Bump Mapped Vector Fields

Alex Pang and Naim Alper

Baskin Center for
Computer Engineering & Information Sciences
University of California
Santa Cruz, CA 95064

ABSTRACT

We present alternative ways of looking at vector fields that complement existing flow visualization methods. These techniques are based on bump mapping and are simple, easy to compute and fast to render. For example, in one of the techniques a surface from the vector field is extracted and the directions of the vector values on the surface are normalized and used as surface normals. The shading of the surface provides directional information. By manipulating the position and color of light sources, different regions with particular vector directions can be highlighted or hidden allowing direction based selection. Since the technique simply uses the directions of the vector as surface normals to bump the surface, it can also be used on irregularly sampled flow fields as well as for visualizing flow directions on curved surfaces. The magnitudes of the vector values can be optionally mapped to color.

Keywords: bump mapping, flow visualization, vector field

1 INTRODUCTION

There are several features that may be of interest when analyzing a vector field. The most common would be the *magnitudes* and *directions* of the vectors at different positions. Other features of interest include *twists* and *divergence* as well as general *flow patterns* in the the vector field. These features can be visualized by a host of 2D and 3D flow visualization techniques. It is difficult to say that any one particular technique is superior as they all have advantages and disadvantages depending on the attributes that are desired. Different users may prefer different tools and often, the use of more than one technique will result in better insight.

In this paper, we present alternative ways of visualizing vector fields that add to this collection of techniques. We have investigated several techniques that were inspired by the computer graphics technique of bump mapping. Our techniques are simple, easy to compute and fast to render. In one, a surface is first extracted from the vector field and the direction of the vectors at the surface are normalized and used as the normals to the surface. The resulting shading highlights directional changes and regions of similar directions. By manipulating light sources, vectors can be hidden or highlighted based on their directions. A variation on this adds the normalized vector to the normal of the surface rather than replacing it. A third technique uses the magnitude of the vector field as a bump map.

In section 2 we discuss the main vector field visualization techniques and compare them in their effectiveness in displaying the different flow features. Next we describe our techniques and show some sample visualizations in section 3.

2 RELATED WORK

There is a wide selection of flow visualization techniques. These techniques can be categorized using different criteria such as: the data dimensionality supported, whether advection is required, whether the method extends to time varying fields, the types of graphics primitives they generate, and the types of flow features the methods are designed to highlight. For example, Max *et al.*[1] compare several techniques by a variety of parameters such as dimensionality, hardware acceleration, etc. In this section, we first group various flow visualization methods into three major groups. We then provide a qualitative comparison of the methods under each group as to their strengths in visualizing the different flow features.

We choose to organize the different flow visualization methods according to the visual representation that is used to present the flow data. The three groups that we have identified are those that: (a) map graphics attributes, (b) use iconic representations, and (c) generate geometric primitives.

Attribute Mappping

These methods are characterized by the mapping of physical flow parameters to attributes such as color, roughness or texture. The most common example in this group is pseudocoloring. For example, Johannsen and Moorhead[2] map vector direction to hue, and vector magnitude to saturation or value in an HSV color model. The *LIC*[3] and *spot noise*[4] techniques use textures instead. Direct volume rendering techniques for vector visualization may also be considered as one of these methods as flow parameters are simply mapped to visual attributes and rendered. Those methods that use auxiliary information such as lighting and viewing conditions may likewise be grouped here. The method proposed in this paper falls under this category in that flow information is mapped as surface attributes (surface normal) and together with lighting conditions can selectively highlight different aspects of the flow field.

Iconic Representation

This group of methods is characterized by their use of specialized glyphs to represent flow information. The field to be visualized is usually made up of sparse and discrete measurements, although this is not a requirement. The most popular method in this group is the *hedgehog* approach where each point in the flow field where a measurement is available is represented by an arrow. The length of the arrow (sometimes also overloaded with color) is used to represent the vector magnitude. In special domains such as meteorology, specialized glyphs called *wind barbs* are used to encode flow information. More recent work in this category include the use of moving iconic objects called *boids*[5] and flow probes.[6] Some of these glyphs may also be extended to include other information such as uncertainties in the measurements.[7]

Geometric Primitives

This group of methods is characterized by the geometric primitives that are generated during the visualization process and left behind in the field. Geometric primitives may be as simple as points and lines or more complicated such as tetrahedrals. Unlike the discrete and sparse nature of iconic representations, these methods usually work with denser fields. In fact, most of the effort in flow visualization research today is directed at developing methods that would belong in this group. Examples in this category include: surface particles,[8] streamlines and its numerous variants: stream ribbons,[9] stream polygons[10] and streamballs.[11]

Comparison

Faced with a wide selection of flow visualization techniques, users need to make intelligent choices as to which techniques would best suit their needs. We give a qualitative comparison of the different techniques according to the features in the flow field that the users may be interested in extracting and understanding. The flow features that are commonly of interest are: flow magnitude, flow direction, general flow pattern, twists and vortices, and regions of divergence and convergence. Other features such as acceleration, shear, fronts, and others may also be of interest but are not considered in this comparison. The entries in the following table are qualitatively ranked as $+$, $\surd$, and $-$. Those with $+$ indicate strengths in visualizing the desired features. Those marked with $\surd$ indicate that the methods can be used to visualize those features, while those with $-$ indicate weakness or deficiency of the method to visualize that particular flow feature.

Features/Methods	lic/spot/hsv/bump	particles/lines	ribbons/polys/tubes/balls	icons
magnitude	$\surd$	$\surd$	$\surd$	$\surd+$
direction	$\surd$	$\surd$	$\surd$	$\surd+$
flow pattern	$+$	$\surd-$	$\surd$	$\surd-$
twist/rotation	$-$	$\surd-$	$+$	$\surd-$
convergence/divergence	$+$	$\surd$	$\surd$	$\surd-$

The rankings above are very qualitative in nature and need additional information and qualification to be effective as a selection mechanism. For example, while the streamlines technique can give an indication of flow pattern, it is sensitive to the seeding positions of the lines (see figures 2 and 3). Dimensionality of the field needs to be taken into account as some methods do not extend well, e.g. 3D arrow glyphs need to be projected onto a 2D display and both magnitude and direction can be distorted. The effectiveness of the different methods also vary depending on the spatial density of the data to be visualized. For example, hedgehogs can provide an indication of the flow pattern only if the measurements in the field are dense. Most of the methods above can also be made more effective by adding animation. The comparisons are also made among groups of methods instead of among individual methods and thus some generalizations are introduced. Some methods have also been left out of this grouping. At any rate, the table provides a first pass at selecting a method appropriate for visualizing the flow features of interest.

In the next section, we present a brief description of bump mapping and the different options of how to map the different flow parameters to surface and lighting attributes.

3 BUMP MAPPED VECTOR FIELDS

The vector field visualization techniques presented in this paper are related to a computer graphics technique called *bump mapping.* Bump mapping was proposed by Blinn to make smooth surfaces appear rough or wrinkled.[12] It is a technique where the normal to the surface at a point is perturbed by an amount dependent on a perturbation function. Given a point on a surface $\mathbf{Q}(u, w)$, the normal n at that point is given by

$$\mathbf{n} = \mathbf{Q_u} \otimes \mathbf{Q_w}$$

where $\mathbf{Q_u}$ and $\mathbf{Q_w}$ are the partial derivatives in the parameter directions u, w. A new displaced point can be defined by adding some amount along the normal at that point:

$$\mathbf{Q}^{'}(u, w) = \mathbf{Q}(u, w) + P(u, w)\frac{\mathbf{n}}{\mid \mathbf{n} \mid}$$

where $P(u, w)$ is a perturbation function. The new normal to the perturbed surface is then given by

$$\mathbf{n}^{'} = \mathbf{Q}^{'}_{\mathbf{u}} \otimes \mathbf{Q}^{'}_{\mathbf{w}}$$

which can be shown to reduce to

$$\mathbf{n}^{'} = \mathbf{n} + \frac{P_u(\mathbf{n} \otimes \mathbf{Q_w})}{\mid \mathbf{n} \mid} + \frac{P_w(\mathbf{Q_u} \otimes \mathbf{n})}{\mid \mathbf{n} \mid} \tag{1}$$

if P is small. In other words, a new normal is calculated based on the original normal and the local effect of P. The perturbation function can be mathematically defined or can be a two dimensional lookup table. Even though the surface $\mathbf{Q}(u, w)$ is not actually displaced, the use of the perturbed normal $\mathbf{n}^{'}$ in conjunction with lighting calculations give the illusion of a bumpy surface.

Bump mapping can be applied to vector field visualization. In this context, the goal is not image synthesis but information extraction. Therefore, rather than trying to make the flow field appear bumpy, we attempt to map flow parameters to equation 1 in order to highlight flow features. Specifically, we experimented with three variations. In one, we simply replace the surface normal at a point by the normalized direction of the vector at that point (equation 2). In the second, we add the normalized vector to the normal at the surface (equation 3). A third technique is more faithful to the bump-mapping idea and uses the vector magnitude as the perturbation function (equation 4). The respective equations are as follows:

$$\mathbf{n}^{'} = \mathbf{v} \tag{2}$$

$$\mathbf{n}^{'} = \mathbf{n} + \mathbf{v} \tag{3}$$

$$P(u, w) = |\mathbf{v}| \tag{4}$$

where $\mathbf{n}$ is the normal, $\mathbf{v}$ is the vector and $P(u, w)$ is the value of the perturbation function at the surface point.

In the first two variations, the shading resulting from the rendering of the surface possesses the vector field's directional information. Only the vectors that point towards a light source will be lit. By controlling the number, type and positions of light sources, vectors can be displayed selectively: regions where the vector field points towards a light source in general will be highlighted while those pointing away will appear dark. For example, regions of easterly flows can be isolated from the rest of the flow field by placing a light source to the east of the field; regions of "horizontal" flows (east-west) can be highlighted by placing a light source to the east of the field and another to the west of the field; regions of inward (versus outward) flows may also be quickly identified by placing a light source on the appropriate side.

There is little difference between the first two variations. In the simplest case, the field is 2D or a slice from a 3D field. In these situations, the surface normals are all alike and do not contribute to distinguishing features in the field. Thus, the first variation of simply replacing surface normals with vectors is sufficient. On the other hand, if the surface is 3D such as when visualizing flow fields near surfaces (e.g. over an airfoil or on an isosurface), the surface normal contributes to providing some indication of surface curvature. If the surface normals were simply discarded as in the first variation, then the rendered surface will appear flat. On the other hand, when both surface normals and vectors are combined as in the second variation, care must also be taken in the interpretation as it becomes difficult to isolate shading differences due to surface curvature separately from those due to vector directions. In this case, it is probably necessary to study the surface prior to visualizing the flow on the surface.

The third variation uses the vector magnitude in the field to perturb the surface containing the flow information. The resulting images produce a bumpy looking field where the heights of the bumps correspond to vector magnitudes. Since the vector directions are not included in this method, they must be overloaded using other methods such as mapping directions to hue values in an HSV color scheme (see figure 8). In addition, because of the difficulty to parameterize an arbitrary 3D surface, we have only applied this method to 2D rectilinear vector fields.

Aside from the three different variations discussed above, we also have many parameters to work with as far as the light sources are concerned. They can be point, infinite, or spot lights. They can also be white or colored. The positions of the light sources can be fixed or movable relative to the field. We can further change the material definition of the surface to increase or decrease specularity.

Results

We tested our methods and compared them with existing techniques using data from the Naval Research Laboratories' (NRL) Navy Operational Regional Atmospheric Prediction System (NORAPS).[13] The 3D vector fields are made up of two-component vectors stacked up on each other. A horizontal slice of this field is illustrated using vector glyphs in figure 1. Only every third sample along each axis have been selected. Notice how the image gets cluttered even though we have subsampled the data. Also, in three-component vectors in 3D, the directions and magnitudes would be hard to interpret.

Figure 2 and figure 3 show the same slice using streamlines. The streamlines have been seeded at columns $i = 30$ and $i = 80$. We obtain different images for the same slice by seeding the streamlines at different places. Also, the direction of flow is not clear (left to right or right to left), although this can be remedied by placing intermittent glyphs along the streamlines.

Figure 4 shows a pseudocolored slice of the field for reference. The magnitude of the vectors are mapped to color while the directions are not mapped and there are no lights. If we now replace the surface normals with the vector directions at the vertices of the slice and place an infinite light coming from the east, we obtain figure 5. It displays the vectors that are mainly towards the east. Figure 6 shows the same with a second light source placed to the west . The dark areas remaining mean that vectors in those areas are mainly in the north-south direction.

If the shape of the surface is nonplanar, such as when visualizing flow information on an isosurface, it is best to know the surface geometry before overloading flow information. In this case, the second technique of adding vector direction to the normals retains the surface information better while displaying the flow information. Figure 7 shows an isosurface of a temperature field with the flow field visualized on it.

Figure 8 shows an example where the vector magnitude is used to perturb the surface normals while vector directions are mapped to hue. The "higher" bumps have greater vector magnitudes.

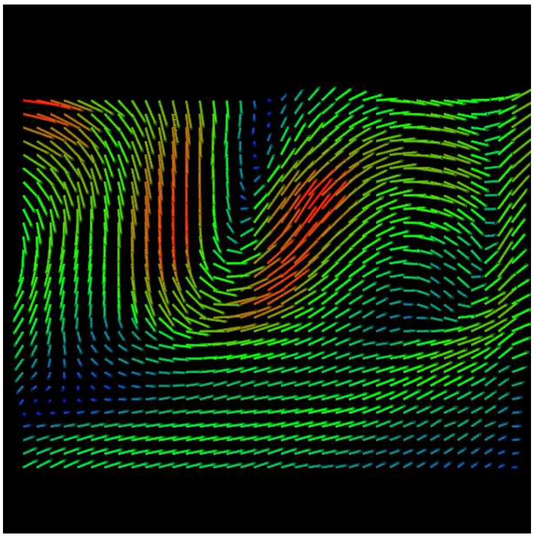

Figure 1: Visualization of a vector field using vector glyphs. To reduce image clutter, every third sample has been taken.

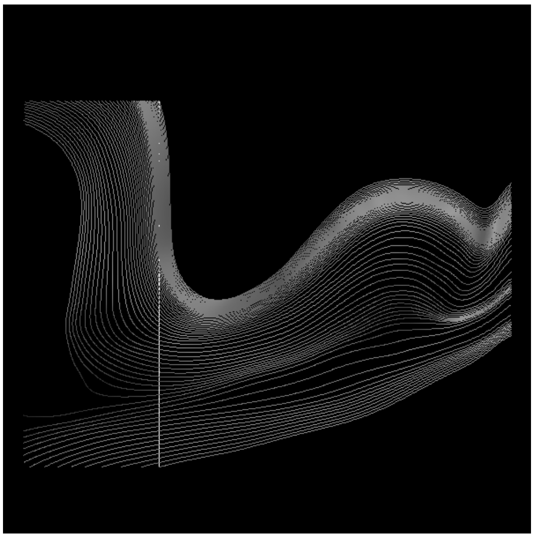

Figure 2: Visualization of a vector field using streamlines. A series of streamlines are seeded at column i=30.

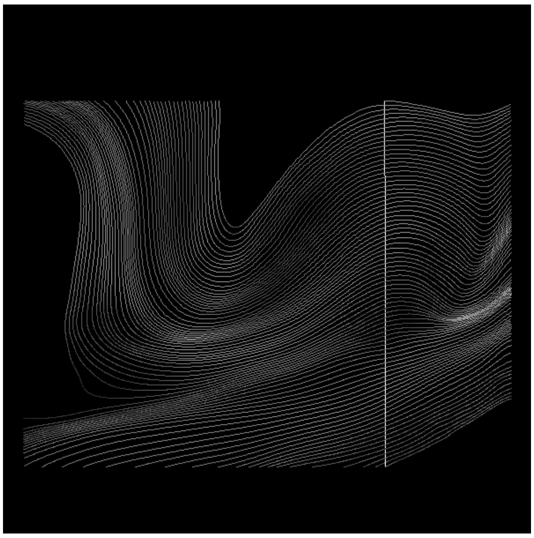

Figure 3: Streamlines seeded at column i=80 produces a different result of the same field, illustrating the sensitivity of streamlines to seed points.

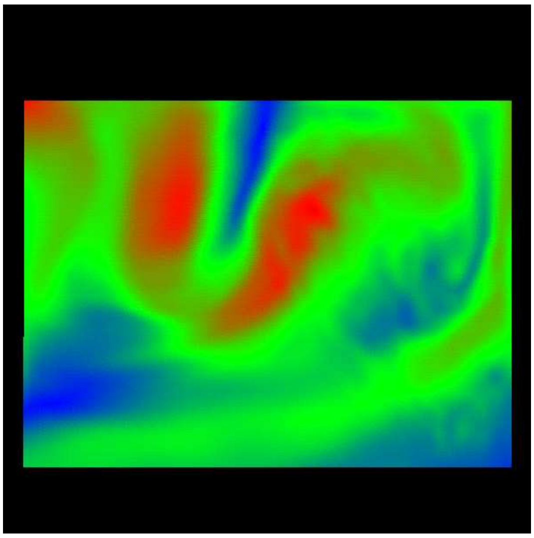

Figure 4: This image just maps the magnitude of the vectors to a colormap. Red is high, blue is low. There is no direction mappping.

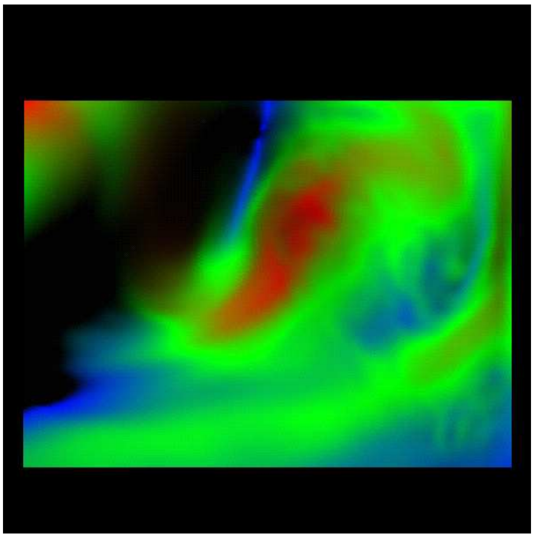

Figure 5: Here, the normalized vector values are used as normals. In this image, there is a single, white, infinite light source coming from the right.

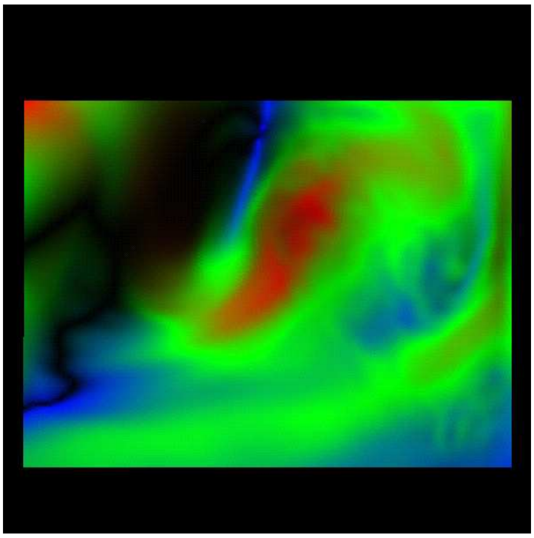

Figure 6: When a second, similar light is added to the left, the vectors pointing to the left are also highlighted.

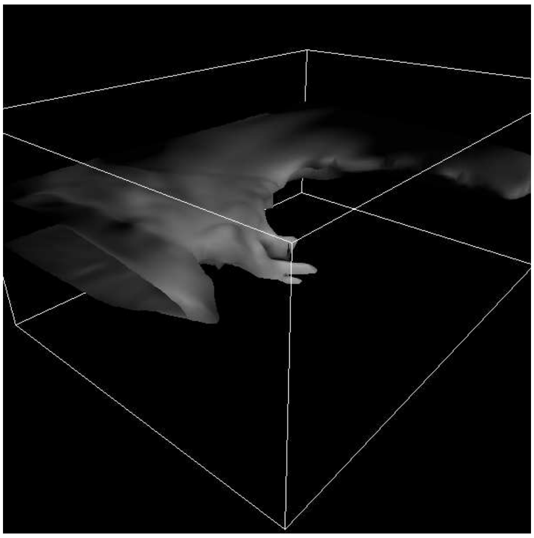

Figure 7: This image shows an iso-surface of a temperature field where the vector magnitude is used to pseudocolor the surface. The vector directions are added to the normals at the surface.

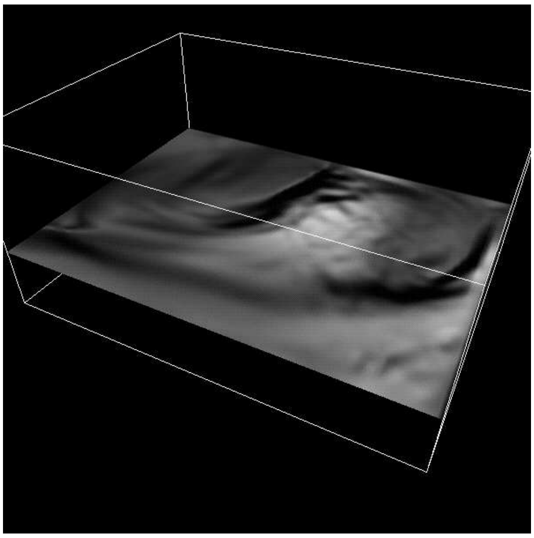

Figure 8: In this image, the vector magnitude is used as a bump map and the vector direction is mapped to hue of an HSV color space where S and V are both 1.

4 SUMMARY

We have presented alternative ways of visualizing flow fields that allow easy identification of regions with similar flow directions. They are based on mapping flow parameters to visual attributes and lighting conditions. Among the advantages of our methods are:

- cheap to compute
- can do direction based region selection
- can be overloaded using more visual attributes and lighting parameters

ACKNOWLEDGEMENTS

Spray[14] facilitated the investigation of bump mapped vector fields and we would like to thank the other members of the *Spray* team: Jeff Furman, Elijah Saxon, Tom Goodman, and Craig Wittenbrink. We would also like to thank Professor Wendell Nuss and Dr. Paul Hirschberg for providing NRL's NORAPS model output for this work. This work is funded in part by ONR grant N00014-92-J-1807.

5 REFERENCES

[1] Nelson Max, Roger Crawfis, and Charles Grant. Visualizing 3d velocity fields near contour surfaces. In *Proceedings: Visualization '94*, pages 248 – 255. IEEE Computer Society, 1994.

[2] A. Johannsen and R. Moorhead. Visualization of mesoscale flow features in ocean basins. In *Proceedings: Visualization '94*, pages 355 – 358. IEEE Computer Society, 1994.

[3] Brian Cabral and Leith(Casey) Leedom. Imaging vector fields using line integral convolution. *Computer Graphics (ACM Siggraph Proceedings)*, 27(4):263 – 270, August 1993.

[4] J. J. van Wijk. Spot Noise: Texture synthesis for data visualization. *Computer Graphics*, 25(4):309 – 318, 1991.

[5] G. D. Kerlick. Moving iconic objects in scientific visualization. In *Proceedings: Visualization '90*, pages 124 – 129. IEEE Computer Society, 1990.

[6] Willem C. de Leeuw and Jarke J. van Wijk. A probe for local flow field visualization. In *Proceedings: Visualization '93*, pages 39 – 45. IEEE Computer Society, 1993.

[7] C. M. Wittenbrink, E. Saxon, J. J. Furman, A. Pang, and S. Lodha. Glyphs for visualizing uncertainty in environmental vector fields. In *IS&T/SPIE Proceedings: Visual Data Exploration and Analysis II*, volume 2410, 1995. In these proceedings.

[8] J. J. van Wijk. Rendering surface particles. In *Proceedings: Visualization '92*, pages 54 – 61. IEEE Computer Society, 1992.

[9] J. P. M. Hultquist. Constructing stream surfaces in steady 3d vector fields. In *Proceedings: Visualization '92*, pages 171 – 177. IEEE Computer Society, 1992.

[10] W. J. Schroeder, C. R. Volpe, and W. E. Lorensen. The Stream Polygon: A technique for 3D vector field visualization. In *Proceedings: Visualization '91*, pages 126 – 131. IEEE Computer Society, 1991.

[11] Manfred Brill, Hans Hagen, Hans-Christian Rodrian, Wladimir Djsatschin, and Stanislav V. Klimenko. Streamball techniques for flow visualization. In *Proceedings: Visualization '94*, pages 225 – 231. IEEE Computer Society, 1994.

[12] James F. Blinn. Simulation of wrinkled surfaces. *Computer Graphics (ACM Siggraph Proceedings)*, 12:286 – 292, 1978.

[13] Richard M. Hodur. Evaluation of a regional model with an update cycle. *Monthly Weather Review*, 115(11):2707–2718, Nov. 1987.

[14] A. Pang. Spray rendering. *IEEE Computer Graphics and Applications*, 14(5):57 – 63, 1994.