Relativistic ray-tracing: simulating the visual appearance of rapidly moving objects

Andrew Howard Sandy Dance Les Kitchen

24 July 1995

Abstract

Many of the effects that arise when objects are moving close to the speed of light are counter-intuitive. Not least of these is the effect of large velocities on the visual appearance of moving objects. While this effect has been discussed mathematically, sophisticated computer simulations have not previously been available. In this report we show how the visual appearance of rapidly moving objects can be simulated by incorporating a relativistic correction into a standard ray-tracing algorithm. The images generated by this algorithm show interesting effects that require subtle interpretation.

1 Introduction

Special relativity was introduced by Einstein in 1905 to resolve the apparent contradictions that existed at that time between Newtonian dynamics and electrodynamics. The basic problem was that a long series of experiments had shown that the speed of light is the same for all non-accelerated observers. This result is not consistent with Newtonian dynamics. In special relativity, the uniformity of the speed of light with respect to non-accelerated observers is taken to be axiomatic, and is in fact extended into a much broader statement, known as the *equivalence postulate*, that the laws of physics must be identical for all non-accelerated observers. Satisfying this postulate requires a revision of the Newtonian ideas of space, time and simultaneity. Primarily, observers with large relative velocities will measure different lengths for the same objects, will measure different times between events and may even observe events in different orders. Relativity also has implications for the visual appearance of rapidly moving objects. Perhaps the best known of these is the so-called 'red-shift' effect. Observers will see the light emitted by a moving objects as being frequency shifted towards either the read or blue end of the spectrum, depending on whether the object is moving towards or away from the observer. Another effect is the apparent distortion of objects.

The reason for this distortion is quite simple. An image of an object is formed by a set of photons emitted by the object which arrive at the camera at the same time. This implies that the photons where *not* emitted by all points on the object at the same time. In this report, we show how this distortion effect can be simulated by taking a standard ray-tracing algorithm and modifying it to allow for the relative motion of the camera and the objects in the scene. No attempt is made to simulate the red-shift effect, as this requires much more sophisticated colour models than are available in standard ray-tracing algorithms.

2 A brief review of special relativity

In this section the basic results of special relativity are presented in a very simplistic fashion. Detailed and rigorous introductions to special relativity can be found in many, many places; see [3, 2]. The results presented here are derived from Goldstein [3].

Consider two coordinate systems (also called *frames of reference*) which we shall call the *primed* and *unprimed* coordinate systems. The origins of both systems coincide at time zero, as measured by observers placed at the origin of each system. The unprimed system has a velocity v along the z axis of primed system. An observer travelling with the unprimed system records an *event* at time t and location (x, y, z), which we indicate by the four-vector (x, y, z, t). Special relativity tells us that an observer travelling with the primed system will record the event (x', y', z', t'), where the relationship between primed and unprimed coordinates is given by:

$$\begin{aligned} x' &= x \\ y' &= y \\ z' &= \frac{z + vt}{\sqrt{1-\beta^2}} \\ t' &= \frac{t + vz/c^2}{\sqrt{1-\beta^2}} \end{aligned} \tag{1}$$

Where $\beta = v/c$. This is the *Lorentz* transform. This result can be written much more compactly as

$$\begin{aligned} x' &= x \\ y' &= y \\ z' &= \gamma(z + \beta ct) \\ ct' &= \gamma(ct + \beta z) \end{aligned} \tag{2}$$

where

$$\gamma = \frac{1}{\sqrt{1-\beta^2}} \tag{3}$$

Note that when v is small ($\beta \ll 1$), the Lorentz transform reduces to the classical Galilean transform:

$$\begin{aligned} x' &= x \\ y' &= y \\ z' &= z + vt \\ t' &= t \end{aligned} \tag{4}$$

These equations can easily be generalised in such a way that the z axis is not singled out. Let the unprimed coordinate system have some arbitrary velocity $\mathbf{v}$ with respect to the primed coordinate system. In this case

$$\begin{aligned} \mathbf{x}' = \mathbf{x} + (\gamma - 1)\frac{(\boldsymbol{\beta}\cdot\mathbf{x})\boldsymbol{\beta}}{\beta^2} + \gamma\boldsymbol{\beta}ct \\ ct' = \gamma ct + \gamma(\boldsymbol{\beta}\cdot\mathbf{x}) \end{aligned} \tag{5}$$

where $\mathbf{x}$ indicates the spatial components (x, y, z) and $\boldsymbol{\beta} = \mathbf{v}/c$.

3 Relativistic raytracing

Ray-tracing is a simple method for forming images by mathematically simulating the passage of photons through a scene [4, 1]. Each 'ray' corresponds to one possible path that photons might take from a light source to the image plane of a camera. Standard ray-tracing algorithms assume that objects in the scene do not have high velocities with respect to the camera; however, they can be easily extended to simulate the effects of such velocities.

In practice, to reduce the amount of computation required to render a scene, ray-tracing algorithms project rays *backwards* through the scene. Consider the pinhole camera shown in Figure 1. The path of a photon which passes through the pinhole and is absorbed at the image plane at location (x_i, y_i) can be described by a ray. For a camera of focal length f pointing along the positive z axis, this ray is defined by two points:

$$\begin{aligned} r_0 &= (0, 0, 0) \\ r_1 &= (x_i, y_i, -f). \end{aligned} \tag{6}$$

The standard ray-tracing algorithm recursively calculates intersections of this ray with objects in the scene and uses the surface properties of these objects to determine the colour of the pixel at point (x_i, y_i). In effect, it simulates the passage of a photon from a light source, bouncing off one or more objects, to eventually be absorbed at the image plane.

We wish to extend the ray-tracing algorithm to a situation in which the camera is moving with respect to the scene, so we must extend the definition of a ray to allow for the dynamic nature of the problem. The path of a photon which passes through the pinhole at time t_0 and is absorbed at the image plane at location (x_i, y_i) at time t_1 can be described by a 4 dimensional ray. This ray is defined by two four-vectors:

$$\begin{aligned} r_0 &= (0, 0, 0, ct_0) \\ r_1 &= (x_i, y_i, -f, ct_1). \end{aligned} \tag{7}$$

Note that t_0, t_1 and (x_i, y_i) are not independent. From the geometry of the situation it is clear that

$$ct_1 = ct_0 + \sqrt{x_i^2 + y_i^2 + f^2}. \tag{8}$$

The above equations define the ray as seen by an observer travelling with the camera, ie an observer in the *camera frame*. If the camera is moving with respect to the scene, the ray (7) must be transformed into the *scene frame* using the Lorentz transform. Specifically, if the camera is moving with velocity v along the z axis of the scene frame, the ray is defined by the pair of four-vectors:

$$\begin{aligned} r'_0 &= (0, 0, \gamma\beta t_0, \gamma c t_0) \\ r'_1 &= (x_i, y_i, -\gamma f + \gamma\beta c t_1, \gamma c t_1 - \gamma\beta f). \end{aligned} \tag{9}$$

Since we are assuming that only the camera moves with respect to the scene, and that no objects within the scene are moving, the standard ray-tracing algorithm can be applied to the transformed ray (9).

One additional issue which must be considered is the definition of a single image. We can extend the pinhole camera by adding a conceptual 'shutter': a single image is formed by all photons passing through the shutter during the brief instant for which the shutter is open. Consider a shutter placed directly behind the pinhole. A single image is formed by those photons which pass through the pinhole at the same time. These photons will reach the image plane at different times, however, since photons striking near the edge of the image plane have farther to travel than photons striking the center of the image plane. If the shutter is opened at time t_s (as measured in the camera frame), then an image can be rendered by considering the set of rays (defined in the scene frame):

$$\begin{aligned} r'_0 &= (0, 0, \gamma\beta t_s, \gamma c t_s) \\ r'_1 &= (x_i, y_i, -\gamma f + \gamma\beta c t_i, \gamma c t_i - \gamma\beta f). \end{aligned} \tag{10}$$

where

$$ct_i = ct_s + \sqrt{x_i^2 + y_i^2 + f^2}. \tag{11}$$

Other shutter models are possible; one alternative is to place the shutter directly in front of the image plane. The images in this report where generated using the pinhole shutter model.

For simplicity, the analysis presented in this section has assumed that camera motion is along the scene's z axis and that the camera points along the positive z axis. This analysis can easily be extended to an arbitrary motion for an arbitrarily located and orientated camera if one uses the general form of the Lorentz transform (5). Also, one could analyse the situation in terms of a moving scene and a stationary camera, but the result (which is, of course, equivalent to the moving camera case) is much more difficult to implement in a standard ray-tracer.

4 Experiments

4.1 Implementation with POV-Ray

The sample images in this report were generated using a modified version of the freeware Persistence of Vision Ray Tracer (version 2.0) [5]. POV-Ray is particular well suited to this problem as it is available in source form and is capable of rendering superb images using a flexible scene description language. We have modified POV-Ray's rendering algorithm to make relativistic corrections to generated rays, and the scene description language has been extended to allow an arbitrary camera velocity to be specified. The general form for the relativistic correction has been implemented and for simplicity we set $t_s = 0$.

4.2 The view forwards

The series of images in Figure 4 represent the results of rendering a simple 'grid' against a sky-like background with different camera velocities. The camera is moving towards the grid. We describe this camera as *forward facing* since it points in the direction of motion. The first thing to note about these images is that the distance from camera pinhole to the center of the grid is the same in all images, as measured by an observer in the scene frame. We can see that at very high velocities the camera behaves as if it has a 'fish-eye' lens. This make sense if we remember that from the point of view of an observer in the scene frame the camera focal length will be reduced due to relativistic length contraction. However, there is much more to this effect than simple shortening of the effective focal length. Consider the series of images in Figure 5. In one of these images ($\beta = 0.90$) the pinhole of the camera has actually passed through the center of the grid, as measured by observers in both frames. We can see that at this velocity the back side of the grid (which is not illuminated by any light source and hence is dark) is visible in the image; that is, we are seeing things which are *behind* the camera! One way of explaining this phenomenon is to imagine the camera travelling through the scene frame, 'sweeping up' photons as it goes. A photon which has left the back side of the grid and is travelling across the path of the camera may pass through the pinhole and be 'swept-up' by the image plane. The path of this photon corresponds to a ray which is pointing in a direction opposite to that of the camera. Consider a ray defined in the scene frame (10); for $t_s = 0$ this becomes:

$$\begin{aligned} r'_0 &= (0,0,0,0) \\ r'_1 &= (x_i, y_i, -\gamma f + \gamma\beta c t_i, \gamma c t_i - \gamma\beta f). \end{aligned} \tag{12}$$

where

$$ct_i = \sqrt{x_i^2 + y_i^2 + f^2}. \tag{13}$$

This ray will point backwards if and only if the z component of r'_1 is greater than the z component of r'_0, ie if

$$-f + \beta\sqrt{x_i^2 + y_i^2 + f^2} > 0, \tag{14}$$

or

$$x_i^2 + y_i^2 > f^2/\beta^2 - f^2. \tag{15}$$

Therefore, for any positive non-zero value of β there will be some finite value of $x_i^2 + y_i^2$ which corresponds to 'backward' pointing rays. The effect arises because of the finite speed of light.

Another way of looking at this phenomenon is to ask: in what part of the image plane will all objects which are in front of the camera appear? In this case we must consider all rays up to and including those which are tangential to the image plane. Rearranging (15), we can see that all objects in front of the camera will appear in a circle of radius s where

$$s = \frac{f}{\beta\gamma}. \tag{16}$$

Figure 6 illustrates this nicely. The scene in this case consists of an infinite plane placed at a very large distance in front of the camera. For any non-zero positive velocity, the projection of this plane onto the image plane must have a finite radius. For very large velocities ($\beta = 0.90$ for example), the radius of the projected plane is quite small. The rest of the image must be filled up with something, and in this case it is filled with the projection of objects which are *behind* the camera.

We can calculate an effective *field-of-view* for a moving camera. We define the field-of-view to be some angle between 0 and 2π, where 0 indicates that only those objects directly ahead of the camera will be projected onto the image plane, and 2π indicates that objects in every direction will be projected onto the image plane. Consider a ray projecting from a point on the image plane with a radial distance s from the center of the image plane. For $s > f/(\beta\gamma)$, the angle θ between this ray and a ray projecting from the center of the image plane (see Figure 1) is given by

$$\theta = \pi/2 + tan^{-1}\left(\frac{-\gamma f + \gamma\beta\sqrt{s^2 + f^2}}{s}\right). \tag{17}$$

The factor of $\pi/2$ indicates that these rays must be pointing backwards. All objects in the scene which have a bearing between $-\theta$ and $+\theta$ (relative to the direction of the camera) will be projected onto the image plane within a circle of radius s. Therefore, the field-of-view for the circle of radius s is 2θ. The total field-of-view Θ is found by taking the limit $s \rightarrow \infty$:

$$\Theta = \pi + 2tan^{-1}(\gamma\beta). \tag{18}$$

For $\beta = 0$ the field-of-view is exactly π. In the limit $\beta \rightarrow 1$, the field-of-view becomes 2π. That is, as velocity increases the field-of-view of a forward facing camera increases until, at $v = c$, all objects in the scene will be projected somewhere on the image plane. Figure 2 shows a plot of field-of-view versus β.

4.3 The view backwards

The series of images in Figure 7 depict the same simple grid object as Figure 4, except that this time the camera is moving away from the object. This camera is *backward facing* since it points opposite to the direction of motion. In all the images the camera is at the same distance from the object (as measured in the scene frame); only the velocity varies. Note that for high velocities the camera acts like it has a 'zoom' lens. This seems paradoxical, since a zoom lens would normally correspond to an increased focal length, and in this case the effective focal length of the camera is *shortened* due to Lorentz contraction. This paradox can be resolved if we remember that in the case of a forward facing camera, objects which are behind the camera will appear in the image, ie there is an increase in the effective field of view. By symmetry, there must be a corresponding *decrease* in the field of view of a backward facing camera. It is easily shown that the field of view in this case is given by

$$\Theta = \pi - 2tan^{-1}(\gamma\beta). \tag{19}$$

In the limit $\beta \rightarrow 1$, Θ becomes 0, ie all objects in the scene will be projected onto the image plane of a forward facing camera and nothing will be projected onto the image plane of a backwards facing camera. Figure 2 shows a plot of field-of-view versus β for a camera point backwards.

4.4 The view sideways

The series of images in Figure 8 depict a situation in which the motion of the camera is perpendicular to the direction in which the camera is pointing. The camera is moving from left to right across the grid object (or equivalently, the grid is moving from right to left across the field of view of the camera). A quick calculation shows that an object travelling at $\beta = 0.90$ will be shortened to 44 percent of its rest length in its direction of travel. However, the interesting thing to note in this series of images is that the object appears to be *rotated* rather than being contracted. This effect was first first noted by Terrell [6] and by Weisskopf [7], who mathematically analysed a number of simple cases.

4.5 Animations

Animations of the experiments described here are available electronically through the World Wide Web at "http://www.cs.mu.OZ.AU/ andrbh/raytrace.html".

5 Conclusion

One of the most striking conclusions that can be drawn from the preceeding discussions and examples is that, counter to what one might expect, the Lorentz contraction is *not* the dominant effect that one observes at large relativistic velocities. The Lorentz contraction definately occurs, but most of the distortion in the images is due to a combination of the finite speed of light and the geometry of the camera. Different camera geometries may well give rise to quite different distortion effects. The human eye, for example, is not a pinhole camera with a image plane. A camera model which more accurately follows the shape of the human eye (particularly the curved retina) might produce quite different results.

As a final observation, note that the whole concept of taking photographs of objects which are moving close to the speed of light is quite unrealistic. For example, in order to obtain an image of an object at a distance of 10 metres without significant blurring, one would require a camera with a shutter speed of the order of 1^{-10} seconds! Even if such a camera existed, the required film (or flash) is unlikely to be available in your local camera store.

References

[1] Foley, J. D., van Dam, A., Feiner, S. K., Hughes, J. F., and Phillips, R. L. *Introduction to Computer Graphics.* Addison-Wesley, New York, 1994.

[2] French, A. P. *Special Relativity.* Van Nostrand Reinhold, Wokingham, Berkshire, England, 1987.

[3] Goldstein, H. *Classical Mechanics.* Addison-Wesley, New York, 1980.

[4] Harrington, S. *Computer Graphics.* McGraw-Hill, New York, 1987.

[5] POV-Ray Team. *Persistence of Vision Ray Tracer, Version 2.0, User's Documentation.*

[6] Terrell, J. The invisibility of the Lorentz contraction. *Phys Rev*, 116 (1959), 1041.

[7] Weisskopf, V. F. The visual appearance of rapidly moving objects. *Physics Today*, 13 (1960), 24.

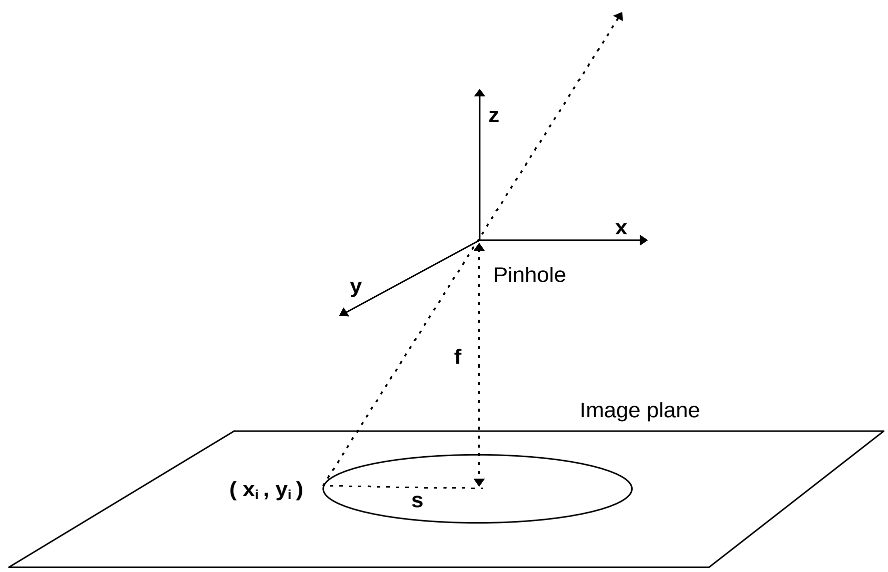


Figure 1: *Camera geometry*

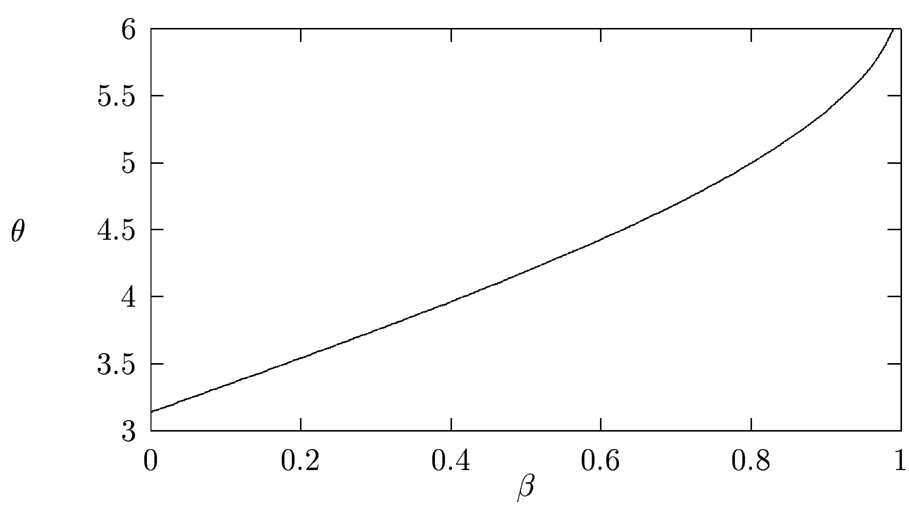


Figure 2: *Field of view versus velocity for a forward looking camera*

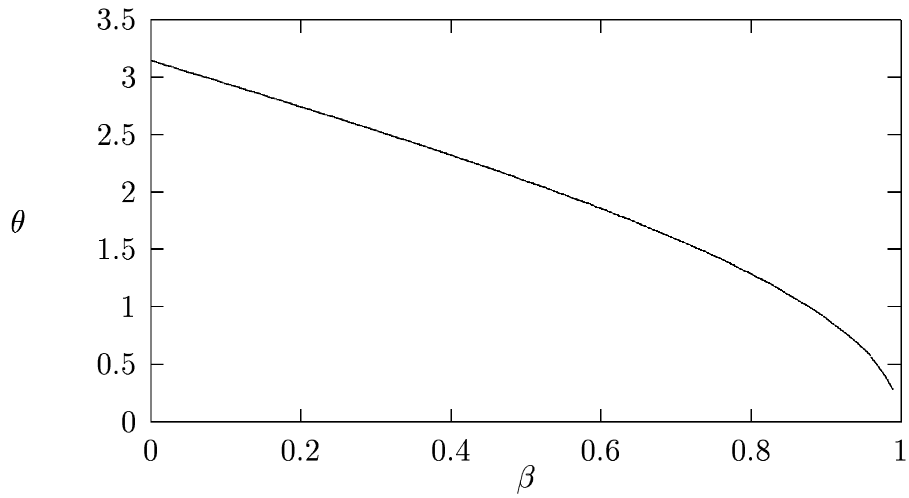
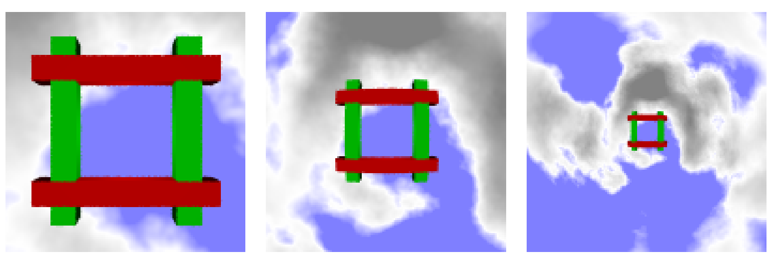
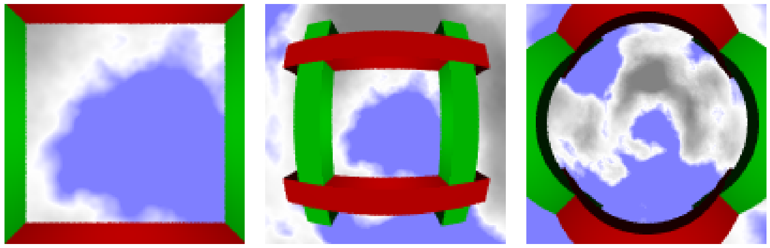


Figure 3: *Field of view versus velocity for a backward looking camera*



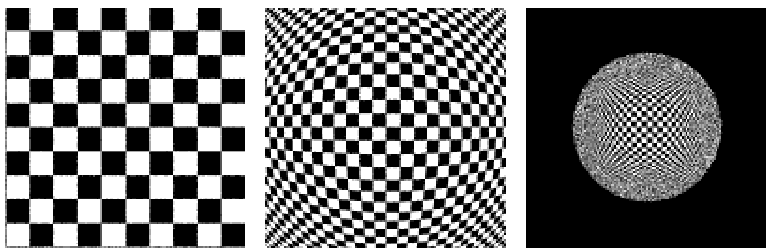
(a) (b) (c)

Figure 4: *Simple grid, looking forward. (a)* $\beta = 0.00, \Delta z' = -4$*; (b)* $\beta = 0.50, \Delta z' = -4$*; (c)* $\beta = 0.90, \Delta z' = -4$



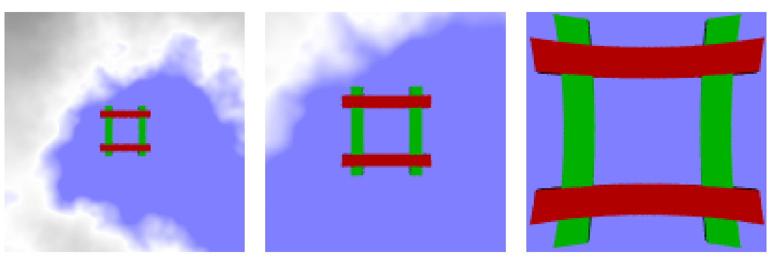
(a) (b) (c)

Figure 5: *Simple grid, looking forward. (a)* $\beta = 0.00, \Delta z' = -1.732$*; (b)* $\beta = 0.50, \Delta z' = -1.732$*; (c)* $\beta = 0.90, \Delta z' = +0.109$



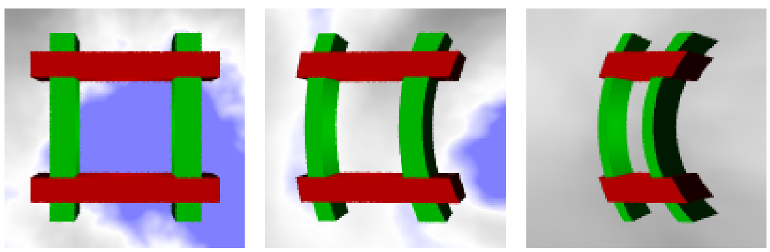
(a) (b) (c)

Figure 6: *Infinite plane. (a)* $\beta = 0.00$*; (b)* $\beta = 0.50$*; (c)* $\beta = 0.90$



(a) (b) (c)

Figure 7: *Simple grid, looking backward. (a)* $\beta = 0.00, \Delta z' = -14.4$*; (b)*$\beta = 0.50, \Delta z' = -14.4$*; (c)* $\beta = 0.90, \Delta z' = -14.4$



(a) (b) (c)

Figure 8: *Simple grid, looking sideways. (a)* $\beta = 0.00, \Delta x' = 0$*; (b)* $\beta = 0.50, \Delta x' = 1.73$*; (c)* $\beta = 0.90, \Delta x' = 9.17$