Graphical Video Representation for Scalability

Kumi Jinzenji Hisashi Kasahara

NTT Human Interface laboratories
1-2356 Take Yokosuka-shi Kanagawa 238-03 JAPAN
Telephone +81-468-59-3034
Facsimile +81-468-59-2829
E-mail kumi@nttvdt.hil.ntt.jp

Abstract

This paper proposes a new concept in video called Graphical Video. Graphical Video is a content-based and scalable video representation. A video consists of several elements such as moving images, still images, graphics, characters and charts. All of these elements can be represented graphically except moving images. It is desirable to transform these moving images graphical elements so that they can be treated in the same way as other graphical elements. To achieve this, we propose a new graphical representation of moving images using spatio-temporal clusters, which consist of texture and contours. The texture is described by three-dimensional fractal coefficients, while the contours are described by polygons. We propose a method that gives domain pool location and size as a means to describe cluster texture within or near a region of clusters. Results of an experiment on texture quality confirm that the method provides sufficiently high SNR as compared to that in the original three-dimensional fractal approximation.

1. Introduction

This paper describes a new method of video representation for scalability. Conventional video representation is a gathering of pixel-composed frames taken by a camera; each frame is divided into non-overlapped blocks; and in its compression scheme, information within the blocks is compressed using the uneven distribution of pixel intensity; and inter frames are compressed using motion compensation. Basically, the representation of conventional video entirely do not have to do with the semantics of the video contents themselves. This, then, is not perfectly scalable so that it is required for the spatial and temporal resolution to be almost always fixed.

Some new approaches to video representation have been proposed with a few new coding schemes proposed [1][2][3]. The aims in these new approaches are reuse and reference / indexing of video. In the newly proposed schemes, the real three-dimensional world is reconstructed utilizing motion information, but the quality depends on the level of motion detection accuracy. Moreover, it is impossible to exactly reproduce the three-dimensional world from two-dimensional raster image data. Therefore, the application of those schemes has been limited to either video which is taken with no camera motion or containing few moving objects.

In this paper, we suppose that video has been structured automatically or by interactive manipulation. In the existing methods mentioned above, objects are automatically extracted and represented at the same time, but in our method the aim is only to represent the structured objects as graphical elements. The advantage of considering extraction and representation separately is that we do not have to concern ourselves with the extraction quality. The final video comprises a set

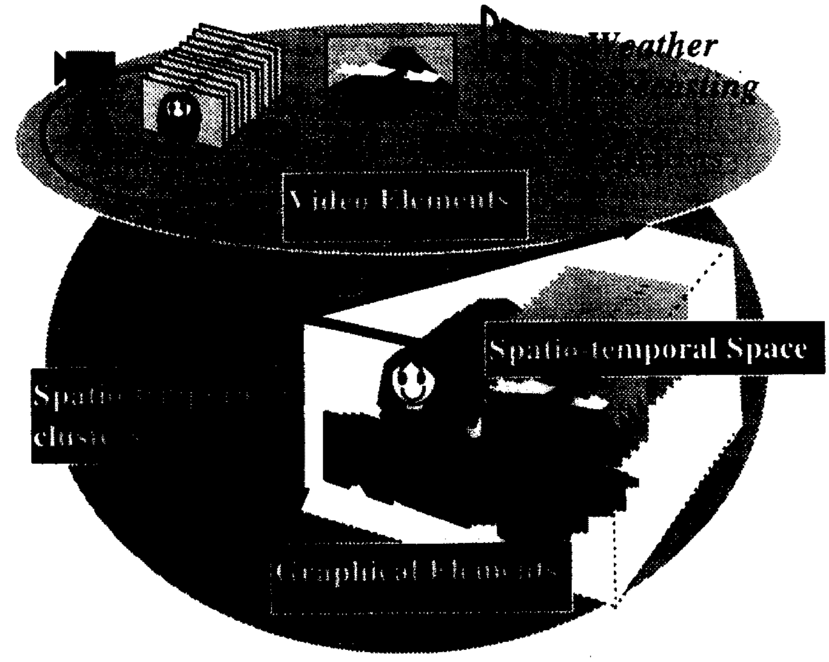

Fig. 1 The Concept of Graphical Video.

of video elements. Each video element is transferred into a graphical element which is scalable and content-based. These graphical elements · have feature, texture and layout information in the spatio-temporal space.

2. Graphical Video: spatio-temporal model

Moving images can automatically or interactively be divided into a group of imaging elements which compose the video as a whole. The important thing in constructing a video is to enable each element to be represented within the element itself. It must be possible to reconstruct the original video in its entirety using only video element characteristics and layout information.

Figure 1 illustrates the concept of the Graphical Video. Video is represented by a gathering of spatio-temporal video elements and layout information in the spatio-temporal space. All pixels are regarded not as a gathering of frames but as boxels in three-dimensional space. The advantage of the Graphical Video representation is, thus enabling the graphical video "friendly" handling of video with characters, still images, charts which are already graphically and scalably represented in a process of creating and reproducing video. Moving images which can be graphically represented are useful for both reuse and reference / indexing.

The advantages of this model are summarized as follows:

I. Video can be represented by a group of elements, allowing the partial contents of the video to be comprehended without transmitting and decoding the entire video stream.
II. Each partial element can be prioritized in its processing.
III. Video can be dynamically scaled for terminals and network capabilities because of a consequence of scalable representation.

3. Scalable representation of spatio-temporal clusters

In making the original video elements into graphical elements, an essential condition is that the represented objects have to be scalable in spatio-temporal space not just in spatial plane. We have studied the spatio-temporal clusters created by moving objects in video, and tried to represent those using texture and contours. The texture and contours of the spatio-temporal clusters are separately represented in our method because transformation of the texture and contours of the same object in video do not always agree with each other as the result of occlusion. Here, the contours are approximated by polygons, while the texture is approximated by three-dimensional fractal coefficients. The texture usually covers an area larger than the original area, because it is difficult to exactly approximate near contours because contour features are too complicated to be systematically calculated. Therefore, after the calculation of texture and contours, the contours are used to extract the desired region from the texture as shown in Fig. 2.

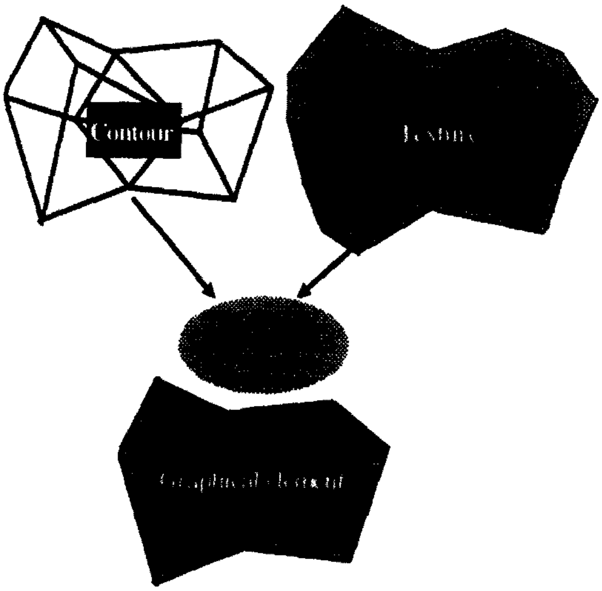

Fig. 2 The graphical elements.

3-1 Spatio-temporal contour: polygonal approximation

First, a voluntary spatio-temporal cluster is extracted from the original three-dimensional boxels. Generally, only a sliced x-y surface provides good visual quality, so the contour polygons are calculated as follows:

I. Extract key frames, whose features significantly changed in the x-t or the y-t image, as shown in Fig. 3.
II. Approximate the features of the key frames using polygons [4].
III. Compensate other frames between key frames with transference, rotation and scaling by referring tothe key frames.

In general, these key frames and compensation parameters are automatically calculated in the extraction process. In the final step, the spatio-temporal contours are approximated into a scalable figure using polygons.

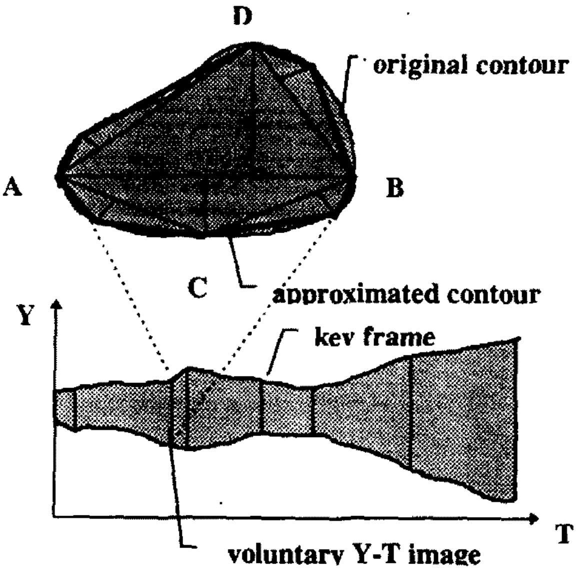

Fig. 3 Contour approximation method.

3-2 Spatio-temporal texture: three-dimensional fractal approximation

A voluntary spatio-temporal texture is approximated using three-dimensional fractal coefficients. Conventionally, x-y planes are sequentially encoded and compressed temporally with motion compensation (MC). A unique approach that excludes MC and regards all the pixels as boxels in the CG space was proposed in [5]. But they used cubes in splitting and merging clusters, and this can worsen the quality of the x-y plane. On the other hand, fractal approximation of images can be found in [6] and it was extended to three dimensions in [7][8]. The methods in these papers also describe video without MC. In this work, we chose the simple fractal coding in [7], which was directly extended to three dimensions in [6]. The merit of using fractal approximation is that it is itself a scalable representation because of its self-similarity. Fractal coefficients are composed of locations, pixel shuffling and intensity scaling. Time and spatial frequency are disregarded. Fig. 4 shows the fundamental structure of a three-dimensional fractal coding. In the approximation, first, whole pixels are divided into range cubes that do not overlap with each other, while a domain cube scans in the domain pool. Secondly, a voluntary domain cube, which is twice as large as the range cube, is scaled down by half, and then the range of intensity is scaled. Then all pixels in the scaled-down domain cube are shuffled. The similarity between the original range cube and the transferred domain cube is then estimated. The most appropriate location, pixel shuffling and intensity scaling parameters are saved as the fractal coefficients of the original range cube. The location and size of the domain pool is voluntarily decided. Usually, domain pool having enough space is located around the range cube.

Here, the most serious problem is the location and size of the domain pool because the amount of calculation and appropriation process depends on these factors. It is simply suggested in [8] that we can get the most appropriate domain cube around the range cube. Fractal approximation is a self-similar function, so this suggestion is reasonable. By the way, a very important condition is that the most appropriate domain cube is found in or near the spatio-temporal cluster to construct region-based representation. This is why the location and size of domain pool is set up as shown in Fig. 5. The region expansion process is as follows:

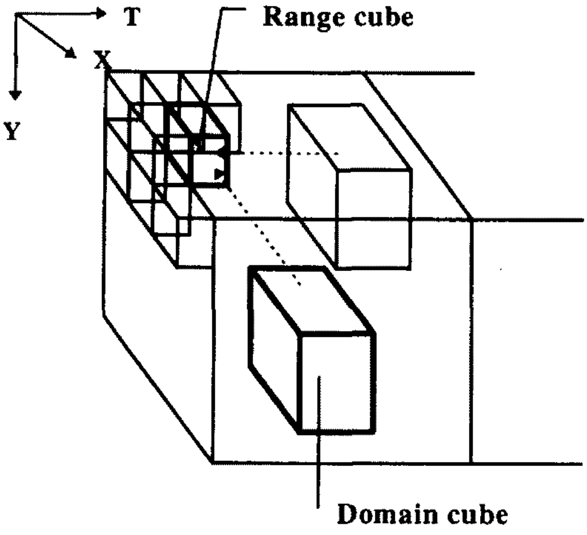

Fig. 4 Concept of the three-dimensional fractal in reference [7].

I. Calculate the maximum distance in each direction X, Y and T, and locate range cubes to fill the object. Only cubes which cover the spatio-temporal clusters are valid. This is the first region expansion in Fig. 5.

II. The cubes next to the expanded region are also valid. This is the second region expansion.

III. Limit the searching area from the center of range cube to W*H*T.

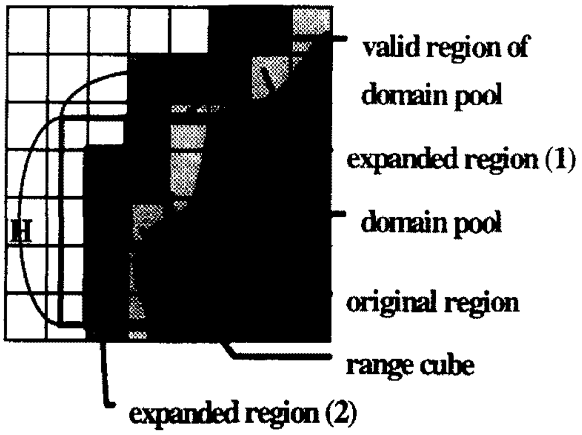

Fig. 5 Region expansion method. (described in two dimensions)

The second regional expansion might be effective in improving the visual quality of texture, because, especially in the boundaries, the domain pool whose center coincides with the center of the range cube is guaranteed as an appropriate candidate. In this way, the texture of the spatio-temporal cluster can be described in a region-based manner.

4. Experiments and study on spatio-temporal texture

The experiment for the texture approximation was conducted with a video with 180*120*40 pixels.

First, as shown in Fig. 6, we prepared four simple examples of spatio-temporal clusters from the video. The spatio-temporal contours were automatically determined. Then, using the SNR, we compared the region-based method (our method) with the normal method (wider domain pool) with respect to domain pool size and location. We used three models for the comparison:

A. whole 40*40*20 [pel] non-region-based

B. region 1 20*20*15 [pel] region-based
C. region 2 20*20*15 [pel] region-based

Among the above, A is the normal three-dimensional fractal approximation. In B and C, the domain pool is located near spatio-temporal clusters. But in B, only the first regional expansion described in the previous section is achieved; while both the first and second regional expansion is achieved in C. As a result, the domain pool in C is narrower than that in B.

Other importance fractal parameters are

- the size of the range cube 4*4*4 [pel]
- the size of the domain cube 8*8*8 [pel]

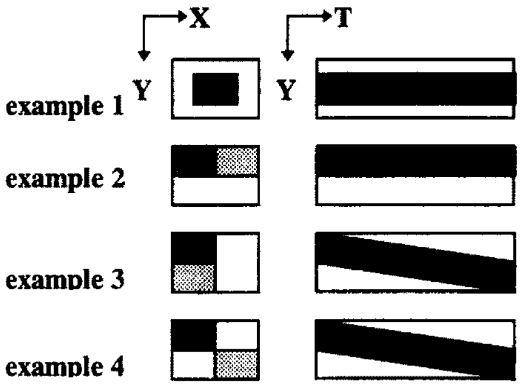

Fig. 6 Examples of spatio-temporal clusters

The SNR within the spatio-temporal cluster was calculated frame by frame, and the results are shown in Fig. 7(graphs of two samples shown). There is little difference (less than 2 dB) between A and C, but B is significantly worse especially at the beginning and the end of the sequence. It is found that the second regional expansion is effective for improving the visual quality of texture, because the domain pool whose center coincides with the center of the range cube is guaranteed as an appropriate candidate, and an appropriate domain cube is found near a range cube. In this way, the texture of the spatio-temporal cluster can be described region-based. I this way, our study shows the proposed method is sufficient for texture approximation, although the size of the domain pool is less than that of existing method.

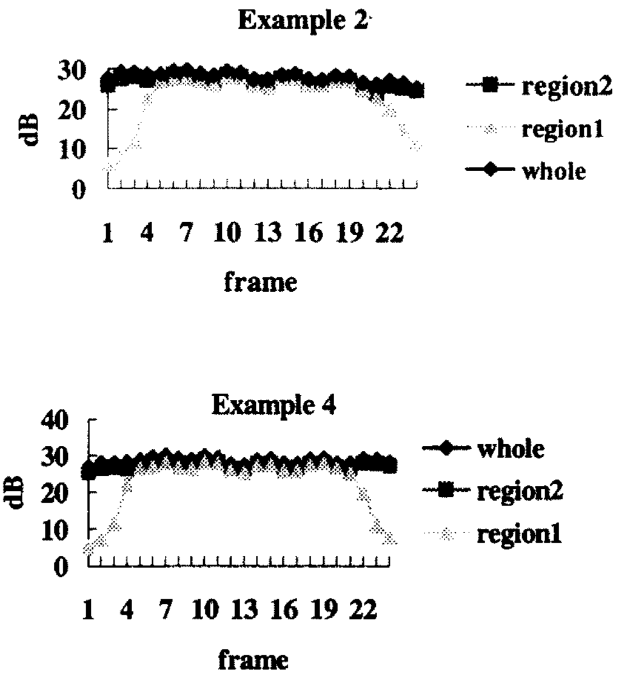

Fig. 7 Experimental results.

5. Conclusion

This paper studied a new method of video representation.

First, the concept of Graphical Video, which is content-based and scalable, was introduced. This was followed by the proposal of a method for the approximated representation of spatio-temporal clusters using polygons and three-dimensional fractal coefficients. Finally, a study of the characteristics of spatio-temporal texture approximation was presented using frame-by-frame SNR measure.

In the future, we will evaluate coding efficiency of whole video, while this paper describes a part of the video, and discuses applications, such as reuse and reference / indexing.

Reference

[1] R. S. Jasinschi et al., "Content-based Video Sequence Representation", ICIP '95, Vol. 3 pp. 229 -232.

[2] John Y. A. Wang et al, "Representing Moving Images with Layers", IEEE Trans. on Image Processing, Vol. 3, No. 5, September 1994.

[3] John Y. A. Wang et al, "Applying Mid-level Vision Techniques for Video Data Compression and Manipulation", SPIE Vol. 2187 VCIP 1994.

[4] Franke et al., "Region Based Image

Representation with Variable Reconstruction Quality", SPIE Vol. 1001 VCIP 1988.

[5] Patrice Willemin et al., "Image Sequence Coding by Split and Merge", IEEE Trans. on Communications, Vol. 39, No. 12, December 1991.

[6] Arnaud E. Jacquin, "Image Coding Based on a Fractal Theory of Iterated Contractive Image Transformations", IEEE Trans. on Image Processing, Vol. 1, No. 1, January 1992.

[7] Lazar et al., "Fractal Block Coding of Digital Video", IEEE Trans. on Circuits and Systems for Video Technology, Vol. 4, No. 3, June 1994.

[8] Kai Uwe Batthel et al ,"Three-Dimensional Fractal Video Coding", ICIP'95 Vol. 3 pp. 260-263.