Pacific Graphics (2018)
H. Fu, A. Ghosh, and J. Kopf (Editors)

Short Paper

Modeling Detailed Cloud Scene from Multi-source Images

Yunchi Cen[1] Xiaohui Liang[1] Junping Chen[1] Bailin Yang[2] Frederick W. B. Li[3]

[1]State Key Laboratory of Virtual Reality Technology and System, Beihang University, Beijing 100191, China
[2]School of Computer Science and Information Engineering, Zhejiang Gongshang University, Hangzhou 310018, China
[3]Department of Computer Science, University of Durham, Durham, DH1 3LE, United Kingdom

Abstract

Realistic cloud is essential for enhancing the quality of computer graphics applications, such as flight simulation. Data-driven method is an effective way in cloud modeling, but existing methods typically only utilize one data source as input. For example, natural images are usually used to model small-scale cloud with details, and satellite images and WRF data are used to model large scale cloud without details. To construct large-scale cloud scene with details, we propose a novel method to extract relevant cloud information from both satellite and natural images. Experiments show our method can produce more detailed cloud scene comparing with existing methods.

CCS Concepts

•***Computing methodologies*** → ***Modeling methodologies; Shape modeling;***

1. Introduction

Being able to model natural cloud scenes can significantly improve the realism of many computer graphics applications, such as flight simulation, city walkthrough and cultural heritage site simulation. Existing cloud modeling methods can be divided into three classes based on the type of data used, namely natural image [DSY10, YLH*14], satellite image [DNYO98, DYN09, ZLYL17], and WRF (Weather Research and Forecasting Mode) data. Methods based on natural images can generate clouds with detailed surface, but being incapable of constructing large-scale cloud scenes, which are typically required by practical applications. Methods based on satellite images can easily model large-scale cumulus cloud scene due to its extensive spatial coverage. However, the limited spatial resolution (hundreds to tens of meters) makes it difficult to recover cloud surface details.

To model a large-scale cloud scene with detailed cloud surface incorporated, we propose a novel data-driven framework using both satellite images and natural images. Our main contributions are:

- An novel cloud modeling framework based on natural images and satellite images.
- A cGAN with improved object function for generating natural cloud images based on the contours of coarse cloud models.

2. Related Work

Realistic natural phenomena and scenes, such as smoke, fire, fluid and cloud, are challenging to reconstruct. In the area of cloud modeling, Dobashi et al. [DSY10] proposed a method for modeling small-scale cloud scene. Yuan et al. [YLH*14] alternatively utilized a simplified single scattering model to recover cloud shape. These methods could only model a single cloud or a small cloud scene. In contrast, Dobashi et al. [DNYO99] modeled large-scale cloud scenes using metaballs as well as adopting a single scattering model. In 2009, Dobashi et al. [DYN09] proposed another method for modeling large-scale cloud scenes by constructing cloud surface from infrared image. Recently, Zhang et al. [ZLYL17] proposed a modeling framework based on high-resolution landsat8 images. Despite this method surpassed [DYN09] by reconstructing physically more realistic and visually better results, the generated cloud scenes were still lacking cloud surface details.

To tackle the problems, we propose a novel cloud modeling framework, utilizing high-resolution feature of natural images and large-scale, multi-band feature of satellite images to model large-scale cumulus cloud scenes with detailed cloud surfaces.

3. Approach

In this section, we present the framework of our work. As shown in Figure 1, we first generate a normalized natural cloud image dataset (section 3.1). We then model coarse 3D cloud models from satellite images following by generating the corresponding natural cloud images for each coarse 3D cloud model (section 3.2). The surface details of each generated natural cloud image are then modeled and transferred onto the surface of the corresponding coarse cloud model (section 3.3).

DOI: 10.2312/pg.20181278

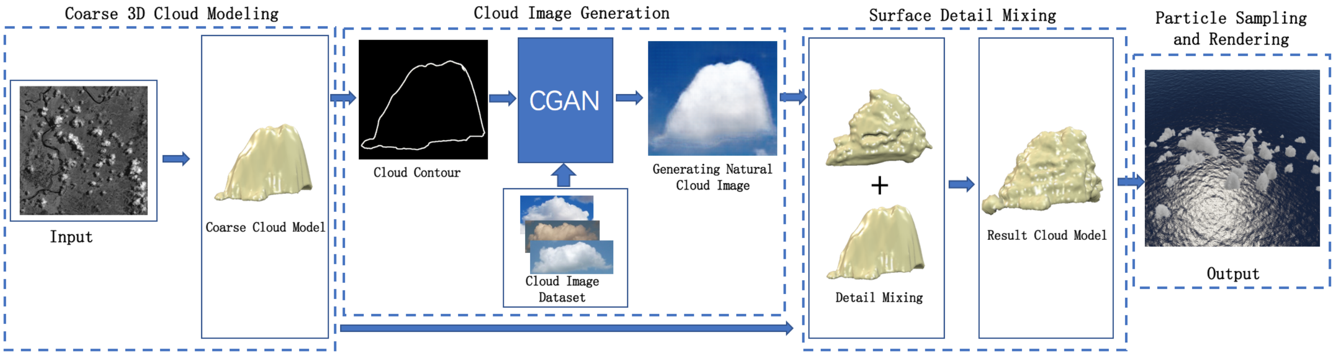

Figure 1: *Overview of our framework.*

3.1. Natural Cloud Image Dataset Generation

In order to model detailed 3D cloud scene, we use cGAN to generate natural cloud images from the contours of 3D coarse clouds modeled from satellite image (section 3.2), and construct a natural cloud image dataset to train the cGAN. We collect natural cloud images from crowdsourcing. Each natural cloud image is checked manually, ensuring that gathered natural cloud images meet the requirements. We then extract individual non-occluded clouds from each natural image, generating the natural cloud image dataset. Finally, we obtain a natural cloud image dataset, where each image in the dataset contains only a single cloud.

3.2. A Contour-based Natural Cloud Image Generation

We take both cloud information from satellite images and natural cloud images as inputs for cloud scene modeling. The main problem involved is that there is no native correspondence between the natural cloud images and the coarse 3D cloud models. We address the problem as a contour-to-image generation problem, generating natural cloud images based on the contour of coarse 3D cloud model. Surface details of each generated natural cloud image are then modeled and transferred onto the surface of the corresponding coarse cloud model. For this purpose, we adopt the method proposed by Zhang et al. [ZLYL17] to generate coarse 3D cloud models from satellite images.

After the coarse 3D cloud model scene has been reconstructed from the satellite images, for each coarse cloud model, we sample viewpoints at 180° apart. When the viewpoint have been sampled, we render the cloud model at these viewpoints. The canny operator is then used and contours of the cloud in the images are extracted.

To address the generation problem, we adopt the conditional generative adversarial network (cGAN) and improve the object function to generate natural cloud images based on the contours of coarse cloud models. cGAN is a variation of generative adversarial network (GAN) [GPAM*14], which learns a conditional generative model [MO14]. It consists of two main components: a generator $G(x,z)$ and a discriminator $D(x,y)$, where x represents an image contour extracted from the natural cloud image dataset, z represents a Gaussian noise vector, and y represents an image generated by the generator.

The main idea is to devise a system where generated natural cloud images produced by the generator become indistinguishable from real cloud images. The objective of a conditional GAN can be expressed as:

$$L_{cGAN}(G,D) = \mathbf{E}_{x,y}[\log D(x,y)] + \mathbf{E}_{x,z}[\log(1 - D(x,G(x,z)))] \quad (1)$$

where generator G tries to minimize this objective against an adversarial D that tries to maximize it. Previous approaches have found it beneficial to mix the GAN objective with a more traditional loss, such as L1 distance [IZZE17]. To generate more realistic cloud image, we adopt a total variation (TV) L_1 norm, since it can stimulate the high frequency component of the sensitivity map.

The final objective function, L_{final} is defined as:

$$L_{final} = \arg\min_G \max_D L_{cGAN}(G,D) + L_{l1}(G) + TV_G \quad (2)$$

Besides, $L_{l1}(G)$ is defined as:

$$L_{l1}(G) = \mathbf{E}_{x,y,z}[\| y - G(x,z) \|_1] \quad (3)$$

The TV_G regularization as:

$$TV_G(s) = \frac{1}{H \cdot W} \sum_{i,j} (sobel_h(s)^2 + soble_v(s)^2)^{\beta/2} \quad (4)$$

where H and W are the height and width of the last feature map in the decoder of the generator, respectively, and i and j are the pixel coordinates on the feature map.

We adopt a modified version of the "U-nets" [RFB15] as a generator. The generator contains 16 CNN blocks, where 8 of those are used for encoding and the remaining 8 are used for decoding. The generator architecture is shown in Figure 2.

Similarly, we adopt a CNN based classifier "patchGAN" [IZZE17] as a discriminator. During the training of the proposed method, we follows the standard approach [IZZE17, GPAM*14].

3.3. Surface Detail Blending

In order to extract the details of a cloud image, we use the method proposed by Yuan et al. [YLH*14] to model cumulus cloud from a single image. This method can generate a fine cloud model closely resembling an input cloud image as well as its cloud surface details

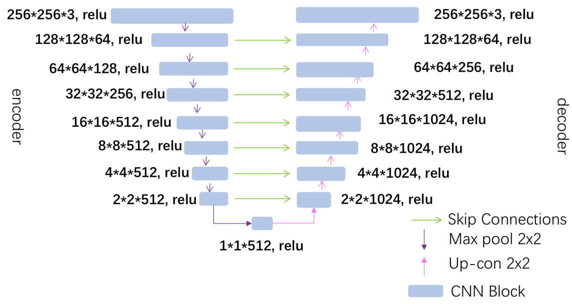

Figure 2: *The generator architecture.*

geometrically. We refer the reconstructed model by this method as the fine cloud model in the following. Table 1 summarizes the main notations used in the following.

Table 1: *Main notations*

Symbol	Definition
S	a fine cloud model
$\tilde{S}$	a smoothed model associate with S
U	a corresponding coarse cloud model to S
δ_i, $\tilde{\delta}_i$	the Laplacian coordinates of the vertex i in S and $\tilde{S}$, where $\delta_i \in \delta$, $\tilde{\delta}_i \in \tilde{\delta}$
v_i	the position in R^3 for vertex i
N_i	the neighborhood ring of vertex i
d_i	the number of vertexes in N_i
(u,v)	the (u,v) pair represents the vertex of cloud model projected into the cylinder surface
ξ_i	the details of vertex i is peeled from S
$\xi_i^{'}$	the details of vertex i is rotated by R_i, where $\xi_i^{'} \in \xi^{'}$
R_i	a rotation operation of vertex i
L	the Laplacian matrix of U
$U^{'}$	a model with transferred details

Instead of directly processing absolute coordinates of the cloud models, we describe their geometry using a set of differentials $\Delta = \{\delta_i\}$ called Laplacian coordinates [Sor05] as these coordinates allowing us to detach fine geometrical details from the fine geometry of a 3D model. The Laplacian coordinate δ_i of vertex i is defined as the difference between v_i and the average of its neighbors:

$$\delta_i = v_i - \frac{1}{d_i} \sum_{j \in N_i} v_j \tag{5}$$

We smooth S in order to get a low-frequency model $\tilde{S}$ associated with S. We smooth the fine cloud model as follows:

$$v_i = \frac{1}{d_i} \sum_{j \in N_i} v_j \tag{6}$$

Our smooth approach filters out the high-frequency details on the model surface yet the connectivity of the model is reserved. Then the Laplacian coordinates δ_i and $\tilde{\delta}_i$ of the vertex i in S and $\tilde{S}$ are calculated. The detail of the model surface S is therefore defined by $\xi_i = \delta_i - \tilde{\delta}_i$. Now the detail is encoded and peeled from the fine cloud model.

To apply detail blending, we need to define a mapping between S and U. We have experimented with several mapping methods, finding that the cylindrical mapping method yields the best result. Based on this, we put S and U in the same space by linear transformation, they are normalized to the same scale and aligned by the viewpoint gained in Section 3.2. We then project both models to be surrounded by a cylinder:

$$\begin{cases} u = y \\ v = \arccos(x/\sqrt{x^2+z^2}) \end{cases}$$

Thus the vertexs are projected into a (u,v) plane, each of the (u,v) pair represents a point on the cylinder surface. For each (u,v) pair that is projected from the coarse cloud model, we find the nearest neighbor on the (u,v) plane as the corresponding vertex of the fine cloud model. Then the details ξ_i should be rotated with respect to the target to compensate for the different local surface orientations of corresponding points in the source and target models $\xi_i^{'} = R_i(\xi_i)$. Having all the R_i associated with the ξ_i, the coating transfer from S onto U is expressed as follows:

$$U^{'} = L^{-1}(\delta + \xi^{'}) \tag{7}$$

Now we have a detailed cloud model.

4. Experiments

This section shows results of natural cloud images generation and cloud surface details transfer, as well as a comparison with Zhang's method [ZLYL17] on cloud modeling.

4.1. Natural Image Generating

We use cGAN to generate natural images. To train the cGAN, the natural cloud image dataset is divided into 240 training images and 75 testing images, with corresponding contour images. The original size of all these images is 256×256. Each natural cloud image and corresponding contour image are merged into a single image with the size of 256×512. We set batch size to 1 and learning rate to 0.0002. Each CNN block uses batch normalization regularization and L1+TV regularization. The generated natural images are shown in Figure 3.

4.2. Comparison of Cloud Scene Modeling methods

We compare our modeling results with [ZLYL17]. Figure 4 shows some examples of the mesh models generated by Zhang and detailed cloud models generated by our method. From the experiment results, with our method, surface details are accurately mapped and transferred onto the corresponding coarse cloud models.

Figure 5 shows the comparisons of the clouds modeled and rendered via Zhang's method and the same clouds rendered via our method. Figure 6 shows the rendering results of a large-scale cumulus cloud scene generated by our method.

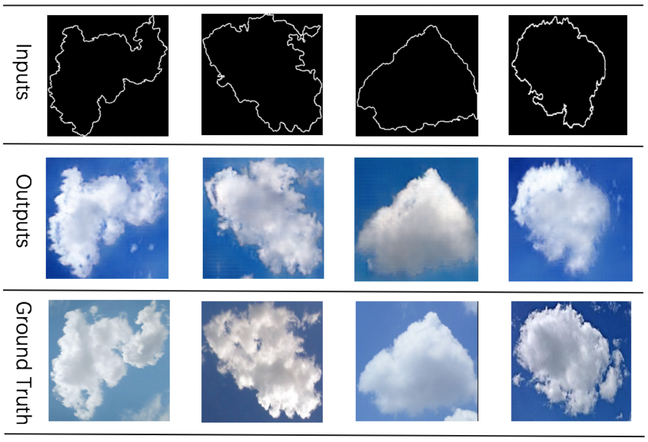

Figure 3: *Different input cloud image contours extracted from natural cloud image dataset, the generator synthesized realistic cloud images (outputs).*

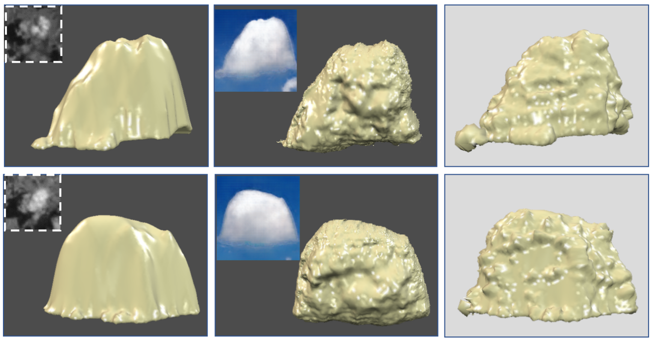

Figure 4: *Comparison of mesh models. Images on the left side are coarse cloud models generated by Zhang, while images in the middle column are corresponding fine cloud models, and images on the right side are our results.*

5. Future Work

In the future, we are interested in improving the natural image generation process. Current cloud image generation method only uses the boundary of a cloud as prior. With the progress on adversarial network, more cues can be applied to the process.

6. Acknowledgments

This work was supported in part by the National Key R&D Program of China under Grant 2017YFB1002702, in part by the National Nature Science Foundation of China under Grant 61572058, and in part by the National Natural Science Foundation of China under Grant 61472363.

(a) **(b)**

Figure 5: *Comparison of modeling and rendering results. (a) clouds modeled and rendered by Zhang's method; (b) same clouds modeled and rendered by our method.*

Figure 6: *Large-scale cloud scene modeling by our method.*

References

[DNYO98] DOBASHI Y., NISHITA T., YAMASHITA H., OKITA T.: Modeling of clouds from satellite images using metaballs. In *Computer Graphics and Applications, 1998. Pacific Graphics' 98. Sixth Pacific Conference on* (1998), IEEE, pp. 53–60. 1

[DNYO99] DOBASHI Y., NISHITA T., YAMASHITA H., OKITA T.: Using metaballs to modeling and animate clouds from satellite images. *The Visual Computer 15*, 9 (1999), 471–482. 1

[DSY10] DOBASHI Y., SHINZO Y., YAMAMOTO T.: Modeling of clouds from a single photograph. In *Computer Graphics Forum* (2010), vol. 29, Wiley Online Library, pp. 2083–2090. 1

[DYN09] DOBASHI Y., YAMAMOTO T., NISHITA T.: Interactive and realistic visualization system for earth-scale clouds. In *Pacific graphics* (2009). 1

[GPAM*14] GOODFELLOW I., POUGET-ABADIE J., MIRZA M., XU B., WARDE-FARLEY D., OZAIR S., COURVILLE A., BENGIO Y.: Generative adversarial nets. In *Advances in neural information processing systems* (2014), pp. 2672–2680. 2

[IZZE17] ISOLA P., ZHU J.-Y., ZHOU T., EFROS A. A.: Image-to-image translation with conditional adversarial networks. *arXiv preprint* (2017). 2

[MO14] MIRZA M., OSINDERO S.: Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784* (2014). 2

[RFB15] RONNEBERGER O., FISCHER P., BROX T.: U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (2015), Springer, pp. 234–241. 2

[Sor05] SORKINE O.: Laplacian mesh processing. In *Eurographics (STARs)* (2005), pp. 53–70. 3

[YLH*14] YUAN C., LIANG X., HAO S., QI Y., ZHAO Q.: Modelling cumulus cloud shape from a single image. In *Computer Graphics Forum* (2014), vol. 33, Wiley Online Library, pp. 288–297. 1, 2

[ZLYL17] ZHANG Z., LIANG X., YUAN C., LI F. W.: Modeling cumulus cloud scenes from high-resolution satellite images. *Computer Graphics Forum 36*, 7 (2017), 229–238. URL: `http://dx.doi.org/10.1111/cgf.13288`, `doi:10.1111/cgf.13288`. 1, 2, 3