Generating Various Flow Fields using Principal Component Analysis

Syuhei Sato*
Hokkaido University

Yoshinori Dobashi
Hokkaido University/JST CREST

Kei Iwasaki
Wakayama University/UEI Research

Hiroyuki Ochiai
Kyushu University/JST CREST

Tsuyoshi Yamamoto
Hokkaido University

Tomoyuki Nishita
UEI Research/Hiroshima Shudo University

1 Introduction

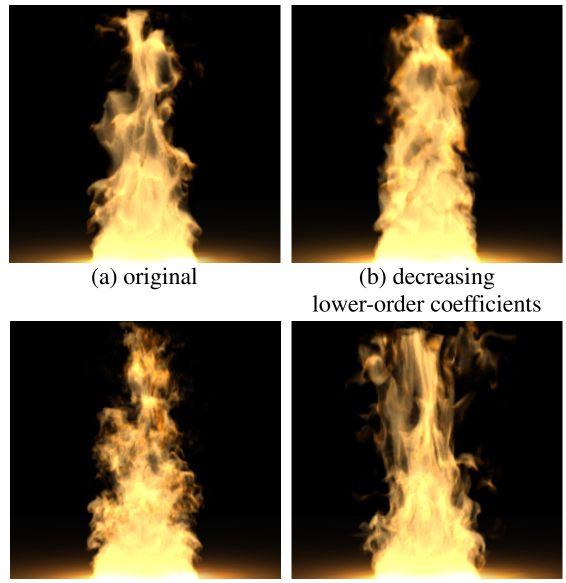
The visual simulation of fluids has become an important element in many applications, such as movies and computer games. In these applications, large-scale fluid scenes, such as fire in a village, are often simulated by repeatedly rendering multiple fluid flows. In these cases, animators are requested to generate many variations of a fluid flow. Previously, we developed a method to help animators meet such requirements [Sato et al. 2013]. However, the method was limited to 2D fluid simulation. In the previous method [Sato et al. 2013], we use Laplacian eigenfunctions as the basis functions. However, Laplacian eigenfunctions are too expensive in computation and storage costs for 3D fluid simulation. Furthermore, Laplacian eignfunctions force us to use slip boundary conditions and this makes the method less practical. In order to address these problems, we introduce Principal Component Analysis (PCA) for the basis functions. In generating the principal components of the input velocity fields, we use a subspace approach [Treuille et al. 2006; Kim and Delaney 2013]. The variations are generated by modulating the coefficient of each principal component.

2 Our Method

First, our method precomputes basis functions from the input velocity fields. We compute basis functions by applying PCA to input velocity fields. After constructing a matrix consisting of input velocity fields $\mathbf{V} = [\mathbf{v}_0, \mathbf{v}_1, \cdots, \mathbf{v}_{M-1}]$, principal components of the input velocity fields $\mathbf{P} = [\mathbf{\Phi}_0, \mathbf{\Phi}_1, \cdots, \mathbf{\Phi}_{N-1}]$ are calculated, where $\mathbf{v}_m(m = 0, 1, \cdots, M-1)$ is an input velocity field at m-th frame, M is the number of frames of the input dataset, $\mathbf{\Phi}$ is principal components, N is the number of basis functions with $M > N$. Next, we represent the input flow field $\mathbf{v}_m$ by a linear combination of principal components $\mathbf{\Phi}_i$. The coefficient w_i^m for the i-th principal component is calculated using the following equation, $w_i^m = \mathbf{v}_m \cdot \mathbf{\Phi}_i$, where $\cdot$ is the dot product between two vectors. To generate various flow fields $\mathbf{u}_m$, our method modulates w_i^m for each principal component as follows, $\mathbf{u}_m = \sum_{i=0}^{N-1} g_i w_i^m \mathbf{\Phi}_i$, where g_i represents the gain with which w_i^m is modulated.

3 Results and Future Work

Fig. 1 shows examples created using our method. The number of grid points for the velocity fields is $128 \times 128 \times 192$, the number of frames of the input velocity fields is 200, and the number of the principal components is 64. The flow field is visualized by advecting the fire temperature. As shown in these examples, our method successfully generates similar but slightly different fire. The videos corresponding to these examples and other results (smoke examples) can be found in the supplementary material.



(a) original (b) decreasing lower-order coefficients (c) increasing higher-order coefficients (d) increasing lower-order coefficients

Figure 1: *Modulating coefficients of a fire animation by our method. (a) shows the input fire animation. (b) through (d) show the results generated by modulating coefficients of original animations.*

One of the limitations of our method is the fact that the flow fields generated by our method might not conform to the laws of fluid flow, if the degree of modulation by g_i is too large. To address this problem, we calculate the Navier-Stokes equations for velocity fields generated by our method, then compare two velocity fields. In future work, we will make experiments to evaluate these flow fields.

*e-mail:sato@ime.ist.hokudai.ac.jp

References

KIM, T., AND DELANEY, J. 2013. Subspace fluid re-simulation. *ACM Transactions on Graphics 32*, 4, Article 62.

SATO, S., DOBASHI, Y., IWASAKI, K., OCHIAI, H., AND YAMAMOTO, T. 2013. Generating flow fields variations by modulating amplitude and resizing simulation space. In *Proceedings of ACM SIGGRAPH ASIA 2013 Technical briefs*, 13.

TREUILLE, A., LEWIS, A., AND POPOVIC, Z. 2006. Model reduction for real-time fluids. *ACM Transactions on Graphics 25*, 3, 826–834.