# $\mathcal{A}$

**Jean-Paul Balabanian[1] and Eduard Gröller[2,1]**

1 **University of Bergen**
**Bergen, Norway**
`jean-paul.balabanian@ii.uib.no`

2 **Vienna University of Technology**
**Vienna, Austria**
`groeller@cg.tuwien.ac.at`

**Abstract**

This paper describes the concept of $\mathcal{A}$-space. $\mathcal{A}$-space is the space where visualization algorithms reside. Every visualization algorithm is a unique point in $\mathcal{A}$-space. Integrated visualizations can be interpreted as an interpolation between known algorithms. The void between algorithms can be considered as a visualization opportunity where a new point in $\mathcal{A}$-space can be reconstructed and new integrated visualizations can be created.

**1998 ACM Subject Classification** I.3.7 Three-Dimensional Graphics and Realism

**Keywords and phrases** illustrative visualization, integrated visualization

**Digital Object Identifier** 10.4230/DFU.Vol2.SciViz.2011.36

## 1 Introduction

Illustrative visualization has been quite successful in recent years. The idea of illustrative visualization is to mimic the traditional illustrators' styles and procedures. Many techniques have been developed that span a wide range of traditional styles. These techniques include lighting models that resemble illustrative styles, exploded views, labeling, ghosting, and halos and have been successful at simulating the original illustrators' results. One strategy of illustrators is in principle to blend together very different styles. For example in one part of an illustration a photo realistic representation of the object is shown while in another part of the drawing the object is shown using ghosting effects, halos or outlines. Figure 1 demonstrates this heterogenous blending of different styles with several examples of a car.

This approach is similar to what illustrative visualization is doing and the idea of *blending different styles* is a concept that can be transferred, in a metaphorical way, to *blending of different algorithms*. Merging the results from one visualization with the results from another visualization, in a non-trivial way, can be considered as blending between the two algorithms. A simple example of this concept can be derived from slicing and volume rendering. These are two different visualization techniques for volume data. Blending between the two techniques could result in a visualization where the slice is integrated into the volume rendering. Figure 2 shows an example of what an integrated visualization combining direct volume rendering and slicing may look like.


Scientific Visualization: Interactions, Features, Metaphors. *Dagstuhl Follow-Ups, Vol. 2.*
Editor: Hans Hagen; pp. 36–47
Dagstuhl Publishing
Schloss Dagstuhl – Leibniz Zentrum für, Germany

**Figure 1** An illustration of a car with different parts ghosted. Image courtesy of Alan Daniels. Copyright beaudaniels.com

Integrated visualizations solve a limitation typical for linked views. In a linked-views setup the number of views increases with the complexity of the data. As the complexity increases the number of interesting aspects of the data also increases and more views are necessary to convey all of the important parts. Integrated visualizations alleviate this problem by providing a single frame of reference for all visualizations. They incorporate all of the important aspects of the data into the same view. Creating an integrated visualization is not straightforward, though, and so far only rather ad hoc approaches are known. A more systematic approach to create integrated views might be $\mathcal{A}$*-space*. $\mathcal{A}$-space is a space where all visualization algorithms reside. In $\mathcal{A}$-space every algorithm is represented by a unique point. $\mathcal{A}$-space is sparsely covered by the known visualization algorithms and there are many voids. Filling the voids between the points leads to reconstruction in $\mathcal{A}$-space and new integrated visualizations.

In the next section we will give examples of different visualizations that are reconstructions in $\mathcal{A}$-space and we will describe what type of integration each example is. In Section 3 we will describe $\mathcal{A}$-space in more detail and we will conclude in Section 4.

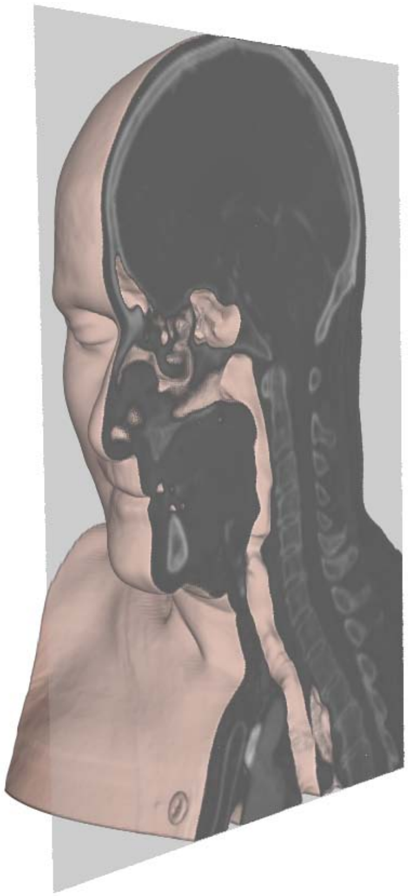
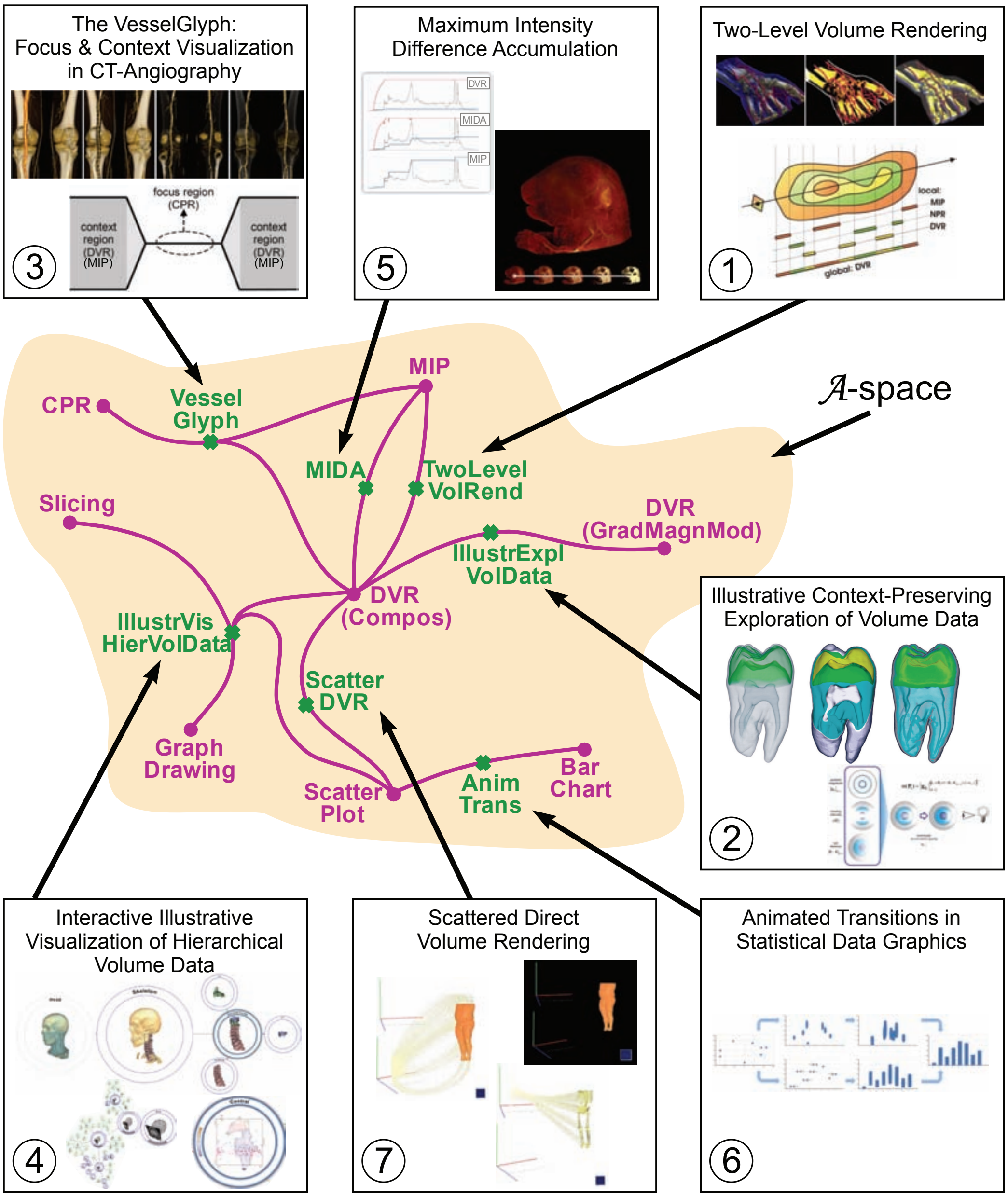


**Figure 2** A volume-rendering with an integrated sagittal slice.

## 2 Examples of Reconstruction in $\mathcal{A}$-Space

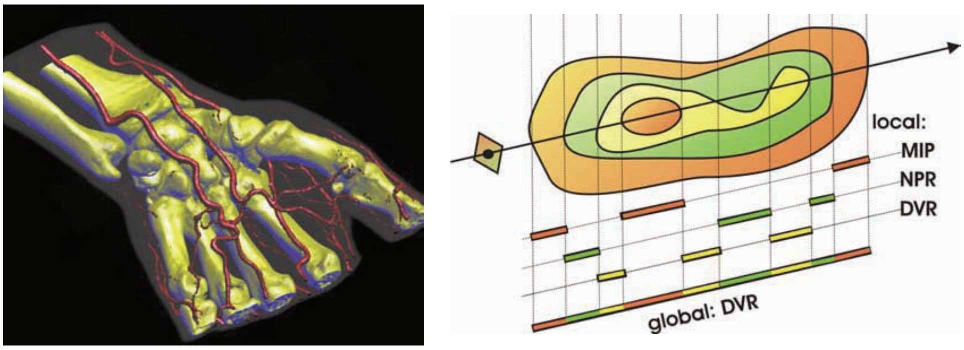
In this section we will showcase several examples of visualizations that are a *blending of algorithms*. The integrated visualizations presented can be considered as reconstructions in $\mathcal{A}$-space with known starting points. We will show where in $\mathcal{A}$-space these visualizations reside and the component algorithms required to perform the reconstructions. Figure 3 is a schematic of $\mathcal{A}$-space with several algorithms indicated. The pink points represent well known algorithms that in principle are not integrated visualizations. Between these points paths have been drawn with green crosses indicating the reconstructed algorithms. An interesting observation is that between MIP and DVR there are two different paths. A path represents one way of blending algorithms. In $\mathcal{A}$-space several paths may exist between algorithms and may result in fundamentally different visualizations. In the following sections we will describe all of the example visualizations that are present in Figure 3. We will discuss them in the order indicated by the number in the lower left of each frame. The chosen examples are just a subjective selection to illustrate a few nice places in $\mathcal{A}$-space. A comprehensive overview on previous integrated views is beyond the scope of this paper.




Legend

CPR: Curved Planar Reformation
DVR: Direct Volume Rendering
MIP: Maximum Intensity Projection

**Figure 3** 𝒜-space with example population.




(a) (b)

**Figure 4** (a) A result from the Two-Level Volume Rendering [5, 4] visualization technique. One specific technique is used for the bone, another one for the skin and a third one for the vessels. (b) A schematic of the algorithm selection during rendering (NPR: Non-Photorealistic Rendering).

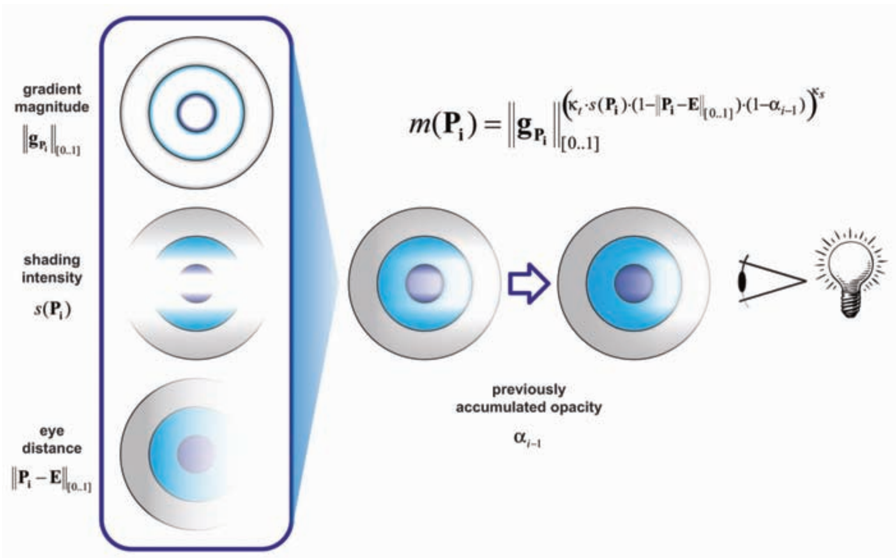
## 2.1 Two-Level Volume Rendering

The Two-Level Volume Rendering approach proposed by Hauser and Hadwiger [5, 4] is a merger of several visualization techniques. The idea is to use different rendering techniques depending on the underlying data. The techniques available for the rendering are Maximum Intensity Projection (MIP), Direct Volume Rendering (DVR) and others. During volume rendering the appropriate visualization techniques for the underlying segmented regions are chosen. The integration is a spatially coarse one since there is no smooth transition between the techniques and the resulting pixel is a composite of the visual representations produced by the different techniques. Figure 4a shows an example of this visualization approach where one technique is used for the bone, another technique for the vessels, and a third technique for the skin. Figure 4b indicates that for different spatial regions different algorithms are employed.

## 2.2 Illustrative Context-Preserving Exploration of Volume Data

The Illustrative Context-Preserving Exploration of Volume Data technique proposed by Bruckner et al. [2] is a visualization technique that enhances interior structures during volume rendering while still preserving the context. During volume rendering one or several structures may occlude the one of interest. Many techniques exist that can help in reducing the occlusion. Reducing the opacity of the occluding structures or applying clipping are two such techniques. The problem with these techniques is that they might remove the context of the interesting feature. The proposed approach combines DVR based on compositing with Gradient Magnitude Modulated DVR to reduce the opacity of less interesting areas in a selective manner. Two parameters are used to decide how to continuously *interpolate* between the two algorithms based on the input data. The results are illustrative volume renderings where contextual structures are outlined and the focused structures are kept in a prominent way. Figure 5a shows an example image produced by this technique. The center of the hand is semi transparent showing, among other details, the blood vessels quite prominently. The edges of the hand are not ghosted and thus retain the context. As indicated in Figure 5b the technique is a smooth and seamless integration of gradient-based opacity modulation, selective occlusion removal, fuzzy clipping planes and multiple transparent layers' handling.

(a)



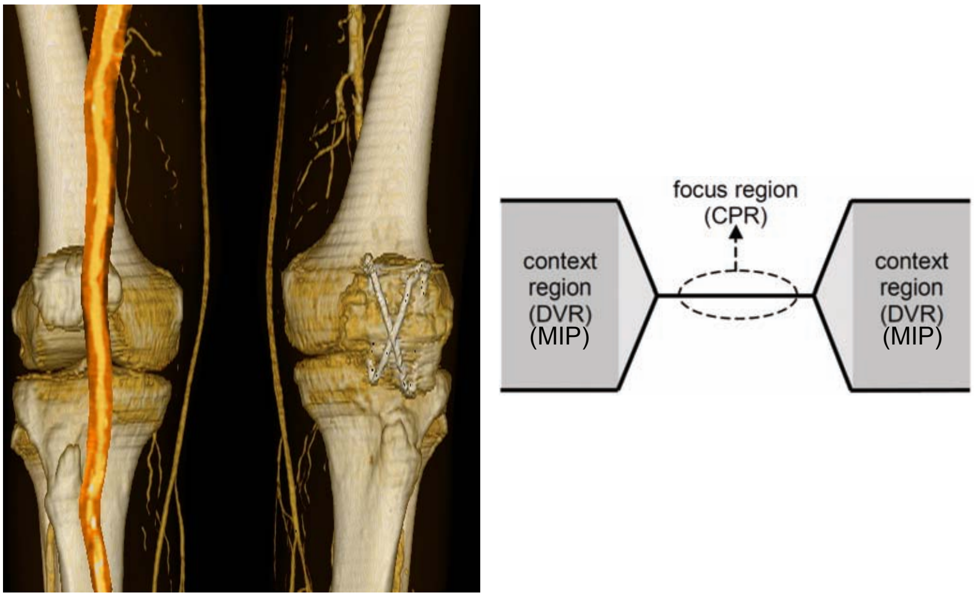

(b)

**Figure 5** (a) The context-preserving volume rendering [2] of a CT-scanned hand produces similarities to the ghosting effect used by illustrators. (b) Illustrates how different rendering techniques are seamlessly integrated while preserving the context [2].

## 2.3 The VesselGlyph: Focus & Context Visualization in CT-Angiography

The two previous examples have shown integration between techniques that all operate in 3D. The following example is an integration between a 2D technique and a 3D technique. The result is a visualization that exploits the complementary strengths and avoids the complementary weaknesses of both.

The VesselGlyph, proposed by Straka et al. [8], is a technique that combines Curved Planar Reformation (CPR) with DVR. CPR is a technique that takes a feature like a blood vessel and cuts it with a curved surface revealing the inside structures. The shape of the




**(a)** **(b)**

**Figure 6** (a) The VesseGlyph [8] in action. The CPR, the vertical orange and red band on the left side, is projected onto the DVR of the same structure. (b) The concept of the VesselGlyph where CPR, the focus region, is integrated smoothly into the DVR, the context region.

surface is adapted so that it follows the curving and twisting of the structure. The resulting visualization is a 2D slice of the inside of the vessel. The VesselGlyph technique incorporates the CPR slice into the DVR visualization of the context structures. The resulting visualization shows interior details of blood vessels with CPR presented in the correct context rendered with DVR. The type of integration employed in this visualization is the merging of two spatially registered visualizations using for example image compositing techniques. Figure 6a shows an example of this type of visualization. With DVR alone the interior calcifications of blood vessels would not show up appropriately. With CPR alone the context region would be sliced arbitrarily which greatly reduces overview. Figure 6b depicts the concept of the VesselGlyph within an axial slice where the blood vessel would show up as a circular region in the slice center. CPR is considered for the focus region and is smoothly integrated into the DVR which is considered for the context region. It is also indicated that the context could alternatively be visualized using MIP.

## 2.4 Interactive Illustrative Visualization of Hierarchical Volume Data

The following example is more complex than the previous ones. The visualization performs integration between 3D and 2D techniques and also between scientific visualization and information-visualization techniques. The result is a visualization that in 𝒜-space blends more than two different algorithms.

Hierarchical Visualization of Volume Data by Balabanian et al. [1] is an integrated visualization that uses graph drawing to visualize the hierarchical nature of structures in a volumetric dataset. Graph drawing in 2D is used as a guiding space where other 2D or 3D visualizations are embedded. The nodes in the graph drawing are enlarged and serve as a canvas for the other visualizations. These visualizations include DVR, slicing, and scatter

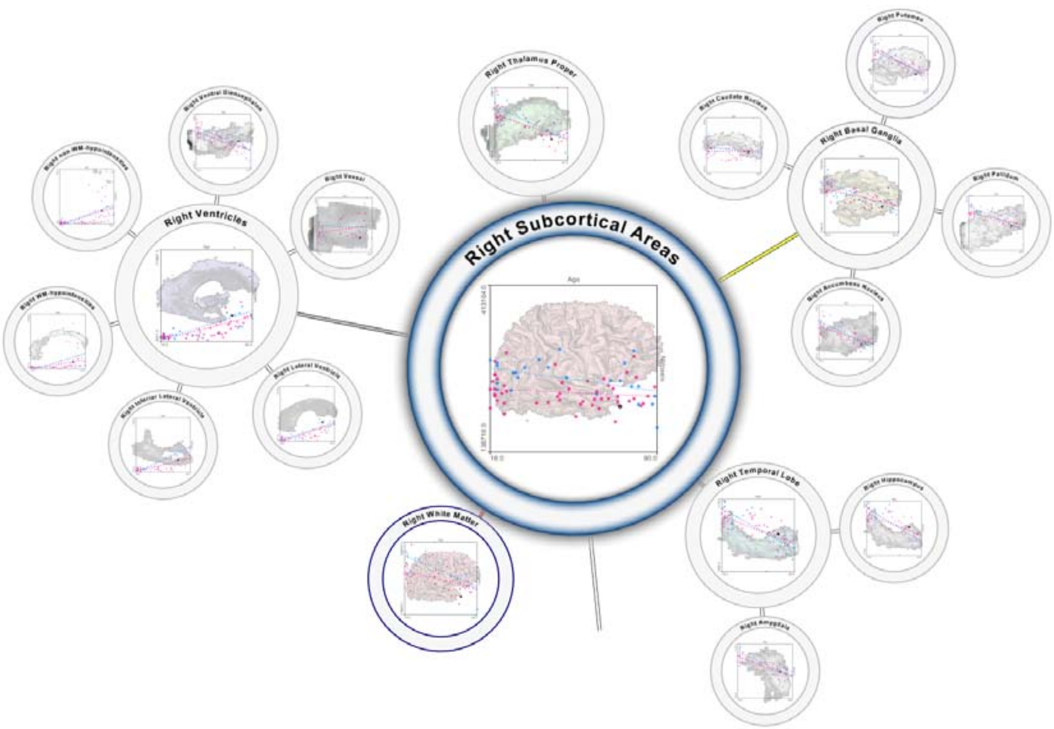



**Figure 7** Hierarchical rendering of volume data [1]. A node-link diagram represents the hierarchical structuring with embedded volume renderings and scatter plots.

plots and are all integrated into one space. The type of integration employed in this visualization is at different levels. DVR and slicing are integrated in the same way as shown in Figure 2. The object that is actually rendered is defined by the hierarchical structure visualized by the graph drawing. The graph drawing is specified by the hierarchy information and every node is rendered as a circle. With statistical data available for the structures a scatter plot is added. Figure 7 shows an example of the subcortical areas of the brain (slicing not included here). The hierarchy of the substructures is visible with semi-transparent scatter plots on top of the embedded volume renderings.

In this example the integration is steered by the graph drawing. The abstract data is used to create a structure to present both the abstract and spatial data. It is also possible to envision an approach that uses the scientific-visualization space as the embedding space. In Figure 8 we have sketched the interpolation between the two spaces that are part of the visualization, i.e., the abstract and the spatial space. The red circle indicates where this work is located but using scientific visualization as the embedding space will result in a visualization located in the dashed square. Such an integrated view might be an exploded view in 3D space where the abstract hierarchical relationships are indicated through arrows.

## 2.5 Maximum Intensity Difference Accumulation

We now present another example of DVR-MIP integration in $\mathcal{A}$-space. This demonstrates that there are more than one possibilities to perform *interpolation* between points in $\mathcal{A}$-space. Maximum Intensity Difference Accumulation (MIDA) is a technique proposed by Bruckner and Gröller [3]. It is a volume rendering technique that integrates MIP and DVR. The integrated visualization preserves the complementary strengths of both techniques, i.e.,

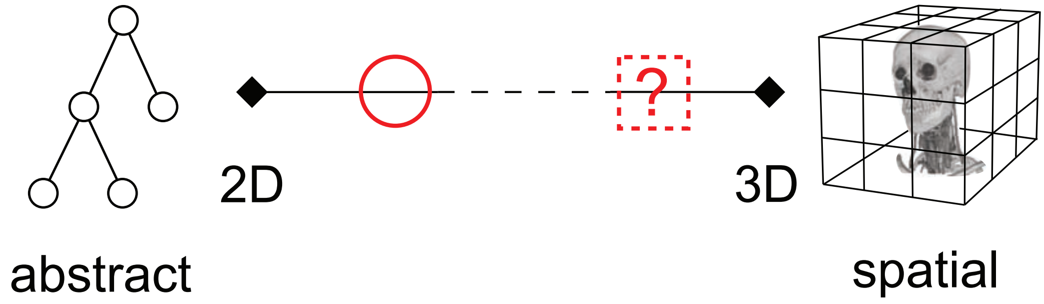



**Figure 8** The red circle indicates the location where the interpolation between spaces takes place in the work by Balabanian et al. [1] while the dashed square indicates an alternative approach to this work.

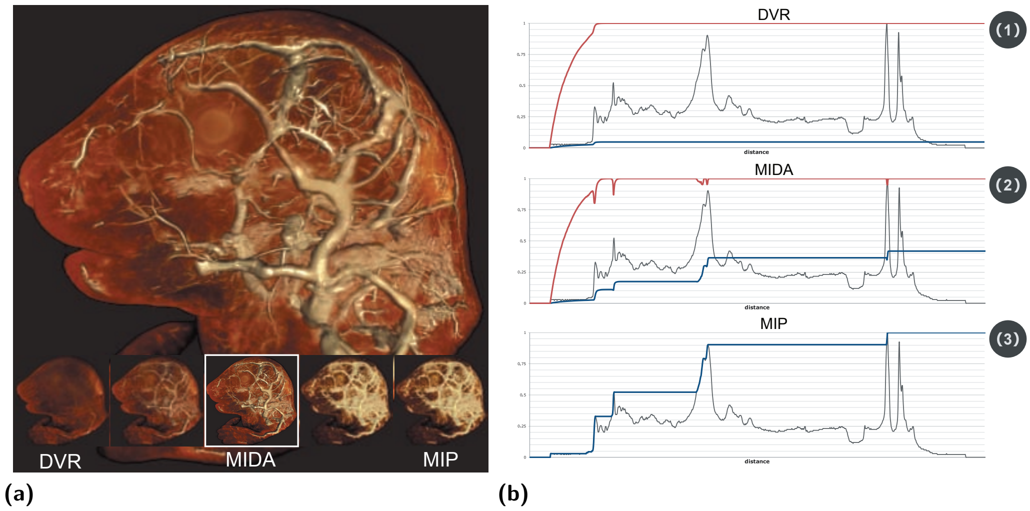



**Figure 9** (a) Ultramicroscopy of a mouse embryo showing the MIDA [3] rendering enhanced with possible *interpolations* from DVR to MIP in the bottom. (b) Shows typical ray profiles for (1) DVR, (2) MIDA and (3) MIP [3].

efficient depth cuing from DVR and parameter less rendering from MIP. Since some datasets look better with DVR and others are best viewed with MIP, MIDA lets the user interpolate smoothly between DVR and MIP. Compared to the two-level volume rendering technique described in Section 2.1, MIDA is a spatially fine-grained integration approach and provides smooth transitions between the techniques. At each spatial position elements of both techniques are incorporated, whereas in two-level volume rendering algorithms are applied spatially disjoint. Figure 9a shows the result of using MIDA on an ultramicroscopy of a mouse embryo. Figure 9b shows the typical ray profiles generated with the different techniques.

## 2.6 Animated Transitions in Statistical Data Graphics

The Animated Transitions in Statistical Data Graphics proposed by Heer and Robertson [6] is a 2D to 2D integration performed in the information-visualization domain. The visualization techniques created provide smooth transitions between different visualizations of

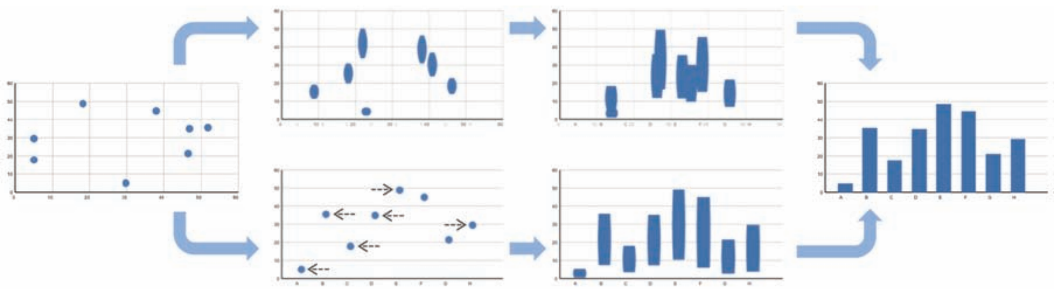


**Figure 10** A schematic overview showing two possible transitions from a scatter plot to a bar chart [6].

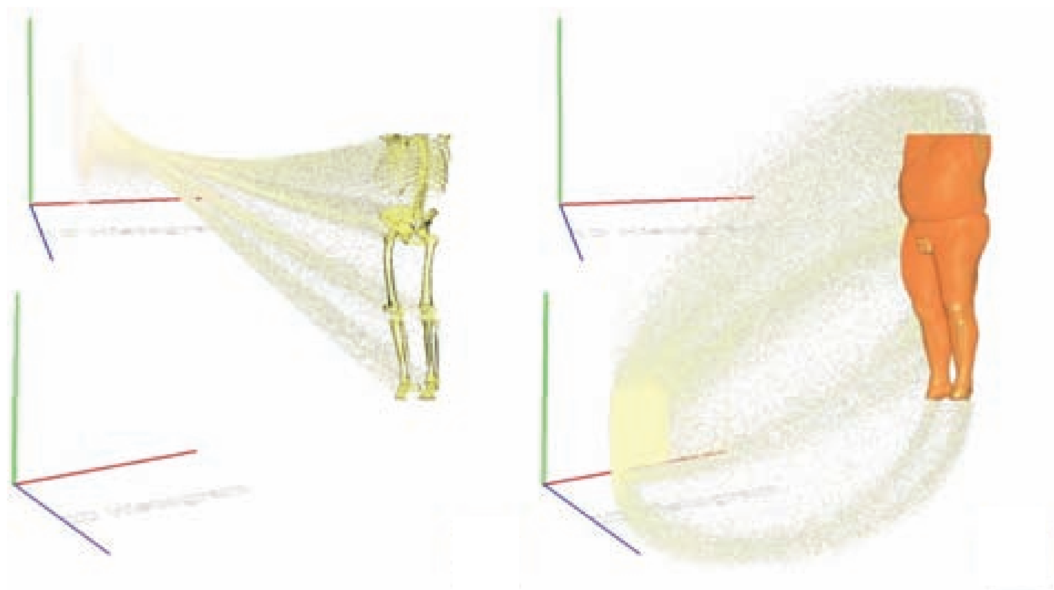


**Figure 11** Two separate frames of the transition from DVR to scatter plots. [7]

statistical data. The example reconstructed in $\mathcal{A}$-space is an integration between scatter plots and bar charts. A benefit of this visualization is that the spatial relationship between sample points is visualized in the transition. The technique allows several different transitions between the statistical visualizations. Figure 10 shows in a schematic way two possible transitions from a scatter plot to a bar chart. This visualization technique is just one example of many approaches that exist for blending 2D to 2D algorithms.

## 2.7 Scattered Direct Volume Rendering

Our last somewhat speculative example from $\mathcal{A}$-space is the work by Rautek and Gröller [7] where a quite unusual integration is taking place. The integration is between 3D and 2D, specifically between DVR and scatter plots. On one side a volume rendering is shown and on the other side a scatter plot. In an animated transition the voxels are moving from the 3D volume-rendering space to their appropriate location in the 2D scatter plot and vice versa. Figure 11 shows two separate frames of the animated transition between DVR and scatter plots.

## 3 On the Nature of $\mathcal{A}$-Space

In the previous sections we have sketched the concept of $\mathcal{A}$-space. Via examples we have shown how visualization algorithms are points in $\mathcal{A}$-space and reconstruction is the process of blending between algorithms. However this concept entails more than this and in this section we will indicate further aspects of $\mathcal{A}$-space and open issues. $\mathcal{A}$-space is not a space in the strict mathematical sense. It shall act as a thought-provoking concept. Real-world phenomena are increasingly measured through several heterogeneous modalities with quite different characteristics. We believe that this increased data complexity can be tackled through integrated views. $\mathcal{A}$-space may help to more systematically explore the possibilities to blend together diverse visualization approaches with complimentary strengths. In the following we shortly discuss various open issues concerning $\mathcal{A}$-space.

**Interpolation and reconstruction** In the examples we have shown various types of blending algorithms together. Loosely we have called this blending interpolation and reconstruction. Can other types of interpolation be transferred to $\mathcal{A}$-space? Would it be possible to use barycentric coordinates to smoothly and simultaneously interpolate between three or more algorithms? Having several data sources for the same phenomenon makes fusion often a necessity. Fusion at the data and image level have been around for a long time. The visualization pipeline, however, consists of many more steps from the data to the final image. It is here where algorithm fusion and $\mathcal{A}$-space come into play. We cannot fight increased data complexity with increased visual complexity. Therefore a sensible step would be to move from linked to integrated or combined views.

**Dimensionality and units** What is the dimensionality of $\mathcal{A}$? What are the dimensions of $\mathcal{A}$? Would knowing the *coordinates* of an algorithm give some insight into the possibilities of reconstruction or the compatibility of algorithms. For example do algorithms in the same plane share some features? What kind of units does $\mathcal{A}$-space use? Is $\mathcal{A}$ a metric space? What is the distance between two algorithms and how does one measure this distance? Currently algorithms are often categorized according to their spatial and temporal asymptotical complexity. Could visualization algorithms be categorized according to other measures like visual complexity, number of algorithms integrated, algorithm length? Does $\mathcal{A}$-space have a set of basis algorithms where all other visualization techniques can be reconstructed from?

**Transformations** Which transformations make sense in $\mathcal{A}$-space? Let us assume we are starting with a linked view as the initial visualization where all the component algorithms are known points in $\mathcal{A}$-space. Is it possible to create a generic transformation that would convert such a visualization into an integrated visualization based on the linking between the views? Would that process reconstruct a new point in $\mathcal{A}$-space or create a mapping to an already known point?

**Iso-algorithms** Iso-surfaces are very important in the scientific-visualization domain. Analogously are there iso-algorithms in $\mathcal{A}$-space, with the visual complexity as iso-value for example?

**Subspaces** Into what subspaces can $\mathcal{A}$-space be subdivided? Would the subspaces correspond to natural categorizations such as 2D and 3D techniques or information visualization and scientific-visualization techniques?

**Local neighborhood** Given a point in $\mathcal{A}$-space how would the local neighborhood look like? Example measures might be gradient, divergence, curl. What would be the gradient, divergence or curl of DVR or MIP, i.e., $\nabla DVR$, $\nabla MIP$?

**Interaction** Given the typically dense overlapping in integrated views, interaction has hardly been explored in this context. An interaction event might simultaneously navigate in several spaces. How can the user be supported to efficiently interact with integrated views?

The above list of aspects and open issues of $\mathcal{A}$-space is for sure not complete. Creating integrated visualizations is currently done in an ad hoc way. Sparse data is reasonably simple to integrate, but the difficulty of integration increases with the density of the data. Dense data may have many important features collocated both spatially and temporally and currently there is no general way of solving this. Maybe some sort of exploded views in space and time could be an answer for this?

Categorizing the algorithms in $\mathcal{A}$-space may help in defining the boundaries of $\mathcal{A}$. The categorization may be to differentiate between fine and coarse visualization integration or to differentiate at what stage the integration is performed, i.e., data stage, algorithm stage, or image stage.

## 4 Conclusion

Integrated visualization will become more important in the future. Integrated views are a not yet fully explored area and they are one answer to cope with increased data complexity. We have shown some of the possibilities in $\mathcal{A}$-space and we think it may be a new direction on how to look at ways to perform visualization integration. $\mathcal{A}$-space might be a useful tool for classifying and indicating the possibilities of integrated visualizations. There already exist many integrated visualizations that may benefit to be localized in $\mathcal{A}$-space. Increasing the population of $\mathcal{A}$-space would also indicate untapped regions where reconstruction is a possibility and could lead to new integrated visualizations. Fill in the holes of $\mathcal{A}$!!!

### References


1 Jean-Paul Balabanian, Ivan Viola, and Eduard Gröller. Interactive illustrative visualization of hierarchical volume data. In *Proceedings of Graphics Interface 2010*, 2010.

2 Stefan Bruckner, Sören Grimm, Armin Kanitsar, and Eduard Gröller. Illustrative context-preserving exploration of volume data. *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1559–1569, 2006.

3 Stefan Bruckner and Eduard Gröller. Instant volume visualization using maximum intensity difference accumulation. *Computer Graphics Forum (Proceedings of EuroVis 2009)*, 28(3):775–782, 2009.

4 Markus Hadwiger, Christoph Berger, and Helwig Hauser. High-quality two-level volume rendering of segmented data sets on consumer graphics hardware. In *VIS'03: Proc. of the 14th IEEE Visualization 2003 (VIS'03)*, pages 301–308. IEEE Computer Society, 2003.

5 Helwig Hauser, Lukas Mroz, Gian Italo Bischi, and Eduard Gröller. Two-level volume rendering. *IEEE Trans. on Visualization and Computer Graphics*, 7(3):242–252, 2001.

6 J. Heer and G.G. Robertson. Animated transitions in statistical data graphics. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1240–1247, 2007.

7 Peter Rautek and Eduard Gröller. Scattered direct volume rendering. Personal communication, 2009.

8 Matúš Straka, Michal Červeňanský, Alexandra La Cruz, Arnold Köchl, Miloš Šrámek, Eduard Gröller, and Dominik Fleischmann. The VesselGlyph: Focus & context visualization in CT-angiography. In *VIS '04: Proceedings of the 15th IEEE Visualization 2004 (VIS'04)*, pages 385–392. IEEE Computer Society, 2004.