Video Synthesis From Still Images Using 3-D Flow Models

Yutao Zhao and Yuncai Liu

***Abstract*—In this letter, we present a new approach to synthesize video from still images. The method first models 3-D flows, and then it proposes image modulation. Finally, the 3-D flow models are projected onto image planes to produce the final video. We use some results of computational flow dynamics to model the flows and propose image matting and inpainting methods that are more appropriate to our animation. Our algorithm can effectively handle still images that have flows. Experiments demonstrate that our synthesis method can produce good-looking videos.**

***Index Terms*—Computational flow dynamics, image matting and inpainting, still images, video synthesis, 3-D flow models.**

I. Introduction

RECENTLY, there are considerable interests in video or animation editing. In these approaches, the basic idea consists of direct simulation in computer graphics (CG) and video editing. We have come to appreciate the direct simulation of flow behavior in films. Many papers follow this idea [1]–[3], trying in different ways to simulate flows. However, direct simulation of flow is difficult, because it requires too much computation power and many parameters have to be set up. Also in video editing, the resolution is low.

Compared with direct simulation and video editing, video synthesis from still images has superiorities of low cost and higher resolution. The still pictures can be modulated to match the director's intentions, and modifying physical setting is possible. The warping tool [4] can animate an image to some degree; however, its domain is limited to smoke and the final animation is unnatural. What we want is good-looking video synthesized from still picture while its natural appearance can also be maintained.

In order to get animation from still flow picture, we present an image-based flow texture analysis and synthesis method. To this end, we construct appropriate flow models based on the demand of processing still images and research results of flow mechanics. For the image processing part in our synthesis, we put forward reasonable matting and inpainting techniques to satisfy both accuracy and speed. We get flow animation through image matting, image inpainting, and projecting the models of 3-D flow onto still image.

This letter is organized as follows: Section II introduces our approach of image-based flow analysis and synthesis approach. In this section, Part A describes ways of constructing different flow models, and Part B discusses image modulation and projection methods. Section III gives the experimental results, and conclusions are drawn in Section IV.

II. Proposed Approach

Given a still natural image that has flows, we first extract flows using image matting technique. Then, the image inpainting is performed to fill the interstices caused by the movement between flows and their borders. Also finally, construction of flow models and the projection procedure can produce the output video. In this section, we will discuss the construction of flow models firstly, and then image modulation and projection procedures will be introduced.

A. Construct Models of Flows

Some researchers have tried to model the water wave [5], [6]. In this letter, we adopt a method based on water wave spectrum that uses observation and research results of oceanography. The height of water wave can be generalized to a complex profile by summing a set of cosine waves.

Select a set of wave vectors k_i, amplitudes A_i, frequencies ω_i, angles θ_i, and phases ϕ_i, for $i = 1, \ldots, N$, to get the expressions

$$h(x,y,t) = \sum_{i=1}^{\infty} A_i \cos\left[\omega_i t - k_i(x\cos\theta_i + y\sin\theta_i) + \phi_i\right]. \tag{1}$$

Because $\omega_i^2 = k_i g$

$$h(x,y,t) = \sum_{i=1}^{\infty} A_i \cos\left[\omega_i t - \frac{\omega_i^2}{g}(x\cos\theta_i + y\sin\theta_i) + \phi_i\right]. \tag{2}$$

We select a analytical semi-empirical model, Pierson–Moscowitz (P-M) spectrum for the wave spectrum

$$P(\omega) = \frac{ag^2}{\omega^2}\exp\left[-\beta\left(\frac{g}{U\omega}\right)^4\right] \tag{3}$$

where$a = 8.1 \times 10^{-3}$, $\beta = 0.74$, U is the speed at the height of 19.5 m. Due to $\partial P(\omega)/\partial\omega = 0$, we can get the maximal frequency $\omega_m = 8.565/U$.

A spatial spectrum can be denoted by

$$P(\omega,\theta) = \frac{1}{\Delta\omega}\sum_{\omega}^{\omega+\Delta\omega}\frac{1}{2}A_i^2 \tag{4}$$

$$P(\omega,\theta) = P(\omega)G(\omega,\theta) \tag{5}$$

Manuscript received December 5, 2007; revised April 9, 2008. This work was supported by the National Basic Research Program of China (No. 2006CB303103). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Mila Nikolova.

The authors are with the Institute of Image Processing and Pattern Recognition, Shanghai Jiaotong University, Shanghai 200240, China (e-mail: zhaoyutao@sjtu.edu.cn; whomliu@sjtu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LSP.2008.925753

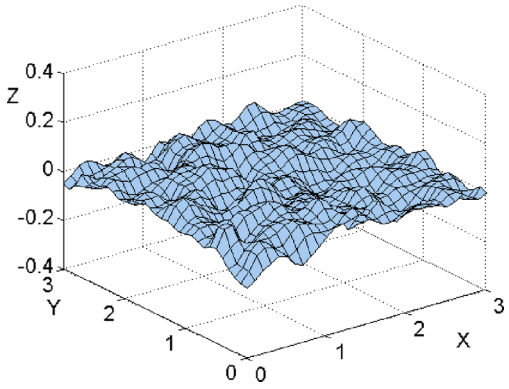

Fig. 1. Simulation result of water wave model at a time.

where $G(\omega,\theta)=(1/\pi)(1+p\cos 2\theta+q\cos 4\theta)$ is the direction function. Fig. 1 shows the simulation result of water wave model at a time

$$p=\left\{0.5+0.82\exp\left[-\frac{1}{2}\left(\frac{\omega}{\omega_m}\right)^4\right]\right\}$$
$$q=0.32\exp\left[-\frac{1}{2}\left(\frac{\omega}{\omega_m}\right)^4\right],\ |\theta|\le\frac{\pi}{2}. \tag{6}$$

As to the model of waterfall, we regard its velocity as two parts, $v=v_1+v_2$, v_1 reflects the flow trait, which can be described by Navier–Stokes (N-S) equation, while v_2 is caused by wind and gravity

$$\frac{\partial v_1}{\partial t}=-(v_1\cdot\nabla)v_1-\frac{1}{\rho}\nabla p+\upsilon\nabla^2 v_1+F \tag{7}$$

where $\nabla=(\partial/\partial x,\partial/\partial y,\partial/\partial z)$, $v_1=(v_{1x},v_{1y},v_{1z})^T$, x, y, and z are directions in rectangular coordinates, ρ is the density, υ is the viscous dissipation rate, and F is external force

$$v_{2x}=v_x+v_w\cos\theta \tag{8}$$
$$v_{2y}=v_y-\int_0^t\left(g-\frac{f}{m}\right)dt \tag{9}$$
$$v_{2z}=v_z+v_w\sin\theta \tag{10}$$

where v_x, v_y, and v_z are the speeds of x, y, and z directions, θ is wind direction, v_w is the speed of wind, f is resistance, and m is mass.

The discussed above are models of water wave and waterfall. Next we model the basic flow primitives that can be added to create complex flow. N-S equation (7) can be simplified in the case where the flow is incompressive ($\nabla\cdot v=0$), inviscid ($v=0$), and irrotational ($\nabla\times v=0$); then it satisfies the Laplace equation in the following:

$$\nabla\cdot v=\nabla\cdot\nabla\phi=\nabla^2\phi=0. \tag{11}$$

Based on this equation, we can model the basic flow primitives, i.e., uniform, source, sink, and rotation. Here we use cylindrical coordinates.

Fig. 2. We draw two lines in foreground and background to do image matting.

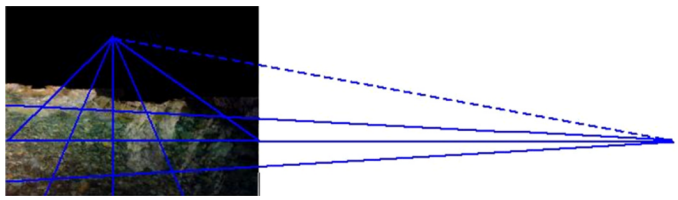

Fig. 3. We draw lines in water plane to get vanish points and vanish line.

Uniform

$$\phi=vr,\quad v_r=v,\quad v_\theta=0,\quad v_z=0. \tag{12}$$

Source and sink

$$\phi=\frac{Q}{2\pi}\ln r,\quad v_r=\frac{Q}{2\pi r},\quad v_\theta=0,\quad v_z=0. \tag{13}$$

Rotation

$$\phi=\frac{Q}{2\pi}\theta,\quad v_r=0,\quad v_\theta=\frac{Q}{2\pi r},\quad v_z=0. \tag{14}$$

If the flow is compressive and rotational, then it will not satisfy $\nabla\cdot v=0$ and $v=\nabla\phi$. For example, if we set

$$v_r=0,\quad v_\theta=\frac{Q}{2\pi r}z,\quad v_z=0 \tag{15}$$

then $\nabla\cdot v=z$, here $\nabla=((\partial/\partial r),(1/r)(\partial/\partial\theta),(\partial/\partial z))$, $v=(v_r,v_\theta,v_z)^T$. The motion of this model will be a cone or frustum of a cone according to the coordinates.

B. Image Modulation and Projection

We applied our method to images that have water, waterfall, and smoke. Before projecting flow models onto still image to get video, we need to do preliminary image processing. Firstly, the flows should be extracted using image matting technique. We assume an input image I is a composite of foreground image F and background image B

$$I_i=\alpha F_i+(1-\alpha)B_i \tag{16}$$

where α is the alpha matte that represents the pixel's foreground opacity and i means the ith pixel. Our goal is to get F_i, B_i, and α from a given image. In this letter, we use Levin's method [7]. Suppose α is a linear function of the image I

$$\alpha_i\approx aI_i+b,\quad \forall i\in w \tag{17}$$

where a and b are coefficients, $a=1/(F-B)$, $b=-(B/(F-B))$, and w is a small image window.

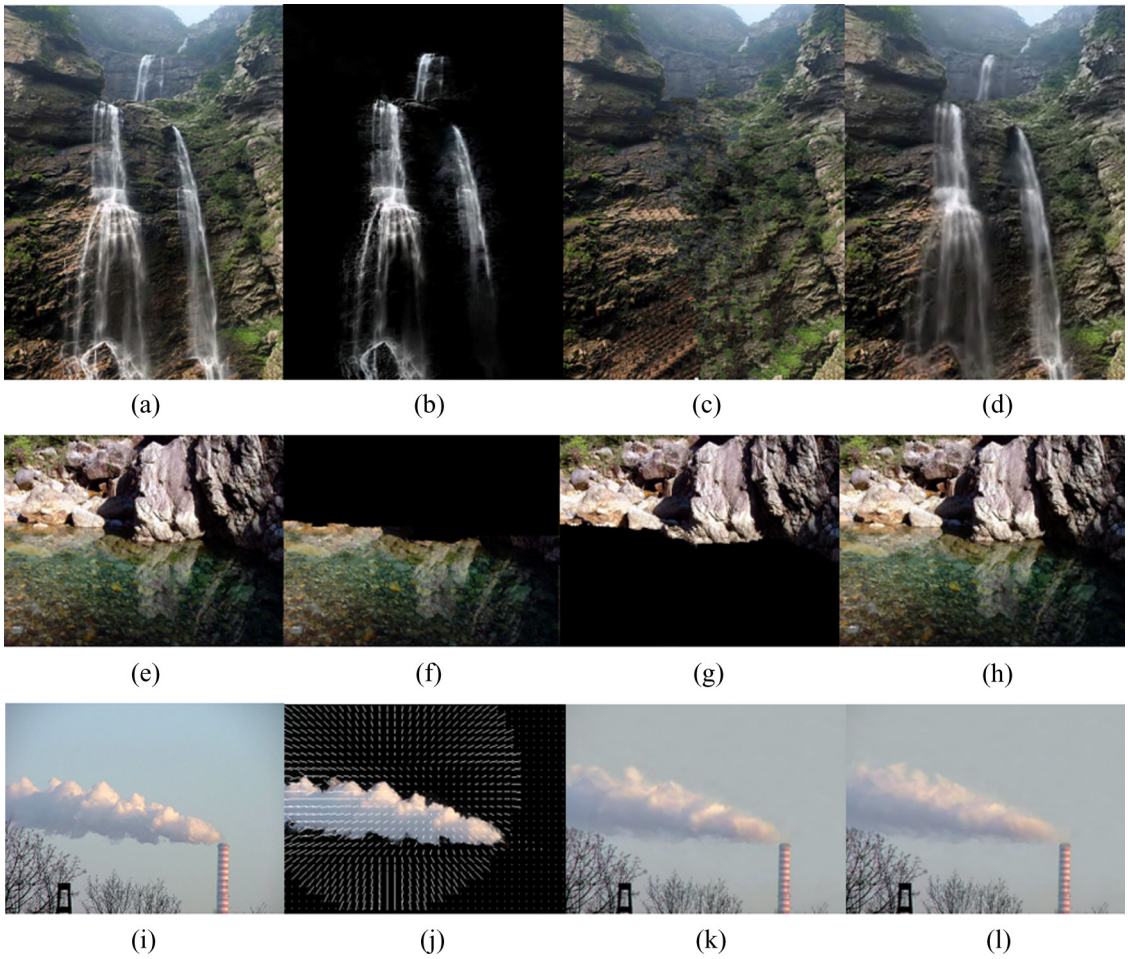

Fig. 4. Animate procedure. (a) Input a still image that has waterfall. (b) Segment waterfall using image matting. (c) Fill the interstice between the waterfall and its border using image inpainting, then map the model onto the waterfall parts and recombine the sequence into background to get final video (d). (e)–(h) are the procedure of synthesizing a still image that contains water. (i)–(l) is the synthesis procedure of image that have smoke. We have to combine different basic flow primitives and project them onto foreground that has been segmented (j). The other procedures are the same compared with the process of synthesizing still images that have waterfall and water wave. (k) and (l) are two frames in final video.

We minimize the cost function below to find α, a, and b

$$J(\alpha, a, b) = \sum_{j \in I} \left(\sum_{i \in w_j} (\alpha_i - a_j I_i - b_j)^2 + \varepsilon a_j^2 \right) \tag{18}$$

where w_j is a small image window and ε is a coefficient. a and b can be eliminated and denoted by the mean and variances of the intensities in the window w_j. In implementation of image matting, we only need to draw two lines in foreground (white) and background (black) that want to be segmented (Fig. 2). We solve the cost function to extract α matching the input lines.

After image matting, we adopt image-inpainting technique to fill the holes caused by flow and its border. Our approach is based on Criminisi's exemplar-based inpainting [8]. The main idea of this approach is an isophote-driven image-sampling process. Given a patch ψp in the region that needs to be filled, find the most similarities patch with ψq and copy the pixels from ψq to fill ψp. The criterion of similarity between patches is judged by the sum of squared differences of the pixels in the two patches. ψp is a patch centered at the point p, and we set the patch size of 9×9 pixels in our experiments. The algorithm performs the search through a best-first filling strategy that depends entirely on the priority values that are assigned to each patch. The patch priority $P(p)$ (for point p) is defined as

$$P(p) = C(p) \frac{|\nabla I_p|}{\alpha} \tag{19}$$

where $C(p)$ is the confidence term, ∇I_p is the gradient of point p, and α is a coefficient. Because low-cost and speed must be taken into account, we set $C(p)$ to 1 in the source region, and 0.5 in the target region by our experimental experience. In Criminisi's method, the search for an appropriate patch is a global one. In order to reduce the computational cost, we change the search area to a fixed size.

When preliminary image processing is done, we project the 3-D flow models onto the flow part of the still image. We use the pinhole camera model to map flow models onto images that have waterfall and smoke. As to the images that have water, the projection matrix can be solved by vanish line and vanish points [9]. As seen in Fig. 3, we draw lines in water plane to get vanish points (intersections) and vanish line (broken line). We specify these lines based on the angle of view of the image. The water plane here is the segmented part of Fig. 4(e).

III. Experimental Results

In this section, we present a set of results using images that have only one type of flow. For our experiments, we select still images that have water wave, waterfall, and smoke. Fig. 4 shows the procedure of our method. As shown in Fig. 4(a)–(h), we segment the waterfall and water wave parts using image matting, and then fill the gap between the flows and their borders by image inpainting. As can be seen in Fig. 4(c) and (g), the degree of image inpainting is diverse since the scope of interstices caused by water wave and waterfall are different. After preliminary image processing, models of waterfall and water wave are projected onto image planes. We set the physical parameters of models and combine the edited frames to the original image to get the final frame sequence, and finally, the new frames are connected to produce the final video. The parameters are important since they affect the scope and direction of animation. For the synthesis of still image that has smoke, there is a little difference. We have to add the flow primitives to create complicated flows and map these flows onto smoke part of the image (Fig. 4(i)–(l)).

IV. Conclusion

This letter proposes a novel video synthesis approach from still images using 3-D flow models. The proposed procedure is herein illustrated on images that have flow parts. The method is easy to implement. Given a still flow image, continuous video can be gained through image matting and inpainting, modeling flows, and projection. The experiment results show that our method is effective and efficient. In the future, we will attempt to extend our research to other textures.

References

[1] J. Stam, "Turbulent wind fields for gaseous phenomena," in *Proc. SIGGRAPH*, 1993, pp. 369–376.
[2] N. Foster and R. Fedkiw, "Practical animation of liquids," in *Proc. SIGGRAPH*, 2001, pp. 23–30.
[3] D. Q. Nguyen, R. Fedkiw, and H. W. Jensen, "Physically based modeling and animation of Fire," *ACM Trans. Graph.*, vol. 21, no. 3, pp. 721–728, 2002.
[4] A Warping Tool. [Online]. Available: http://www.debugmode.com/winmorph/.
[5] L. S. Jensen and R. Golias, "Deep-water animation and rendering," in *Proc. Game Developer's Conf. Europe'01*, 2001. [Online]. Available: http://www.gamasutra.com/gdce/2001/jensen/jensen_01.htm.
[6] A. Fournier and W. T. Reeves, "A simple model of ocean waves," *Comput. Graph.*, vol. 20, pp. 65–74, 1986.
[7] Levin, D. Lischinski, and Y. Weiss, "A closed form solution to natural image matting," in *Proc. IEEE CVPR*, 2006, pp. 61–68.
[8] A. Criminisi, P. Perez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1200–1212, Sep. 2004.
[9] A. Criminisi, I. Reid, and A. Zisserman, "Single view metrology," *Int. J. Comput. Vis.*, vol. 40, no. 2, pp. 123–148, 2000.