PHOTOGRAPHIC TEXTURE AND CIG: MODELING STRATEGIES FOR PRODUCTION DATA BASES

L. Charles Clark
Thomas C. Brown

Evans and Sutherland Computer Corporation
Simulation Division
Salt Lake City, Utah

ABSTRACT

The addition of texture to Computer Image Generation (CIG) systems has increased the potential for realism and cuing effectiveness in visual data bases used for flight simulation. While the visual simulation industry has already embraced texture technology, most of its attention has been focused on synthetic or statistical patterns. The use of photographic texture has been demonstrated and shows great promise, but it has not yet been thoroughly exploited in the production environment. Although photographic texture can significantly enhance the realism of a data base, its indiscriminate use often introduces unrealistic visual anomalies into the scene. However, when it is applied correctly, photographic texture can improve the efficacy of current and future CIG systems. The enhanced realism in flight simulation which accrues from the proper use of photographic texture provides a critical advantage in training effectiveness.

This paper discusses the scope of usefulness for photographic texture in production data bases, particularly for constructing self-repeating texture patterns. The results of new modeling strategies which mitigate or eliminate some of the visual anomalies inherent in the use of photographic texture are also described. Finally, examples are given of how photographic texture can be exploited to meet some specific training requirements for current and future flight simulators.

INTRODUCTION

The concept of texture, as discussed in this paper, refers to a way of modifying the surface characteristics of a polygon based on information stored in a two-dimensional array of data called a *texture map*. Each texture element in the two-dimensional array is called a *texel*. Generally, more than one texture map can be used to modify a polygon. In the Evans and Sutherland CT6 image generator, a set of maps applied to a single polygon is called a *mapset*, and may include up to four maps. By varying the scales and positions of the maps in a mapset, a great variety of patterns can be achieved with only a few maps.

In CT6 systems, texture is available in two forms: *modulation texture* and *contour texture*. Modulation texture varies a polygon's attributes almost continuously from one state or region to another, while contour texture is used to define a distinct edge between one state and another (like a cookie cutter). A combination of these two types of texture on a single polygon can create a very realistic image. An example is a tree that has subtly varying foliage colors while maintaining crisp branch and leaf shapes regardless of the proximity of the eye to the tree.

A polygon has three basic surface characteristics that can be modified with texture: intensity, color, and transparency. The standard application of texture is to vary intensity. In this case, a texture map varies the brightness across a polygon, texel by texel, based on the corresponding intensities stored in the map. Color can also be varied smoothly across the surface of a polygon, blending from one color to another. Finally, texture can be used to vary the transparency of a polygon. Both contour and modulation texture may vary each of these three characteristics, and both types may be combined in a mapset to provide a wide variety of visual results. (7) In this paper we will focus on the construction and use of texture maps for terrain color modulation, i.e., texture which modulates between colors on terrain surfaces.

Approaches to Making Texture Maps

Photodigitizing is one of two general approaches to making texture maps. The other approach, *synthetic generation*, is well understood in the visual simulation industry, and synthetically generated texture maps have been applied successfully in production data bases. Some approaches to generating synthetic texture include hand digitizing, procedural methods for creating simple patterns, statistical methods for generating patterns which imitate nature in the frequency domain, and fractals for imitating nature in the spatial domain (4).

Photodigitizing, on the other hand, is not as well understood nor has it been used as extensively, except perhaps to produce marketing material. The extent of the difficulties inherent with the use of photo-derived texture depends on what class of texture pattern is being created.

Classes of Texture Patterns

There are two general classes of texture patterns: *discrete* and *self-repeating*. Discrete patterns are used for modeling single occurrences of distinct objects (e.g., trees, people, aircraft insignia, etc.). Such patterns are typically easy to construct with either synthetic or photodigitizing procedures, and applying them in a visual data base is very straightforward. Although photodigitized discrete patterns are simple to construct and apply, they are not without problems. However, these problems, (which include obtaining orthographic source photos, and editing their background, color, and content registration) are easily managed in a "piece work" fashion.

Since the on-line memory for storing texture maps in a CIG system is always limited (5), it is virtually impossible to texture an entire production-size data base using discrete texture patterns. Some textured surfaces cover very large areas -- for example, terrain, clouds, and water. If such areas were textured with discrete patterns, an exorbitant amount of on-line memory would be required to store them. As a result, large surface areas are usually textured with self-repeating patterns. However, these self-repeating patterns are almost always synthetically generated, not photodigitized. The photo-derived texture often demonstrated in marketing material usually consists of local splashes of discrete patterns, possibly iterated over and over throughout a data base. We have found that this use of texture falls short of the real potential for using photo-derived texture in production data bases.

Unfortunately, photo-derived texture patterns which self-repeat are much more difficult to construct than are discrete patterns. In a self-repeating texture map, the edges must be blended to match left to right and top to bottom. This makes the boundaries between replications of the pattern inconspicuous. In addition, the contents of the map must be kept as homogeneous as possible to prevent the replication of some dominant feature in the texture map from emphasizing the inherent periodicity of the self-repeating pattern. (6)

The Challenge: Creating Photo-Derived Self-Repeating Patterns

Self-repeating texture can be readily constructed using the synthetic methods mentioned previously which allow the modeler to control the blending of edges and the homogeneity of content as the map is generated. However, in a photograph, the pre-existing edge and content conditions are usually far from optimal for natural self-repetition. One seemingly obvious solution to this problem would be to "set up" the source photos so that they match. However, this process would be expensive and complex.

Another possible approach would be to alter the content of the photos in the darkroom or with an airbrush so that the edges match. Techniques for manually altering photos have been around for as long as photography, and have been improved by modern electronic technology. However, manual editing would be tedious, time consuming, and extremely expensive, especially for a large number of photos.

The most rational approach to generating self-repeating patterns from photos is undoubtedly to use some digital method to blend the opposite edges of an image so they match in a natural looking way. The evolution of possible approaches to this edge-blending problem is discussed next.

FINDING SOLUTIONS: AN OVERVIEW OF EXISTING METHODS

Although the addition of texture to CIG systems has just recently brought the edge-blending problem into focus for the simulation industry, much thoughtful progress toward its solution has already been achieved in the field of image processing. For example, photomosaics have been constructed with multiple overlapped images from interplanetary space probes, earth satellites, and telescope photography. (1) Existing techniques for constructing photomosaics are the basis for solving the edge-blending problem.

The Multiresolution Spline Approach

A technical problem common to the construction of all photomosaics is joining together two images so that the edge between them is not visible. This problem can be solved with an image spline. An image spline is any digital technique for gently distorting the edges of adjacent and usually overlapping images so that they may be joined together with a smooth seam. Probably the single most important contribution to image spline technology is the recognition that edge blending is best done as a separate process for each frequency band in an image. This approach, called the *multiresolution* spline, was conceived and developed by Burt and Adelson (1). In the multiresolution spline, each image is first decomposed into a set of band-pass frequency images called a Laplacian pyramid. Then the Laplacian pyramids for the images are spliced together into a new Laplacian pyramid where texel values are averaged along the splice boundary. Finally, the new Laplacian pyramid is recomposed, yielding a successfully splined image.

Burt and Adelson showed that the multiresolution spline can be used to successfully join the following types of images.

- *Similar overlapped images*, e.g., two Landsat images with identical content but different gray levels.
- *Dissimilar overlapped images*, e.g., pictures of an apple and an orange. Although these images are dissimilar in the frequency domain, they are spatially symmetrical and therefore lend themselves to being splined.
- *Similar non-overlapped images*, e.g., images composed of pixel-block subsets resulting from data compression. Although these pixel block subsets are non-overlapped, they do originate from a single image. This results in similar spatial frequencies and content along the shared boundaries of adjacent pixel blocks, which makes them easier to spline than a set of unrelated images.

Although the multiresolution spline successfully blended these categories of images, there is still one case unaddressed by Burt and Adelson -- the need to create a self-repeating texture pattern from a non-repeating photograph. This problem falls into the category of *dissimilar non-overlapped images*. We tried using the multiresolution spline for this case, and we found that a single image could not be splined with itself using this method without noticeable discontinuities appearing between repetitions.

Yang et al.(6) also concluded that the use of the multiresolution spline to blend the opposite edges of a single image did not achieve acceptable results. They found that in the absence of image overlap, the extrapolation that occurs in the process of *decomposing and recomposing the Laplacian* pyramid placed too much weight on edge and corner texels and the edges did not blend inconspicuously. When the original image was overlapped with itself by 50%, blending along edges was somewhat improved. However, this approach resulted in changes to the general image character. In addition, it only worked for carefully selected images and was not considered to be a general solution.

The Multiresolution Image Pyramid

Another approach to solving the edge-blending problem was suggested by Zimmerman (7). He found that the multiresolution spline could be used to create a self-repeating pattern from a non-repeating image if an edge-blending filter was integrated into the process. In his implementation of the multiresolution spline, called the Multiresolution Image Pyramid (or MRIP), Zimmerman applied an edge-blending filter to the perimeter texels in each frequency band before the image was recombined.

Although we observed that MRIP was more successful at splining dissimilar non-overlapped images than any previous approach, we found that the edge filter dampened the image intensity range along the boundaries. This made the image appear blurry and gray along its edges. The problem of splining dissimilar non-overlapped images still remained.

A SOLUTION: MRIP WITH SOME NEW TWISTS

To generate self-repeating patterns from non-repeating photographs, we borrowed the basic ideas of the MRIP process, and added some new twists. The two most difficult technical challenges in our implementation included: 1) the design of a successful edge-blending filter and 2) the correct integration of the filter algorithm into the MRIP process.

After experimenting with a variety of edge-blending filters, we developed an intelligent filter which yields acceptable results. When this filter is applied along the edges of the band-pass image, a self-repeating pattern is generated from a non-repeating image by blending the top and bottom, and left and right, sides. This filter gently distorts opposite edges toward similarity while minimizing blurring or other changes to the fundamental character of the image. We were able to optimize the success of this filter by adjusting critical functions in the MRIP algorithm.

The new edge-blending filter, combined with the enhanced implementation of the MRIP algorithm, provides exciting results. Figures 1, 2, and 3 compare examples of non-repeating images of desert terrain scanned directly from an aerial photo to examples of self-repeating texture patterns generated from these images using our implementation of MRIP. These photographs illustrate how the intelligent edge-blending filter gently coerces a non-repeating pattern to be self-repeating while preserving the original image character along the newly filtered edges. The figures also show how the filter allows important features which are inevitably truncated at an image edge to grow or extend naturally into the opposite edge region.

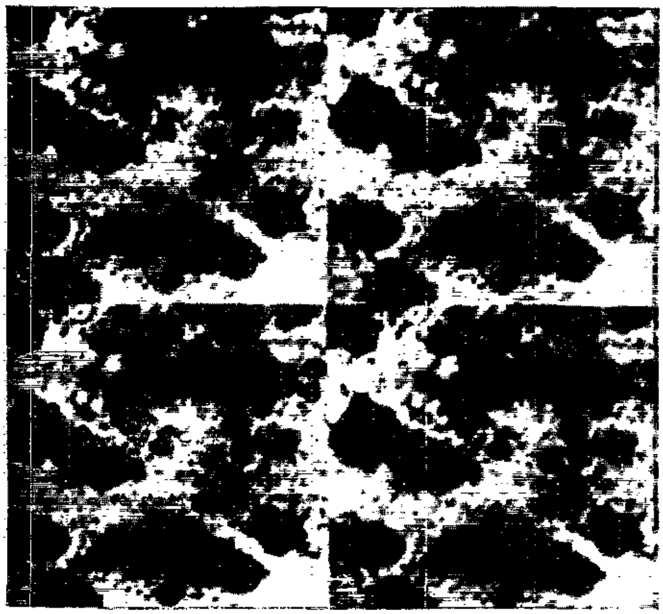

Figure 1a

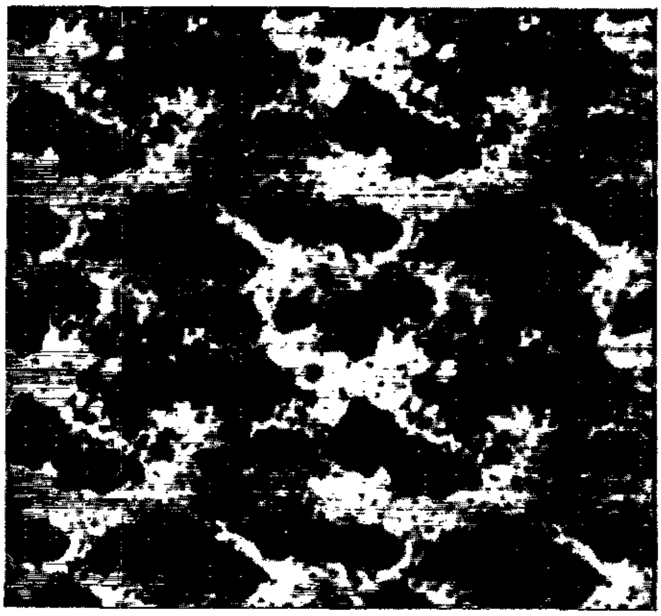

Figure 1b

Figures 1a and 1b show an aerial view of a small area covered with piñon/juniper trees. Dark objects in the photo are the tops of the trees. Figure 1a contains four copies of the photo and shows the problem of edge truncation in photo texture. Figure 1b is the same photo after the MRIP procedure has blended the edges to self-repeat. Note the truncated edge objects in Figure 1a and the subtle changes imposed in Figure 1b to facilitate edge blending.

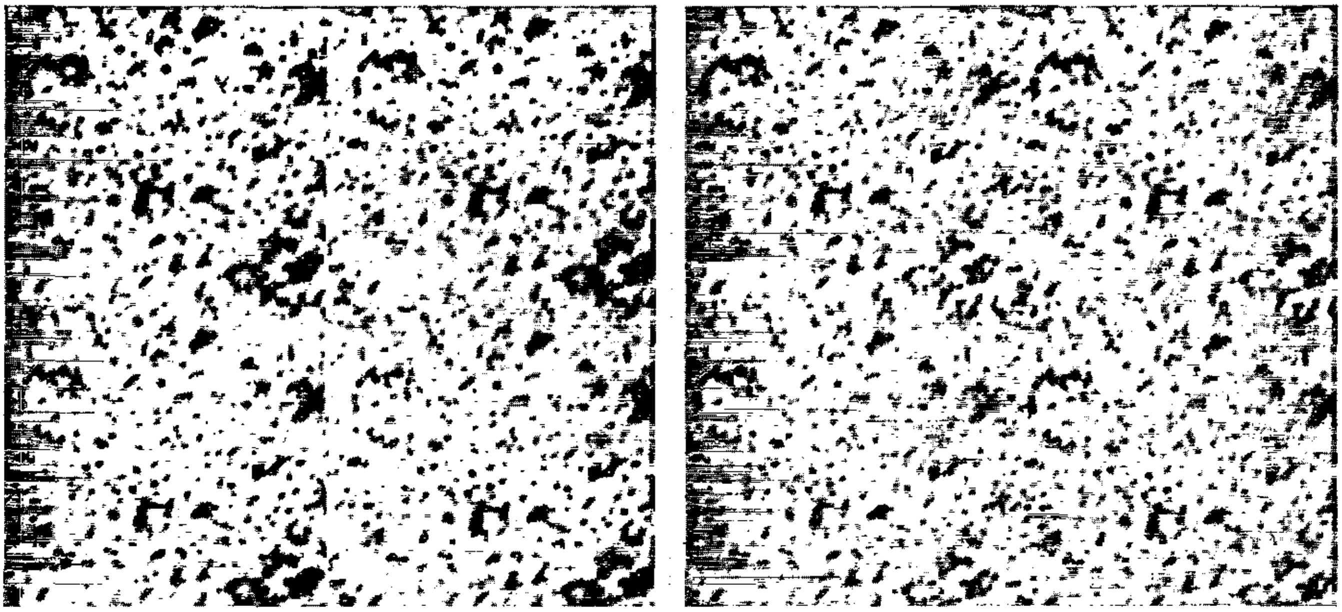

Figure 2 a

Figure 2b

Figures 2a and 2b show an aerial view of desert sagebrush. The sagebrush has a higher spatial frequency content than does the piñon map. Notice that in Figure 2a the higher frequency results in an original photo with nearly matching edges. Figure 2b is the sagebrush map, with fully matching edges, after the MRIP process is applied.

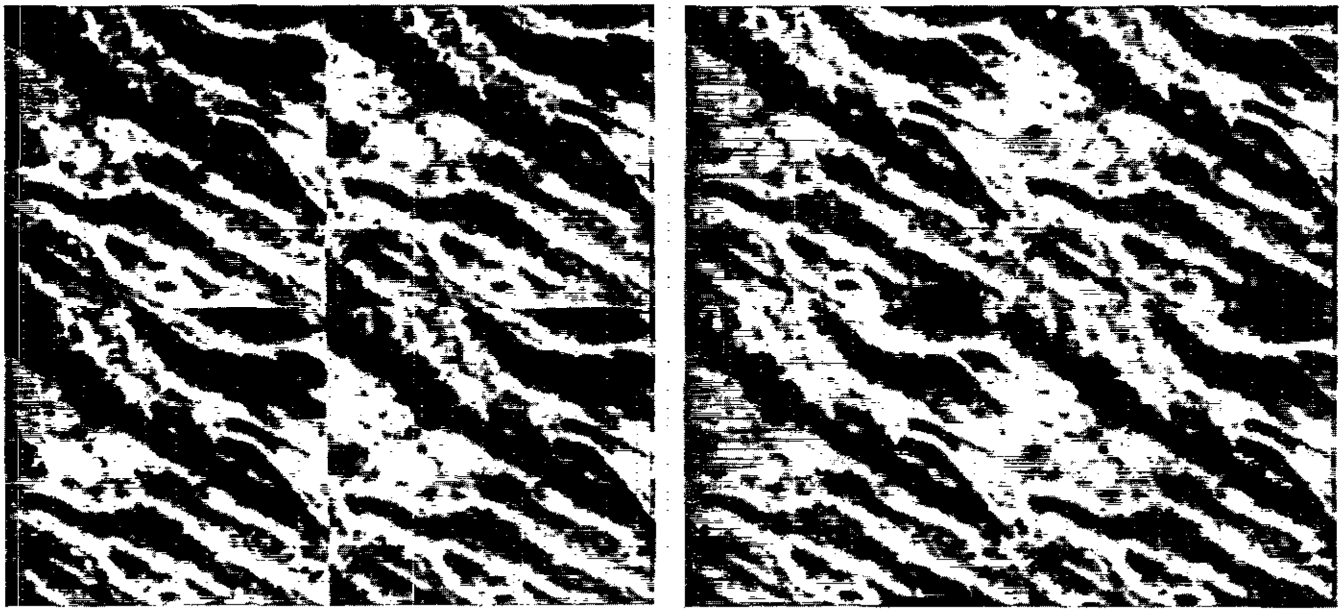

Figure 3a

Figure 3b

Figures 3a and 3b are taken from an aerial view of desert relief. This view covers a much larger area than the view used for Figures 1 or 2. This arroyo pattern is applied at the largest scale of the mapset to give the lowest level of desert detail a greater illusion of topographic relief.

In addition to desert, we have found the MRIP procedure to be very successful at blending many other kinds of modulation and contour patterns. For example, Figures 4 and 5 illustrate the MRIP procedure applied to create photo-derived texture patterns of ocean surfaces.

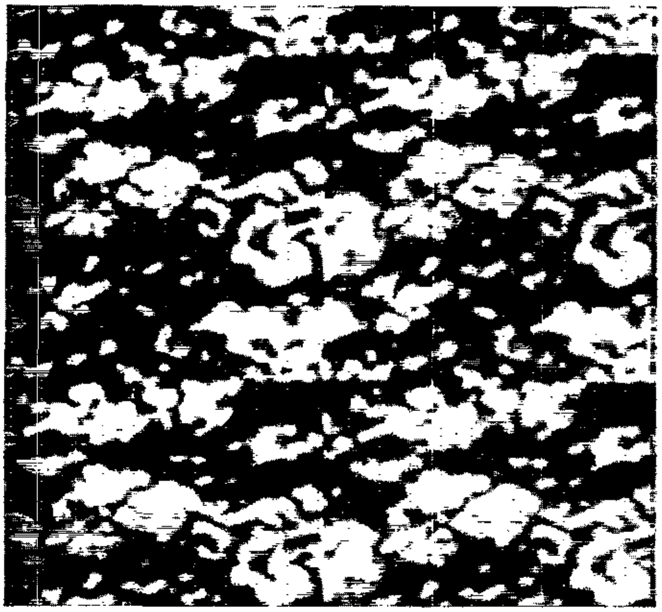

Figure 4a

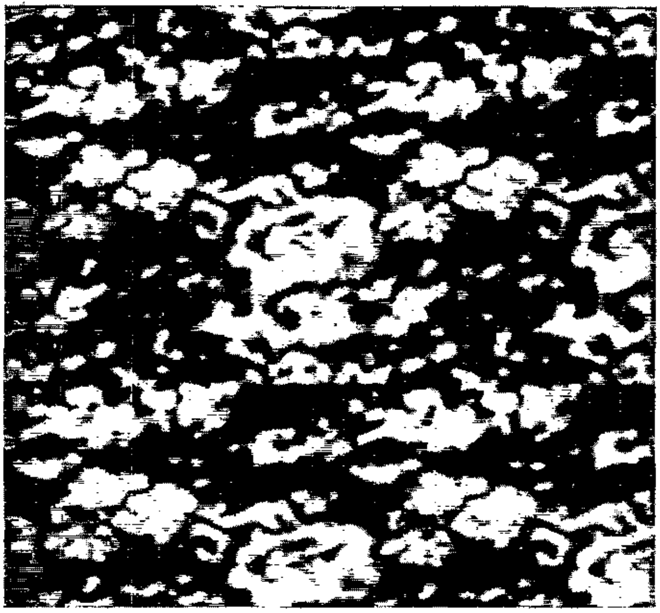

Figure 4b

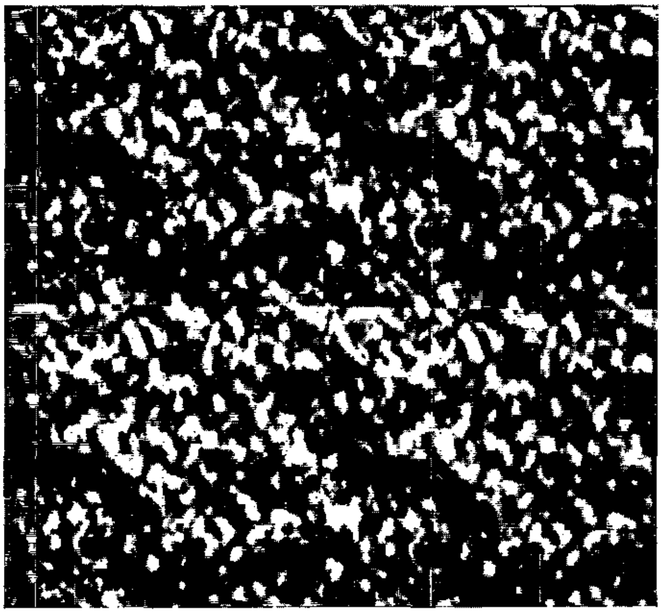

Figure 5a

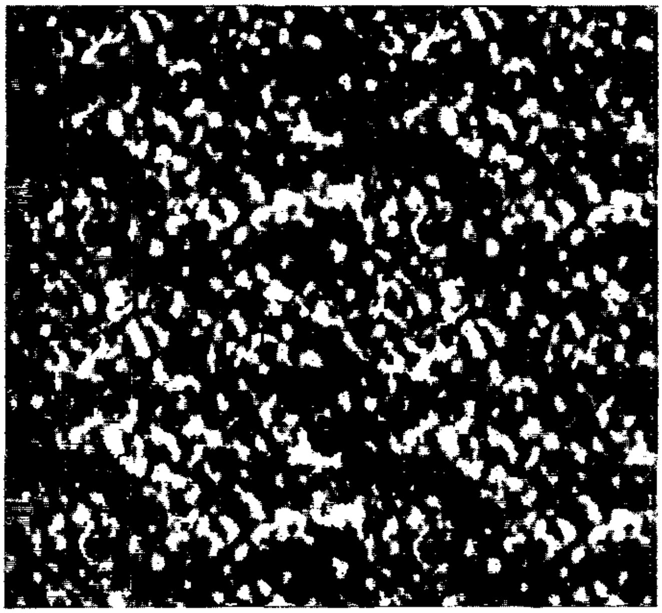

Figure 5b

Figures 4 and 5 are two photo textures of ocean taken from different altitudes, and consisting of slightly different sea states. Figure 4a is a photo of the ocean from a lower altitude and Figure 4b is the resulting texture map after the MRIP process. Figure 5a is a photo from a higher altitude with a calmer sea state and 5b is the resulting texture map.

IMPLEMENTATION STRATEGIES: OPTIMIZING VISUAL IMPACT

We have shown that the MRIP technique is very useful for creating self-repeating texture patterns from non-repeating photographic images. However, to insure that self-repeating texture maps yield optimal visual results, the following processes which relate to the construction and use of self-repeating patterns must be handled correctly.

- Acquiring the correct picture
- Selecting the correct image patch
- Determining texture levels of scale
- Applying self-repeating patterns in data bases

Acquiring the Correct Picture

The first problem in creating photo-derived texture maps from a photodigitized source is to acquire the "correct" (used as a relative term) photograph. Pictures for discrete patterns are easy to find, or just as easy to take. However, photographs which are suitable as image sources for self-repeating terrain texture patterns are not as trivial to acquire.

The most straightforward picture to use for generating terrain texture is usually taken from a high altitude with an orthographic view and precise control of scale. It is impractical for most of us to go take this kind of a picture. Fortunately, a large variety of scales and series of orthographic aerial photographs covering all 50 states is available from the Agricultural Stabilization and Conservation Service of the United States Department of Agriculture. These pictures contain images of topography, vegetation, water surfaces, urban areas, etc. They are invaluable image sources for generating terrain texture patterns.

Selecting The Correct Image Patch

Once the correct picture is acquired, the next step is to select the correct image patch. Rarely will an entire photograph be useful as a texture map even if it is the correct picture. More often, the best source for a texture map is some image patch within the picture.

For discrete patterns, the selection step is easy. The image patch can simply be selected, framed, and digitized. However, in the case of self-repeating patterns, image-patch selection is much more difficult and is critical to maximizing success. Selecting the correct image patch for use in terrain texture modeling requires the identification of patches which show the greatest potential for self-replication.

Two general approaches to selecting correct image patches include *statistical methods* and *inspection*. Statistical methods are particularly useful for identifying dominant features in an image patch which could potentially aggravate the periodicity problem. Such methods work on the assumption that an image patch which possesses strong potential for self-repetition exhibits the following characteristics: (6)

- The image patch has relatively constant correlational statistics over its entire surface.
- No subset of the patch can be easily and unambiguously segmented from the rest of the image.

These criteria generally limit image patches to isotropic content, i.e., similar frequency and spatial content in all directions. As a result, the texture patterns surviving these tests tend to be boring and bland. However, texture in the real world is often anisotropic, having form and content which result from and imply directionality. For example, arroyos in the desert, waves in the ocean, and cloud formations in the sky are evidence of anisotrophism in nature.

With anisotropic texture, the challenge in creating a self-repeating pattern is to align directional texture features at opposite map boundaries. This is best accomplished by inspection. In general, selecting the correct image patch is more a matter of art than science, and the best image patch selection tool is the human eye, along with a process of trial and error.

Determining Texture Level Of Scale.

A proven approach to masking the periodicity inherent in a self-repeating texture pattern is to combine two different scales of the pattern on a polygon. With this technique, each replication of the smaller pattern is subjected to a slightly different modulation of the large scale map than every other replication. (6) When four scales of the same pattern are combined together in a CT6 mapset, the resulting texture pattern begins to exhibit fractal-like behavior.

Fractals rely on the notion of self-similarity to produce models that mimic the natural world. The general concept of fractals is that most natural forms, such as a shoreline or the outline of a cloud, exhibit detail no matter how close the observer, and the form of that detail is similar regardless of scale. (7)

Although a fractal-like mapset (with four scales of the same texture pattern) on terrain polygons provides flight cues through a broad range of altitudes, it does not result in a truly realistic appearance. This is because fractals merely emulate a model of nature, rather than reproduce it, and there are too many natural phenomena to be defined in a set of fractal rules (4). For example, the macro-topography of an entire mountain range may be very similar in form to the micro-topography of a small foothill. However, in a textured terrain model, it is more important to show topographic texture patterns at large scales (or low frequencies), while it is more important to show vegetation patterns at small scales (or high frequencies).

Although fractals offer an interesting and useful approach to analyzing texture, we have found that the designation of texture map content for each level of scale in a mapset is best done through careful observation of nature, and common sense. In addition, the CT6 capability to combine four texture levels of scale in a single mapset yields extremely realistic texture effects.

Applying Self-Repeating Patterns in Data Bases

Given a mapset of self-repeating terrain texture patterns with appropriate content and level of scale, CT6 systems enable some very useful techniques for applying texture in a visual data base. These capabilities include the use of *macro texture*, and the unique ability of the CT6 to apply *smooth shading* to textured polygons.

Macro Texture. A common terrain modeling approach for CT6 data bases is the subdivision of *terrain into a regular grid. The elevation data for this* grid is typically derived from DMA terrain elevation data. (4) If texture is applied such that mapset repetitions begin and end at polygon boundaries, the repetition of the mapset and the faceting of the terrain model mutually accentuate each other. This undesirable interaction highlights rather than disguises the polygon boundaries in the terrain model. To remedy this problem, CT6 systems use macro texture.

Macro texture is the use of a mapset repetition interval which is larger than that of the polygonal subdivision of the terrain. In macro texture, the mapset is projected onto a multi-polygon terrain surface such that texture detail is continuous and unique across polygon boundaries. As a result, the largest scale pattern in the mapset covers many polygons. Up to three other texture maps at smaller scales may be overlaid onto this pattern to produce a broad range of texture scales. This produces a homogeneous texture pattern across the terrain which effectively masks texture map repetition and polygon boundaries.

Smooth Shading. The Evans and Sutherland CT6 is unique in its ability to render smooth-shaded textured polygons. When a mapset of photo-derived self-repeating texture patterns is applied as macro texture to a smooth-shaded terrain model, the illusion is complete. Polygon boundaries in the terrain model become virtually invisible. The result is a quantum leap in image realism, as shown in Figure 6.

OTHER APPLICATIONS FOR PHOTO-DERIVED TEXTURE

Most examples and explanations given in this paper have illustrated the use of photo-derived texture patterns for modeling desert terrain. Beyond the modeling of desert, an immediate potential exists for vastly increasing the realism of a variety of other terrain types modeled in visual data bases. The use of self-repeating photo-derived topography and vegetation patterns for models of forest, agricultural, and urban areas offers great promise.

In addition to terrain, other large surface areas in visual data bases which may be textured with self-repeating patterns include ocean and clouds. Figures 4 and 5 showed how the MRIP procedure could be successfully applied to ocean patterns. In addition to using MRIP, CT6 offers two capabilities that allow an ocean model to simulate sea states. (The term *sea states* refers to the frequency and magnitude of ocean waves.)

- Given a number of photo-derived texture maps representing a variety of sea states, the simulated sea state in a model can be changed by reloading texture maps.

- In addition, texture motion can be used to animate whatever texture maps are applied. (4)

Figure 7 shows the result of modeling an ocean surface using these capabilities with the texture patterns shown in Figures 4 and 5.

Great long-range potential exists for modeling terrain with photo-derived *discrete* patterns because of anticipated capabilities of future CIG systems. For example, CIG texture memory capacity will continue to increase. This capability, combined with dynamic updating of texture memory, will virtually eliminate texture memory limitations. This and other features will make it feasible to apply a mosaic of discrete map-correlating texture patterns to large terrain regions in a visual data base. The MRIP technology will play an important part in this capability, accomplishing the task of blending the edges of a large number of adjacent images.

Applications will be found for both contour and modulation maps in terrain image mosaics. Modulation maps will continue to be exploited for representing topography and vegetation patterns. Contour maps will prove invaluable for modeling sharp-edged features such as coastline. Both kinds of maps will be generated from a variety of sources, i.e., they may be photo-derived, or they may instead be generated from photo-like sources such as remotely sensed digital imagery (2) or DMA data.

CONCLUSION

The addition of texture capabilities to CIG systems has made possible the application of texture patterns to large terrain surface areas in visual data bases. However, the relatively limited texture map memory in today's CIG systems does not allow these *large terrain regions to be textured with discrete* (non-repeating) patterns, so self-repeating patterns must be used instead. Because the creation of self-repeating texture patterns from non-repeating photographic images has offered long-standing technical challenges, photo-derived texture patterns have not been as generally applied in visual data bases as have synthetically generated patterns.

A method for generating self-repeating texture patterns from non-repeating photographic images has finally been developed. This process, known as MRIP, is based on the concept that blending opposite image edges, as is required in generating a self-repeating pattern, is best done independently for each frequency band in an image. The key to the success of the MRIP approach is the use of an intelligent edge-blending filter which gently distorts the opposite edges of an image toward similarity while preserving the original image character along the newly filtered edges.

Given the capability to create self-repeating photo-derived texture maps, several implementation strategies insure their successful application in terrain data bases. The processes of selecting an

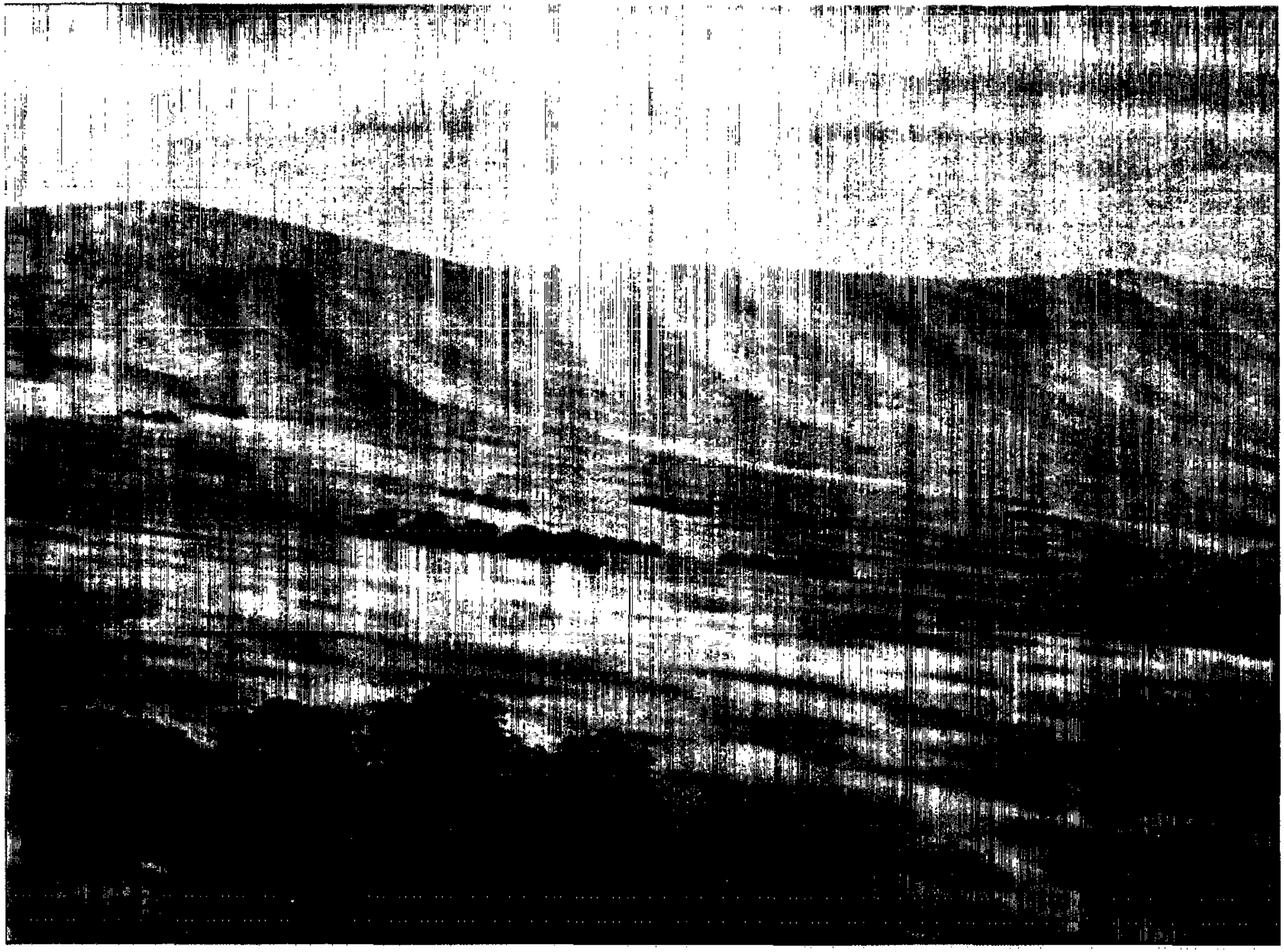

Figure 6

Figure 6 was photographed directly from the display of a CT6 real-time image generator. This view of the data base includes only 42 visible ground polygons, and is part of a completed production data base which contains over 100,000 polygons in its high-level-of-detail terrain model. Notice the piñon/juniper texture in the foreground and the arroyo map in the distant background. The 3D textured trees are added to allow the eye to calibrate to the correct scale of the piñon texture.

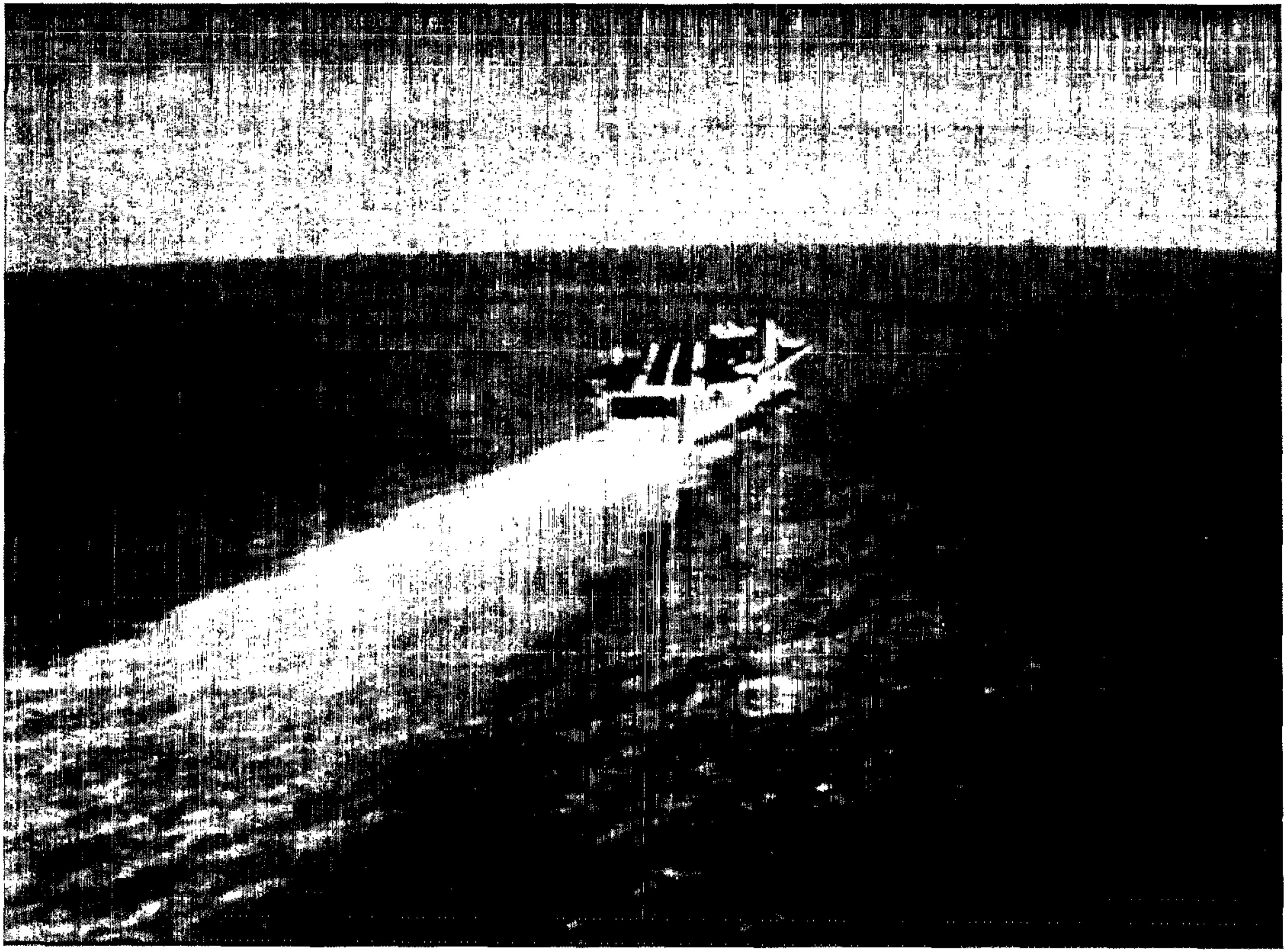

Figure 7

Figure 7 was also photographed directly from the display of a CT6 real-time image generator. The textured ocean as viewed in this photograph is derived from the application of the texture maps shown in Figures 4b and 5b onto a flat polygonal surface. The ocean can be textured in this fashion with a single polygon.

appropriate source photograph and an appropriate image patch subset of a source photo must be executed correctly. Texture mapsets must be composed of patterns with appropriate content and level of scale, and must be applied to the terrain in combination with macro texture and smooth shading to maximize the illusion of realism.

Photo-derived texture promises great potential for enhancing the realism of current and future data bases. Virtually all terrain surfaces in current data bases can now be more realistically represented. In addition, photo-derived texture can also be applied to ocean and cloud models. New capabilities on future CIG systems will make it possible to replace self-repeating patterns with a mosaic of map-correlated discrete texture maps. As the simulation industry becomes more and more fluent with the technology of photo-derived texture, the scene realism and resulting training effectiveness of visual data bases will grow by quantum leaps and bounds.

REFERENCES

1. Burt, P.J., and E.H. Adelson. "A Multiresolution Spline with Application to Image Mosaics." *ACM Transactions of Graphics*, Vol. 2, No. 4 (October, 1983), pp. 217-236.
2. Faintich, M. "A System for All Sources." *Computer Graphics World*, Vol. 10, No. 6 (June, 1987), pp. 91-92.
3. Mandelbrot, B.B. *Fractals: Form, Chance and Dimension.* San Francisco: Freeman Press, 1977.
4. Robinson, J. and S. Zimmerman. "Exploiting Texture in an Integrated Training Environment." *7th IITEC Proceedings*, (November 19-21, 1985), pp. 113-121.
5. Yan, J. "Texture Generation in Real-Time Computer-Generated Imagery." *SIGGRAPH '85 (Course No. 14: High Performance Image Generation Systems)*, San Francisco, (July 22-26, 1985).
6. Yang, S., S. Wood, and D. Wallner. "Generation of Texture Patterns for Realistic Visual Simulation." *8th I/ITSC Proceedings*, Vol. 1, (November 18-20, 1986), pp. 78-85.
7. Zimmerman, S.A. "Applying Frequency Domain Constructs to a Broad Spectrum of Visual Simulation Problems." *1987 Image Conference IV Proceedings*, (June 23-26, 1987).

ACKNOWLEDGMENT

Many individuals at Evans and Sutherland have encouraged the development of the concepts presented in this paper. In particular, the work of Stephen Zimmerman served as a springboard for our effort, and we appreciate his interest and support.

ABOUT THE AUTHORS

L. Charles Clark is the Supervisor of the CT Exploitation Group in the Simulation Division of Evans and Sutherland, and has been involved in the development of CT5A and CT6 data base technology for the past five years. He obtained a B.A. in Geography from the University of Utah in 1980.

Thomas C. Brown is a Software Engineer for Evans and Sutherland's Simulation Division where he is currently assigned to the CT Exploitation Group. He received a B.S. in Computer Science from the University of Utah with an emphasis in medical imagery. His work with medical imagery and computer assisted surgery has been published. Tom has devoted the last ten years to the simulation product line at Evans and Sutherland and has been a major contributor to the development of photo-based texture for the CT6 Data Base Engineering Group.