Level Set Advection of Free Fluid Surface Modified by Surface Tension

Israel Pineda*, Oubong Gwun**

Abstract

Fluids appear in innumerable phenomena; therefore, it is interesting to reproduce those phenomena by computer graphics techniques. However, this process is not trivial. We work with a fluid simulation that uses Navier-Stokes equations to model the fluid, a semi-Lagrangian approach to solve it and the level set method to track the surface of the fluid. Modified versions of the Navier-Stokes equations for computer graphics allow us to create a wide divers ity of effects. In this paper, we propose a technique that allows us to integrate a force inspired by surface tension into the model. We describe which information we need and how to modify the model with this new approach. We end up with a modified simulation that has additional effects that might be suitable for computer graphics purpo ses. The effects that we are able to recreate are small waves and droplet-like formations close to the surface of the fluid. This model preserves the overall behavior governed by the Navier-Stokes equations.

Keywords : fluid simulation | surface tension | level set | fluid surface

Ⅰ. INTRODUCTION

Computer graphics simulations continuously need new methods to enhance their results. Sometimes these methods are physically inspired; however, they are not always physically accurate because the ultimate goal is to generate visually plausible results. This is an important distinction from other related disciplines like computational fluid dynamics (CFD). Computer graphics simulations also need control techniques that allow us to manipulate how the simulation behaves. This is particularly true for fluid simulations. In fluid simulations, the overall behavior of the fluid must be defined by the physical equations. But, it is also desirable to have a method to further control the simulation at a lower level.

Different categorizations are possible for fluid simulation but the one we are interested in is the scale. According to scale there are two kinds of fluid simulations, small scale and large scale. This distinction is important because at different scales the fluid behaves differently and therefore the effects that we reproduce are also different.

Small scale simulations deal with very small domains where even small forces can lead to important effects. One important example is surface tension. Surface tension is a force that, on this scale, show very interesting results. On the other hand, large scale simulations deal with big amounts of fluid. On this scale some forces are more important than others, specifically, surface tension is not considered to have an important contribution.

This paper describes the design and implementation of a new method that includes low scale simulation techniques in a large scale simulation, inspired by surface tension force. In this way, we improve the

---

* PhD Student, Division of Computer Science & Engineering, Chonbuk National University
** Professor, Division of Computer Science & Engineering, Chonbuk National University

Manuscript Received : 2015.05.20.
Revised : 2015.06.24.
Confirmation of Publication: 2015.06.27.
Corresponding Author : Oubong Gwun
e–mail : obgwun@jbnu.ac.kr

surface of the fluid by adding additional information to the simulation. We present how the Navier-Stokes set of equations that govern the fluid should be updated to contain a new logic. We report and analyze our results showing the modified and enhanced free surface of the fluid.

This paper is a following up to the work presented in "The International Conference on Computer Graphics, Multimedia and Image Processing (CGMIP2014)" [1].

Ⅱ. RELATED WORK

The evolution of fluid free surfaces has been tackled from different perspectives considering the different scales of the underlying phenomenon. For example, small scale simulations have successfully recreated drop formations [2] and motion of soap bubbles[3]. Bubble growth and detachment has been studied [4] as the effect of surface tension. CFD has a wide range of literature in this regard. In the case of large scale simulations, there is also a wide variety of work. However, we will concentrate on the level set based fluid simulation. The usage of level set methods to track the surface of the fluid has been proved to generate nice free surfaces[5]. Additionally, the interaction of multiple liquids [6] has also been investigated where different level sets are used within one simulation. The coupling between solid and fluid [7] has also been reported. CFD and computer graphics have both done major contributions to this topic.

Approaches considering surface tension exist, like the extension of the Ghost Fluid Method (GFM) [8] where viscosity, surface tension, and gravity are considered. These methods need to store additional data in what is called "ghost cells". In addition, there is some study of the interface of multiphase flows [8]. Some related topics are also discussed in a survey of physically based fluid animations[9] where different approaches are exposed.

Ⅲ. PROPOSED METHOD

1. Eulerian and Semi-Lagrangian Simulation

The Eulerian approach of a fluid simulation uses a fixed set of points in the domain and then observes how the quantities related to the fluid change in those positions over time. Those quantities could be scalars or vectors and they are usually viscosity, pressure and local velocity of the fluid. The set of points where the information is observed is known as the grid. Thus, the grid is the data structure that contains all the information. One special grid, very popular in computer graphics is the mark-and-cell (MAC) grid [10] which is a staggered grid that stores information in different locations of the cell, its roots are from CFD. For the sake of completeness, we have to mention that more sophisticated and modern grids exist, but we are not exploring them in this work.

The semi-Lagrangian approach [11] is an Eulerian simulation that updates the values in the grid in a very specific way. It uses the idea of a particle traveling through the fluid in order to compute the new values. This model has been proved to be stable and is widely used in physically based animations for computer graphics.

2. Navier-Stokes Equations

The physical behavior of a fluid is represented as a set of Partial Differential Equations (PDE) known as the Navier-Stokes equations. These equations represent the fluid in terms of its local velocity. Most of the understanding and techniques to solve the Navier-Stokes equations are borrowed from CFD. The first equation, known as the momentum equation, models the behavior of the fluid, see Equation 1.

$$\frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} + \frac{1}{\rho} \nabla p = \vec{g} + \nu \nabla \cdot \nabla \vec{u} \qquad (1)$$

In Equation (1), $\vec{u}$ represents the local velocity and it is the output that we need to calculate, ρ represents the density, p the pressure, $\vec{g}$ the gravity force and ν is the viscosity. The operators are the gradient ($\triangledown$)

and the Laplacian (▽ • ▽). The second equation, known as the incompressibility equation, ensures a constant volume of the fluid over time enforcing a divergence free vector field, see Equation (2).For this last equation the operator involved is the divergence (▽ •).

$$\nabla \cdot \vec{u} = 0 \tag{2}$$

These equations are solved using finite differences [12]. In particular, we use splitting. This technique solves independent terms of the Navier-Stokes equations and then adds the output of each intermediate stage to get the final solution.

3. Level Set Equation

Solving the Navier-Stokes equations is one step of the simulation, we later need to know how to extract the free surface of the fluid from the vector field that the Navier-Stokes equations yielded. Different methods exist to track the free surface of the fluid, namely Volume Of Fluid (VOF) [13] and Level Set Method [14]. In this work we use the level set method to track where the surface of the fluid is at a given time.

The level set method allows us to integrate over time a given distance function, this distance function contains the distance from any point within the domain to the actual surface of the fluid. In other words, if we have the distance functionwe can obtain the surface by evaluating the isocontour . To advect this function, we have to solve the differential equation known as the level set equation, see Equation (3). In Equation (3) we use the vector field from the Navier-Stokes equations to direct the advection.

$$\phi_t + \vec{u} \cdot \nabla \phi = 0 \tag{3}$$

This distance function handles very efficiently even complex movements of the free surface that cannot be reproduced by other techniques.

4. Surface Tension

When two bodies of fluid interact with each other, the generated interface is affected by a force called surface tension. It is a force derived from the interaction of the molecules of the fluids. Basically, the molecules of the fluid are more likely to stay together with other molecules of the same fluid instead of crossing the interface.

Therefore, it looks like the surface of the fluid has a solid component. Surface tension occurs between two or more fluids and it has a measurable magnitude, the surface tension coefficient.

The effects of surface tension can be observed in a variety of circumstances in nature. For example, insects walking in a puddle, or paper clips floating on the water are some of the most famous ones. Another related idea to surface tension is capillarity. Capillarity is the movement of the liquid that happens in small spaces due to surface tension and adhesive forces.

Navier-Stokes system of equations, as presented in Equation 1 and Equation 2, can be derived from conservation of mass and momentum from classic mechanics. These equations do not include information about surface tension. In order to include surface tension the common approach is to model using the Young–Laplace equation and the contact angles law. However, this adds more complexity to the model and it is not the path we follow in this paper.

5. Design and Implementation

Within the context of a semi-Lagrangian simulation, we included an approximation of surface tension and analyzed the effect that this model has in the free surface of the fluid. We made this in a specific region of the simulation, close to the interface. In order to restrict the effect of the new force to the surrounding of the surface, we used a technique inspired by the narrow band of the level set method. Thus, the computation of the Navier-Stokes equation is modified only near the surface boundary. See Fig. 1 for an overview of the method.

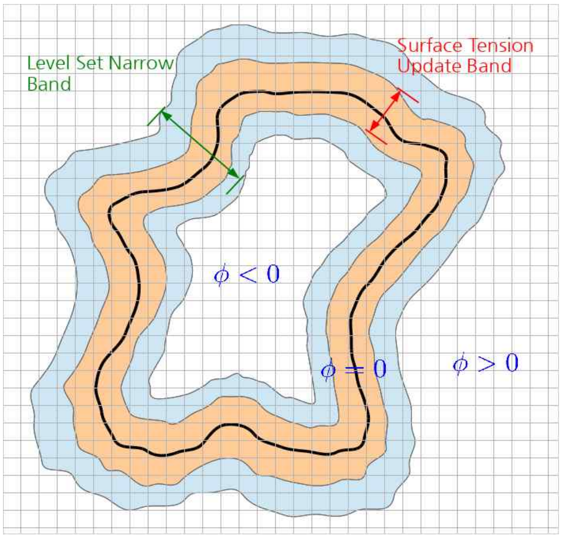


Fig. 1.Overall scheme of the surface tension model

In Figure 1, it is possible to see that our scheme uses two different bands, the first one is the narrow band of the level set and the second is the surface tension update band. They are defined independently from each other. However, it is better if the update band for the surface tension algorithm is smaller than the narrow band of the level set in order to reduce computation time. The length of the threshold is defined by the T parameter, some typical way to assign this parameter is $T = 5\Delta x$. The applicability of this value depends on the resolution of the grid. Everything outside this threshold is not affected by our method. Every time step is defined by the Courant-Friedrichs-Lewy (CFL) condition, this guarantees that the simulation does not blow up while trying to integrate for time steps that are too big. In each time step we called each of the following procedures:

1. Calculate the time step
2. Advect the velocity
3. Compute the body forces
4. Apply viscosity
5. Apply pressure
6. Advect fluid surface (zero level set)

The reason why we used all these steps is that we used splitting to solve the Navier-Stokes equations. Splitting consists in solving different terms of the Navier-Stokes equations independently. The third step of the simulation flow shown before, computing the body forces, is used to modify the velocity field with some external forces, usually gravity. However, here we included the logic of surface tension to compute the total force as $\vec{F} = \vec{g} - \gamma\kappa \bullet \vec{N}$. Where $\vec{g}$ represents gravity, γ represents the surface tension coefficient between the two interacting fluids, κrepresents the mean curvature and $\vec{N}$ is the normal vector.

Equation 1 needed to be partially redefined as presented in Equation 4. We added the force in what is known as the body forces block. This modified version uses additional information for the computation.

$$\frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} + \frac{1}{\rho}\nabla p = (\vec{g} - \gamma\kappa \cdot \vec{N}) + \nu \nabla \cdot \nabla \vec{u} \tag{4}$$

We directly affect the velocity field which determines the behavior of the fluid. Changing the velocity field means that in the posterior processing, viscosity and pressure will lead to different results. Bear in mind that if the difference inserted is too abrupt, it may cause the next iteration of the solver to fail at solving either the viscosity or the pressure. In other words, the change that happens due to our method should not be too far away from the physical constraints.

Our method uses an in-place algorithm, therefore, all the operations are executed using the already created MAC grid and do not need additional space. Although surface tension is a characteristic of the interface, not of the material itself, in our approximation we included this force within the fluid. More precisely, we included surface tension in a narrow band near the interface boundary defined by the parameter T.

Computer graphics simulations often neglect the gas, usually air, outside the liquid body, however, because our implementation is based on the implementation of multiple fluids [5], we could use two level sets, one for water and one for the air, in our experiments we have only dealt with one level set. Ignoring the gas outside the fluid is actually one of the differences between a computer graphics and a CFD simulation.

In contrast with our model, real surface tension makes the pressure to be different at each point of the fluid which leads to a big increment in computation time. In our approximation that is not the case because we consider the pressure to be uniform all over the fluid. That intrinsically means that we only have experimented with incompressible fluids, constant pressure.

This technique of using the body forces block to modify the behavior of the fluid has been previously used to modify the behavior of the fluid. For example, Stam [11] uses this term to manipulate the movement of a smoke simulation using the computer mouse, and Fattal and Lischinski [15] study in more detail how to modify this block in order to drive the flow towards a target object. We neglect any computation that goes beyond the boundaries of the domain.

Ⅳ. RESULTS

Now we present some results we obtained from our experimentation with the proposed method. For the discretization process we used different grid resolutions, however, a grid with 100 cells in each dimension, gave good results while it was still viable to execute it on a regular computer. We used only squared fixed grids, thus, it took a long time to compute. To create our testing scene we used a modeling application (Blender) and imported it as an OBJ file. We started the process by creating a distance function. Then, we executed the simulation as presented in Section 3.5. After the execution of our simulation, we analyzed and visualized its results using the Visualization Toolkit (VTK). See Fig. 2 to see how a full simulation looks like.

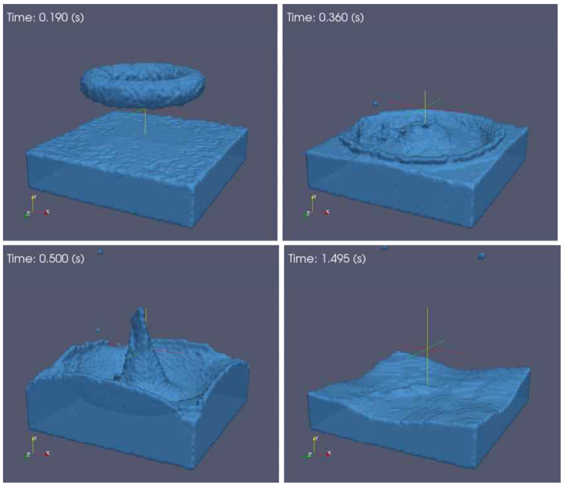


Fig. 2. Fluid simulation using surface tension at different times.

Next we show the visualization of two close-ups of the fluid for two different simulations of the same scene, see Fig. 3; we found that they are clearly different from each other. The simulation featuring our proposed method included low scale irregularities like small waves, but at the same time the overall simulation kept the expected flow. We tested different fluids. Here, we present some results using water properties. The viscosity of water is $0.001\ Ns/m^2$, the surface tension factor depends on what are the two interacting fluids, for example, for the interaction of water and air the surface tension factor is $0.073\ N/m$. We used these values to configure the simulations presented in Fig. 2 and Fig. 3.

In Fig. 4, we show the simulation in context with the underlying grid, the distance function and the surface of the fluid. In addition, we compare how the distance function of the level set is affected within the given threshold.

Different values of surface tension factor make the output to be more or less similar to the original simulation. In order to analyze numerically how each surface differ from one another depending on the surface tension factor, we have found a displacement value from the original surface without surface tension value. That displacement is computed using Equation 5. In this equation, grid1 and grid2 are the MAC grids with surface tension and without surface tension respectively. The value we used is the distance function stored in the center of the grids.

$$displacement = \sum_{i=1}^{n} \frac{grid1(i) + grid2(i)}{2} \tag{5}$$

We executed the same simulation using the standard value of surface tension factor and additionally we have doubled and also halved that value to see how different the simulations were from each other. This showed the average displacement of the simulations, see Fig. 5.

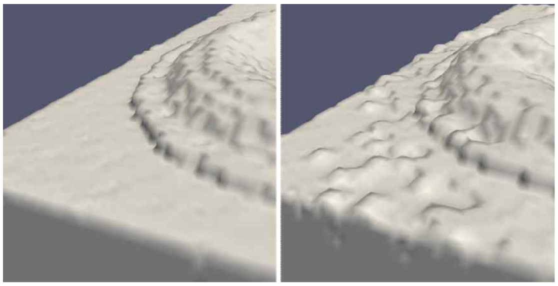
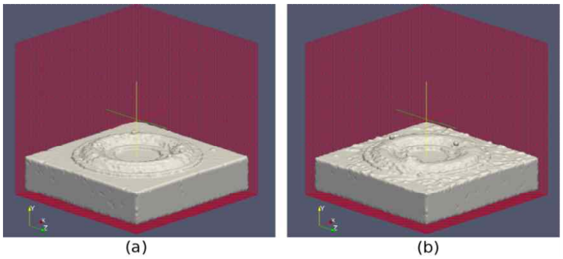


Fig. 3. Close-ups comparing the small scale differences introduced when surface tension is included. Left: without surface tension. Right: with surface tension.

We can see that at the beginning of the simulation they are all the same, no difference, however, as the simulations progress they start to differ from one another. Nevertheless, we found that the rising and falling pattern of the displacement is the same. Also, we saw that if we double the surface tension parameter a proportional displacement happens. Likewise, with half of the surface tension factor a smaller displacement occurs. Another important result is the difference of the vector fields of the simulations. Even though, they follow the same pattern it is important to notice how they differ in the low scale and how our method include some turbulence to the execution. See Fig 6.



(a) (b)

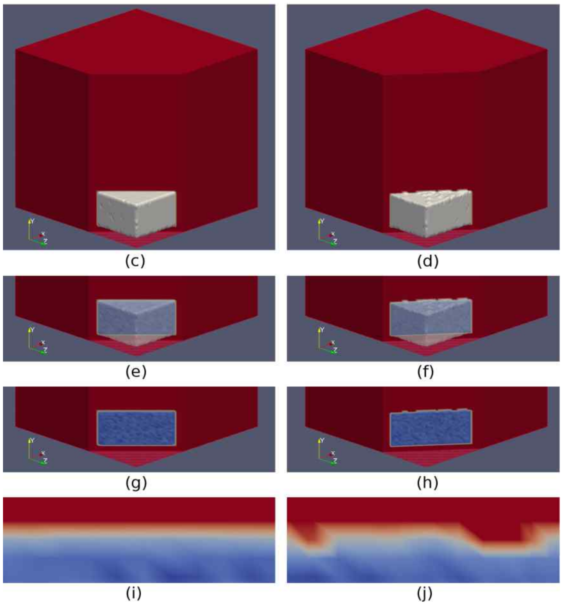


Fig. 4. Left column shows results without surface tension, right column shows results with surface tension. a & b show the surface in context with the grid. c & d show how we clipped the simulation for analysis. e & f zero level set with transparency. g & h distance function without surface, i & j show the distance function difference of the two simulations.

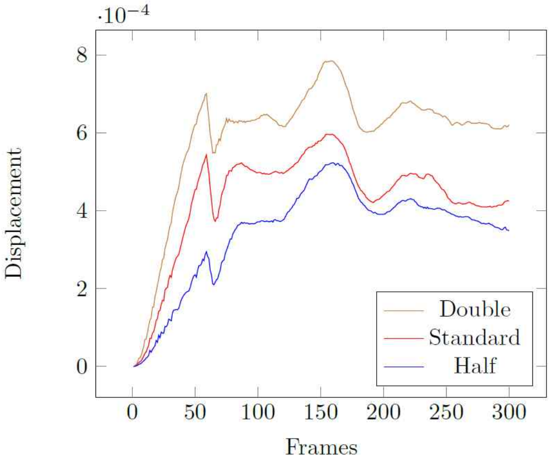
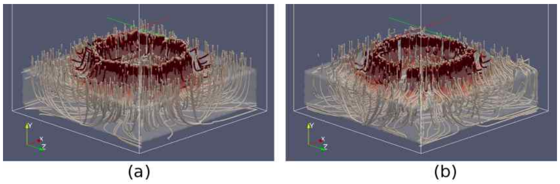


Fig. 5. Average displacement for different values of surface tension. Standard surface tension coefficient at room temperature is *0.073 N/m*, halved and doubled values are calculated from this value.



(a) (b)

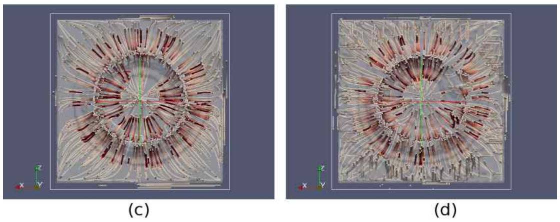


Fig. 6. Vector field comparison from perspective and top view. Left without surface tension: the stream trace is very smooth. Right with surface tension: the stream trace includes turbulence.

Ⅴ. DISCUSSION

We created a semi-Lagrangian simulation that included surface tension parameter for computer graphics purposes. Our simulation used the level set method to track the surface of the fluid. We used Navier-Stokes equations to control the fluid and in addition we modified the body forces block with a computation inspired in the physical behavior of surface tension. We described what data we needed, how we computed surface tension and how we integrated this approximation in the context of a semi-Lagrangian fluid simulation. This has led to a more visually realistic simulation.

We can see the reproduction of a variety of fluid effects like shivering, water pouring and container shaking. Small waves are created in the surface when the fluid has moved.

Depending on the target application, our algorithm can be more or less suited to improve the results. For example, we believe that it has potential in applications like movie production and game development. However, it might not be very well suited for applications where accuracy is more important that appearance.

The simulation could blow up if the change in the vector field is too big because it will not be able to solve the pressure or the viscosity of the system. Therefore, it is important to keep the values of surface tension not too far away from the physical constraints.

Even though other methods for surface tension exist, ours is unique because it does not work with triangle meshes, it is a fast approximation, and avoid the complexity of dealing with variation in pressure. We made sure to restrict that the new modification is only executed near the boundary of the surface, this keeps the computational cost low and at the same time enforces the idea of surface tension as a characteristic of the interface.

We conclude that the simulation has been improved with the proposed method by the inclusion of low scale irregularities like small waves and drop-like formations around the surface of the fluid. Using a threshold for the distance we were able to restrict the change so it took place only near to the boundary.

Ⅵ. LIMITATIONS AND FUTURE WORK

The effects of our method over the surface can be controlled through the parameters like surface tension factor and the threshold. However, other forms of control could provide artists or other users more flexibility. Those other forms could include additional thresholds. The computational cost of a simulation using a MAC grid is huge. Therefore, it might hide some additional benefits of our method, but at least it makes it difficult to experiment with high resolution, this makes data collection and analysis extremely slow.

We omitted rendering techniques from the present work; however, we believe that a proper rendering could showcase those low scale irregularities. Because fluids include transparency, refraction and reflection, raytracing could be an excellent alternative.

REFERENCES

[1] Israel Pineda, Oubong Gwun, “Surface Tension Approximation in Semi-Lagrangian Level Set Based Fluid Simulations for Computer Graphics”, *CGMIP 2014 The International Conference on Computer Graphics, Multimedia and Image Processing (Malaysia, 2014).*

[2] Huamin Wang, Peter Mucha, Greg Turk, “Water Drops on Surfaces”, *ACM SIGGRAPH 2005 Papers (New York, NY, USA: ACM, 2005)*, pp. 921-929.

[3] Myungjoo Kang, "A Level Set Approach for the Motion of Soap Bubbles with Curvature Dependent Velocity Or Acceleration" (1996).

[4] A. Albadawi, D.B. Donoghue, A.J. Robinson, D.B. Murray, Y.M.C Delauré, "Influence of surface tension implementation in Volume of Fluid and coupled Volume of Fluid with Level Set methods for bubble growth and detachment", *International Journal of Multiphase Flow 53 (2013),* pp. 11-28.

[5] Douglas Enright, Ronald Fedkiw, Joel Ferziger, Ian Mitchell, "A Hybrid Particle Level Set Method for Improved Interface Capturing", *Journal of Computational Physics 183, 1 (2002)*, pp. 83-116.

[6] Frank Losasso, Tamar Shinar, Andrew Selle, Ronald Fedkiw, "Multiple Interacting Liquids", *ACM SIGGRAPH 2006 Papers* (New York, NY, USA: ACM, 2006), pp. 812-819.

[7] Christopher Batty, Florence Bertails, Robert Bridson, "A Fast Variational Framework for Accurate Solid-fluid Coupling", *ACM SIGGRAPH 2007 Papers* (New York, NY, USA: ACM, 2007).

[8] Ronald Fedkiw, Tariq Aslam, Barry Merriman, Stanley Osher, "A Non-oscillatory Eulerian Approach to Interfaces in Multimaterial Flows (the Ghost Fluid Method)", *Journal of Computational Physics 152, 2 (1999),* pp. 457-492.

[9] Jie Tan, XuBo Yang, "Physically-based fluid animation: A survey", *Science in China Series F: Information Sciences* 52, 5 (2009), pp. 723-740.

[10] Francis Harlow, Eddie Welch, "Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface", *Phys. Fluids* 8 (1965), pp. 2182-2189.

[11] Jos Stam, "Stable Fluids", *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 1999), pp. 121-128.

[12] Nick Foster, Dimitri Metaxas, "Realistic Animation of Liquids", *Graphical Models and Image Processing* (, 1996), pp. 23-30.

[13] C. W. Hirt, B. D. Nichols, "Volume of fluid (VOF) method for the dynamics of free boundaries", *Journal of Computational Physics* 39, 1 (1981), pp. 201-225.

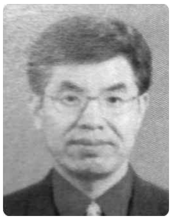
[14] Stanley Osher, Ronald Fedkiw, "Level Set Methods and Dynamic Implicit Surfaces 2003 edition" (New York: Springer, 2002).

[15] Raanan Fattal, Dani Lischinski, "Target-driven Smoke Animation", *ACM SIGGRAPH 2004 Papers* (New York, NY, USA: ACM, 2004), pp. 441-448.

Authors

Israel Pineda

He received his B.Eng. degree in Computer Systems from Universidad Politécnica Salesiana, Ecuador, in 2011 and his M.S. degree in Computer Science and Engineering from Chonbuk National University, South Korea, in 2015. Currently he is a PhD student at the same institution.



Oubong Gwun

He received his B.S. and M.S. degrees in Electrical Engineering from Korea University in 1980 and 1983, and his PhD degree in Engineering and Science from Kyushu University, Japan, in 1993. Since 1993 he has been a professor in the Division of Computer Science and Engineering at Chonbuk National University, Korea.