

Inverse Approach for Visual Simulation of Clouds

Dobashi, Yoshinori
Hokkaido University | JST CREST

<https://hdl.handle.net/2324/1430818>

出版情報 : MI lecture note series. 50, pp.78-83, 2013-10-21. 九州大学マス・フォア・インダストリ
研究所
バージョン :
権利関係 :



Inverse Approach for Visual Simulation of Clouds

Yoshinori DOBASHI

Hokkaido University, Japan/JST CREST

Clouds play an important role for creating realistic images of outdoor scenes. There are two important factors in synthesizing realistic images, that is, shapes and colors of clouds. Many methods have therefore been proposed for modeling and rendering clouds. One of the promising approaches is to numerically simulate the actual physical phenomena. However, realistic images cannot be generated unless the user chooses appropriate parameters involved in the numerical simulation, which is not an easy task. The shapes and colors of the simulated clouds depend on many parameters and it is generally difficult and time-consuming to adjust those parameters manually. This paper presents an inverse approach to address this problem. For cloud shapes, we present a method for controlling the simulation of cloud formation so that the simulated shapes become similar to those specified by the user. For colors of clouds, a method for automatically adjusting the parameters for computing realistic colors by using user-specified photographs of real clouds is presented.

1. INTRODUCTION

Clouds are important elements when synthesizing images of outdoor scenes to enhance realism. Many methods have therefore been proposed for visual simulation of clouds [6, 9, 4, 8, 7, 1, 10]. These methods are used in many applications such as flight simulators, movies, computer games, and so on. With the rapid development of the computers, recent researches focus on the numerical simulation of the actual physical phenomena. However, one of the problems is that it is often difficult to achieve desired appearances of the clouds. The appearances of the clouds are determined by the shapes and colors. Animators need to adjust many non-intuitive parameters manually by a trial-and-error process. The expensive computation cost of the numerical simulation makes this process much more difficult. One of the solutions to this problem is to accelerate the simulation process. This can be considered as a fast solution to the forward problem: the corresponding output image is computed in real-time using the given parameters, allowing one to efficiently find the appropriate parameters that produce a desired appearance. However, even using this approach, a repetitive trial-and-error process is still required until satisfactory results are obtained. Our aim is to remove the manual trial-and-error process by solving an inverse problem.

For the shapes of clouds, we present a method for controlling the simulation of the cloud formation process [5]. In this method, the user specifies the contours of the clouds viewed from a specific camera position. The simulation process is controlled in order to form the desired shape of the clouds. One straight forward approach to achieve this goal is to apply the previous methods for controlling smoke or water to clouds. However we found that this approach did not produce convincing results. The reason is that there are several physical processes, such as the phase transition from water vapor to water droplets (i.e. clouds), that are not present in other phenomena.

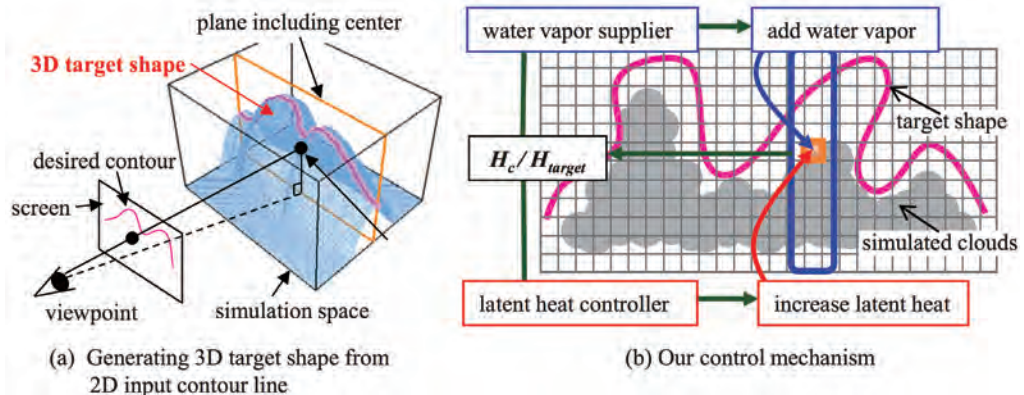


FIGURE 1. Overview of our control method.

We therefore developed a new method that controls the physical parameters affecting the cloud formation process. This results in natural cloud shapes and motion.

For the colors of clouds, we present a method for automatically adjusting the parameters such that the colors of the synthetic clouds are similar to a specified photograph of real clouds [3]. Our purpose is not to estimate physically correct parameters but to find the parameters that can produce an image that is visually similar to the clouds in the input photograph. We take into account important optical phenomena affecting the colors of clouds such as scattering and absorption of light inside the clouds. We use a color histogram to measure the visual difference between the synthetic image and the photograph. Solving the inverse rendering problem, however, is not trivial because the intensity of clouds is a highly nonlinear function of the parameters used to render them. Furthermore, there is seldom a unique solution to this problem: many different sets of parameters can produce similar images. We chose genetic algorithms (GAs) to address this problem because of their two capabilities: 1) they can find the optimal parameters efficiently even for such a highly nonlinear problem and 2) they can find a number of candidates for the optimal parameters during the optimization process.

In the following, we briefly explain these two methods and show some examples to demonstrate the effectiveness of our inverse approach.

2. CONTROLLING SIMULATION OF CLOUD FORMATION

There are two requirements in designing our method for the inverse modeling of realistic cloud shapes: (1) realistic shapes of clouds have to be generated and (2) the shapes should closely match to the desired shape specified by the user. For the first requirement, we employ numerical simulation of cloud formation processes based on the atmospheric fluid dynamics. For the second requirement, we use a feedback control mechanism to automatically adjust some of the simulation parameters.

The physical processes for the cloud formation are as follows. First, where there are no clouds in the sky, the ground is heated by the sun. Then, the air near the ground is heated and air parcels start to move upward due to the thermal buoyance forces. The temperature of the rising air parcels decreases due to adiabatic cooling, so vapor in the air parcels causes a phase transition, coagulates, and water droplets are generated. The water droplets are perceived as the cloud. At that time when the phase

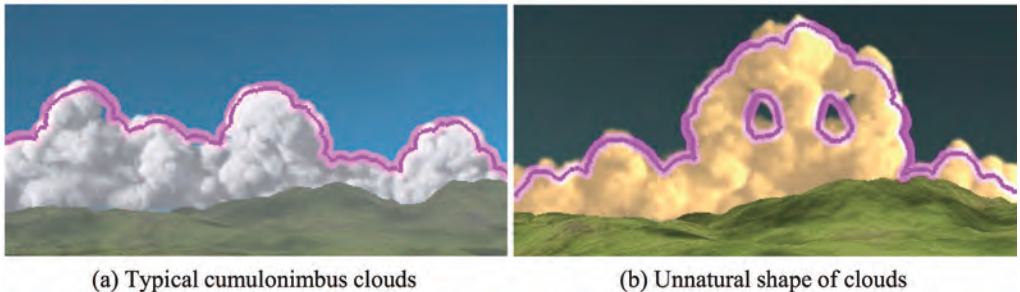


FIGURE 2. Examples of clouds generated by our control method.

transition occurs, the latent heat is liberated, which creates additional buoyancy forces and promotes further growth of the clouds. These processes can be expressed by five partial differential equations [5]. Two of them are the NS equations. The other three correspond to temperature, water vapor, and clouds. By simulating these processes numerically, realistic animation of the cloud formation can be created.

In order to control the simulation, the user specifies a contour line of the desired shape of clouds as indicated by the pink curve in Fig. 1(a). Then a three-dimensional target shape is generated from the contour line. The simulation is controlled so that the difference between the target shape and the simulated clouds become zero. We developed two controllers, a latent heat controller and a water vapor supplier, to automatically adjust the amount of the latent heat and the amount of water vapor to form the target shape. We chose the latent heat and the water vapor as control variables based on our experimental investigation. Other parameters remain fixed throughout the simulation. The control mechanism is shown in Fig. 1(b). To measure the difference between the target shape and the simulated clouds, the height ratio of the height of the simulated clouds H_c to the height of the target shape H_{target} is calculated. The height ratio is fed back into the latent heat controller and the water vapor supplier. The latent heat controller increases the latent heat and the water vapor supplier adds water vapor in the regions where the clouds have not reached the top of the target shape. By combining these controllers, the vertical development of the clouds and the generation of clouds are controlled until the target shape is formed. An important aspect of our control mechanism is that the simulation is *implicitly* controlled. No external forces nor cloud densities are explicitly generated. This prevents our controller from destructing the cloud dynamics and results in realistic cloud formation.

Fig. 2 shows the clouds generated by our method. The shape of the clouds generated by our method is almost the same as the desired shape indicated by the pink curves. Fig. 2(a) shows a typical shape of cumulonimbus clouds generated by using our control method. In Fig. 2(b), the user specifies an unnatural shape of clouds, a skull. Our method can successfully generate realistic clouds even for this unnatural shape.

3. AUTOMATIC ADJUSTMENT OF PARAMETERS FOR RENDERING CLOUDS

In order to display realistic clouds, the light scattering inside clouds need to be simulated. However, realistic images are not generated unless the user specifies good parameters used for the simulation of the light scattering. Our method addresses this problem by solving an inverse problem. That is, we let the computer search for the parameters that produce the realistic image. However, in order to do this, we have

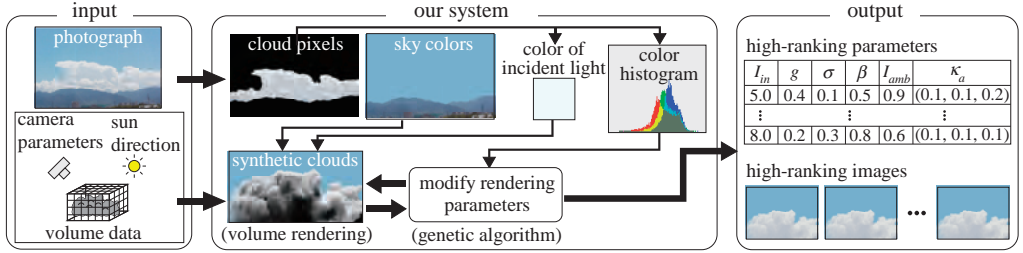


FIGURE 3. Overview of our method for automatic adjustment of cloud colors.

to tell the computer about the definition of the realism. In our approach, we use a photograph of real clouds and let the computer search for the parameters so that the appearance of the synthetic clouds look similar to those in the photograph.

An overview of our system is illustrated by Fig. 3. The inputs to our system are volume data representing the density distribution of synthetic clouds, and a photograph of real clouds. The direction of the sunlight and the camera parameters used to render synthetic clouds also need to be specified by the user. Our system then searches for the optimal parameters that minimize the following objective function O :

$$(1) \quad \arg \min_{\mathbf{c}} O(\mathbf{I}_{cg}(\mathbf{c}), \mathbf{I}_{usr}),$$

where \mathbf{c} is a vector consisting of the parameters used for rendering the synthetic image \mathbf{I}_{cg} . \mathbf{I}_{usr} is the photograph specified by the user. The objective function O measures the visual difference between \mathbf{I}_{cg} and \mathbf{I}_{usr} . We use color histograms to compute the visual differences. O is defined by:

$$(2) \quad O = \frac{1}{3} \sum_{\lambda=R,G,B} \sum_{n=0}^{n_L-1} |h_{cg}(n, \lambda) - h_{usr}(n, \lambda)|,$$

where λ is the wavelength sampled at the wavelength corresponding to the RGB color channels, n_L is the number of intensity levels, and h_{cg} and h_{usr} represent histograms of \mathbf{I}_{cg} and \mathbf{I}_{usr} , respectively. h_{cg} and h_{usr} are normalized by dividing them by the number of pixels. These histograms are computed using only the pixels corresponding to the clouds.

The intensity of clouds in the synthetic image \mathbf{I}_{cg} is calculated based on the rendering equations for the clouds [9, 2, 11]. The intensity of clouds depends on many parameters, such as the intensities of the sunlight and the skylight, and the optical properties of atmospheric and cloud particles. In our method, the only light source illuminating the clouds is the sun. However, the intensity of light directly reaching the viewpoint from the sky behind the clouds is taken into account. The attenuation and scattering of light due to atmospheric particles between the clouds and the viewpoint are also taken into account.

The minimization problem defined above is solved by rendering the clouds repeatedly with various parameter settings using GAs. To render clouds, we take into account both single and multiple scattering. The scattering and absorption due to atmospheric particles between the clouds and the viewpoint are also taken into account. We employ the simplest model where the density of the atmospheric particles is assumed to be uniform and the intensity of scattered light due to atmospheric particles is assumed

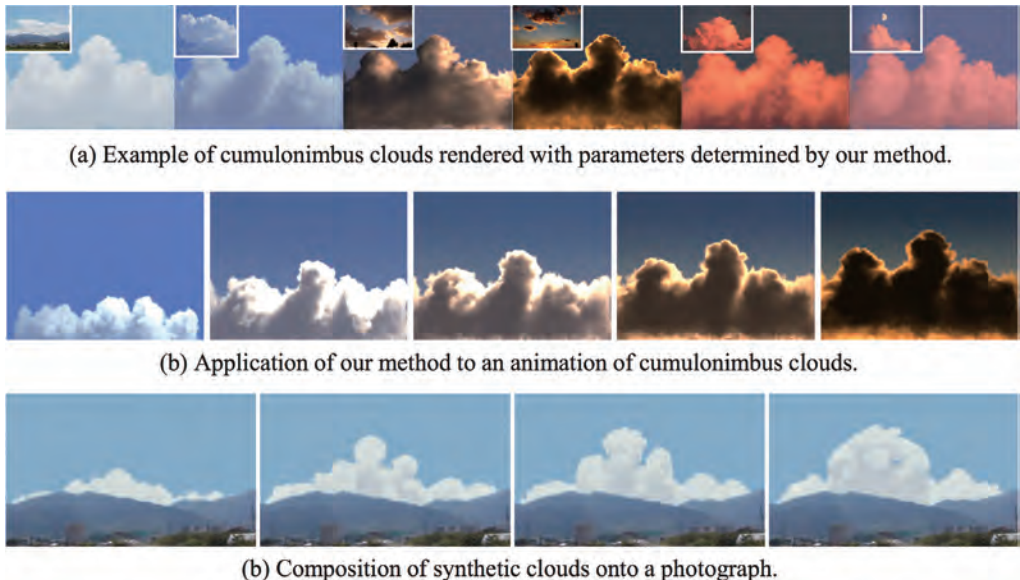


FIGURE 4. Examples of our method for adjusting parameters for rendering clouds.

to be constant. Under these assumptions, the intensity of light reaching the viewpoint for a pixel is a blended intensity of the clouds and the sky behind the clouds. Before using GAs, our system extracts cloud pixels from the input photograph and estimates the color of the incident light $c_{in}(\lambda)$ and the intensity of the sky behind the clouds $L_{sky}(q, \lambda)$. The color of the incident light is different from that of the sun because the sunlight is attenuated and scattered by atmospheric particles before reaching the clouds. The color histogram of the input photograph is also calculated using the extracted cloud pixels. The rest of the parameters are then estimated using GAs. The images of the synthetic clouds are repeatedly created by using volume rendering techniques with different parameter settings. GAs compute the objective function for each of the candidate parameter sets to measure the quality of the parameters and modify the parameters. Each set of parameters is ranked by the objective function and high-ranking parameter sets are stored. The output of our system is a set of high-ranking parameters and their corresponding images.

Fig. 4 shows examples of clouds rendered by using our method. Fig. 4(a) shows an example of cumulonimbus clouds generated by fluid simulation [8]. The inset in each image is the input photograph of the clouds. By estimating the parameters for rendering the clouds, the subtle color variations observed in the photograph are reproduced in the synthetic clouds. Fig. 4(b) shows an application of our method to create an animation of dynamic clouds. We used two parameters to render the daytime and the sunset clouds in Fig. 4(a) to create the animation with the position of the sun changing. The parameters were linearly interpolated. Fig. 4(b) shows snapshots from the animation. Realistic color transitions are realized. Fig. 4(c) shows an example of unnatural clouds. The clouds were generated using controlled simulation described in the previous section. In this example, we replaced the real clouds in the input

photograph with the synthetic clouds, rendered using the optimized parameters. The synthetic clouds are naturally composited onto the real photograph.

4. CONCLUSION

We have presented an inverse design approach for visual simulation of clouds. Our approach takes two-dimensional information as input and computes three-dimensional information to create realistic images of clouds. For the cloud shapes, two-dimensional contour line of the desired shapes of clouds is used to generate three-dimensional shapes of clouds. For the cloud colors, a photograph of real clouds is used to find optimal parameters that can produce realistic images of the synthetic clouds. These inverse problems are highly nonlinear and hard to solve. We used the feedback control mechanism and genetic algorithms. The future work includes extension of the methods to visual simulation of other natural phenomena such as fire, smoke, water, and so on.

REFERENCES

- [1] Antoine Bouthors, Fabrice Neyret, Nelson Max, Eric Bruneton, and Cyril Crassin. Interactive multiple anisotropic scattering in clouds. In *Proceedings of ACM Symposium on Interactive 3D Graphics and Games*, pages 173–182, 2008.
- [2] E. Cerezo, F. Perez, X. Pueyo, F. J. Seron, and F. X. Sillion. A survey on participating media rendering techniques. *The Visual Computer*, 21(5):303–328, 2005.
- [3] Yoshinori Dobashi, Wataru Iwasaki, Ayumi Ono, Tsuyoshi Yamamoto, Yonghao Yue, and Tomoyuki Nishita. An inverse problem approach for automatically adjusting the parameters for rendering clouds using photographs. *ACM Transactions on Graphics*, 31(6):Article 145, 2012.
- [4] Yoshinori Dobashi, Kazufumi Kaneda, Hideo Yamashita, Tsuyoshi Okita, and Tomoyuki Nishita. A simple, efficient method for realistic animation of clouds. In *Proceedings of ACM SIGGRAPH 2000*, pages 19–28, 2000.
- [5] Yoshinori Dobashi, Katsutoshi Kusumoto, Tomoyuki Nishita, and Tsuyoshi Yamamoto. Feedback control of cumuliiform cloud formation based on computational fluid dynamics. *ACM Transactions on Graphics*, 27(3):Article 94, 2008.
- [6] Geoffrey Y. Gardner. Visual simulation of clouds. *Computer Graphics (Proceedings of SIGGRAPH 1985)*, 19(3):297–304, jul 1985.
- [7] Mark J. Harris and Anselmo Lastra. Real-time cloud rendering. *Computer Graphics Forum*, 20(3):76–84, 2001.
- [8] Ryo Miyazaki, Yoshinori Dobashi, and Tomoyuki Nishita. Simulation of cumuliiform clouds based on computational fluid dynamics. In *Proceedings of EUROGRAPHICS 2002 Short Presentations*, pages 405–410, aug 2002.
- [9] Tomoyuki Nishita, Yoshinori Dobashi, and Eihachiro Nakamae. Display of clouds taking into account multiple anisotropic scattering and sky light. In *Proceedings of ACM SIGGRAPH 1996*, pages 379–386, 1996.
- [10] Yonghao Yue, Kei Iwasaki, Bing-Yu Chen, Yoshinori Dobashi, and Tomoyuki Nishita. Unbiased, adaptive stochastic sampling for rendering inhomogeneous participating media. *ACM Transactions on Graphics*, 29(6):Article 177, 2010.
- [11] Kun Zhou, Zhong Ren, Stephen Lin, Hujun Bao, Baining Guo, and Heung-Yeung Shum. Real-time smoke rendering using compensated ray marching. *ACM Transactions on Graphics*, 27(3):Article 36, 2008.