The Journal of the Society for Art and Science, Vol.15, No.3, pp. 111 – 119 (2016)

Waterfall Simulation with Spray Cloud in different Environments

Nobuhiko Mukai[1)] **Yuto Hizono**[1)] **Youngha Chang**[1)]

1) Graduate School of Engineering, Tokyo City University

{mukai, hizono, chang} (at) vgl.cs.tcu.ac.jp

Abstract
This paper reports the simulation of waterfall including spray cloud in different environments. Visualization of liquid behavior on physically based simulation is one of the most challenging issues in computer graphics, especially the simulation of waterfall needs huge amount of computer resources such as calculation power and much memory. Then, we have been trying to visualize the waterfall simulation by dividing the model into three parts: water stream, splashing spray, and spray cloud. Among them, the behavior of spray cloud has much effects on the waterfall appearance as well as the water stream. Therefore, we have simulated the behavior of a waterfall including spray cloud in different environments, and found that the environment in front of the waterfall affects the spray cloud behavior.

1 Introduction

There are many kinds of simulations by using computer graphics for natural phenomena. Although its reality is very important for visualization, precise simulation is also needed to validate the simulation model and to improve the realism of the image. One of the most challenging issues in computer graphics is visualization of natural phenomena, especially liquid behavior because liquid changes its shape easily, and also its boundary is so clear. Then, there have been many researches studied for visualization of liquid behavior based on physical simulation such as ocean, rivers, bubbles, and spray. However, there were not so many researches on waterfall, although there are some works limited to part of falling water.

Therefore, we have been working on the model and the method to visualize the whole behavior of a waterfall based on physical simulation. A waterfall is divided into three parts: water stream, splashing spray, and spray cloud. The water falls down from the lip of the waterfall, and splashing spray comes out of the water stream. After the water reached at the basin, spray cloud soars up from it.

In the simulation, we employ a hybrid method of particle and grid. Particle method is used for water stream and splashing spray, because the boundary of them are clear and simulation space is limited. On the other hand, grid based method is used for spray cloud, since spray cloud is vapor, and the boundary is not so clear that simulation space is not limited.

In our previous method, we tried to visualize spray cloud soaring up from the basin; however the spray cloud was not soaring so high. Therefore, in this paper, we reports the simulation results that have been performed in different envrionments, especially different slopes and heights of obstacles in front of the waterfall. We also have introduced the force generated from the air density difference and the interaction between the spray cloud and the splashing spray velocities through the air velocity.

2 Previous Works

Related to water simulation with computer graphics, Mould et al. explained strengths and shortcomings of water modeling [1], and Iglesias surveyed computer graphics techniques that has been developed during 1980s and 1990s for modeling, rendering and animation [2]. In addition, Darles et al. introduced the latest technologies for ocean simulation [3]. Each technique of the researches is as follows including other works.

Hinsinger et al. and Cui et al. proposed methods to represent the waves on ocean surface by using an adaptive mesh model [4, 5]. Dupuy and Bruneton also presented ocean scenes with whitecaps [6]. These methods basically visualize only the continuous surface of ocean so that mesh model is suitable although noncontinuous surfaces (whitecaps) appear on the surface when it breaks. Mesh model, however, is not suitable for the simulation where the topology of fluid changes such as river stream break by rocks. Then, Müller et al. used SPH (Smoothed Particle Hydrodynamics), which is one of particle methods, to represent interactive animation of pouring water [7], and Kipfer and Westermann also performed interactive simulation of rivers with SPH [8].

There are two kinds of methods for treating liquid objects: Eulerian (grid-based) and Lagrangian (particle) methods. Chentanez and Müller used Eulerian method to simulate water flow in real time with the help of GPU power [9], and Nishino et al. visualized small air bubbles that appear on freezing ice with grid-based method [10]. On the other hand, Foster and Fedkiw employed modified semi-Lagrangian method to represent liquid animation [11], and Busaryev et al. employed particle method and Voronoi diagram to visualize coke foam [12].

It is easy to represent a freely transformable object with Lagrangian (particle) method; however, it is difficult to determine the boundary of the object and some methods such as level-set are used to determine the boundary. On the other hand, Eulerian (grid-based) method needs huge amount of memory to store every physical quantity at each grid point; however, it is easy to visualize the simulation result. Then, Hong et al. proposed a hybrid method that combines particle and grid-based approaches to visualize both large scale of water and many small bubbles [13]. This hybrid method is so efficient that Chentanez and Müller used the hybrid method to visualize a large scale of flowing river along a valley [14].

There are mainly two reasons for the difficulty of fluid simulation. One is that its shape changes easily and the other is that its boundary is clearer. Particle method is very useful for handling the shape change. Then, Miller proposed particle system for animating viscous fluids [15], and Sims introduced a parallel particle rendering system to model dynamic phenomena such as wind, snow, water and fire [16]. However, it is difficult to make the boundary so clear that level-set method is used to determine the boundary. Greenwood and House used PLS (Particle Level Set) algorithm to treat bubble-bubble interfaces and represented two or three bubble clusters [17], and Geiger et al. used PLS to visualize the main body of ocean and splash [18]. In addition, Kim et al. presented bubbles staying for a long time on the water surface with level-set and volume control methods [19], and Losasso et al. proposed two-way coupled simulation framework that uses PLS and SPH to represent large scale of ocean scene [20].

Related to waterfalls, Mallinder proposed waterfall model with a method of "string texture", which is a memory space reduced from 3D to 1D [21]. Particle method needs much memory to store some parameters for each particle so that this paper described a method to store the data not explicitly but implicitly. In addition, Foster and Metaxeas developed a system with which animators can control a fluid animation without knowledge of equations for dynamic simulation [22]. On the other hand, Howes and Forrest visualized waterfalls by using particle method [23]. They represented horseshoe falls with mist; however, the simulation was not based on physics and the image was not so realistic. Sakaguchi et al. created a tool for waterfall that is based on volume rendering; however the rendering requires two passes so that it takes a lot of time to visualize waterfalls [24]. Hardie also visualized falling water with particle simulation; however, it also was not based on physical simulation but used noise function [25]. On the other hand, Takahashi et al. proposed CIP (Cubic Interpolated Propagation) method to visualize the seamless connection between splash and foam [26]. Hoetzlein and Höllerer proposed a technique for extracting surface of falling water stream combined with many particles by using sphere scan conversion [27], and Miyashita and Funahashi proposed a real-time liquid manipulation model that handles water falling from a cup in a virtual space [28]. In addition, Nielsen and Østerby visualized water spray that appears in waterfalls, water jets, and stormy seas with Eulerian two-continua simulation of air and water [29].

Waterfalls are mainly constructed with three parts: water stream, splashing spray, and spray cloud; however, the previous works did not describe the method to visualize the whole behavior of waterfalls on physically based simulation. Then, we have proposed a method to visualize the whole behavior of a waterfall based on precise physical simulation [30]. We have also proposed the method to visualize spray cloud soaring from the basin by considering the environment conditions such as vapor density and the terrain around the basin [31]. Especially, in the paper, we have realized the interaction between the splashing spray and the spray cloud by utilizing the velocity calculated in the splashing spray simulation as the velocity in the spray cloud simulation and vice versa. In this paper, we report the simulation results of the spray cloud behavior in different environments, especially different slope angles and heights of obstacles in front of the waterfall.

3 Waterfall Model

Fig.1 shows the proposed waterfall model, which is composed of three parts: water stream, splashing spray and spray cloud. The water is modeled with par-

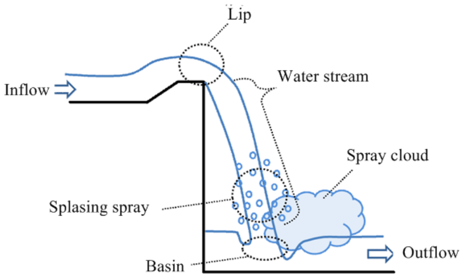


Figure 1: Waterfall model.

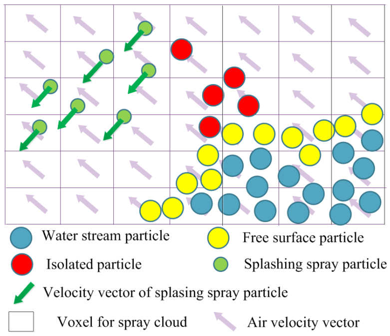


Figure 2: Particles that construct waterfall model.

ticles, and water particles flow into the lip and they fall down to the basin. The stream from the lip to the basin is the main water stream of a waterfall, and splashing spray comes out of the water stream. Spray cloud is also generated from the splashing spray.

On the other hand, Fig.2 shows particles that construct a waterfall model. The water stream is composed of three kinds of particles: water stream, free surface and isolated particles, and they are differentiated by the density as follows.

$$\rho_{sur} = \alpha_{sur}\rho_{main} \tag{1}$$

$$\rho_{iso} = \alpha_{iso}\rho_{main} \tag{2}$$

$$0 < \alpha_{iso} < \alpha_{sur} < 1 \tag{3}$$

Where, ρ_{main}, ρ_{sur} and ρ_{iso} are the densities of water stream, free surface and isolated particles, respectively. ρ_{main} keeps almost the same value because water is incompressible fluid. In order to analyze the behavior of these particles, particle method is useful [32, 33, 34] and its behavior obeys Navier-stokes equation described in the following section.

On the other hand, the splashing spray particle comes out of the isolated particle, and the isolated particle disappears after it has generated some splashing spray particles. The splashing spray particle has an initial velocity that is the same value as the isolated particle, and is affected by air velocity. The splashing spray particle is a tiny particle and its density is so small that the behavior does not obey Navier-stokes equation. In this paper, we employ equation of motion for small particle to analyze the behavior of the splashing spray [35].

The spray cloud comes out of the splashing spray particles in the water stream and/or the basin, and spreads out into larger space so that grid-based method is useful for the analysis. In this paper, we use the method of Stable Fluid to analyze the behavior of the spray cloud, because Stable Fluid gives an unconditionally stable model for complex flow so that it becomes possible to take larger time steps and achieve faster simulation [36].

4 Method

4.1 Water Stream

We employ SPH method to analyze the behavior of the water stream, which includes water stream, free surface, and isolated particles. Navier-stokes equation is shown as the following [33].

$$\frac{\partial \boldsymbol{u}}{\partial t} = -\frac{1}{\rho}\nabla p + \nu\nabla^2\boldsymbol{u} + \boldsymbol{f} \tag{4}$$

Where, $\boldsymbol{u}$ is velocity, t is time, ρ is density, p is pressure, ν is viscosity, and $\boldsymbol{f}$ is external acceleration. A physical quantity $\phi(\boldsymbol{x}_i)$, its gradient $\nabla\phi(\boldsymbol{x}_i)$ and its Laplacian $\nabla^2\phi(\boldsymbol{x}_i)$ at a position $\boldsymbol{x}_i$ are defined respectively as follows.

$$\phi(\boldsymbol{x}_i) = \Sigma_j m_j \frac{\phi_j}{\rho_j} W(\boldsymbol{x}_i - \boldsymbol{x}_j) \tag{5}$$

$$\nabla\phi(\boldsymbol{x}_i) = \Sigma_j m_j \frac{\phi_j}{\rho_j} \nabla W(\boldsymbol{x}_i - \boldsymbol{x}_j) \tag{6}$$

$$\begin{aligned}\nabla^2\phi(\boldsymbol{x}_i) &= \Sigma_j m_j \frac{\phi_j}{\rho_j} \nabla^2 W(\boldsymbol{x}_i - \boldsymbol{x}_j) \\ &= \Sigma_j m_j \frac{\nabla\phi_j}{\rho_j} \nabla W(\boldsymbol{x}_i - \boldsymbol{x}_j) \\ &= \Sigma_j m_j \frac{\phi_j - \phi_i}{\rho_j} \frac{\boldsymbol{x}_j - \boldsymbol{x}_i}{|\boldsymbol{x}_j - \boldsymbol{x}_i|} \nabla W(\boldsymbol{x}_i - \boldsymbol{x}_j)\end{aligned} \tag{7}$$

Where, m_j, ρ_j and $\boldsymbol{x}_j$ is the mass, the density and the position of a particle j, respectively. W is called kernel function and the kernel functions of density W_d, pressure W_p and viscosity W_v are defined respectively as follows [7, 34].

$$W_d(\boldsymbol{r}_{ij}) = \frac{315}{64\pi r_e^9}(r_e^2 - |\boldsymbol{r}_{ij}|^2)^3 \tag{8}$$

$$\nabla W_p(\boldsymbol{r}_{ij}) = -\frac{45}{\pi r_e^6}(r_e - |\boldsymbol{r}_{ij}|)^2 \frac{\boldsymbol{r}_{ij}}{|\boldsymbol{r}_{ij}|} \tag{9}$$

$$\nabla W_v(\boldsymbol{r}_{ij}) = \frac{45}{\pi r_e^6}(r_e - |\boldsymbol{r}_{ij}|) \tag{10}$$

Where, r_e and $\boldsymbol{r}_{ij}$ are the radius of influence and the vector from particle j to particle i, respectively. With the above equations, the density $\rho(\boldsymbol{x}_i)$ at a position of $\boldsymbol{x}_i$, the pressure term ∇p and the viscosity term $\nu\nabla^2\boldsymbol{u}$ are calculated as follows.

$$\rho(\boldsymbol{x}_i) = \Sigma_j m_j W_d(\boldsymbol{x}_i - \boldsymbol{x}_j) \tag{11}$$

$$\begin{aligned}\nabla p_i &= \Sigma_j m_j \frac{p_j}{\rho_j} \nabla W_p(\boldsymbol{x}_i - \boldsymbol{x}_j) \\ &\approx \Sigma_j m_j \frac{p_i + p_j}{2\rho_j} \nabla W_p(\boldsymbol{x}_i - \boldsymbol{x}_j)\end{aligned} \tag{12}$$

$$\nu\nabla^2\boldsymbol{u} = \nu\Sigma_j m_j \frac{\boldsymbol{u}_j - \boldsymbol{u}_i}{\rho_j} \frac{\boldsymbol{x}_j - \boldsymbol{x}_i}{|\boldsymbol{x}_j - \boldsymbol{x}_i|} \nabla W_v(\boldsymbol{x}_i - \boldsymbol{x}_j) \tag{13}$$

Where, p_i is the pressure to particle i. The pressure works between particles i and j; however, the pressure to particle i from particle j is different from the pressure to particle j from particle i so that the pressure to particle j (p_j) is approximated with the average pressure of $\frac{p_i+p_j}{2}$ in Eq.(12). In addition, p is defined with the following equation.

$$p = \begin{cases} k(\rho - \rho_0) & \rho \geq \rho_0 \\ 0 & otherwise \end{cases} \tag{14}$$

Where, k and ρ_0 are gas constant and the initial particle density. Finally, the gravity and the air resistance are considered as the external acceleration $\boldsymbol{f}$, which is defined as the following.

$$\boldsymbol{f} = \boldsymbol{g} + \frac{1}{\rho_p V_p}\boldsymbol{F}_{air} \tag{15}$$

$$\boldsymbol{F}_{air} = \frac{1}{2}\pi\left(\frac{D_p}{2}\right)^2 \rho_a C_r |\boldsymbol{u}_p| \boldsymbol{u}_p \tag{16}$$

Where, $\boldsymbol{g}$ is gravity, and ρ_p, V_p, D_p, and $\boldsymbol{u}_p$ are density, volume, diameter and velocity of a particle, respectively. In addition, ρ_a and C_r are the density of air and the coefficient of resistance, respectively. The air resistance is calculated by approximating the particle as a sphere.

4.2 Splashing Spray

In order to analyze the behavior of the splashing spray, we use equation of motion for small particle, which is described as follows [35].

$$\boldsymbol{F}_r + \boldsymbol{F}_b = \rho_p \frac{\pi}{6} D_p^3 \frac{d\boldsymbol{u}_p}{dt} \tag{17}$$

$$\boldsymbol{F}_r = \frac{1}{2}\pi\left(\frac{D_p}{2}\right)^2 \rho_a C_r |\boldsymbol{u}_a - \boldsymbol{u}_p| (\boldsymbol{u}_a - \boldsymbol{u}_p) \tag{18}$$

$$\boldsymbol{F}_b = (\rho_p - \rho_a)\frac{\pi D_p^3}{6}\boldsymbol{g} \tag{19}$$

Where, ρ_a and $\boldsymbol{u}_a$ are density and velocity of air, respectively and other parameters are the same as those in Eqs.(15) and (16). The splashing spray particle comes out of an isolated particle so that the initial velocity of the splashing spray particle has the same value as that of the isolated particle. On the other hand, the velocity of air $\boldsymbol{u}_a$ has the same value as that of the spray cloud, which is calculated in the next section.

4.3 Spray cloud

The spray cloud comes out of the water stream and/or the basin and spreads out into larger space so that we employ grid-based method to analyze the behavior. The density of the spray cloud is defined for each voxel as follows.

$$D_c(\boldsymbol{x}) = \beta n(\boldsymbol{x}) \tag{20}$$

Where, $D_c(\boldsymbol{x})$, $n(\boldsymbol{x})$ and β are density of spray cloud, number of particles in the voxel at a position $\boldsymbol{x}$, and parameter that defines the density of the spray cloud. The velocity of the voxel is also defined as the average velocity of the particles that exist in the voxel.

There are many grid-based methods; however, some does not guarantee stable behavior. Stable Fluid [36] is one of the grid-based methods and guarantees stable behavior of fluid. Then, we use Stable Fluid to analyze the behavior of the spray cloud in this paper. Navier-stokes equation for spray cloud is as follows, which is different from Eq.(4) because advection term $(\boldsymbol{u}\cdot\nabla)\boldsymbol{u}$ should be considered in grid-based method.

$$\frac{\partial \boldsymbol{u}}{\partial t} = -(\boldsymbol{u}\cdot\nabla)\boldsymbol{u} - \frac{1}{\rho}\nabla p + \nu\nabla^2\boldsymbol{u} + \boldsymbol{f} \tag{21}$$

$$\nabla\cdot\boldsymbol{u} = 0 \tag{22}$$

According to Helmholtz decomposition, any vector $\boldsymbol{w}$ can be decomposed to a vector with no divergence $\boldsymbol{v}$ and the gradient of a scalar q as follows.

$$\boldsymbol{w} = \boldsymbol{v} + \nabla q \tag{23}$$

The above equation can be written as the following with P, which is an operator that projects $\boldsymbol{w}$ to $\boldsymbol{v}$.

$$P\boldsymbol{w} = \boldsymbol{v} = \boldsymbol{w} - \nabla q \tag{24}$$

If we apply Eq.(21) to Eq.(24), we can obtain the next equation.

$$P\{-(\boldsymbol{u}\cdot\nabla)\boldsymbol{u} + \nu\nabla^2\boldsymbol{u} + \boldsymbol{f}\} = \frac{\partial \boldsymbol{u}}{\partial t} \tag{25}$$

$$= \{-(\boldsymbol{u}\cdot\nabla)\boldsymbol{u} + \nu\nabla^2\boldsymbol{u} + \boldsymbol{f}\} - \frac{1}{\rho}\nabla p \tag{26}$$

Eq.(25) does not have the pressure term $\frac{1}{\rho}\nabla p$ and this becomes the basic equation for the spray cloud.

On the other hand, we can obtain the following equations from Eqs.(24) and (26) because $\nabla \cdot \boldsymbol{v} = 0$.

$$\nabla \cdot \boldsymbol{w} = \nabla^2 q \quad (27)$$

$$\nabla \cdot \{-(\boldsymbol{u} \cdot \nabla)\boldsymbol{u} + \nu \nabla^2 \boldsymbol{u} + \boldsymbol{f}\} = \frac{1}{\rho}\nabla^2 p \quad (28)$$

Then, the pressure term can be solved with Eq.(28). Eq.(25) can also be rewritten as follows.

$$\boldsymbol{u} = P\{-(\boldsymbol{u} \cdot \nabla)\boldsymbol{u} + \nu \nabla^2 \boldsymbol{u} + \boldsymbol{f}\}dt \quad (29)$$

In order to visualize the soaring-up of spray cloud, we consider vapor density. Vapor density (ρ_v) is calculated with the following equation [37].

$$\rho_v = 1.293 \frac{273.15}{273.15 + T} \frac{p}{1013.25} (1 - 0.378 \frac{e}{p}) \quad (30)$$

Where, $1.293[kg/m^3]$ is the density of dried air, $273.15[degree]$ is a coefficient of translation from Celsius temperature to absolute temperature, $T[degree]$ is Celsius temperature, $p[hPa]$ is pressure, $1013.2[mb]$ is air pressure, 0.378 is a coefficient of translation from the gravity of dried air to the gravity of wet air, and $e[hPa]$ is vapor pressure.

The density difference generates the force that makes spray cloud move up. Then, the following acceleration is added to $\boldsymbol{f}$ in Eq.(29) as the external acceleration of lower grid.

$$(\frac{\rho_l - \rho_u}{\rho_l})\boldsymbol{g} \quad (31)$$

Where, ρ_l and ρ_u are the densities of lower and upper grids, respectively, and $\boldsymbol{g}$ is gravity. Then, the algorithm to calculate the velocity $\boldsymbol{u}$ is as follows, where the initial velocity $\boldsymbol{u}_0$ is defined as the velocity of the splashing spray ($\boldsymbol{u}_p$ in Eq.(18)). The calculated velocity $\boldsymbol{u}$ also affects the air velocity $\boldsymbol{u}_a$ in Eq.(18).

<Algorithm for spray cloud>

1. Define the initial velocity $\boldsymbol{u}_0$.
2. Calculate the velocity $\boldsymbol{u}_1$ updated by the external acceleration with $\boldsymbol{u}_1(\boldsymbol{x}, t) = \boldsymbol{u}_0(\boldsymbol{x}, t) + \boldsymbol{f}(\boldsymbol{x}, t)dt$.
3. Calculate the velocity $\boldsymbol{u}_2$ updated by the advection with $\boldsymbol{u}_2(\boldsymbol{x}, t) = \boldsymbol{u}_1(\boldsymbol{x} - \boldsymbol{u}_1 dt, t)$.
4. Calculate the velocity $\boldsymbol{u}_3$ updated by the viscosity with $\boldsymbol{u}_3(\boldsymbol{x}, t) - \nu \nabla^2 \boldsymbol{u}_3(\boldsymbol{x}, t)dt = \boldsymbol{u}_2(\boldsymbol{x}, t)$.
5. Calculate the pressure p according to Eq.(28) with $\frac{dt}{\rho}\nabla^2 p = \nabla \cdot \boldsymbol{u}_3(\boldsymbol{x}, t)$.
6. Calculate the velocity $\boldsymbol{u}_4$ updated by the pressure with $\boldsymbol{u}_4(\boldsymbol{x}, t) = \boldsymbol{u}_3(\boldsymbol{x}, t) - \frac{dt}{\rho}\nabla p$.

In addition, we have considered the terrain around the basin of the waterfall. The basin of waterfall is like a pond that is surrounded by rocks so that we have generated the environment terrain as shown in Fig.3. The simulation space is divided into 80^3 voxels. The left object in Fig.3 is the wall behind the waterfall, along which water falls down. On the other hand, the right object is a block of rocks that is placed in front of the basin of the waterfall, where $Width$ and $Height$ change according to the conditions such as different heights of the rock with the same slope angle.

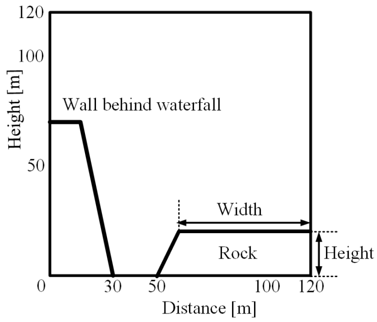


Figure 3: Terrain around the waterfall.

5 Simulations

With the above methods, we can analyze the behavior of a waterfall, which includes water stream, splashing spray and spray cloud. The water stream is also composed of three types of particles: water stream, free surface and isolated particles. Water stream, splashing spray and spray cloud are analyzed by particle method (SPH), equation of motion for small particle, and grid-based method (Stable Fluid), respectively. Here, the water stream can be calculated without the simulation results of splashing spray and spray cloud, because water stream is not affected by splashing spray and spray cloud. Therefore, we analyze the behavior of water stream first and then analyze splashing spray and spray cloud later with the simulation result of the water stream. Then, the simulation algorithm is composed of two parts of A and B as the following.

<A: Algorithm for the water stream>

A1 Initialize and set parameters.

A2 Add the water stream particles to the lip and remove others from the basin.

A3 Detect the free surface and the isolated particles.

A4 Analyze the behavior of the water stream with SPH.

A5 Calculate positions, velocities and densities of the water particles.

A6 Continue from $A2$ to $A5$.

< B: Algorithm for splashing spray and spray cloud>

B1 Input positions, velocities and densities of the water particles.

B2 Generate the splashing spray and remove them after a constant time.

B3 Analyze the behavior of the splashing spray with equation of motion for small particles.

B4 Generate the spray cloud and remove them after a constant time.

B5 Analyze the behavior of the spray cloud with Stable Fluid.

B6 Update the air velocity, which affects the splashing spray, with the velocity of the spray cloud for the next time step.

B7 Continue from $B2$ to $B6$.

6 Simulation Results

The simulation was performed with a normal PC that has Intel Core i7 2.8GHz CPU, 4GB main memory, and GeForce GTX 670 with 8GB memory. The maximum number of particles of the water stream and the splashing spray were both 262,144, and the grid size for the spray cloud was 80^3. The parameters of α_{sur}, α_{iso} and β were 0.79, 0.45 and 500, respectively. Simulation time was 1,096[s] for 1,000 steps (575[s] for water stream, and 521[s] for splashing spray and spray cloud). On the other hand, the rendering time was 614[s] for 1,000 step. In this simulation, 1 step corresponds to 2[ms].

Simulation Results are shown in Fig.4. All images are captured at 15 seconds after the simulation has begun. (a1)-(a3) are the simulation results with the same slope angle (60[degrees]) and different heights of the rock, while (b1)-(b3) are the other simulation results with the same width (110[m]) and different heights of the rock. From the figures, we can see that the spray cloud is soaring up along the rock, spreading out to the front and rotating over the rock. As the rock height becomes lower, the spray cloud is more spreading out over the rock rather than soaring up. On the other hand, as the height becomes higher, the spray cloud is more soaring up along the rock rather than spreading out over the rock. It seems that the boundary affects the soaring-up of the spray cloud; however, the spray cloud that goes out of the boundary is just removed so that the spray cloud does not soar up along the boundary. Instead of that, the spray cloud soars up by the force generated from the air density difference.

Fig. 5 shows the comparison of the simulation result with a real waterfall called “Kegon no Taki” that is very famous waterfall in Japan. We can see that both waterfalls have the same kind of water stream that has some water blocks and spray cloud that soars up from the basin.

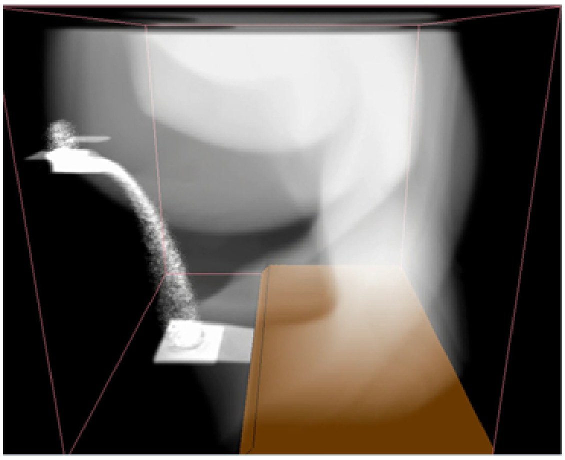
(a) Simulation result (b) Kegon no Taki

Figure 5: Comparison of the simulation result with a real waterfall.

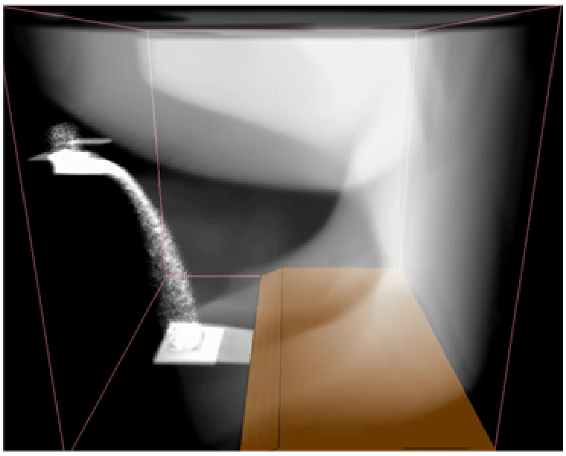
7 Conclusions and Future Works

We have simulated waterfall behavior with different environments. In our model, waterfall is composed of three parts: water stream, splashing spray and spray cloud, and also the water stream is constructed with three types of particles: water stream, free surface, and isolated particles. In addition, water stream has been analyzed with SPH, splashing spray has been analyzed by solving equation of motion for small particles, and finally spray cloud has been analyzed with Stable Fluid. Especially, we have introduced the interaction between the splashing spray and the spray cloud by considering the air velocity that has the role of the bridge between the spray cloud and the splashing spray. With the comparison of the different kinds of simulation results, we have found that the spray cloud soars up along the rock located in front of the waterfall, spreads out to the front and rotates over the rock. We have also seen that the spray cloud is more soaring up near the rock edge rather than spreading over the rock as the rock slope becomes steeper, while the spray cloud is more spreading over the rock as the rock slope becomes gentler.

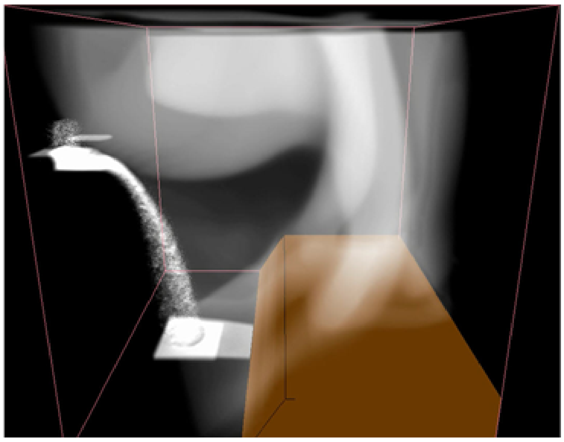
In the future, we have to consider mass conservation when one particle generates some other particles, and also the method to generate many kinds of waterfalls.



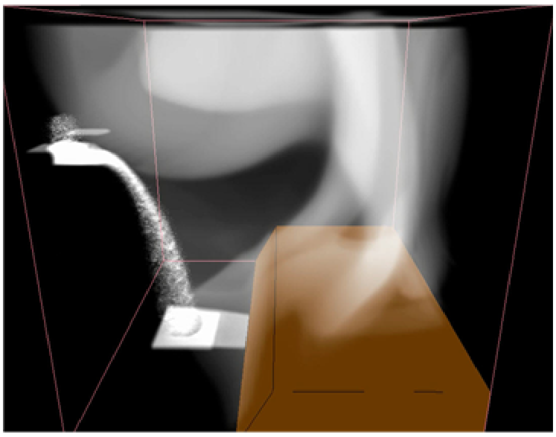
(a1) Angle 60°, Width 117m, Height 5m



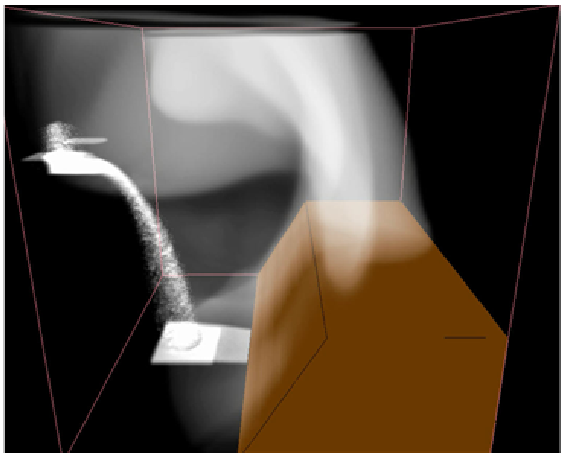
(b1) Angle 27°, Width 110m, Height 5m



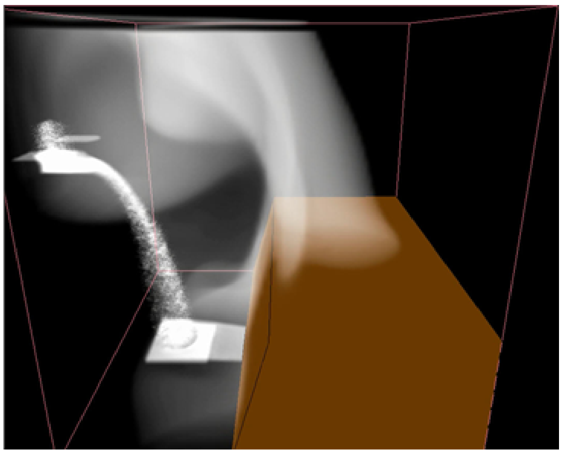
(a2) Angle 60°, Width 108m, Height 20m



(b2) Angle 63°, Width 110m, Height 20m



(a3) Angle 60°, Width 97m, Height 40m



(b3) Angle 76°, Width 110m, Height 40m

Figure 4: Simulation Results.

References

[1] D. Mould and Y.H. Yang. Modeling water for computer graphics. *Computers & Graphics*, 21(6):801–814, 1997.

[2] A. Iglesias. Computer graphics for water modeling and rendering: A survey. *Future Generation Computer Systems*, 20(8):1355–1374, 2004.

[3] E. Darles, B. Crespin, D. Ghazanfarpour, and J.C. Gonzato. A survey of ocean simulation and rendering techniques in computer graphics. *Computer Graphics Forum*, 30(1):43–60, 2011.

[4] D. Hinsinger, F. Neyret, and M. P. Cani. Interactive animation of ocean waves. *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on computer animation*, pages 116–166, 2002.

[5] X. Cui, J. Yi-cheng, and L. Xiu-wen. Real-time ocean wave in multi-channel marine simulator. *Proceedings of the 2004 ACM SIGGRAPH international conference on virtual reality continuum and its application in industry*, pages 332–335, 2004.

[6] J. Dupuy and E. Bruneton. Real-time animation and rendering of ocean whitecaps. *Proceedings of the SIGGRAPH Asia 2012 Technical Briefs*, page Article No.15, 2012.

[7] M. Müller, D. Charypsr, and M. Gross. Particle-based fluid simulation for interactive applications. *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on computer animation*, pages 154–159, 2003.

[8] P. Kipfer and R. Westermann. Realistic and interactive simulation of rivers. *Proceedings of Graphics Interface 2006*, pages 41–48, 2006.

[9] N. Chentanez and M. Müller. Real-time eulerian water simulation using a restricted tall cell grid. *ACM Transactions on Graphics*, 30(4):82:1–82:10, 2011.

[10] T. Nishino, K. Iwasaki, Y. Dobashi, and T. Nishita. Visual simulation of freezing ice with air bubbles. *Proceedings of the SIGGRAPH Asia 2012 Technical Briefs*, page Article No.1, 2012.

[11] N. Foster and R. Fedkiw. Practical animation of liquids. *Proceedings of the ACM SIGGRAPH 2001*, pages 23–30, 2001.

[12] O. Busaryev, T. K. Dy, H. Wang, and Z. Ren. Animating bubble interactions in a liquid foam. *ACM Transactions on Graphics*, 31(4):63:1–63:8, 2012.

[13] J. M. Hong, H. Y. Lee, J. C. Yoon, and C. H. Kim. Bubbles alive. *ACM Transactions on Graphics*, 27(3):48:1–48:4, 2008.

[14] N. Chentanez and M. Müller. Real-time simulation of large bodies of water with small scale details. *Proceedings of the 2010 ACM SIGGRAPH/Eurographics symposium on computer animation*, pages 197–206, 2010.

[15] G. Miller. Globular dynamics: A connected particle system for animating viscous fluids. *Computers & Graphics*, 13(3):305–309, 1989.

[16] K. Sims. Particle animation and rendering using data parallel computation. *Proceedings of the ACM SIGGRAPH 90*, 24(4):405–413, 1990.

[17] S. T. Greenwood and D. H. House. Better with bubbles: Enhancing the visual realism of simulated fluid. *Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on computer animation*, pages 287–296, 2004.

[18] W. Geiger, M. Leo, N. Rasmussen, F. Losasso, and R. Fedkiw. So real it'll make you wet. *Proceedings of the 2006 ACM SIGGRAPH Sketches, Article No.20*, 2006.

[19] B. Kim, Y. Liu, I. Llamas, X. Jiao, and J. Rossignac. Simulation of bubbles in foam with the volume control method. *ACM Transactions on Graphics*, 26(3):98:1–98:10, 2007.

[20] F. Losasso, J. O. Talton, N. Kwatra, and R. Fedkiw. Two-way coupled sph and particle level set fluid simulation. *IEEE Trans. on Visualization and Computer Graphics*, 14(4):797–804, 2008.

[21] H. Mallinder. The modelling of large waterfalls using string texture. *The Journal of Visualization and Computer Animation*, 6(1):3–10, 1995.

[22] N. Foster and D. Metaxas. Controlling fluid animation. *Proceedings of the 1997 Conference on Computer Graphics International*, pages 178–188, 1997.

[23] A. T. Howes and A. R. Forrest. Visual simulation of waterfalls and other water phenomena. *Proceedings of the ACM SIGGRAPH 97 Visual Proceedings: The art and interdisciplinary programs of SIGGRAPH '97*, page 146, 1997.

[24] R. Sakaguchi, T. Dufor, J. Zalzala, P. Lambert, and A. Kapler. End of the world waterfall setup for "pirates of the caribbean 3". *Proceedings of the ACM SIGGRAPH 2007 Sketches, Article No.90*, 2007.

[25] P. Hardie. Falling water. *Proceedings of the ACM SIGGRAPH 2007 Art Gallery*, page 270, 2007.

[26] T. Takahashi, H. Fujii, A. Kunimatsu, K. Hiwada, T. Saito, K. Tanaka, and H. Ueki. Realistic animation of fluid with splash and foam. *Computer Graphics Forum*, 22(3):391–400, 2003.

[27] R. Hoetzlein and T. Höllerer. Interactive water streams with sphere scan conversion. *Proceedings of the 2009 symposium on interactive 3D graphics and games*, pages 107–114, 2009.

[28] S. Miyashita and K. Funahashi. Falling water with key particle and envelope surface for virtual liquid manipulation model. *Proceedings of the 18th ACM symposium on virtual reality software and technology*, pages 197–198, 2012.

[29] M. B. Nielsen and O. Østerby. A two-continua approach to eulerian simulation of water spray. *ACM Transactions on Graphics*, 32(4):67:1–67:10, 2013.

[30] N. Mukai, Y. Sakai, and Y. Chang. Waterfall simulation by using a particle and grid-based hybrid approach. *Proceedings of 2014 International Conference on CyberWorld*, pages 23–30, 2014.

[31] N. Nishibe, N. Mukai, and Y. Chang. Spray cloud simulation by considering environment conditions. *Proceedings of NICOGRAPH International 2015*, 2015.

[32] S. Koshizuka. *No.5 in Computational Mechanics Series: Particle Method*. Maruzen, 2006.

[33] S. Koshizuka. *Particle Method Simulation - Physics based CG Introduction*. Baifukan, 2008.

[34] M. Fujisawa. *Basics of Physics Simulation for CG*. Mainabi, 2013.

[35] T. Ushijima. Dispersion and mixing of small particles, droplets and bubbles in turbulent flows. *Nagare*, 23:191–201, 2004.

[36] J. Stam. Stable fluids. *Proceedings of the ACM SIGGRAPH 99*, pages 121–128, 1999.

[37] J. Kondo. *Meteorology of Aquatic Environment - Income and Expenditure of Water and Heat on Ground Surface*. Asakura Shoten, 1994.

Nobuhiko Mukai

Nobuhiko Mukai is a professor of Graduate School of Engineering, Tokyo City University. He received a B.E., M.E., and Ph.D degrees from Osaka University in 1983, 1985, and 2001 respectively. He started to work at Mitsubishi Electric Corporation and changed to work as an associate professor for Musashi Institute of Technology in 2002. He is currently a professor of Tokyo City University from 2007. His research interests are computer graphics and image processing. He is a member of ACM, SAS, VRSJ, IEICE, ITE, IPSJ, IIEEJ, and JSUM.

Yuto Hizono

Yuto Hizono is a graduate student in Engineering, Tokyo City University. His research interests are computer graphics and physically based simulation.

Youngha Chang

Youngha Chang is a Lecturer of Graduate School of Engineering, Tokyo City University. She received a B.E. from Ewha Woman's University in 1998. She also earned Ph.D from Tokyo Institute of Technology in 2004. She became a researcher, a research associate, and an assistant professor at Tokyo Institute of Technology in 2004, 2006, and 2007, respectively. She is currently a lecturer at Tokyo City University from 2012. Her research interests are image processing and color science. She is a member of SAS and IPSJ.