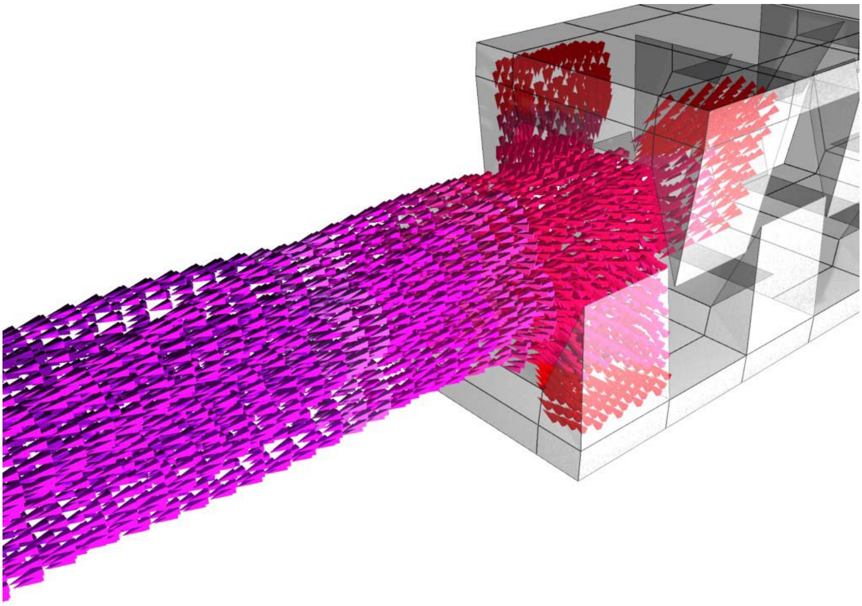

Figure 6. Visualization of magnetic fields with dynamic particle systems with illuminating and shading effects

Therefore, as the field size (i.e. length of the vector field) at the point of the space in direct visualization methods is provided using color it was appropriate to translate one value (the size of the field) to another (color). Knowing the normalized value of the field and having the appropriate color palette, we can easily calculate the color for that value. An additive color is expressed as an RGB triplet (Red, Green, Blue), each component of which can vary from zero to a defined maximum value. This triplet (RGB) can represent the three-dimensional coordinate of the vector. The corresponding algorithms for processing the input data and displaying the results on the screen is customizable. These configurations correspond to parameters that can be changed at run time without restriction for recompilation. The user has more influence on the interpretation of the present field. The program and library, implemented and presented in this paper, contain the basic tools to visualize the input data. Modular design of the project can be readily extended to the new methods. The developed class library may be adopted in any existing solution.

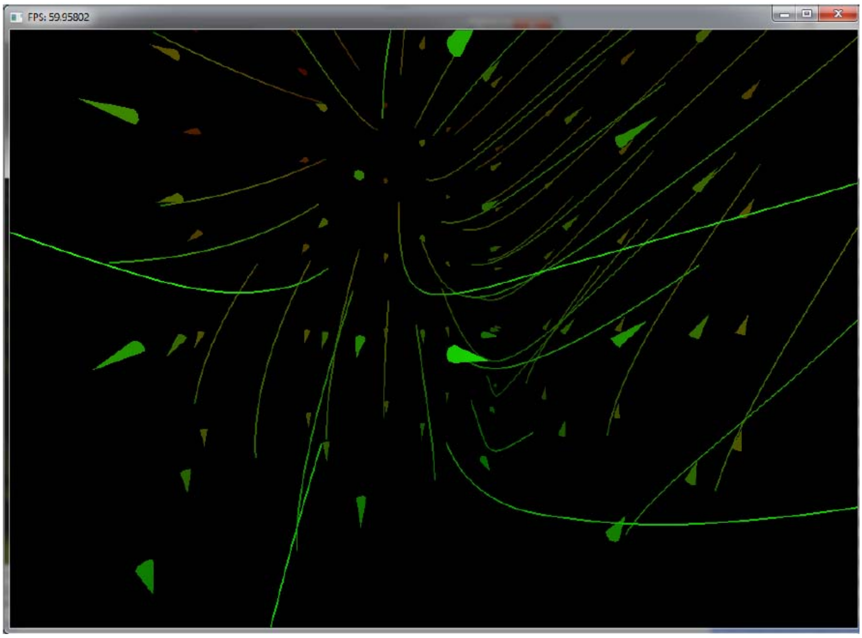

Figure 7. Animation of particle motion

The project was realized in the form of interactive 3D applications. The program was written in C in version 4.0. It is compiled into intermediate code statically and strongly typed high-level language. Applications written with this language are executed in the runtime system. Our solution was created by combining their own source code with the .NET Framework, OpenTK (for building a graphical user interface) and OpenGL (for visualization effects, this library offers several stages that are fully programmable using GLSL [6].)

4. Conclusions

Comparing our solution to other existing methods (with regard to both computational expenses and rendering performance on multiple data sets), we demonstrate that our solution is better suited than existing methods to the task of interactively visualizing the distribution of the vector in the context of complex geometry. The proposed rendering approach permits the rendering of large, time-dependent simulation datasets at interactive rates. The presented results show that the animation can be generated using the previously calculated values. The visualization method allows the exploration of virtual space in real time, with the possibility of examining the calculated values. Currently data set are provided to the program from external files. The program uses it to quicken visualization with the possibility of space exploration in real time. Parameters such as particle size and transparency will depend on the position of the observer. Changing these parameters significantly affects the convenience of observation of the phenomena occurring in the device.

4.1. Acknowledgements

This research was supported under TEWI project.

References

[1] Stoll, C., Gumhold, S., and Seidel, H. P., *Visualization with stylized line primitives*, In: Visualization, 2005. VIS 05. IEEE, Oct., pp. 695–702.

[2] Sparavigna, A. and Montrucchio, B., *Vector field visualization with streamlines*, CoRR, Vol. abs/cs/0610088, 2006.

[3] Chen, Y., *Techniques for 3D scalar and vector field visualization with error evaluation*, Ph.D. thesis, HE JOHNS HOPKINS UNIVERSITY, 2009, (in English).

[4] Zhang, Y., Duckworth, R., Ha, T., List, F., Gouge, M., Chen, Y., Xiong, X., Selvamanickam, V., and Polyanskii, A., *AC Loss Reduction in Filamentized YBCO Coated Conductors With Virtual Transverse Cross-Cuts*, Applied Superconductivity, IEEE Transactions on, Vol. 21, No. 3, June, pp. 3301–3306.

[5] Szajerman, D., *GPU-BASED RENDERING AND ANIMATION OF WATER SPREADING ON FLAT SURFACES*, Metody Informatyki Stosowanej Kwartalnik Komisji Informatyki Polskiej Akademii Nauk Oddział w Gdańsku, Vol. 29, No. 4, 2011, pp. 231–240.

[6] Szajerman, D. and Pietruszka, M., *Real-time ice visualisation on the GPU*, Journal of Applied Computer Science, Vol. 16, No. 2, 2008, pp. 89–106.

[7] Sundquist, A., *Dynamic Line Integral Convolution for Visualizing Streamline Evolution*, IEEE Transactions on Visualization and Computer Graphics, Vol. 9, No. 3, 2003, pp. 273–282.

[8] Lum, E. B., Ma, K. L., and Clyne, J., *Texture hardware assisted rendering of time-varying volume data*, In: Proceedings of the conference on Visualization '01, VIS '01, IEEE Computer Society, Washington, DC, USA, 2001, pp. 263–270.

[9] Polrola, M. and Wojciechowski, A., *Real-time hand pose estimation using classifiers*, Vol. 7594 LNCS of *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2012.