Journal of Korea Multimedia Society Vol. 11, No. 6, JUNE 2008(pp. 729-736)

Flowing Water Editing and Synthesis Based on a Dynamic Texture Model

Qian Zhang†, Kijung Lee††, Taegkeun Whangbo†††

ABSTRACT

Using video synthesis to depict flowing water is useful in virtual reality, computer games, digital movies and scientific computing. This paper presents a novel algorithm for synthesizing dynamic water scenes through a sample video based on a dynamic texture model. In the paper, we treat the video sample as a 2-D texture image. In order to obtain textons, we analyze the video sample automatically based on dynamic texture model. Then, we utilize a linear dynamic system (LDS) to describe the characteristics of each texton. Using these textons, we synthesize a new video for dynamic flowing water which is prolonged and non-fuzzy in vision. Compared with other classical methods, our method was tested to demonstrate the effectiveness and efficiency with several video samples.

Key words: Textons, Dynamic textures, LDS, Statistics method, Textures synthesis

1. INTRODUCTION

Natural phenomena have a complexity and mobility that have been roughly emulated in many recent advances in simulation and rendering. The synthesis of dynamic water flowing in computer simulations is an ongoing research focused on computer graphics and virtual reality. The process of rendering a water surface in real-time computer graphics is highly dependent on the demands on realism. There are main solution for water simulation. One is based on the physics of fluid flows which have been developed since the time of Euler, Navier and Stokes (from the 1750's to the 1850's). These developments have led to the so-called Navier-Stokes Equations, a precise mathematical model for most fluid flows occurring in Nature. In general, these algorithms strive for accuracy and are fairly complex and time consuming. This is because the applications that require these solvers have to be physically accurate. It is obviously crucial that the stresses and drag on an airplane or a bridge are calculated precisely. The other is based on the video. The video will be looked as a sample, and it will be rearrangement for synthesis a new video. The method in this paper is based on the type of solution.

In this paper, we propose a simulation method based on dynamic textures. Our research utilizes the video sample to synthesize water scene flowing smoothly in vision at fixed time. The method is based on video sample. Reference to the MRF (Markov Random Field) in 2-D texture synthesis, we use the dynamic textures to solve the fuzzy phenomenon method [1]. The fuzzy phenomenon occurred in video result of paper [1] means that the flowing water they synthesized looked vague

※ Corresponding Author : Taegkeun Whangbo, Address : (461-701) 5-7 Saeromkwan, Kyungwon Univ., Bokjeong-Dong, Sujeong-Gu, Seongnam-Si, Gyeonggi-Do, Korea, TEL : +82-31-750-5417, FAX : +82-32-757-9508, E-mail : tkwhangbo@kyungwon.ac.kr
Receipt date : May 1, 2007, Approval date : Dec. 11, 2007
† Dept. of Computer Science, Kyungwon University (E-mail : aazhqg@hotmail.com)
†† CTI of Kyungwon University (E-mail : jcm5758@empas.com)
††† Dept. of Computer Media, Kyungwon University
※ This research was supported by the Ministry of Cultural & Tourism and Korea Culture & Content Agency, Seoul, Korea, under Supporting Project of Culture Technology Institute.

in the fog. The textures describe the pixels' distribution model on gray level scale, and show the quality of the object, for instance roughness, smooth, particle degree, randomness, etc. The structural techniques (one of the three principal approaches used to describe texture) characterize texture as being composed of simple primitives called "texton" (texture element), which is regularly arranged on a surface according to some rules. Our method treats the video sample as textons. We calculate the distribution of video textons through analyzing the video sample automatically, and then normalize the flowing water results. The method utilizes linear system (Linear Dynamic System, LDS) to describe texton. Synthesis technique can control the results by giving restrained conditions, in order to avoid the fuzzy phenomenon appearing in the method [1].

The remainder of this paper is organized as follows. After reviewing related work in Section 2, we introduce the main methods and techniques in Section 3, and show how to edit and synthesis the video in Section 4. Several kinds of flowing water synthesis and editing results are shown in Section 5. We conclude our paper in Section 6.

2. RELATED WORKS

Initially, in terms of hydrodynamics, visual effects such as sea level, waves and rivers, were produced through the modeling method of physics. Limited by the CPU's performance, it takes a lot of time to edit flowing water. Consequently, researchers improved the method using parameter models. For example, in order to show the folding effect on the surface of water, a paper [2] based on the Gerstner model adopted the Lagrangian particle to imitate the wave parameter surface, which controls sinusoid through the depth and slope of the sea floor. But the methods based on physics models show particle and net of water are just moved near its initial position. They neither are all unable to display the real flow result, nor couldn't deal with the influence on the surface of water which brought by the water border. In recent years, a lot of scholars use this method to produce the flowing water result for cartoon scene [3] and achieved better results.

More recently, researchers have begun trying to utilize video to produce the flowing water scene, such as Bar-Joseph [4], Wei [5], and Schodl [1,6]. This method analyzes the video samples, and rearranges the frames with high efficiency. Paper [7] rearranged the frames by analyzing the relationship between frames. This method could handle the synthesis of both unstructured textures as well as structured textures. It could cause duplication in vision and, thus, cannot be used to get better results in waves and body motion. The main difference between [4], [5] and [6] is that [5] and [6] are using different ways to reconstruct every frame. Methods [5] and [6] improve the quality of the video result, but efficiency is very low.

Stefano Soatto proposed the Dynamic Textures theory in his paper [1]. In the paper author obtained a dynamic texture through dynamic model, calculated the model's parameters, and made video prolonged. The simulation result kept the original characters basically, but little fuzzy phenomena still existed. Kiran S. Bhat [8] in 2004 adopted human-computer interaction to format and edit dynamic scenes according to the route that users mark. This method, which chases frames and assigns an editor to train the sample pictures one by one, has achieved good experiment results. Our research refers to the studies above to synthesize flowing water.

3. PROPOSED APPROACH

In this section, we will introduce the main algorithms in our system. In order to understand the method clearly, we explain some concepts first, and then we illustrate our model and approaches.

3.1 Several Concepts

We should introduce some concepts occurred in the paper as follows.

Textures show the luminance (gray scale) in 2-D images, which have three criteria: repeat constantly in a certain array area larger than this array; the array is made up at random by the basic part; every part is the same entity in general and there is roughly the same dimension in any locality of the texture area. The basic part of this array is usually called the texton [9]. The dynamic texton proposed in this paper is one part of the video, which shows the vision consents. For simplicity, we refer to dynamic textons as textons in the paper.

Dynamic textures [1] are sequences of images in moving scenes that exhibit certain stationary properties in time; these include sea-waves, smoke, foliage, and whirlwinds but also talking faces, traffic scenes etc. Let {I(t)}, t=1···τ. Suppose that at each instant we can measure a noisy version y(t)=I(t)+ω(t), where ω(t) is an independent and identically distributed sequence drawn from a known distribution. We say that the sequence is a (linear) dynamic texture.

Generally, a system means a set of entities which accept input and produce the output results correspondingly. In a linear system [10], the phase space is the v-dimensional Euclidean space, so any spot in the phase space can be represented by a vector with v dimensional. The analysis of linear systems is possible because they satisfy a superposition principle: if u(t) and w(t) satisfy the differential equation for the vector field (but not necessarily the initial condition), then so will u(t) + w(t). In the paper, we use LDS to describe the dynamic texton. X_t (at time t in frames of texton) shows the state variable, which the system inputs, and Y_t is the system's output, where Y_t is the frame at time t.

3.2 Dynamic textures model

The dynamic textures model is a two-level statistical model with textons [9] and their distribution, as shown in Figure 1. Each texton is represented by a LDS [10] and the initial state distribution. Texton distribution can be represented by a matrix M_{ij}.

Each texton is represented by an LDS with the following state-space model:

$$\begin{cases} X_{t+1} = AX_t + Bv_t \\ Y_t = C_t X_t + w_t \end{cases} \quad (1)$$

Where, X_t is the hidden state variable, Y_t is the observation, and v_t and ω_t are independent Gaussian noises at time t. Then the parameters of an LDS can be represented by Θ = {A, B, C}.

We assume that the distribution of textons satisfies the first-order Markovian dynamics, which could be represented by a transition matrix.

$$M_{ij} = P(l_k = j \mid l_{k-1} = i) \quad (2)$$

Where two continuous textons are labeled as l_k and l_{k-1}, frame j and frame i in the two textons are labeled as j and i. P is the transition probability between l_k and l_{k-1}. The scale of transition matrix M_{ij} is m*n, and m and n labeled as the amount of frames in textons l_k and l_{k-1} respectively.

3.3 Extract the texton

Given $Y_{1:T} = \{Y_1, Y_2, \ldots, Y_T\}$, or the observation

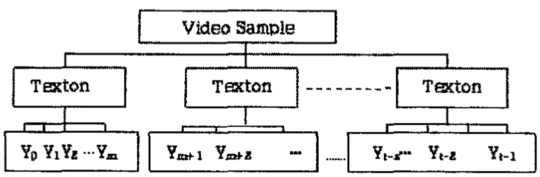

Fig. 1. With the learnt dynamic texture, a video sequence can be divided into multiple segments, labeled as texton 1, texton 2··· texton n. Each segment may have a different length. We use Y_0, Y_1, ···, Y_t to represent continuous frames of the vide sample.

Y_t from frame 1 to frame T, our system calculates the model parameters $\Theta=\{A, B, C\}$. Set $p(t)=x_t/X_t$, here p(t) is the confidence value which used for evaluation of the frame's quality produced by LDS at time t. x_t is the state variable as we using LDS to calculate, and X_t is the ideal statement variable got from Y_t.

Greedy approach

The greedy approach is utilized to incrementally initialize our model. First, we use $T_{\min}$ frames to fit an LDS i, and incrementally label the subsequent frames to segment i until the fitting error is less than p(x). Then, all existing LDS' (from 1 to i) learnt from all preceding segments (possibly more than i) are tested on the remaining unlabeled $T_{\min}$ frames, and the best-fit LDS is chosen. If the smallest fitting error exceeds the given threshold, none of those LDS fits the observation well, we select a new LDS and repeat the above process until the entire sequence is processed.

Fitting an LDS

In order to capture richer dynamics (velocity and acceleration), we use a second-order linear dynamic system:

$$\begin{cases} Y_t = C_t X_t + w_t \\ X_{t+1} = A_1 X_t + A_2 X_{t-1} + D + B v_t \end{cases} \quad (3)$$

Where $v_t \sim N(0,1)$, $\omega_t \sim N(0,R)$, and R is the covariance matrix. A closed form approximated estimation of the model parameters could be derived as follows [1]:

1. We calculate the SVD (singular value decomposition) of the observation sequence $Y_{1:T}$, $[U,S,V]=SVD(Y_{1:T})$, and set:

$$C=U, X=SV^T \quad (4)$$

2. The maximum likelihood estimation of A_1, A_2, D, and B are given by:

$$[A_1, A_2] = [R_{00}, R_{01}] \cdot \begin{bmatrix} R_{11} & R_{21} \\ R_{12} & R_{22} \end{bmatrix}^{-1}$$

$$D = \frac{1}{\tau-2}\left[R_0 - \sum_{i=1}^{2} A_i R_i\right] \quad (6)$$

$$R_i = \sum_{t=2}^{t} X_{t-i}$$

$$R_{ij} = \sum_{t=2}^{\tau} X_{t-i}(X_{t-i})^T - \frac{1}{\tau-2} R_i R_j^T \quad (7)$$

Where, $BB^T=\hat{Q}$, and $\hat{v}(t) = \hat{x}_{t+1} - A_1\hat{x}_t - A_2\hat{x}_{t-1} - D$. $\hat{Q}$ is not the full rank matrix. In order to calculate parameters B in formula (3), we could reduce the rank n through SVD. $\hat{Q}=U_Q\Sigma_Q U_Q^T$, $\Sigma_Q = diag(\sigma_Q(1),\ldots,\sigma_Q(k))$, $k \le n$. The diagonal matrices of Q is composed with $\sigma_Q(1),\ldots,\sigma_Q(k)$, which are the diagonal elements.

Normalize distribution of textons.

When we obtain Mij, the matrix M is then nor malized such that $\sum_{i=1} M_{ij}=1$.

In the paper, an important parameter that the user needs to determine is $T_{\min}$. $T_{\min}$ must be long enough to describe the original wave. In our system, it is suitable to choose 20 ~ 30 frames as $T_{\min}$. The threshold of p(t) is determined by users, and it is related with the video sample's quality.

4. EDITING AND SYNTHESIS

Analyzing the dynamic textures, new video can be synthesized. Moreover, we can edit the video interactively, both at the texton level and at the distribution level.

Editing

We rearrange the textons according to the textons and the user's demand. The flowing water in nature may be influenced by the wind, artificial force, etc. As the system performs, we rearrange the textons by the transition matrix. In our system, we choose the textons randomly when they have nearly the same matrix.

Synthesis

We divide the flowing scene into two types: regular dynamic texture and random dynamic texture. When we conduct the experiment with a different sample, we utilize the different methods to solve them. The regular dynamic texture means that the change of flowing water is repeated and regular, such as s calm sea level, lake level, and so on. The main character is that we can incorporate the two textons directly without producing the new frames at the junction. The random dynamic texture means that the change of flowing water is irregular, such as a spring, brook, river, and so on. The main character is that we cannot achieve the smooth video in vision if we incorporate the two textons directly. So we incorporate the two textons with hard constraints.

When we have the texton sequence, the second step in synthesis is conceptually straightforward. In principle, given a texton and its key frames, a video sequence can be synthesized frame by frame with the learnt LDS and sampled noises. However, the prediction power of LDS decreases after some critical length of the sequence as LDS approaches its steady state. In the system, we called the unjointed textons as synthesis texton and constrained texton according to time. A by-product of the constrained synthesis is the smooth transition between two textons because the two starting poses of the second texton are guaranteed from the synthesis processes of both the first and the second textons

Let W_i and W_j be the two unjointed textons, and $\{x_{i1}, x_{i2}\}$, $\{x_{j1}, x_{j2}\}$ are the first two key frames in the two textons respectively. In order to joint the two textons smoothly in the video, we set the hard constraint as follows:

$$\begin{cases} x_1 = x_{i1}, \ \ x_2 = x_{i2} \\ x_{l-1} = C_i^T C_j x_{j1}, \ \ x_l = C_i^T C_j x_{j2} \end{cases} \tag{8}$$

We should switch C_n to $C_i^T C_j$ through the projection as the hard constraint matrix. The synthesized frames between two textons are recorded as $x_3, x_4, x_5, \ldots, x_{l-2}$. We calculate the transition in the linear system of the first texton based on the LDS for the two textons we have established before.

5. EXPERIMENT

In this part, we demonstrate the performance of our system step by step through several different flowing water video samples.

5.1 The system processing

The synthesis of dynamic textures includes two stages: analyzing and synthesizing. The analyzing is the process in which we divide the video sample into textons, and get the initial state distribution between textons; the synthesis process reconstructs the textons, and gets the smooth, prolonged video according to the user's demands. The algorithm is summarized as follows:

step 1 : Get textons and their distribution

We find the textons in video sample using greedy approach, and get the distribution between two random textons, which is labeled by a matrix M_{ij}.

step 2 : Estimate the LDS for every texton

We describe the texton with a second-order linear dynamic system, and estimate the each LDS's parameters.

step 3 : Synthesize the new video

We edit and plan the new order among textons using dynamic programming approach and incorporate the two textons with hard constrains.

5.2 Experiment results and discussion

We utilized a sea level video and spring video to experiment separately on an Intel Pentium IV 3.0GHz computer with 1G memory. The two results are shown as follows according to the algorithm.

We use our algorithm to analyze the simple video, and show the results in Table 1.

Table 1. Analysis results of simple video

	Video (frame)	Textons (frame)	CPU time(s)	Synthesis result(frame)
1.Sea level	46	Texton 1: 1~13, 22~35 Texton 2: 14~21 Texton 3: 36~44	34.486	134
2.Spring	180	Texton 1: 1~44 Texton 2: 45~156 Texton 3: 157~180	264.420	238

Sea level

We only show the part of textons and the synthesis results in Fig.2 and Fig.3 limited by the space.

Spring

We show three continuous frames in original video sample in Fig. 4, and we calculate three continuous frames using LDS in Fig. 5. From the two figures, we find that there are no special differences between them. It means that we can use LDS to describe dynamic textures, in other words LDS is the flowing spring texture. Then we show the synthesis with two adjacent textons in Fig. 6, in which A is the last frame in first texton, E … H is the smooth connection between two adjacent textons, and I is the first frame in second texton.

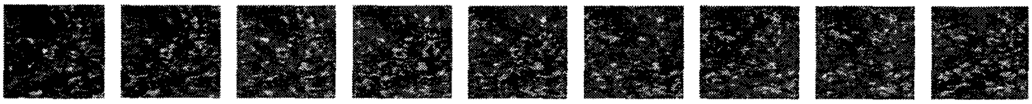

Fig. 2. Continuous frames in a texton, 128*128

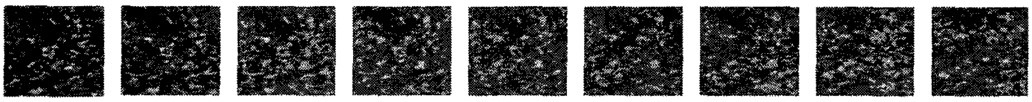

Fig. 3. Synthesis textons, 128*128

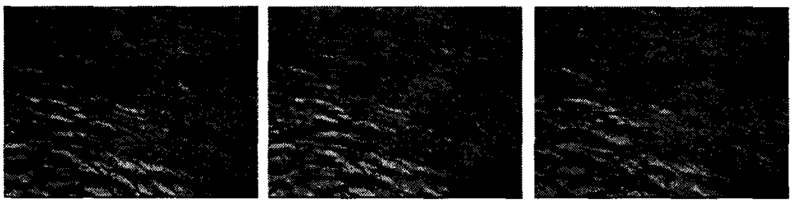

Fig. 4. Three continuous frames in the original video sample, 320*240

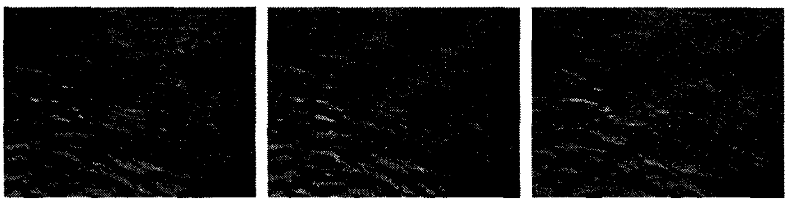

Fig. 5. Continuous frames using LDS we have calculated, 320*240

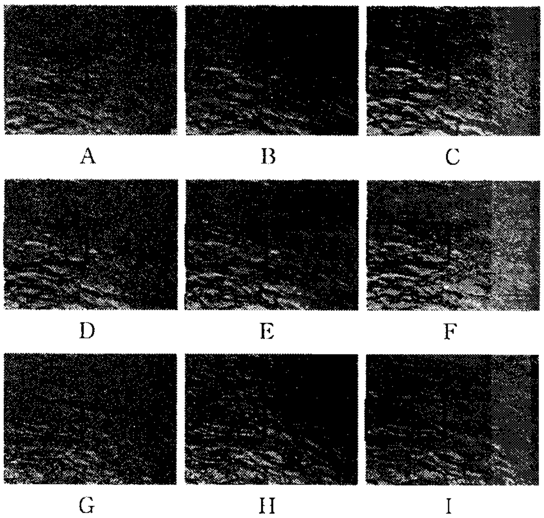

Fig. 6. Synthesize with two adjacent textons.

Flowing water

We divided the flowing water into regular and random, according to the flowing water's texture character. Different results will be produced according to the different type of flowing scene. It is boundless to synthesis the video for regular dynamic texture as long as the sample satisfies the shortest frames. But for the random dynamic texture, when we use the shortest frames, it will cause reiteration in the boundless time. So there is a longest time for synthesized video. The conclusion is shown in Table 2.

In the paper, we proposed a method to synthesize and edit flowing water based on dynamic textures model. The method can handle cases mixed with synthesizing photorealistic and cartoon flowing

Table 2. The results between different type of flowing scene

Flowing water		Shortest frame as synthesis (frame)	Synthesis the longest in time with shortest frame(frame)
regular	Lake	35	boundless
	Sea level	70	boundless
random	Spring	65	174
	River	73	248

Fig.7. Synthesized results in paper [1]

scene. In our system, dynamic textures are represented by a set of textons and their distribution. As calculating LDS of each texton, we can edit and synthesize video easily. Through the hard constraint, we solved the fuzzy phenomenon, synthesized the smooth video and kept the original flowing water characters compared with experiment result in paper [1]. We show the result from paper [1] in Figure 7, in which left part is the original video and right part is the synthesized results. It is obvious that the new one looked vague and lost some natural characters for smoothing through filter. In our system, we achieved satisfactory results through several experiments.

6. CONCLUSIONS

In the system, our objective was to find a reliable method to imitate a flowing scene that could show a dynamic landscape on a digital TV or in computer games, such as a city landscape show, the ditch work demonstration, etc. Our approach was inspired by the proposed model for dynamic textures. We realized the model by effective analysis, optimized LDS and hard constraints. We demonstrate them through several experiments. In experiment part we concluded our algorithm steps one by one. There still exists a longest in time for a sample vedio generally in order to avoid visual repeat. LDS is essential to synthesize the high quantity of video. Sometimes, several vague images will occurred among the generated frames. Currently, we need to conduct further research on the interaction of the system and improve the robust and stability of LDS. In addition, we found that applying different decoding methods for the video will have little influence on our results.

REFERENCES

[1] S. Soatto, G. Doretto, and Y.N. Wu, "Dynamic Textures," *In Proceedings of IEEE International Conference on Computer Vision*, pp. 439-446, 2001.

[2] F. Alain and T.R. William, "A Simple Model of Ocean Waves," *Computer Graphics*, Vol.20, No.4, pp. 75-84, 1986.

[3] N. Foster and R. Fedkiw, "Practical Animation of Liquids," *In Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 23-30, 2001.

[4] B.J. Ziv, "Statistical Learning of Multi-dimensional Textures," *Master's thesis, The Hebrew University of Jerusalem*, 1999.

[5] L.Y. Wei and L. Marc, "Fast Texture Synthesis Using Tree-structured Vector Quantization," *In Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 479-488, 2000.

[6] S. Arno, S. Richard, H.S. David, and E. Irfan, "Video Textures,"" *In Proceeding of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 489-498, 2000.

[7] J. Stam, "Stable Fluids," *In Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 121-128, 1999.

[8] K.S. Bhat, S.M. Seitz, J.K. Hodgins, and P.K. Khosla, "Flow-based Video Synthesis and Editing," *ACM Transactions on Graphics*, Vol.23, No.3, pp. 360-363, 2004.

[9] J.B. Precept, "Effect of Orientation and of Shape Similarity on Perceptual Grouping," *Perceptual psychophysics*, pp. 300-302, 1966.

[10] E.G. Theodore, and C.S. Irvin, "System Theory: Modeling, Analysis and Control," Springer published on 1 Jan, 1999.

Qian Zhang

She received her B.S and M.S. degree from Sandong University, Sandong, China in 2003 and 2006, respectively. She is currently a Ph.D. degree student of the Dept. of Computer Science in Kyungwon University, Korea. Her research interests include Image Processing, Pattern Recognition, Face Recognition.

Kijung Lee

He received his M.S. and Ph.D. degrees from Kyungwon University, Korea in 2003 and 2008. He is currently a researcher of CTI of Kyungwon University, Korea. His research interests include Face Recognition, 3D Game Engine, Image-based Retrieval.

Taegkeun Whangbo

He received his Ph.D. degree from Stevens Institute of Technology, USA in 1995. He is currently a professor of Dept. of Computer Media in Kyungwon University, Korea. His research interests include 3D Game Engine, Image Processing, Face Recognition.