One Step Closer to a Fully Realistic 3D World on the Consumer PC
A White Paper on the 3dfx T-Buffer™

3D Effects, and Why They Matter

This white paper focuses on a new, proprietary 3D rendering technology and the problems it solves. The technology, from 3dfx, is the T-Buffer™. It's designed to bring a new level of realism to 3D graphics on the consumer PC.

The quest for real-time, full-motion photorealism in 3D graphics is the holy grail for the 3D industry. The ultimate goal is to eliminate all visual cues that distinguish synthesized — meaning computer-generated — scenes from photographs or video clips taken of physically real scenes.

Achieving this level of photorealism on a single photograph, or a single frame of a full-motion clip, is relatively straightforward, in that the required rendering algorithms are widely understood. Achieving it on a series of frames to produce a full motion video clip is also fairly straightforward, at least in situations that allow the luxury of taking all the time you need to produce each frame, then joining them together. Reaching this level of photorealism in real-time, generating each frame quickly enough to produce full motion, photorealistic scenes as needed in an interactive game or simulation, for example, is a challenge that has yet to be met on consumer platforms.

The gap between the level of photorealism possible in scenes rendered offline compared to scenes rendered in real-time is dramatic. The best example of this gap can be found at the beginning of almost every game that ships with a pre-rendered video introduction. These cinematic introductions typically run for several minutes. They are meant to introduce the game's starting scenario and draw the player into the game. These intros are painstakingly created frame-by-frame offline by artists, then assembled into full motion clips. However, the scenes in the actual game, created in real-time during game-play, rarely match the introductions for photorealistic effect. The difference can leave users disappointed, as the quality of photorealism falls noticeably once the game starts.

The T-Buffer allows several key digital effects for improving photorealism in real-time 3D graphics rendering. Until now, these effects have not been available on consumer-level PCs at real-time frame rates. Chief among these are full-scene spatial anti-aliasing (to remove the roughness or outright jaggedness from lines or polygon edges over an entire visible scene), motion blur (to add a realistic blur to moving objects that are otherwise defined more sharply than they would be if they were real objects actually photographed on chemical film) and depth of field (to add visual cues that help define the distance to each object in a scene by way of different levels of sharpness, or degrees of focus, at different depths).

Note that the first two rendering techniques — spatial anti-aliasing and motion blur — are non-issues in chemical photography. Chemical film records edges without adding any jaggedness; motion blur is the natural result of photographing moving objects at shutter speeds of $1/250^{th}$ of a second or slower. Adding these effects to a rendered scene is a way to make the scene appear much more realistic.

The third rendering technique — depth of field — is also a natural result of the way cameras work in the real world, but

it differs from the first two in that it's a standard consideration in photography and cinematography. In fact, directors regularly take advantage of this effect. It lets them draw the audience's attention to one part of the screen or another simply by changing the camera's focus. If near objects are in focus, objects further away from the camera become blurry. If far objects are in focus, the objects nearer to the camera become blurry. The audience's attention automatically goes to whatever is in focus.

We'll come back to these rendering techniques, and some others as well, in more depth later in this paper. For the moment, it's enough to know that a key benefit of each of these effects is to make the image on screen more realistic — meaning that it will better match what the viewer sees in real life, in photographs, and in motion pictures.

Precursors to the T-Buffer (A Short history)

Many of the features built into 3D controller chips for the consumer PC today are derived from rendering algorithms developed for high end workstations. As PCs became faster and more sophisticated, the once costly workstation solutions eventually made their way into the inexpensive PC systems we have today. To fully understand the problems the T-Buffer technology addresses, it is useful to take a brief look at some of the breakthrough algorithms developed first for these high end workstations.

In 1984, Robert Cook, Thomas Porter, and Loren Carpenter of the computer division of Lucasfilm Ltd. published a seminal paper, called *Distributed Ray Tracing,* which focused on many of the same problems the T-Buffer addresses. At the time the paper was written, an earlier technique known as ray tracing was able to render sharp shadows, reflections, and refracted light in computer-generated images by determining precise ray

directions. However, the original ray tracing algorithm was limited in its ability to create blurry effects or soft effects, such as the soft edges of most real-world shadows, or the depth of field effect, as shown in Figure 1.

One reason for the limitation was that ray tracing is subject to aliasing artifacts. The breakthrough in distributed ray tracing was that it combined the basic concept of ray tracing with a new anti-
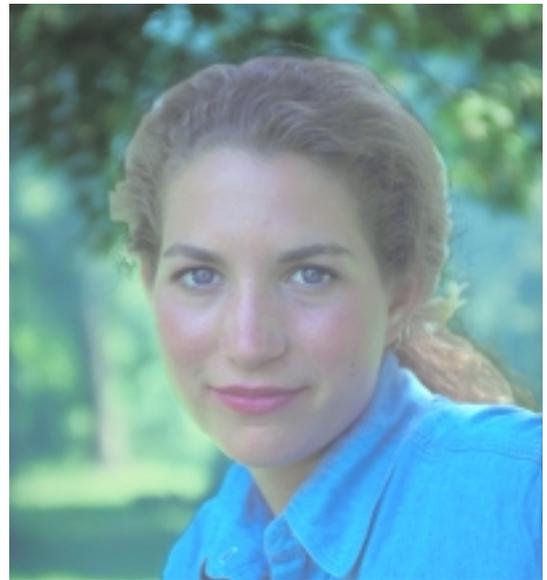
Figure 1: The depth of field effect shows objects at different distances at different degrees of focus. In this image, the woman in the foreground is in focus, while the trees in the background are out of focus, or blurry. This different focus depending on distance is a natural result of the way camera lens work. This effect is difficult to emulate using standard 3D rendering techniques. .

aliasing technique. The resulting technique offered a new approach to synthesizing images that made it far easier to create blurred effects, such as soft shadows, motion blur, and depth of field.

For all practical purposes, this was the first time that these, and related,

effects were available for computer-generated images. These newly available effects brought 3D computer graphics to a new level of realism, but also created new obstacles to overcome.

One of the problems that needed to be solved was how to render these new effects in real-time – at a high frame and fill rate. That problem was addressed in 1990 by Paul Haeberli and Kurt Akeley of Silicon Graphics Computer Systems, in their paper, *The Accumulation Buffer: Hardware Support for High-Quality Rendering.*

The Accumulation Buffer, which we discuss in more detail later in this white paper, is a system architecture that provides an additional buffer to integrate multiple renderings of a scene. Although it was originally conceived to be used for spatial anti-aliasing, it also provides a way to achieve such effects as soft shadows, motion blur, and depth of field. (See Figure 2 for an example of spatial aliasing.) Most important, the Accumulation Buffer can provide these effects in real time.

With these issues solved for high end workstations, the next natural step was to bring similar technologies to consumer PCs.

The Problems the T-Buffer Technology Addresses

Like many new 3D rendering technologies, the primary purpose of 3dfx's T-Buffer technology is to improve image quality. The ultimate goal of computer-generated 3D graphics is to create such realistic images that the computer monitor effectively becomes a window into a lifelike, fully interactive, 3D world. By definition, that lifelike world should not show any artifacts that serve as a distraction and break the illusion of reality. The T-Buffer lets PCs take another key step in that direction.

The challenge is exactly how to narrow this gap between computer-generated 3D graphics and what users typically see in real life, photography, and motion pictures. The T-Buffer attempts to narrow the gap considerably by offering real-time hardware acceleration of spatial anti-aliasing, motion blur, depth of field, and some other, closely-related effects.

It is important to realize that rendering with full-scene spatial anti-aliasing, motion blur, depth of field, and related techniques can make the difference between allowing the viewer to perceive the image as realistic, and jarring him or her
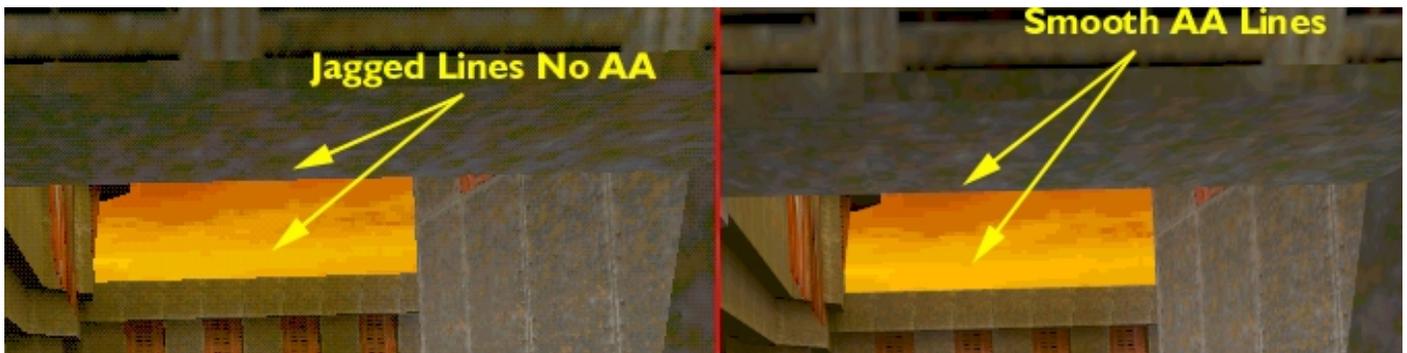

Figure 2: The left side of this figure shows aliasing in the top and bottom edges of the window view of a sunset in Quake2, with no anti-aliasing. The image on the right side is a rendering of the same scene with anti-aliasing applied. Screenshots from Quake2 © 1998 id software, Inc.
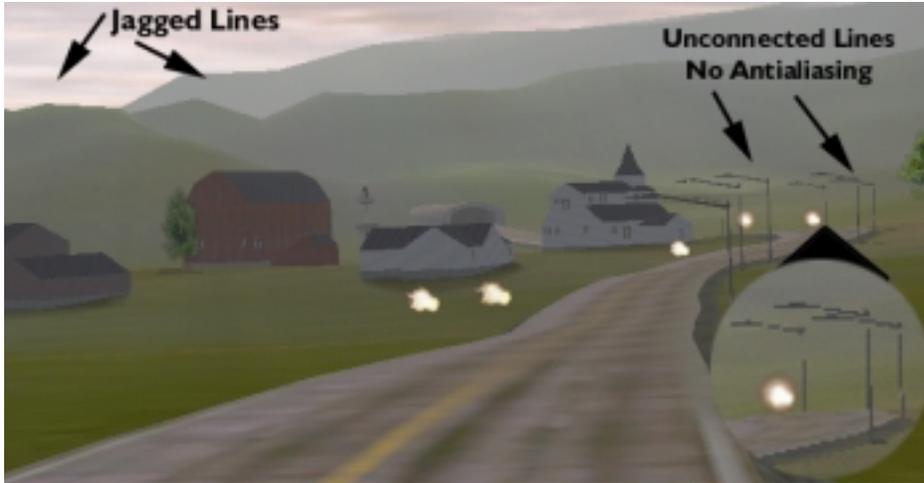
Figure 3 (top): The aliasing in the image to the left shows as both jagged, or stair-stepping, lines and as broken lines. The jagged lines show most clearly in the edges of the mountains on the upper left of the scene. The broken lines show in the light poles in the lower right.

Figure 3 (bottom): Both types of artifacts are dramatically improved with spatial anti-aliasing in the version of the same scene to the right. Screenshots from Barrage © 1998 Mango Grits, Inc.

out of a willing suspension of disbelief. That phrase is most often associated with the state of mind a fiction writer strives to maintain in readers or a director strives to maintain in movie viewers. However, it applies just as well to the state of mind a viewer should be in when viewing a 3D world on screen for interactive entertainment, game playing, or simulations. With this in mind, here's a look at the specific digital effects the T-Buffer addresses.

Full Scene Spatial Anti-Aliasing

The basic concept of spatial anti-aliasing applies to almost every kind of computer-based image. The undesirable artifacts on everything from jagged text characters in a word processor to the stair-stepping of near-horizontal lines in a 3D scene can be characterized as spatial aliasing artifacts. Spatial anti-aliasing is a technique for smoothing those jagged edges. However, there is more to spatial aliasing than just jagged edges, and more to spatial anti-aliasing than simply smoothing those edges.

With today's consumer level 3D graphic boards which only support primitive types of spatial anti-aliasing, you will not only still see jagged edges as a result of spatial aliasing, but you will see

scintillating, or popping, polygons. This problem shows when a small, thin object, like a light pole in the distance, repeatedly appears and disappears as the object moves relative to the screen, or when the object sometimes shows as a solid line and sometimes as a broken line. Both jagged (or stair-stepping) edges and popping polygons fall under the general category of spatial aliasing artifacts. Spatial anti-aliasing is a technique intended to reduce these visual anomalies. Figure 3 shows the dramatic visual difference between a spatially aliased image and a spatially anti-aliased image.

To better understand what causes aliasing, it's important to know a little about how computer generated 3D graphics are created. Rendering a digital image involves a process called *sampling*, and the technical definition of aliasing is *artifacts caused by under-sampling*.

Without getting into details, the need for sampling grows from the reality of memory restrictions, bandwidth restrictions, and the like. Because of these limitations, less image data can be stored than ideally should be stored. The solution is to sample the source image data at regular intervals and store the resulting, incomplete set of data. Because some of the source data gets discarded, this process can result in visual artifacts such as aliasing. To alleviate this problem, rendering techniques often try to increase the sample rate to store more of the original image data. Supersampling is a technique for increasing the sample rate.

It's useful in the context of discussing the T-Buffer to make the distinction between spatial anti-aliasing and anti-aliasing in general. As it happens, almost all the digital effects the T-Buffer makes possible are, in a strict technical sense, variations on general anti-aliasing. When discussing spatial anti-aliasing, we're talking about removing the jagged polygon edges and minimizing polygon popping. The T-Buffer, however, can also perform temporal anti-aliasing (supersampling in time to add motion blur) and focal anti-aliasing (supersampling in focal length to add depth of field).

It's also important to distinguish between *full-scene* spatial anti-aliasing and lesser forms of spatial anti-aliasing that deal only with edges (including lines). Only full-scene spatial anti-aliasing eliminates all spatial aliasing artifacts, including both jagged edges and popping polygons. Some 3D boards available today claim to support spatial anti-aliasing, but there are numerous limitations to the implementations: either the scene needs to be sorted (essentially arranging 3D slices by depth to remove hidden surfaces) or the edges need to be flagged and re-rendered. Either approach puts a large demand on the system's CPU. As a result, software developers have chosen to maintain high frame rates rather than take advantage of currently available "non real-time" spatial anti-aliasing technologies. The T-Buffer is the first consumer technology to offer real-time, full-scene spatial anti-aliasing.

Motion Blur

With traditional computer generated images, a given frame showing an object in motion will render that object with crisp, clean edges in each frame. When viewing a full motion version of the scene based on these images, the result is an unrealistic strobing effect (much like watching someone move underneath a strobe light). This strobing effect is quite different from the continuous, fluid motion of moving objects in real life. Strobing artifacts can also be thought of as temporal aliasing. The

T-Buffer addresses this issue by adding support for real-time motion blur.

To better understand what causes temporal aliasing, it helps to understand how static images are put together in sequence to form a full motion version. As already mentioned, aliasing refers to those artifacts that result from undersampling, which is to say there is more data that needs to be stored than can be stored.

In temporal aliasing, the strobing effect is caused by an object's motion not being sampled frequently enough. So instead of fluid movement the eye sees jerky movement from one frame to the next. The most common way to reduce this strobing artifact is to increase the number of snapshots of an object's motion. The problem with this brute-force approach for real-time 3D rendering is that it dramatically raises the performance requirements to unreasonable levels for a consumer-priced PC.

The T-Buffer technology allows for a much more cost effective approach to this problem by averaging multiple, discrete motion data together. This adds a motion blur to moving objects in each frame within a scene. And instead of seeing one discrete movement at a time, the eye sees multiple samples of movement at once. The result is a substantially improved motion effect which removes the strobing problem that's so typical of computer-generated 3D

Figure 4: The motion blur effect gives a sense of movement even in a still image like this.Copyright ©1984, the Association for Computing Machinery. Permission text as appropriate to be added by 3dfx.

graphics. Figure 4 shows an example of a single frame with motion blur added.

As an example of how much more visually appealing motion blurred sequences are, consider Hollywood movies. Movie projectors update each frame at only 24 frames per second. However, if you tried to play a computer game that updated at 24 frames per second you would find it jerky and utterly unrealistic.

The difference stems from the way cameras take pictures, by holding a shutter open for some period of time, to let light fall on the film, while the moving object continues to move. This automatically adds motion blur as a natural result of the process. And the motion blur, recorded in each frame of the film, lets the audience experience the movie comfortably at 24 frames per second. In comparison, a non-motion-blurred computer-generated image has to update frames at 2 to 3 times that rate to avoid the strobing effect. The T-Buffer technology allows for extremely high frame rates combined with real-time motion blur for an unparalleled visual experience.

Depth of Field

Any amateur photographer whose camera includes a focus control and f-stop settings has at least some familiarity with depth of field issues. Put a camera close to a window pane covered with raindrops, focus it to show the raindrops well, and most of the world beyond the window pane will show as an unrecognizable blur. Change the focus to show the house across the street, and the raindrops will essentially disappear from view. (Figure 5 shows an example of the depth of field effect.)

The human eye works in much the same way, although people typically aren't aware of the blurry parts in their field of view. Move your eye close to that same window with raindrops, focus on the raindrops, and the world beyond the raindrops will be blurry. Look at the house across the street and the nearby window will turn blurry.

Movie and TV directors take advantage of this depth of field effect all the time. With an actor in the foreground, for example, the camera may focus on the actor first, then change focus to something happening in the background. The foreground becomes blurry as the action in the background comes into visual focus, and the audience's attention automatically moves to the part of the screen that's in focus. Similarly, when two characters are at different distances from the camera, the camera focus — and with it, the audience's attention — may shift from one character to

Figure 5: The version of this image on the left shows all the crayons in focus, from the closest to the furthest away. The image on the right adds a depth of field effect that keeps the crayons relatively close to the viewer in focus, while blurring the closest crayons, and those furthest away. This different focus at different depths gives a visual cue to three dimensionality. Permission text to be added

the other as the conversation goes back and forth. Alternatively, the camera may stay focused on one character while the other is talking, to focus the audience's attention on the first character's reaction.

In addition to being able to use the depth of field effect to focus a viewer's attention on some particular area of the screen, the effect gives important visual cues to the distance between two objects in a scene or between the viewer and each object.

Real-time computer generated images today, however, don't show any visual cues to distance, and they don't allow for using this effect to draw the viewer's attention to certain areas of the screen. They show both near and far objects in focus unless you add the appropriate blurriness though a depth of field effect. The T-Buffer technology brings the capability of offering real-time depth of field effects for the first time to the consumer PC.

Other effects (Soft Shadows and Reflectance Blur)

The T-Buffer allows several other effects as well. Two in particular are worth mention: soft shadows and reflectance blur.

Some games and 3D applications are now starting to take advantage of hardware features such as Stencil Buffers to render shadows. The result, however, looks synthetic and is disappointing to the user in that the shadows are drawn with unrealistically sharp edges. In real life, shadows normally have soft edges as in the shadow in Figure 6 (these soft edges are also called blurry penumbras). The T-Buffer can be used to generate much more realistic soft shadow effects in real-time on consumer PCs.

Reflectance blur (also called soft reflectance) is another natural visual phenomenon. In the real world, there are some semi-glossy surfaces, like polished wood or brushed stainless steel, that will reflect objects with different degrees of focus, depending on how close the object is to the surface. Hold a pencil perpendicular to the surface, for example, and the part of the pencil that's closest to the surface will reflect in sharp focus, but the reflection will become increasingly blurry for parts of the pencil that are further and further away from the surface, as shown in Figure 6. Using today's 3D accelerators, the entire reflection of the pencil will be in sharp focus. In contrast, the T-buffer can be used to simulate reflectance blur in real-time on consumer PCs.

What all these effects have in common

Full-scene spatial anti-aliasing, motion blur, depth of field effects, soft shadows, and reflectance blur all have one important thing in common: From a strict technical sense, they are all variations on anti-aliasing. As already mentioned, aliasing is technically defined as artifacts caused by under-sampling. More generally, aliasing artifacts are the visual anomalies that result from needing more data stored and displayed than actually can be stored and displayed.

The T-Buffer allows acceleration of general anti-aliasing by providing more snapshots (for lack of a better term) of the source image data to yield better overall visual quality.

For full-scene spatial anti-aliasing this means providing more samples in the 2D (XY) space to remove the roughness, or jaggedness, from polygon edges. For motion blur, it means providing more

samples over time to remove the strobing effect for fast-moving objects. And for depth of field effects it means providing more focal (or "camera focus") samples to allow 3D rendering to function much more like a real-world camera. Soft shadows and reflectance blur effects are also based on similar techniques using multiple samplings.

Another aspect these new 3D rendering features have in common is that — with the exception of full-scene spatial anti-aliasing — they are relatively subtle.

(Spatial anti-aliasing is far from subtle.) No matter how subtle they are, however, they can make the difference between offering a reasonably good computer-generated image, and one that approaches being indistinguishable from a photograph. Leave them out, and a viewer with an untrained eye will know that the scene looks less than fully realistic, even though he or she may not be able to spot what's wrong with the image. The T-Buffer bridges this realism gap to let consumer
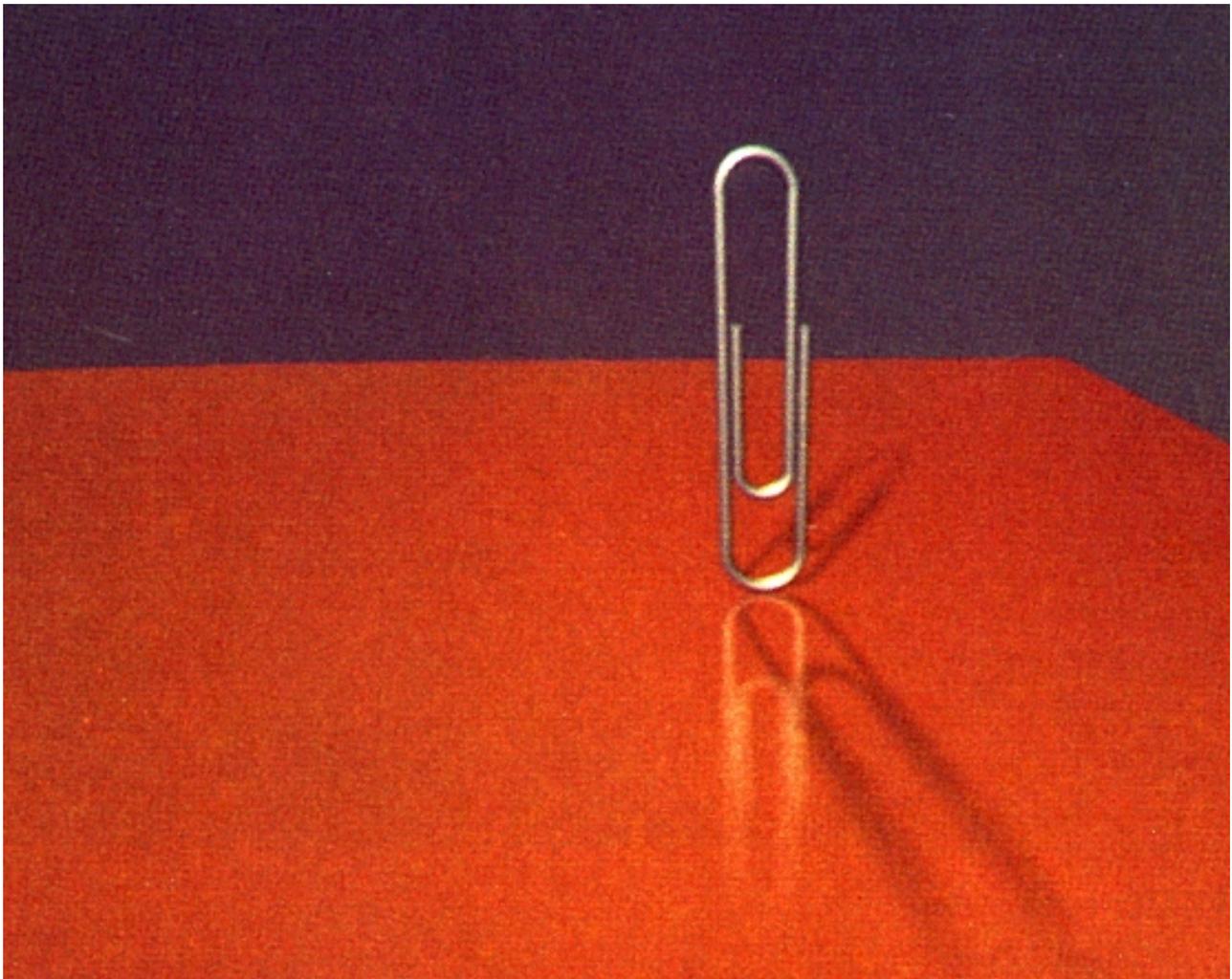
Figure 6: Most shadows in the real world have soft edges as in the shadow of the paper clip shown here. Many semi-glossy surfaces show the kind of reflection shown here, with the reflection showing as increasingly blurry for those parts of the object farther and farther away from the reflecting surface. Copyright ©1984, the Association for Computing Machinery. Permissions to be added by 3dfx.

PCs achieve far more photorealistic images at real-time rendering rates.

How The T-Buffer Works

To best understand the advantages of the T-Buffer technology it's important to understand how it works and how it differs from the Accumulation Buffer that preceded it.

Although there are some variations on Accumulation Buffer implementation, the rendering technique normally uses the following steps:

1. Clear the Accumulation Buffer.
2. Clear the back buffer and render the full scene in the back buffer.
3. Transfer the rendered scene to the Accumulation Buffer, with weighting (this is known as the accumulate step).
4. Repeat steps 2 and 3 as many times as needed. Each repetition combines a new version of the image with the image already in the Accumulation Buffer, so the final image is created as a weighted average result of different images. It's the combination of the multiple images that adds the effects.
5. Transfer, with scaling, the Accumulation Buffer to the back buffer, and flip (or swap) buffer contents. The result becomes the displayed image.

The Accumulation Buffer offers an elegant, but expensive, solution to one of the obstacles to rendering realistic images. However, it's somewhat limited in consumer applications because it's so expensive. The T-Buffer offers substantially the same capabilities, and in some cases better performance, at consumer price points.

Although the T-Buffer does essentially the same thing as the Accumulation Buffer in that it combines multiple different images to create digital effects, it reaches the final result differently. The steps for creating these effects using the T-Buffer are quite simple:

1. Clear the back buffer(or buffers)
2. Render multiple images into the back buffer(or buffers)
3. Swap the front and back buffer for display

For both the Accumulation Buffer and the T-Buffer, the key steps that add the multiple renderings need to be repeated as many times as necessary. It happens that repeating these steps quickly reaches the point of diminishing returns, with the first few repetitions doing most of the work to increase the realism of the final result, and additional repetitions adding less and less improvement.

As a rule of thumb, for the best possible image quality, you need to integrate 16 to 30 renderings of slightly different images. With current technology, however, it's extremely difficult to reach this many repetitions at real-time frame rates. Fortunately, between 4 and 8 repetitions typically yield a high quality image for adding these effects in real time, and both the Accumulation Buffer and the T-Buffer support this mode of operation.

One of the distinguishing characteristics of the T-Buffer is that it does not require an accumulate step nor a copy from an Accumulation Buffer to the back buffer. Rather, the hardware automatically combines the various T-Buffer contents

when a particular image is ready to be displayed on the computer monitor.

A second distinguishing feature of the T-Buffer as compared to the Accumulation buffer is that it lets you apply particular effects to only a selected portion of the image. Consider, for example, a still background with a moving car that needs motion blur added. For any given number of renderings, with the Accumulation Buffer you would need to render the entire image each time. With the T-Buffer you can render the still background once, then repeat the rendering on the car only. Since fewer triangles and pixels are involved in the additional renderings, the T-Buffer offers an automatic performance boost compared to the Accumulation Buffer.

Conclusion

The T-Buffer opens the door to digital effects that have never before been possible in real-time on consumer PCs. The most important of these is the addition of full-scene spatial anti-aliasing at real-time frame rates. Of all the remaining problems to be solved in 3D rendering on consumer PCs, this is the most important. It will, by itself, do more than any other step forward to help preserve the willing suspension of disbelief that is central to the enjoyment of interactive entertainment.

The T-Buffer also opens the door to more effects that we have covered here, because the T-Buffer is a tool, rather than a technique. 3dfx fully expects that software developers will take advantage of this new technology to add effects beyond the ones it was originally developed to accelerate. In fact, we have already discovered another interesting effect that's best described as "double vision" and is readily accelerated by the T-Buffer technology. With such flexibility of the T-Buffer technology, combined with the creative minds of developers, imagery and special effects never seen before in real-time on consumer PCs are only a short time away.

Prepared for 3dfx by MaceLabs Corporation