Spacetime Raytracing for Visualization of Gaseous Volumes Modeled by Particle Systems

Neeharika Adabala [1] and Swami Manohar[1,2]
[1]Supercomputer Education and Research Center
[2]Computer Science and Automation
Indian Institute of Science, Bangalore 560012

Abstract

Gaseous phenomena are challenging to model in computer graphics as they have a fuzzy geometry and exhibit dynamic behavior. Particle systems can capture the fuzzy geometry and dynamic behavior of gaseous volumes with ease. However, a large number of particles are needed in order to generate realistic images. Also the technique of motion blurring individual particles for generating images results in realistic images only for a limited class of applications where particles can be treated as individual sources of light. This technique cannot be applied to gaseous volumes and hence alternative techniques to incorporate temporal information into images of gases have to be investigated. In this paper we introduce a technique of spacetime raytracing four-dimensional particle maps, that results in a reduction the of number of particles required to model gases, and also, allows inclusion of temporal information into images of gases.

1 Introduction

The ubiquity of gaseous phenomena in real world scenes make their modeling of importance in synthetic computer graphics generated scenes[1]. Gases do not possess definite surfaces and hence are said to have a fuzzy geometry. The geometry of gases is represented by textures [2] or particles [3] in computer graphics.

Gases also usually exhibit dynamic behavior and so their geometry varies with time. The dynamic behavior is captured, in the case of gases represented by textures, by either recomputing the texture [2] or by changing the shape of the geometry which is to be textured [4]. The former approaches are time consuming and are difficult to implement because of the absence of a direct relationship between the parameters that vary the appearance of the texture and the parameters that define the motion of the gas. The latter approaches vary in complexity depending on the geometry that is modified to reflect the gaseous dynamics. These techniques become progressively difficult to handle with the complexity of the geometric primitives used.

In the case of particle representation of gaseous volumes the particles are displaced in order to reflect the dynamic behavior of the gas. It is found that use of physics based equations to model the dynamics of gaseous volume reduce the effort needed to design flows that appear visually convincing [5]. The simplicity of incorporating motion based on equations of physics into particle systems as compared to texture based approaches has lead to the favoring of particle representation of gases.

However, particle based methods require a large number of particles to model the gaseous volume and result in images with limited realism if the particles are rendered directly. On the other hand the texture based methods result in realistic images. Therefore techniques have evolved which combine the particle and texture based approaches to obtain the benefits of both the approaches [6] [7]. Foster et. al. [7] represent the fluid as particles and resolve the particle distributions that result after dynamics simulation into densities on a grid before rendering. Such a technique is influenced by the resolution of the grid and requires a large number of particles. An alternative approach of resolving particles into densities is by using particle maps [8]. This is a gridless approach that allows evaluation of densities on the fly during rendering. Stam et. al. [9] model the gas as a distribution of collections of particles namely "blobs". A wind field is used to modify the density distribution. In the above approach all the blobs are diffused in a similar manner by the wind field resulting in regular shape of blobs. This gives an artificial appearance to the gas. This problem is overcome by allowing the wind field to warp the blobs non-uniformly [6]. The warping of blobs by the wind field is achieved by tracing the points in a blob backwards for a short time step. The non-uniform blob shapes result in more realistic images of the gas. Warping blobs in time allows inclusion of temporal information into individual images, and by adopting this approach Stam et. al. [6] were able to reduce the number of blobs required to model the fluid.

The work of Foster et. al. [7] uses a large number of particles, of the order of thousands of particles per second of simulation. No technique for inclusion of temporal information in individual images is used.

In this work we introduce the use of spacetime raytracing for rendering gaseous volumes modeled by particle systems. This technique allows the inclusion of temporal information into individual images and leads to a reduction, by an order of magnitude, in the number of particles required to model the gas. We typically use tens to hundreds of particles for our modeling. This is achieved by retaining previous locations of particles to a few time steps backwards in the dynamics simulation. The three dimensional particle map introduced in [8] now becomes four dimensional as each particle is associated with time. The densities which are evaluated from a particle map are also associated with time. By suitably weighting the "older" densities at a location, a motion blurred image of the dynamic gaseous volume is generated.

In the following section we give a brief overview of spacetime raytracing. Details about four-dimensional particle maps are presented in Section 3, followed by a brief description of a method to combine spacetime raytracing with four-dimensional particle maps in section 4. Section 5 gives the implementation details, and presents the proposed algorithm. Section 6 gives a discussion of results that demonstrate the effectiveness of our technique. Conclusions are given in section 7.

2 Spacetime raytracing

Spacetime raytracing was developed by Glassner [10] as an efficient technique to render scenes for animation. In this technique a spacetime volume of the animation scene is constructed by considering the successive locations that the objects in the scene will occupy with time. A technique of building a bounding volume hierarchy is applied on the space time volume to construct a structure which combines the advantages of adaptive space subdivision and bounding volume techniques. Spacetime raytracing is performed on this structure. Motion blur is naturally achieved by this technique by associating different times with the primary rays.

In this paper we apply the concept of spacetime raytracing to achieve a motion blurred image of a gaseous volume defined by particles. Particles have been individually motion blurred and rendered in the past [3], [11]. However, these techniques result in realistic images for very limited cases like fireworks, explosions, etc., where particles can be treated as sources of light. When the particles define a distribution of density, as in the case when they are modeling gaseous volumes, the problem of motion blurring is non-trivial.

The main difference between the situation considered by Glassner [10] and the problem considered in this paper is that the dynamic objects in our case are particles. It is not feasible to treat the particles by a similar technique as described for objects in a scene and perform spacetime raytracing by intersecting rays with particles. To overcome this problem the level of abstraction is moved up so that the particles are no longer treated individually but instead used to define the distribution of density within the gaseous volume. This concept of the particles defining a density distribution can be implemented by either resolving the particles into densities on a grid as done by Foster et. al. [7] or by using a particle map which allows the estimation of densities on the fly from the results of nearest neighbor searches. The former approach is effected by the choice of grid resolution and fails when few particles are used while the latter is grid free.

An additional advantage of the particle maps approach is the natural way in which it can be extended to accommodate spacetime raytracing. This is achieved by retaining previous locations of particles as the dynamics computation proceeds for successive time steps. All particle positions are associated with the time when the particle exists at the location. These four-dimensional particles are used to construct the particle map. Densities estimated by performing nearest neighbor searches on this spacetime particle map are associated with time. By suitably weighting these densities based on the time step, motion blurred images of the gaseous volume are generated.

The following section describes four-dimensional particle maps.

3 Four-dimensional particle maps

A gaseous volume consists of a distribution of continuously varying densities. Rendering of the gaseous volume involves modeling the way light interacts with the volume. In the case of inhomogeneous media it is necessary to know the density at locations within the medium to determine how light interacts with the media, namely, whether it is scattered, transmitted or absorbed by that particular region. Also, there is a need to perform an integral of densities along the rays of light to determine the amount of light that passes through. Therefore, it can be concluded that the ability to determine density at any location within the model of the gaseous volume is of prime importance in computer graphics for generating a realistic image of a gaseous volume.

Texture based approaches are inherently advantageous for modeling of light interaction as they directly define the continuously varying densities of the gaseous volume. A rendering technique requiring the densities of the gas at any point within the gaseous volume can obtain it in a straightforward manner from the definition of the texture.

On the other hand particle based methods of modeling a gaseous volume are disadvantageous because of their discrete nature which makes them unsuitable for modeling a continuous medium. They have been applied successfully to generate realistic images only in cases like explosions [3] and fireworks [12] where the particles can be treated as sources of light and they do not represent a continuous volume. This problem is overcome by resolving the particle system into densities on a grid, but this approach makes the model dependent on the choice of resolution of the grid. The particle map method is a gridless technique that allows the modeling of a continuous volume with varying density based on a particle system. The technique defines a texture based on the distribution of particles in the particle system.

A particle map is a kd-tree [13], that stores the particle locations of a particle system. Densities within the volume defined by the particle system are estimated by performing a nearest neighbor search and applying a kernel density estimation technique which is a standard statistical method [14]. The integration along the ray is reduced to a single range search by this technique rather than a series of density estimates.

The particle maps technique introduced in [8] considered static images of gaseous volumes. When a gaseous volume is animated a large number of particles are required to capture the motion.

Stam et al. in [6] reduce the number of blobs required to model the gaseous volume by allowing the blobs to warp with the wind. This is achieved by backtracing the points in a blob backwards in time, leading to the inclusion of temporal information into individual images. Temporal information is included into particle maps by associating each particle location with the time at which the particle occupies the location. Thus each particle becomes four-dimensional namely, (x, y, z, t). The particle map becomes four-dimensional because the kd-tree is constructed from the four-dimensional particles.

In the following section we describe how spacetime raytracing is applied to the four-dimensional particle map.

4 Applying spacetime raytracing to four-dimensional particle maps

The extension of the particle maps to include time makes them four-dimensional and it is no longer feasible to perform the nearest neighbor and range searches on the particle map with a three dimensional specification of the ray. A four-dimensional specification of the ray has to be used: each ray is associated with the time it hits the gaseous volume. This allows the estimation of the density it encounters at that particular time instant.

It can often happen that the rest of the scene in which the dynamic gaseous volume is present is static. Therefore, we switch to spacetime raytracing only when the gaseous volume is hit by a ray. This is done by associating rays that hit the gaseous volume with time before searching the particle map. The raytracing algorithm keeps track of the current time step of the dynamic simulation. All search results obtained from the particle map are associated with time. The density at any location within the gaseous volume is computed as a weighted sum of densities at various times at that location. The weights are assigned depending on the time: densities evaluated far from the current time step are given a smaller weight than those close to the current time step. This approach of switching to spacetime raytracing only for the dynamic gaseous volume makes the proposed method different from the spacetime raytracing described in [10]. As images are usually generated with anti-aliased raytracing we integrate the spacetime density evaluations into the super-sampling performed for anti-aliasing. By this we avoid incurring additional cost for the time sampling that is performed for spacetime raytracing.

5 Implementation details

The results obtained from the nearest neighbor and range searches are not directly applicable to rendering. In this section we give the details for the case of low albedo participating medium. The high albedo case is implemented by merging particle maps with photon maps and is described in [15].

The brightness of light that reaches the eye in the case of low albedo participating medium is :

$$B = \int_{t_{near}}^{t_{far}} e^{-\tau \int_{t_{near}}^{t} \rho(x(u),y(u),z(u))du} \times \left[\sum_i I_i(x(t),y(t),z(t))p(cos(\theta))\right] \times \rho(x(t),y(t),z(t))dt \ , \quad (1)$$

where

$$I_i(x,y,z) = e^{-\tau \int_{s_{near}}^{s_{far}} \rho(x(s),y(s),z(s))ds}. \quad (2)$$

The integrals basically compute the light that passes through the medium lying between the limits of integration. In the particle map approach the integral is replaced by a cylindrical range search between the limits of integration. The result of the range search, which is the number of particles that lie in the path of the light, is converted to the amount of light that passes through the medium by using a formula similar to the one given in [16]. The attenuation is estimated by associating a radius r_p with each particle and using the formula :

$$f(N) = 1 - e^{-N(\frac{r_p}{R})^2} \quad (3)$$

where R is the radius of the cylindrical range. This formula gives the probability that a random ray in the cylinder intersects a particle, which is assumed to be a sphere of radius r_p.

The density estimate at any location within the gaseous volume is performed by using the kernel density estimation technique which is a standard statistical method [14]. In this approach the density is estimated by evaluating a kernel function at nearby points and dividing by the area under the kernel. A Gaussian kernel is used by us for density estimation. The density is given by

$$\rho = \frac{1}{n}\frac{\sqrt{k_2}}{\sqrt{\pi}}\sum_{i=1}^{n} e^{-k_2 d^2} \quad (4)$$

where d is the distance of the particle from the location at which the density is being estimated and k_2 is a parameter that takes positive real values. It can be tuned to control the appearance of the image: large values of k_2 result in dull images of the gaseous volume.

It should be noted that these values are associated with time and the final value used in the equation (1) is a weighted sum of these individual values. The weights are chosen proportional to $\frac{1}{t_c - t_e}$ where t_c is the current time step and t_e is the time step at which the estimation is made.

5.1 Algorithm

The following is the algorithm for the rendering of the gaseous volume represented by particle maps for the low albedo case:

1. For each time step repeat steps 2 to 5
2. If first iteration or time step when particle source is active : generate particles of the gaseous volume.
3. Create the particle map by building the optimized kd-tree [13], with the particles.
4. Spacetime raytrace the scene using the equation (1) for rendering the gaseous volume as follows:
   (a) Find the particles that lie along the path of the ray from the eye to the location of interest, point $P(x,y,z,t)$, in the gaseous volume by performing a cylindrical range search and transform the result using equation (3). Use the obtained value in place of the integral in the first term of the brightness equation.
   (b) Find the number of particles along the path of the ray from the light source to $P(x,y,z,t)$, compute attenuation as before and multiply by phase function.
   (c) Estimate the density at $P(x,y,z,t)$ by performing a nearest neighbor search. Use equation (4) to obtain the density value.

   Multiply the values obtained in the steps 4a, 4b and 4c and repeat step 4 by moving the point of interest along the ray in the volume. The time associated with the point of interest is also varied. Compute a weighted sum to obtain the brightness of the ray with weights decreasing as the time associated with the point of interest is further from the current time step.
5. Update particle locations using dynamics equations.

Presence of t in $P(x,y,z,t)$ takes care of spacetime raytracing. Adaptive step integration is used and the point $P(x,y,z,t)$ is jittered both in space and time to avoid aliasing problems. Different values of time, t, are used for the multiple samples evaluated for antialiasing, thus no overhead is introduced due to having to sample in time.

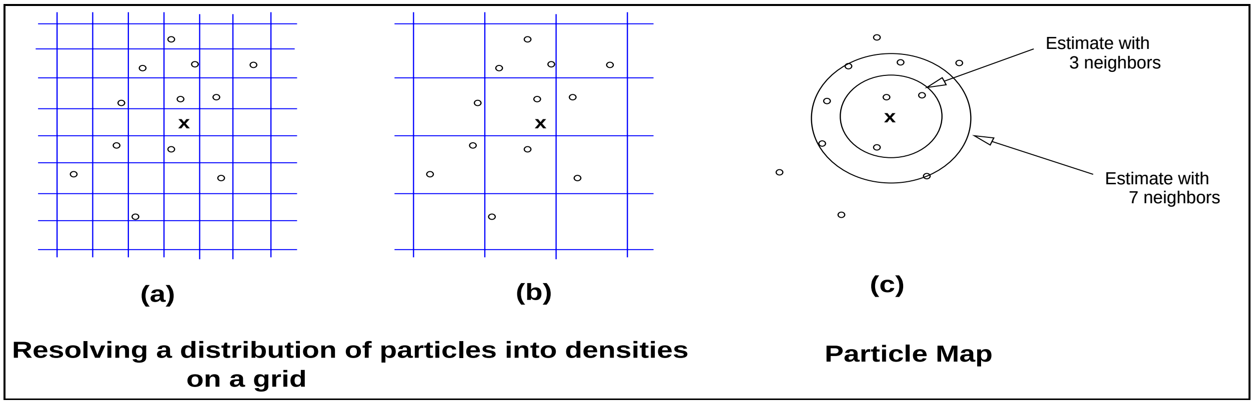

Figure 1: Comparison of grid and particle map methods for resolving a distribution of particles into densities.

6 Results and discussion

Some images that result from the implementation of the technique described in this paper are presented here. The images have been generated on a Pentium III (448 MHz) with 64MB RAM.

The grid-based methods of resolving a particle system into densities has the disadvantage of being dependent on the resolution of the grid used. This is illustrated in the Figure 1 where a distribution of particles at a time step is shown and two possible resolutions of grid sizes are shown. The grids are applied to define the density at the center of each grid-cell. It can be observed that choice of grid resolution can result in significantly different densities at the point of interest marked **x** in the Figure 1 (a) and (b). Even if interpolation techniques are used the value continues to vary. On the other hand the particle map method gives reasonably consistent estimates of the density even when the number of particles used to estimate the density is varied. This is because the kernel used for density estimation reduces the contribution to density of particles far from the point of interest. The technique of estimating the density by using nearest neighbors is illustrated in Figure 1 (c). Cases when three and seven neighbors are used are shown.

Apart from change in estimated density with the resolution of the grid, it can also be observed that a grid-based technique requires a large number of particles in order that the grid cells do not remain empty within a gaseous volume. An empty grid cell results in zero density even though in a non-discretized method the density should have been a small non-zero value at that location. This is unlike the particle map approach which uses the distance of the nearest neighbors to estimate the density and hence can tackle a situation where the concentration of the gas is low within the gaseous volume.

The images presented in the Figure 2 represent frames from an animation of smoke rising from a tepee. These images are created without spacetime raytracing. Each pixel was sampled five times to perform anti-aliasing. The marked "blobby" appearance of the smoke can be observed in these images. The particles are introduced at the rate of about 10 particles per second.

The images presented in the Figure 3 represent the corresponding frames from the same animation generated with spacetime raytracing. It can be seen that there is a noticeable reduction in the "blobby" appearance of the smoke. This is because the density at each point in the gaseous volume is a weighted sum of the densities that existed at that point during the past few time steps, thus including temporal information in the images. The continuity between successive frames can be seen more clearly in these frames than the frames of Figure 2. This results in a smoother animation in the case of images generated by using the spacetime raytracing on particle maps.

The overhead introduced due to spacetime raytracing is minimal when anti-aliased images are considered because the spacetime raytracing is performed during the super-sampling of the pixels. The time taken to render a frame is dependent on the fraction of the image which contains the gaseous volume and it varies from 30 seconds to 25 minutes. The program has a good scope to be optimized and currently we are investigating techniques to speed up the algorithm.

7 Conclusions

Gaseous volumes are often modeled by particles because of the fuzzy nature of their geometry and the ease of inclusion of dynamics into a particle system. But a limitation of modeling with particle systems is that they do not result in realistic images unless a large number of particles are used. Also when motion blurred individual particles are used to render images they result in realistic appearance in very limited cases where the particles can be treated as sources of light. We have presented a new approach for the generation of realistic images of gaseous volumes with the following desirable features :

- The particle map structure is exploited to overcome the limitations of grid-based techniques.
- A reduced number of particles is required to model the gaseous volume. The reduction is from a few thousands of particles per second [7] to few tens of particles.
- It permits the inclusion of temporal information in individual images of an animation sequence leading to smoother animations.
- It allows the spacetime raytracing of just the dynamic gaseous volume without inclusion of the remaining static scene.
- No overhead is incurred due to space time raytracing as the time sampling is included in the super-sampling done during anti-aliasing. Thus, motion blur is achieved at no extra cost as the time sampling is rolled into the anti-aliasing.

The algorithm proposed has a good scope to be optimized and currently we are investigating techniques to speed up the algorithm.

Acknowledgements

Neeharika thanks Dr. Andrew Glassner for his useful suggestions which have helped to improve the paper.

References

[1] H.E. Rushmeier, "Rendering participating media: Problems and solutions from application areas," *Proc. of the 5th Eurographics Workshop on Rendering*, pp. 35–56, 1994.

[2] D. S. Ebert, K. Musgrave, D. Peachey, K. Perlin, and S. Worley, *Texturing and Modeling: A Procedural Approach*, Academic Press, 1994.

[3] W. T. Reeves, "Particle systems – a technique for modeling a class of fuzzy objects," *ACM Trans. Graphics*, pp. 91–108, 1983.

[4] Nelson Max, Barry Becker, and Roger Crawfis, "Flow volumes for interactive vector field visualization," *Proceedings of Visualization*, pp. 261–265, 1993.

[5] J. Stam, "Stable fluids," *Computer Graphics Proceedings, ACM SIGGRAPH '99*, pp. 121–128, Aug. 1999.

[6] J. Stam and E. Fiume, "Depicting fire and other gaseous phenomena using diffusion processes," *Proc. of SIGGRAPH '95*, pp. 129–136, 1995.

[7] N. Foster and D. Metaxas, "Modeling the motion of hot, turbulent gas," *Proc. of SIGGRAPH '97*, pp. 181–188, 1997.

[8] N. Adabala and S. Manohar, "Gaseous volume visualization using particle maps," in *EUROGRAPHICS '99 Short Papers and Demos*, Sept. 1999, pp. 209–213.

[9] J. Stam and E. Fiume, "Turbulent wind fields for gaseous phenomena," *Proc. of ACM SIGGRAPH '93*, pp. 369–376, 1993.

[10] A. S. Glassner, "Spacetime ray tracing for animation," *IEEE Computer Graphics and Applications*, vol. 8, pp. 60–70, Mar. 1988.

[11] Karl Sims, "Particle animation and rendering using data parallel computation," in *Computer Graphics (SIGGRAPH '90 Proceedings)*, Forest Baskett, Ed., Aug. 1990, vol. 24, pp. 405–413.

[12] Teng-See Loke, Daniel Tan, Hock-Soon Seah, and Meng-Hwa Er, "Rendering fireworks displays," *IEEE Computer Graphics and Applications*, vol. 12, pp. 33–43, May 1992.

[13] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Communications of the ACM*, vol. 18, no. 9, pp. 509–517, 1975.

[14] B. Walter, P. M. Hubbard, P. Shirley, and D.P. Greenberg, "Global illumination using local linear density estimation," *ACM Transactions on Graphics*, vol. 16, no. 2, pp. 217–259, Apr. 1997.

[15] N. Adabala and S. Manohar, "Modeling and rendering of gaseous phenomena using particle maps," *to appear in Journal of Visualization and Computer Animation.*

[16] J. F. Blinn, "Light reflection functions for simulation of clouds and dusty surfaces," *Computer Graphics (SIGGRAPH '82 Proceedings)*, vol. 16, no. 3, pp. 21–29, July 1982.

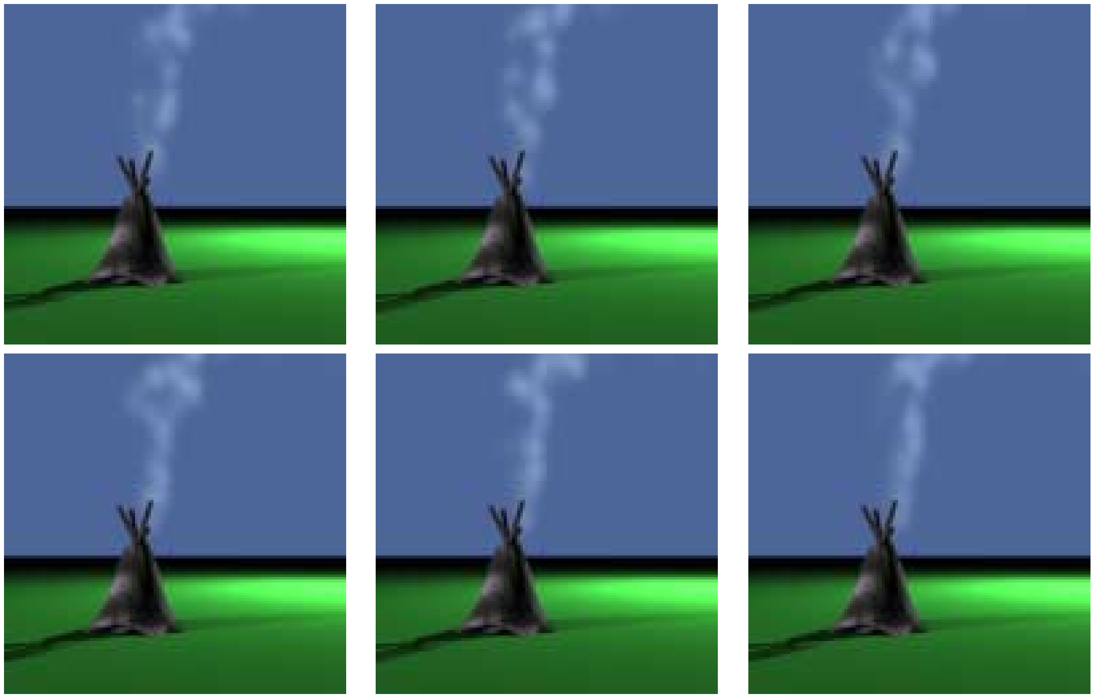

Figure 2: Frames from an animation of smoke rising from a tepee. Particles maps without spacetime raytracing.

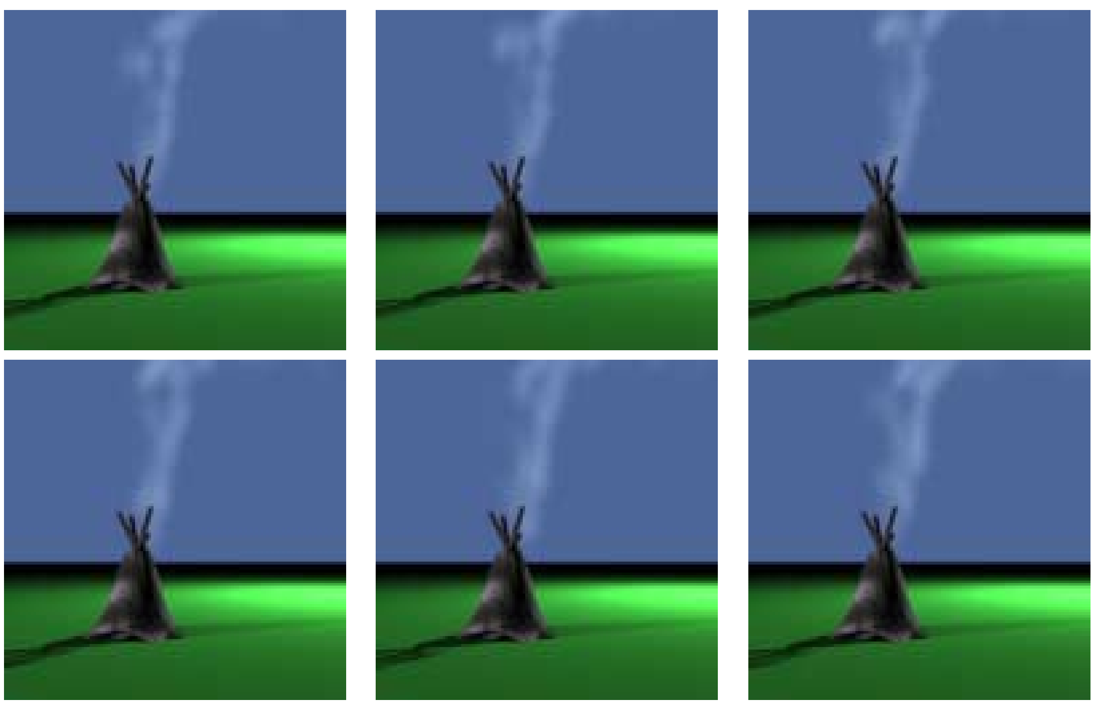

Figure 3: Frames from an animation of smoke rising from a tepee. Particles maps with spacetime raytracing.