VISUAL SIMULATION OF DROPLETS ADHERING TO A SURFACE

面上に付着した液滴の視覚的シミュレーション

by

Nobuyuki Nakata
仲田将之

A Master Thesis
修士論文

Submitted to
the Graduate School of the University of Tokyo
on January 29th, 2013
Department of Complexity Science and Engineering
Graduate School of Frontier Sciences

Thesis Supervisor: Tomoyuki Nishita　西田 友是
Professor of Complexity Science and Engineering

ABSTRACT

Surface tension (interfacial tension) occurs on a common boundary of two different matters. All of solid, liquid and gas materials have surface tensions. Since liquid surface tension is stronger than its of gas and since liquid material is flexible compared to solid material, the surface tension of liquids takes an interesting role in their motions and shapes.

Liquid having either a height or a diameter less than capillary length (the effective length of capillary), its surface tension is dominant over gravity. Because either the height or the size of droplet is less than capillary length, droplet motions and shapes are also under strong influence of surface tensions. In these liquids, droplet is one of the common liquids in our life. The proposed methods take account of the effects of surface tension on droplet motions and on droplet shapes.

In a driver's viewpoint, the visibility through the windshield is seriously affected by the droplets adhesion and their motions on the glass. Driving simulator is commonly used for driver's education. Recently, the hydrophobic coating on the windshield became a solution to keep driver's view clean. The proposed method of real-time droplet animation takes account of the droplet adhesion called contact angle hysteresis due to surface tension. If the proposed method is added on top of the simulator, it will assist to improve the driving safety. This method assumes each relatively large droplet as a mass point and simulates its movement using contact angle hysteresis accounting for dynamic hydrophobicity as well as other external forces such as gravity and air resistance. All of an exponential number of still, tiny droplets are treated together in a normal map applied to the windshield. This method also visualizes the Lotus effect, a cleaning action by the moving droplets.

In the both of the film industry and the construction industry, it is common to construct a realistic scene in computer. Although the realistic scene is highly demanded, measuring a droplet shape is not easy because droplet is a deformable liquid and usually translucent. The proposed method for computing droplet shape can generate droplet shapes from an image of droplets. Because this method is simple yet flexible for computing droplet shapes on the surface of an object, it is a help to construct a more realistic scene in computer. The proposed method can flexibly generate natural-looking droplets on top of artificial outlines, such as logo designs, symbolic icons and character fonts. Animation of the droplets is achieved by applying the method to a series of outlines generated by a 2D controllable fluid simulator. This method also demonstrates the transition of the droplet shape following changes in topology of adhesion region and compare the result to that of a physical experiment.

論文要旨

表面張力 (界面張力) は二種類の異なる物質の境界で発生する．固体や液体，気体のすべての物質は表面張力を持つ．液体の表面張力は気体のそれより強く，液体は固体に比べて柔軟なので，液体の表面張力はその運動や形状にとって重要な役割を担う．

高さや直径が毛管長 (表面張力の有効半径) より小さな液体では，重力より表面張力が支配的になる．液滴の高さまたは大きさが毛管長より小さいため，その運動と形状も表面張力の影響を強く受ける．そのような液体の中で，液滴は我々の生活において一般的な液体の一つである．提案法の一つでは液滴の運動に対する表面張力の影響を考慮する．もう一方の提案法では液滴の形状に対する影響を考慮する．提案法は，ロータス効果と呼ばれる水滴の運動による自浄作用も可視化する．

ドライバーの視点で見ると，フロントガラス上の水滴の付着とその運動は視認性に大きな影響を与える．近年，フロントガラスに対する撥水加工がドライバーの視認性を確保する一つの解決法になってきた．提案するリアルタイムアニメーション手法は表面張力に起因する接触角ヒステリシスという付着力を考慮する．提案法をドライビングシミュレータへの応用すれば，運転の安全性の向上に寄与する．提案法は比較的大き液滴を質点として扱い，外力として重力や空気抵抗だけでなく，接触角ヒステリシスによる動的撥水性を考慮することで水滴の運動をシミュレートする．指数関数的な数の静止した微小水滴は，単一の法線マップに統合してフロントガラスへ適用する．

映画産業や建築産業では，コンピュータ上にリアルなシーンを構築することは一般的である．リアルなシーンの必要性は高いが，液滴は変形する液体であり一般的に半透明であるため，液滴の形状の測定は容易ではない．水滴形状の計算に関する提案法では一枚の水滴画像から水滴形状の生成ができる．提案法は面上の水滴形状の計算においてシンプルで柔軟な方法なので，コンピュータ上によりリアルなシーンを構築するための役に立つ．提案法は，ロゴやアイコンやフォントなど，人工的な付着形状からも自然な見た目の水滴を柔軟に生成できる．液滴のアニメーションも二次元の流体シミュレーション結果を利用して作成できる．付着領域のトポロジー変化に関して物理的な実験と比較する．

Acknowledgements

First and foremost, I would like to thank Professor Tomoyuki Nishita, Assistant Professor Yonghao Yue, Professor Masanori Kakimoto (Tokyo University of Technology) and Yoshinori Dobashi (Hokkaido University). And I thank all of the members in Nishita laboratory for valuable discussions among them.

Contents

List of Figures

Chapter 1

Introduction

Surface tension (interfacial tension) is an intermolecular force. All of solid, liquid and gas materials have surface tensions. Solid material has the relatively strong surface tension and gas material has the weakest surface tension. The surface tension of liquid is intermediate between them. Although liquid surface tension is not much strong as it of solid, its shape is deformable and the surface tension takes a central role in the droplet shape.

Since solid surface tension is strong, if a small liquid is on a solid surface, the liquid is adhered to the solid by its surface tension. Solid surface tension is usually stronger than it of liquid but some of organic materials has weaker surface tension compared to it of liquid. Once an inorganic solid surface is covered with an organic material, the surface holds a function of water-repellent. Even if the surface tension of hydrophobic surface is weak, there is some adhesion force on the surface. This adhesion force caused by the surface tension is called as contact angle hysteresis and it is important for simulating liquid motion to consider the effects of contact angle hysteresis.

Generally, a liquid having either a height or a radius of less capillary length is under strong influence of surface tension. In these liquids, droplets, especially

water droplets, are seen commonly in our daily life. Hence a droplet is a typical liquid dominated by surface tension. Other than the aspect of surface tension, droplet has a visually impressive characteristic for the reflections and refractions.

In a rainy scene, flow of water droplets on the window or windshield surfaces can be seen. Those droplets are commonly used as those rainy scene description in film works and in other types of motion pictures. More recently, computer generated animations of droplets flow on the windshields are realized for advanced video games and driving simulators. Since the glass material has hydrophilic or water-attracting nature, water droplets move along irregular trajectories seeking for water-attracting places of the surface, as we often find on the windows in a rainy day. Most of the existing water droplet animation methods simulated these winding trajectories of the droplets.

In real driving situations, those water droplets trajectories or water-film on the windshields due to the hydrophilicity seriously affect the visibility through the glass. To clear the water droplets, mechanical wipers have been used since the beginning of the automobile history. In addition, as auxiliary measures, coating the windshield with water repellent material became a solution a few decades ago. In the year 2000, the first water-repellent finished windshield became commercially available. Nowadays such hydrophobic windshield products are widely used in the automobile market.

A large amount of research literature on the behavior of water droplets on hydrophobic surfaces is published in chemical and mechanical engineering fields. To the authors' knowledge, however, little work has been done on real-time simulation of water droplets sliding across hydrophobic windshields. In Chapter 3, we address this problem and propose a solution consisting of several practical simulation models for use in games and driving simulators (Figure 1.1).

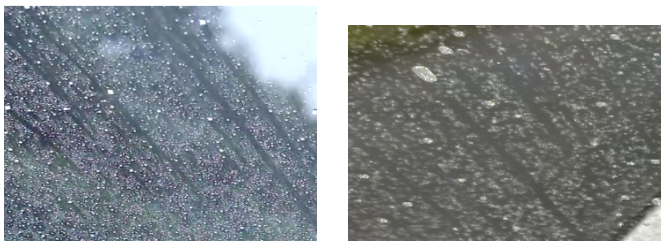

Figure 1.1: Real droplets (left) and droplets of the proposed method (right).

Water attracting or repelling feature of surface material of hydrophobic windshield should be quantified differently in two situations, static and dynamic. The static repellency has been investigated for a long time and the fundamentals have been established. For water droplet animation, knowledge on the dynamic repellency is more important, which is true in engineering analysis of water-shedding phenomena on the windshield. While the dynamic water repellency includes a number of unexplainable phenomena, there are a couple of major factors and indicators characterizing the dynamic repellency. Those include contact angle hysteresis, falling angle (Figure 3.2), falling velocity, and falling acceleration.

The relationship between the contact angle hysteresis and the slope angle has long been investigated. In case of an ideal water droplet shape, the contact angle hysteresis is known to be in proportion to the falling angle. The falling velocity and acceleration vary by the surface material even when the slope angle remains constant. Although the standard methods for evaluating and measuring the falling velocity/acceleration were not established until recently, it is known that the behavior of a falling water droplet on the hydrophobic surface is explainable in terms of rolling and sliding.

In Chapter 3, we take the knowledge on the dynamic repellency into account and propose a real-time animation method for water droplets on the hydrophobic windshield. As the water-repellent coated windshields become standard in the

automobile market, our contribution is to provide video game and simulator developers with a means of reproducing realistic and harmonious motions of the water droplet cluster traveling across the hydrophobic windshield.

Other than on windows or windshields, droplets can be seen commonly. For example, if we get a can of beer out from a refrigerator, we can observe a number of droplets of various sizes and shapes formed on the surface of the can. Seeing these droplets, someone would feel the coldness of the beer. Hence representing these droplets is important not only for realism but also for stimulating a human response. Additionally, the generation of artistically shaped droplets (Figure 1.2 (b)) is also in demand, to make images more expressive.

Shooting a photograph or taking a video with the desired droplet shapes in the real world is not an easy task, since the formation of the shapes is usually an instantaneous phenomenon (*e.g.*, the droplet will soon dry up), and user desired droplet shapes are usually difficult to generate (*e.g.*, a droplet with a crispy corner). Thus, the development of a method for assisting the generation of the images using computer graphics is in demand. In Chapter 4, we focus on modeling droplets, formed (mound) on a surface (Figure 1.2). In particular, we aim at artistic applications, such as logo designs, symbolic icons, character fonts and shape transition animations with topology changes.

In Chapter 4, we assume that the droplets are quasi-static (*i.e.*, they either remain in their original locations or move at slow speeds). In this case, the forces acting on the droplets are gravitational forces and surface tension. The surface of the droplet is determined according to the balance between these two forces. Since these forces are potential forces, solving for the balanced state is equivalent to minimizing the sum of the surface and gravitational energies. Hence, we can faithfully capture the physical requirement by considering an energy optimization

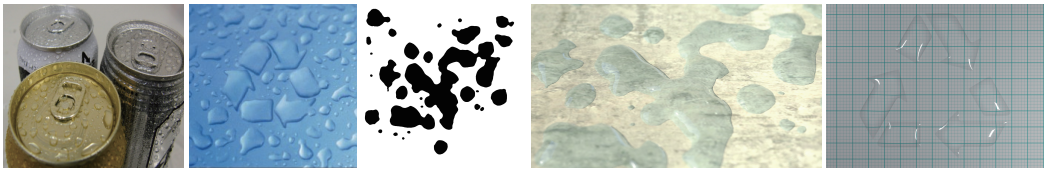

Figure 1.2: (a) An example of droplets on a can of beer. (b) An artistic droplet shaped logo (©Global Water). (c) The user inputs the domain(s) in which the droplet shape(s) are to be generated. (d) The droplet shapes generated by our method using (c) as the input. (e) A logo generated using our method.

problem, that minimizes the sum of the surface and gravitational energies of the droplets, satisfying mass conservation. Compared to previous droplet modeling methods (e.g., [3]) using implicit surfaces like metaballs, our method can generate droplet shapes with crispy corners and with better physically plausible constraints.

The proposed method in Chapter 4 can generate not only natural droplet shapes, like the reproduction of droplet shapes from a photograph (Figure 1.2 (d)), but also artificial and artistic shapes, like logo designs (Figure 1.2 (d)), symbolic icons and character fonts. Since our method is lightweight, the user can edit those shapes interactively. Animation of droplets is achieved by applying our method to a series of outlines generated by a 2D controllable fluid simulator. We also demonstrate transition of the shape of a droplet as a result of changes in topology and compare the result to that of a physical experiment.

Chapter 2

Related Work

In the computer graphics field, several methods have been introduced for animating water droplets. Kaneda et al. [4] proposed a method to describe the movement of the droplets by defining each droplet as a particle and move it with particle dynamics. Since the droplets travel seeking for water-attracting places, their trajectories on the glass surface form complex shapes. Kaneda et al. [5] simulated these motions by a random walk method using random numbers. Recently their method was implemented as a real-time simulator with a GPU computing technique [6]. Fournier et al. [7] depicted the trajectories of droplets using the mass spring model.

Several researchers have developed fluid dynamics based methods for the water droplet dynamics simulation. Wang et al. [8] took into account surface tension, contact angle, and contact angle hysteresis. The surface tension is more dominant in a water droplet than in regular large-scale fluid forms. Thurey et al. [9] introduced the mean curvature flow, which is known as a motion equation for surface boundaries, and evaluated the phenomena caused by the surface tension more appropriately than Wang et al.

Zhang et al. [10] developed a faster shape deformation method for droplets using

the mean curvature flow without other fluid simulations. They ignored the internal fluid flow of the droplets but used the surface tension and other external forces to give deformation, collision and division to each droplet represented as a polygon mesh. They achieved 10-50 fps in the experiment with 10K-50K polygon mesh. However, due to the implicit method for the mean curvature flow computation, the stability of their solution depends highly on the mesh quality and the time step, and the performance optimization is limited.

In order to tackle the problem of the droplet motion on the hydrophobic surfaces, we need to understand dynamic repellency. The structure or the behaviour of the surface molecules are considered to be a source of the dynamic repellency. To figure out the behaviour, Hirvi et al. [11] simulated a droplet consisting of thousands of water molecules using a molecular dynamics calculation technique.

Analyses of real water droplets have been done by several research groups. For example, Sakai et al. [12] measured the velocity and the acceleration of a droplet sliding across water-repellent surfaces. Droplets are known to run down either rolling or slipping on the incline depending on the degree of hydrophobicity [13][14]. Hashimoto et al. [1] measured the relationship between the volume and the velocity of a windswept droplet.

In Chapter 3, we address the problem of dynamic water-repellency taking the contact angle hysteresis into account. In addition, we use the knowledge of the real water drop analyses to verify and compensate our results. We avoided using the fluid dynamics simulation or the mean curvature flow since they are not suitable for real-time visualization. Due to the computing load and the time step limitations, those methods cannot handle sufficient number of droplets on a car windshield. Each droplet is represented as a mass point in our method.

Particle dynamics are common in the real-time simulation field. They are widely

adopted in games and interactive applications. Real-time physics engines in the market are equipped with features of particle dynamics and rigid body dynamics including collision detections as fundamental functions. We implemented our method introduced in Chapter 3 on top of a game engine 'Unity' and added unique behaviours of water droplets running or staying on the hydrophobic surfaces.

Modeling objects, in order to give them user specified and natural appearances, is an important research topic in the computer graphics field. For example, methods have been proposed for modeling smoke [15, 16, 17], water [18], clouds [19], trees [20], *etc.* Using such methods, one can generate artistic results or desired shapes while maintaining physical properties to ensure a natural appearance.

Previous methods for simulating droplets, such as that due to [21, 4, 8, 10], basically aim at generating realistic droplet motion, and do not take into account user specified shapes. In Chaper 4, we propose a simple yet flexible method to compute droplet shapes on an object surface. We propose decomposing the droplet generation process into two steps. In the first step, the user designs the 2D outlines (footprints or masks) of the droplets as shown in Figure 1.2 (c). In the second step, natural-looking 3D droplets are automatically computed from the outlines (Figure 1.2 (d)). The natural look is brought about by considering the physical properties that the droplets should satisfy. In this way, the user can easily generate droplet shapes by specifying just the outlines, even when an artificial 2D outline is specified. Thus, our method is flexible in the sense that it is easy to use.

Chapter 3

Animation of Water Droplets on a Hydrophobic Windshield

Solid surface tension is relatively strong compared to it of liquid. Some of organic materials has weaker surface tension than it of liquid. Covered with a organic material, the solid surface can have a function of water-repellent. Although the hydrophobic surface is very low surface tension, it still has some adhesion force with the droplets. The adhesion force is called as contact angle hysteresis. By considering contact angle hysteresis, the proposed method can simulate the droplet-repellent motion.

3.1 A Practical Model for Water Droplets on Hydrophobic Windshields

When a droplet is on a solid surface, the contact angle is defined as the angle between the solid surface and the droplet surface. The contact angle is determined by the Young equation, which describes the balance of three surface tensions, as shown in Equation (3.1).

$$\gamma_L cos\theta = \gamma_S - \gamma_{SL}, \tag{3.1}$$

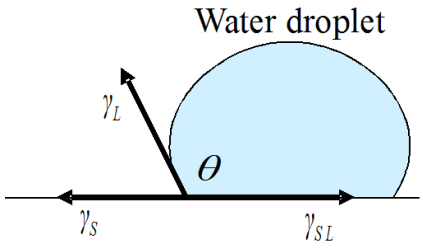

Figure 3.1: Contact angle and tensions of a water droplet.

where θ is the contact angle, γ_L is the surface tension of the water droplet, γ_S is the surface tension of the solid, γ_{SL} is the interface tension between the water and the solid (Figure 3.1).

When the radius of the droplet on hydrophobic surfaces is less than the radius of capillary (2.8mm), the surface tensions are the dominant factors of the water drop shape. Thus the droplet forms a near spherical geometry. Meanwhile, the contact angle of the glass becomes $90° - 100°$ when it is coated with commercially available repellent material.

Based on the above two observations, we assume that each rain droplet is rendered as a hemisphere. In practice, the geometric shape is basically a disc-like plane and the normal vectors for refraction are controlled to make it look hemisphere. Details are described in Section 3.2.1.

3.1.1 Contact Angle Hysteresis

When a thin pipe is inserted into water, the water level in the pipe is raised by the capillary action. This is caused by a force called the capillary force which operates along the triple boundary line among the water, the solid and the air. The capillary force is determined by the Young-Laplace equation.

With regard to a droplet which lies on a solid plane, the capillary forces along the circular triple boundary cancel each other out if the contact angle is constant

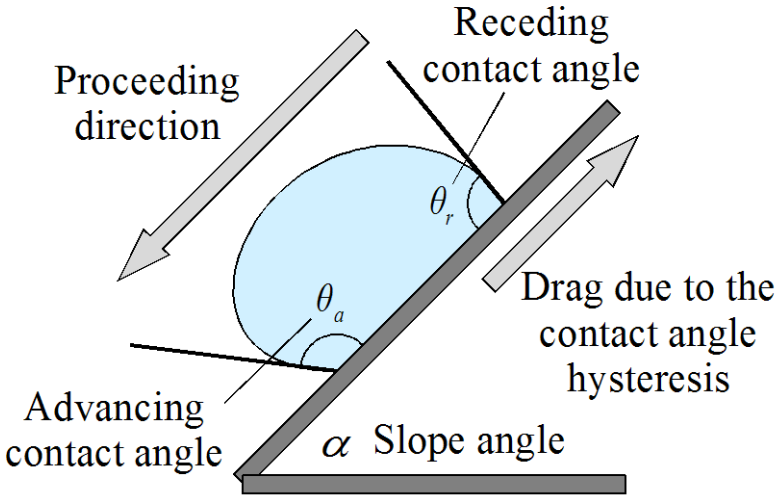

Figure 3.2: Advancing and receding contact angles of a water droplet.

along the circle. When some external forces are put on the droplet and its shape is deformed, the contact angles vary while the droplet stands still until the contact angle variance reaches at a certain value.

The contact angle hysteresis is defined as the difference between the advancing and receding contact angles (θ_a and θ_r, respectively). These two angles are defined as the largest and the smallest contact angles, respectively, at the moment that the water droplet starts moving on the solid plane by the sufficient external force. The slope angle at this moment is called the falling angle. Figure 3.2 illustrates the advancing and receding contact angles for an incline.

While the droplet is moving on the plane, a drag operates on the droplet toward the reversed direction against the proceeding direction. The amount of drag is related to the contact angle hysteresis. Assuming that the shape of the triple boundary is a circle, the drag is approximated with the following equation [22]

$$F_{hys} \simeq \frac{1}{2}\pi r\gamma_L \left(cos\theta_r - cos\theta_a\right), \tag{3.2}$$

where represents the radius of the water droplet. θ_r and θ_a are the receding and

the advancing contact angles, respectively.

3.1.2 Wind Drag

Automobile windshields meet with air resistance, or wind drag, according to the velocity of the running vehicle. The wind drag is defined as follows:

$$F_{wind} = \frac{1}{2}\rho C_D S V^2, \tag{3.3}$$

where ρ is the density of the air, C_D is the coefficient of resistance, S is the projected size of the droplet, and V is the velocity relative to the air.

In Equation (3.3), the droplet is assumed to be floating in the air. Since all droplets in our model are placed on a solid windshield, the equation needs to be modified. We assume that the wind is weakened at places very close to the solid plane. It is known that in such near-boundary layer, the wind velocity changes in a complicated manner.

We employed a simplest compensation to decrease the velocity in the near-boundary layer using an exponential law as shown in the following formula.

$$\tilde{V} = \begin{cases} V\left(\frac{y}{\sigma}\right)^{\frac{1}{2}} & (y < \sigma) \\ V & (y \geq \sigma) \end{cases}, \tag{3.4}$$

where V is the wind velocity out of the boundary layer (relative to the solid plane), y is the height of the droplet, δ is a parameter representing the thickness of the boundary layer, and $\tilde{V}$ is the compensated wind velocity for the droplet.

3.1.3 Viscous Dissipation

When a droplet is moving or rolling, another drag is caused by some in-bulk friction called viscous dissipation [23]. The drag is in proportion to the velocity of the droplet and represented as

$$F_{bulk} = \eta R v f(\theta), \tag{3.5}$$

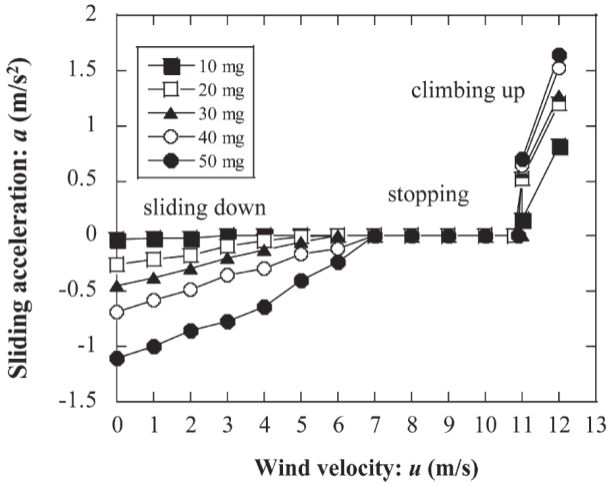

Figure 3.3: A measured relationship between the wind velocity and the acceleration of droplets, using a varying droplet size as a parameter (excerpt from [1]).

where η is the degree of viscosity of the water, R is the radius of the droplet, v is the velocity of the droplet. $f(\theta)$ is a factor dependent on the contact angle.

3.1.4 Wind Speed and the Droplet Acceleration

In the surface finishing engineering discipline, Hashimoto et al. [1] introduced an experiment to measure the acceleration of various volumes of water droplets placed on an angled hydrophobic plane in a wind tunnel. Figure 3.3 quotes from the literature and shows the result of the measured descending or ascending acceleration of the droplets. The contact angle, the slope angle, and the falling angle are 105° , 35° and 10°, respectively.

In the range where the wind velocity is relatively low, moderate but more falling accelerations are observed as the droplet size becomes greater. When the wind velocity is raised beyond a certain value (7m/s in Figure 3.3), the droplet stays still within some range of wind velocities. When the velocity is further raised beyond a higher value (11m/s), rapid ascending accelerations are observed, which

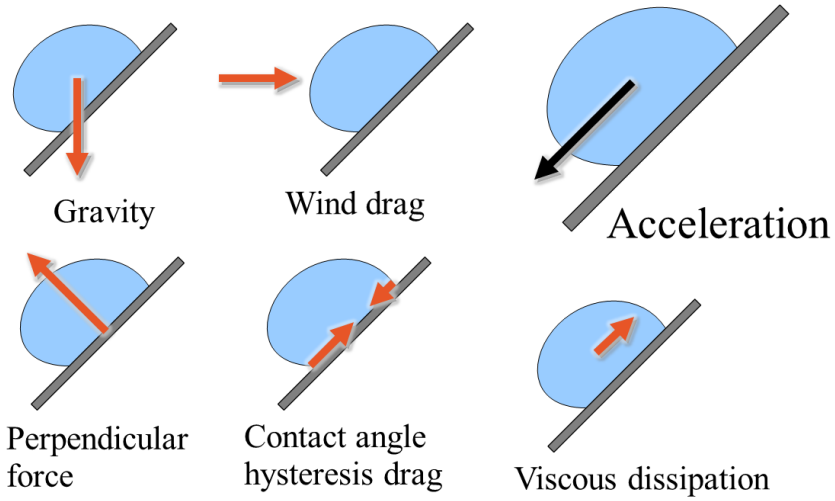

Figure 3.4: External forces added to a droplet and the resultant acceleration. In this example, the gravity is more dominant than the wind drag and thus the droplet slides down.

are greater as the droplet becomes larger.

On the other hand, we simulated the sliding accelerations of a droplet taking the following four forces into account (Figure 3.4).

- Gravity (vertical) F_g
- Wind drag (horizontal) F_{wind}
- Perpendicular force (normal to windshield)
- Viscous dissipation drag (tangential to windshield) F_{bulk}
- Contact angle hysteresis drag (tangential to windshield) F_{hys}

The wind drag has been described in Section 3.1.2. The contact angle hysteresis drag behaves as a resistance force parallel to the windshield, in the same way as the perpendicular force normal to the windshield. The force represented in Equation 3.2 defines the maximum limit of the hysteresis drag.

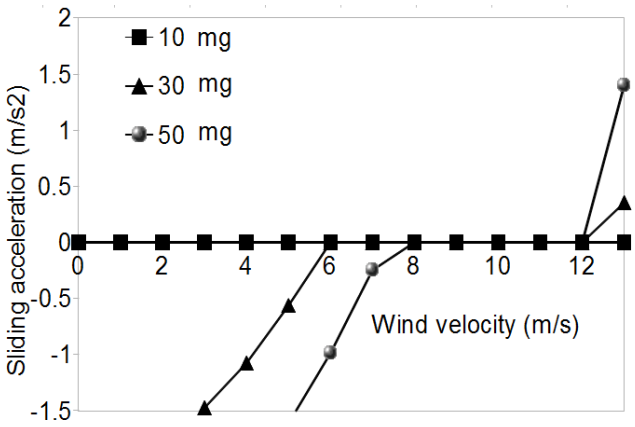

Figure 3.5: Simulation results of the droplet accelerations.

Figure 3.5 shows a simulated result of the accelerations for the varying droplet sizes. The range of wind velocities in which the droplet stays still is reproduced, and the range is very similar to the measured result in Figure 3.3.

The surface tension of the water droplet causes a pressure difference in the droplet. This is known as the Laplace pressure and is greater as the droplet radius is smaller. Therefore, when two water droplets of different sizes collide with each other, the small droplet gets absorbed by the larger one. We implemented this process and it is invoked on droplet collision detection.

3.1.5 Distribution of Raindrop Radii and the Lotus Effect

Lotus effect is a phenomenon which occurs when a water droplet moves across a hydrophobic surface. Lots of very small droplets and contamination spread on the surface are removed by the moving droplet along the trajectory. The same phenomenon is observed on a windshield as demonstrated in the snapshot of Figure 3.6.

Figure 3.7, an excerpt from [2], shows rain droplet radius distributions under 1mm/h, 5mm/h and 25mm/h rainfalls. The graph is with the raindrop diameters

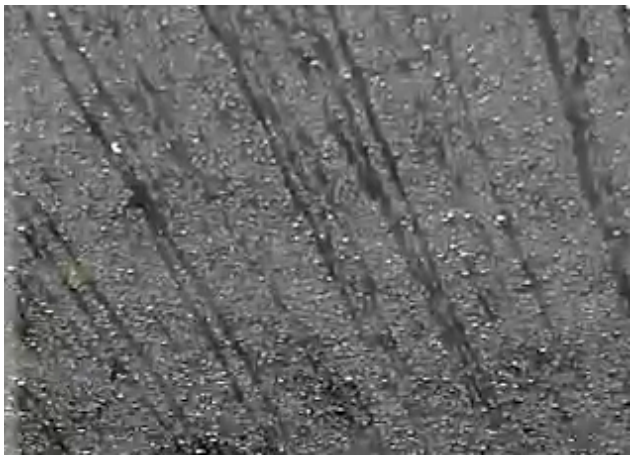

Figure 3.6: Droplet trajectories caused by the Lotus effect (image captured from a live-action movie of a windshield).

as the horizontal axis and the number of raindrops for each diameter as the vertical logarithmic axis. The line indicated as ‘ MP ’ is an exponential distribution model called the Marshall-Palmer distribution [2].

According to the model, the smaller the raindrop diameter is, the greater the number of raindrops becomes. Especially, tiny raindrops of below 1mm are contained with an exponentially large numbers. Therefore, it is impractical to simulate the motion of every droplet. Fortunately, those tiny raindrops do not move at all with our simulation model as shown in Figure 3.5. Thus we apply a single large normal map onto the windshield. The map contains the normal vectors which represents all the small droplets standing still on the windshield.

3.2 Implementation and Results

This section describes implementation of our method proposed in the previous section and demonstrates some results. We implemented the system on top of Unity, a popular game engine. Although our method regards each water droplet as

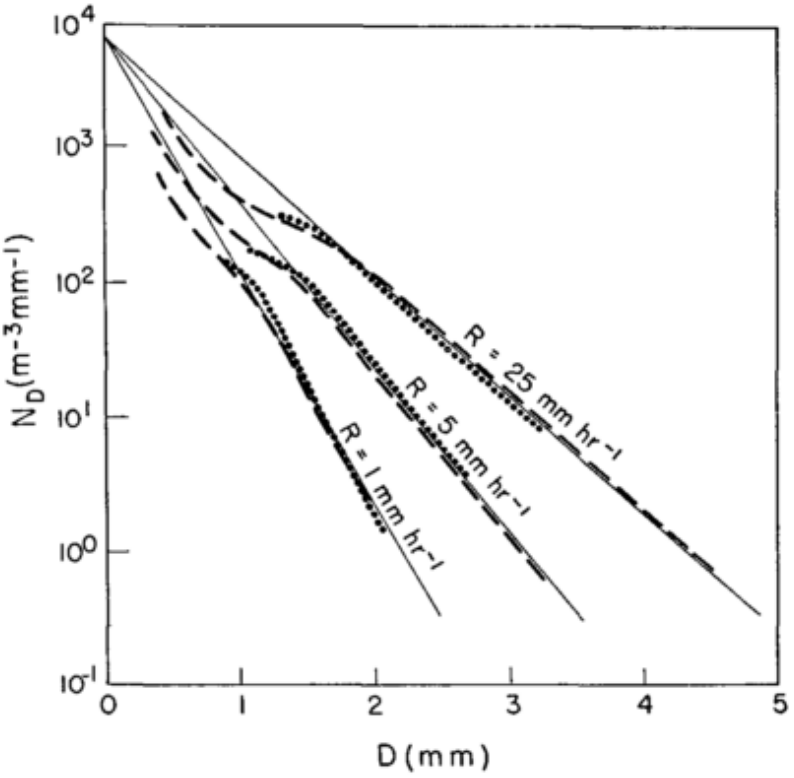

Figure 3.7: Distribution of the number of raindrops for each diameter (drop size distribution)[2].

a particle, we implemented each droplet as a small rigid body which does not rotate. Regarding the rigid body physics engine, we used NVIDIA PHYSX embedded in the Unity system.

In each time step of the simulation, our system calculates the external forces imposing on the water droplets as illustrated in Figure 3.4 in Section 3.1.3.

Regarding the gravity, we added some random noise to the force component parallel to the windshield in order to realize natural motions of the droplets caused by some assumed fluctuation of the running vehicle.

The implementation of viscous dissipation (Section 3.1.3) is a heuristic matter since the factor $f(\theta)$ in Equation (3.5) is not determined. We used a constant value

$\eta f(\theta) = 0.5$ in the equation. The important point is that the viscous dissipation drag is in proportion to the droplet velocity. The above constant value can be used to control the maximum droplet speed.

While the droplets are moved by the external forces, we obtain each collision point and the u-v coordinates and the normal vectors of the colliders from the collision detector of the physics engine. For a droplet being regarded as to be on the windshield, the windshield point corresponding to the droplet is calculated and the refraction map image for the Lotus effect is updated.

In case that a droplet collides with another droplet, the Laplace pressure effect is applied. The system compares the masses of the two droplets. If the difference is greater than the pre-defined threshold, these two will fuse together into one droplet.

3.2.1 Rendering Large and Movable Droplets

Each large water droplet (with over 1mm diameter) is rendered as a disc-shape polygon mesh when it is staying still on the windshield. The normal vectors on the disc surface are controlled so that the refracted environment appears to be mapped on a hemisphere.

While the droplet is moving across the windshield, its shape is deformed to be longer along the moving direction. The normal vectors are controlled so that the lengthened transparent droplet looks like a drug capsule sectioned by a screen-parallel plane. The deformation is controlled so that the assumed volume of the droplet is preserved. Using its normal vectors, the pixel shader calculates the refraction directions and maps the background texture image as the environment. Figure 3.8 is a close-up rendering image of a pseudo-hemisphere water droplet and a deformed pseudo-hemisphere.

Figure 3.8: Droplets rendered as a pseudo-hemisphere (left) and a deformed pseudo-hemisphere (right).

Those large droplets are generated with various sizes according to the Marshall-Palmar distribution shown in Figure 3.7. The number of large droplets generated per frame is set to be five typically. They are accumulated but eventually moved away out of the windshield or collided and fused with others. As a result, a couple of hundred to one thousand large droplets reside in the steady-state situation.

3.2.2 Rendering Small Static Droplets

Small droplets (with less than 1mm diameter) are represented as perturbation in a normal map image for the windshield, as described in Section 3.1.5. The diameters of the generated small droplets vary also according to the Marshall-Palmar distribution. The number of small droplets in our implementation amounts to approximately 200K.

The outside scene image is refracted according to the normal map. The trajectories of large droplets (pseudo-hemispheres) are stored as an image component which is used to suppress the normal map. They are composed in the shader

Figure 3.9: The Lotus effect. Small and still droplets are rendered as a normal map on the windshield. Large and moving droplets are rendered as pseudo-hemispheres.

program and the Lotus effect on the windshield surface is rendered (Figure 3.9).

Figure 3.10: A result with low wind velocity (11.3m/s) and a large contact angle hysteresis with 0.5.

Figure 3.11: A result with low wind velocity (11.3m/s) and a small contact angle hysteresis with 0.05.

Figure 3.12: A result with high wind velocity (15m/s) and a large contact angle hysteresis with 0.5.

3.2.3 Performance

All results referred to in this section are captured snapshots of real-time animations rendered from the driver ' s point of view toward the automobile proceeding direction viewing the outside through the windshield. The source of the outside image is a motion picture shot with a video camera placed between the two front seats of a running car when no rain is falling. The pre-recorded image is mapped as a video texture onto a billboard model placed in front of the windshield model.

Figures 3.10 and 3.11 are the examples with a small wind velocity. In Figure 3.10, a relatively large contact angle hysteresis is specified and thus the adherence is strong that the droplets do not move at all. In Figure 3.11, the adherence is smaller and the droplets move along the windshield curve.

Figure 3.12 is a result with stronger wind and the large droplets climb straight up the windshield. Since the adherence is strong and the boundary layer is set to be thick, the small droplets are made still.

The frame rates for Figures 3.10, 3.11 and 3.12 are 134-153fps, 80-100fps, and 70-100fps, respectively. The scene contains a windshield, large droplets and the video texture billboard shapes, which total approximately 17K vertices.

For the rendering results, we used an Intel Core2 Extreme X9600 (3GHz), NVIDIA GeForce GTX480 Graphics and 8GB main memory. The horizontal field of view is 45 and the distance between the viewpoint and the windshield is approximately 0.5m. The horizontal curvature radius of the windshield geometry is 5m constant and the vertical curvature is 0 (flat). The slope of the windshield is set to be 45.

Chapter 4

Generating Droplet Shapes on Surfaces from Boundaries

Assume that a droplet stays still on a surface. Since the droplet doesn't move, the droplet is under the equilibrium configuration of two different potential energies. It means the sum of these energies are minimized. Once the volume constraint and the boundary condition are added to the minimization problem, the solution will be unique. Although the volume and the boundary condition are the solution of a free boundary problem taking account of the solid interfacial energy, we assume the volume and boundary condition are given by users so that the users can specify the heights and the boundary shapes of droplets. By assuming the volume and the boundary condition as user defined values, the proposed method for computing droplet shapes on a surface will only need an image of the droplets. This assumption makes the method keep simple yet flexible.

4.1 A Simple yet Flexible Method for Computing Droplet Shapes on Surfaces

We assume the surface of the object (on which the droplet sits) is defined as a single-valued function $h(x, y)$, as shown in Figure 4.1, and the droplet footprint

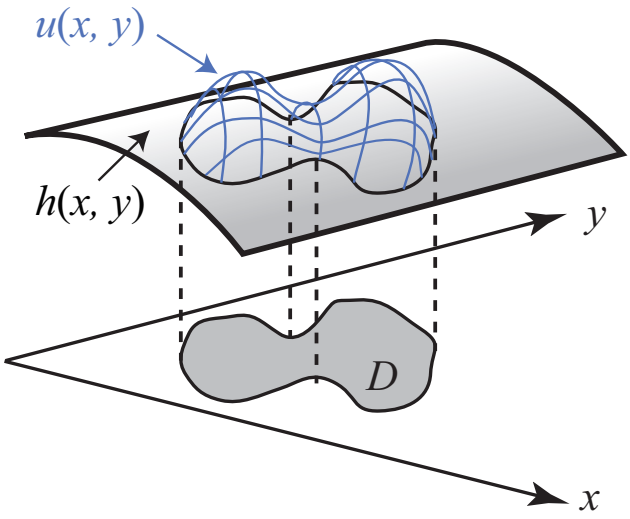

Figure 4.1: An illustration of droplet representation.

(or a mask) is a 2D domain D specified by the user (if the user specifies a mask, our system automatically extracts the outline from the mask). Note that h also represents the bottom of the droplet. We take the x and y axes to closely fit the object surface. We assume the droplet surface is a single-valued function defined as $u(x, y)$ on D. $u(x, y) = h(x, y)$ is satisfied on the boundary ∂D of the domain. Given its volume V, we want to minimize the sum of its surface energy E_s and gravitational energy E_g. For the simplicity of the presentation, in the following, let us assume x and y to be in the horizontal plane, which could be extended to inclined planes.

4.1.1 Definition of Potential Energies

The surface energy E_s is given by

$$E_s = \sigma S, \tag{4.1}$$

where σ is the surface tension coefficient, and S is the surface area of the droplet, given by

$$S = \int_D \sqrt{1 + \left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2} dxdy. \tag{4.2}$$

The gravitational energy E_g is given by

$$E_g = \int_D \frac{\rho g(u^2 - h^2)}{2} dxdy, \tag{4.3}$$

where ρ is the liquid density of the droplet and g is the gravitational acceleration. Finally, the droplet volume V_d is given by

$$V_d = \int_D (u - h) dxdy. \tag{4.4}$$

Hence, to solve the optimization problem we have to minimize $E_s + E_g$, such that $V_d = V$. We solve this problem using a variational approach. That is, we define an energy functional $E[u]$, given by

$$E[u] = E_s + E_g + \lambda(V - V_d), \tag{4.5}$$

where λ is a Lagrange multiplier. Let Θ be a set of parameters used to describe the surface u. The optimal situation can be computed by solving $\frac{\partial E[u]}{\partial \lambda} = 0$ and $\frac{\partial E[u]}{\partial \theta} = 0$ for every parameter $\theta \in \Theta$.

4.1.2 Numerical Computation of Droplet Shapes

To find the function u, we utilize a finite element method as follows.

First, in order to avoid the nonlinear term in Equation (4.2), we utilize the following approximation. Using the Taylor expansion, we have $\sqrt{(1+t)} \approx \frac{1}{2}(2+t)$ when t is small, where t is a real number. The approximation error is less than 10%, when $t < 1.428$. Hence, we have $\sqrt{1 + (\frac{\partial u}{\partial x})^2 + (\frac{\partial u}{\partial y})^2} \approx \frac{1}{2}\left(2 + (\frac{\partial u}{\partial x})^2 + (\frac{\partial u}{\partial y})^2\right)$. This approximation is valid (the error is less than 10%) when the slope of the surface from the xy plane is less than 50°.

Next, we triangulate the simulation domain D containing the droplets. To minimize the energy functional, we use a 1st-order Litz method (a finite element method equivalent to the Galerkin method in our case since the problem has an energy functional). More precisely, the surface $u(x, y)$ is approximated as a weighted summation of basis functions $u_i(x, y), i = 1, ..., N$ as $u(x, y) \approx \sum_{i=1}^{N} w_i u_i(x, y)$, where N is the number of basis functions and w_i is the weight of u_i. In the standard Galerkin method, each basis function u_i is a piecewise function defined at the vertices of each triangle, within one-ring neighborhood of the vertex. The minimization of the energy functional is then equivalent to solving the equations for a linear system, governed by $\frac{\partial E[u]}{\partial \lambda} = 0$ and $\frac{\partial E[u]}{\partial w_i} = 0$ for each basis function.

4.2 Implementation and Results

We implemented our method on Matlab R2012a. Since the matrix of Litz methods's equations is sparse and since the number of elements is typically less than one hundred thousand, we solve this matrix with using Matlab's sparse direct solver.

4.2.1 Examples Generated From Photographs

First, we validated our method by checking if we could recover real droplet shapes from photographs of droplets. We photographed droplets in the real world (Figure 4.2), and then used their footprints as the input to our method (Figure 4.3). Figure 4.4 shows the droplets generated by our method. Comparing the appearances of Figures 4.2 and 4.4, and the refraction patterns, we can see that the droplet shapes generated using our method match the real shapes. Figure 4.5 shows another rendered results of the droplet shapes generated by our method.

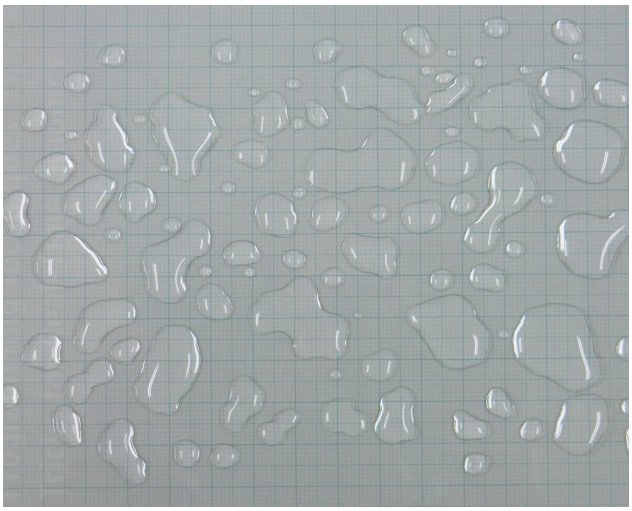

Figure 4.2: A photograph of droplets.

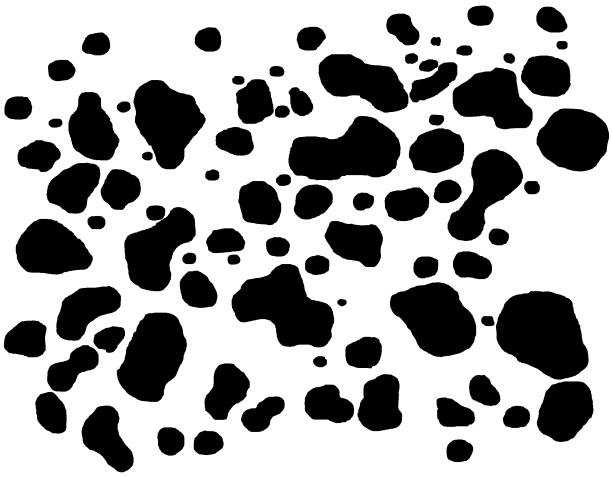

Figure 4.3: The input to our method, created according to Figure 4.2.

4.2.2 Comparison With a Real Example

Next, we demonstrated the usefulness of our method, by showing that we can use it to generate droplet shapes that are difficult to produce in reality. To generate these difficult to produce droplets in the real world, we first sprayed the region within which we wanted to generate them with water repellent, and then carefully

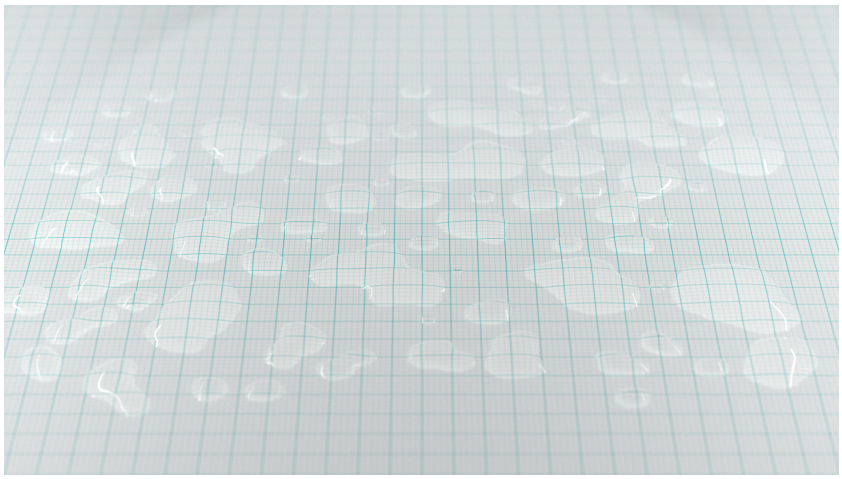

Figure 4.4: The droplet shapes generated by our method from Figure 4.3.

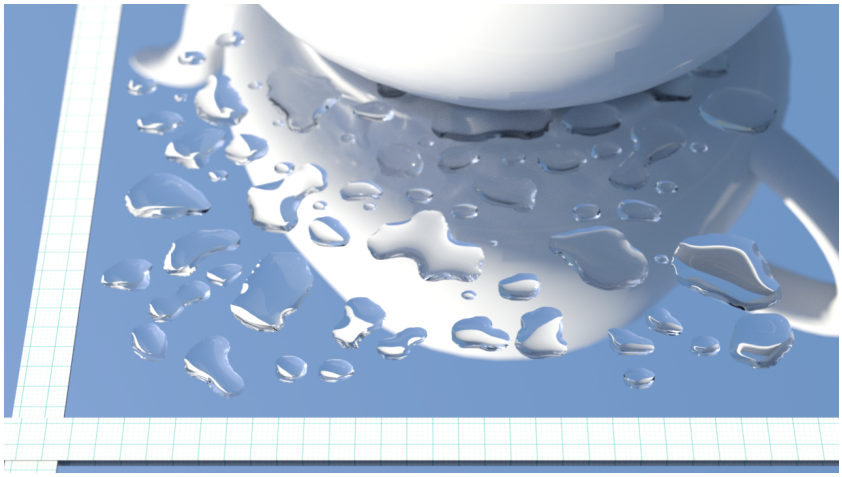

Figure 4.5: Rendered using different settings from Figure 4.4.

injected water onto the region. Even so, it is difficult to generate droplets with crispy corners in the real world (Figure 4.6). Our method can easily generate droplet shapes even for such a case (Figure 4.7).

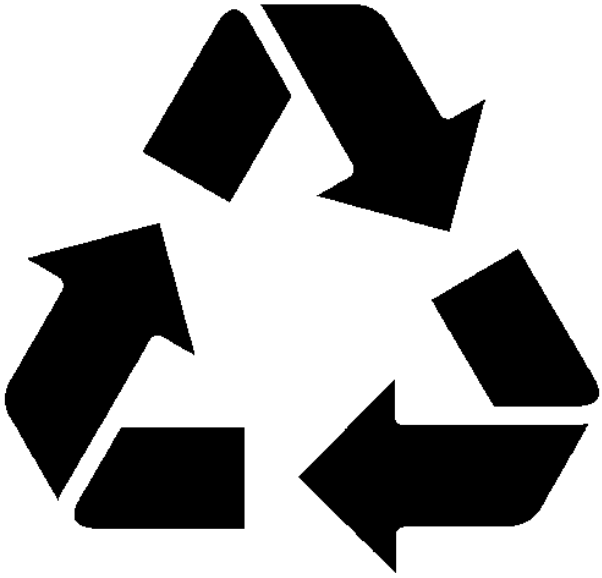

Figure 4.6: The desired shape.

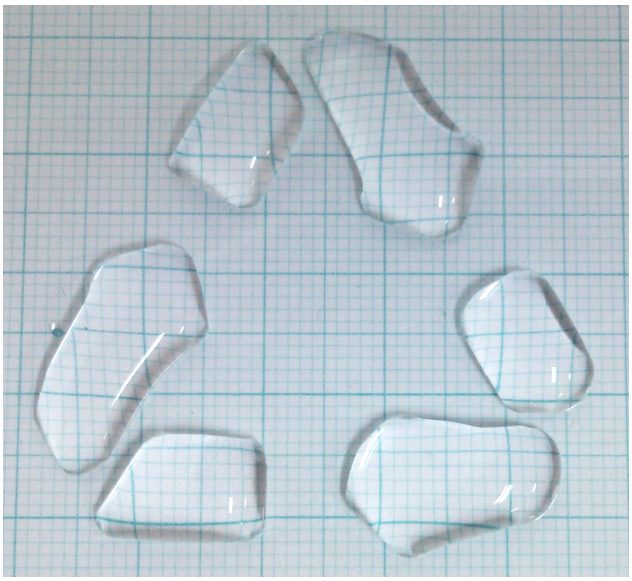

Figure 4.7: Droplets with crispy corners (see Figure 4.6) are difficult to produce in reality.

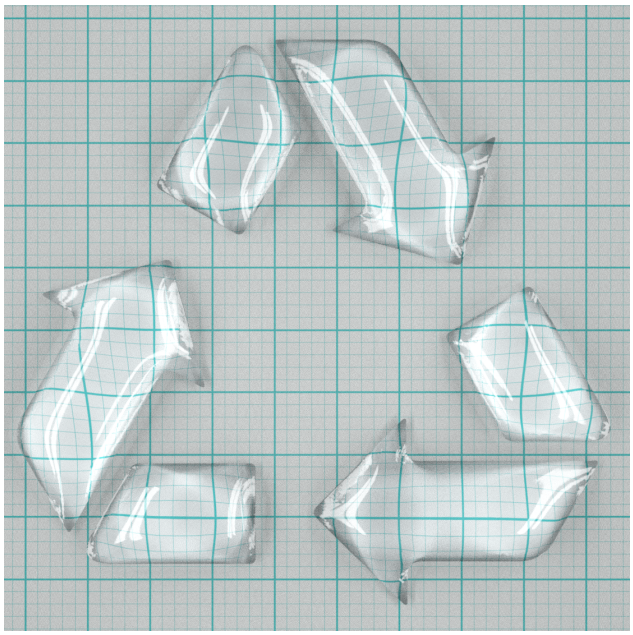

Figure 4.8: Our method can easily generate such droplets (see Figure 4.6).

4.2.3 Artistic Examples

Using our method, artistic examples, such as a logo made from a droplet (Figure 4.9), or a 'droplet font' (Figure 4.10) can be generated. Our method is lightweight since it just solves a 2D finite element method. Hence, we can easily let the user interactively edit the droplet shapes on a laptop. If needed, we can show a low resolution solution during editing in realtime, and present a high resolution result when the editing is finished.

In Figure 4.11, we show our generated droplets on a curved surface.

4.2.4 An Animation Example

We can also generate droplet animations, as shown in Figure 4.12. We used a 2D fluid control method based on [17] and quantized the result to obtain a binary signal to represent the droplet footprints. Then, for each frame, we applied

Figure 4.9: A logo made from a droplet.

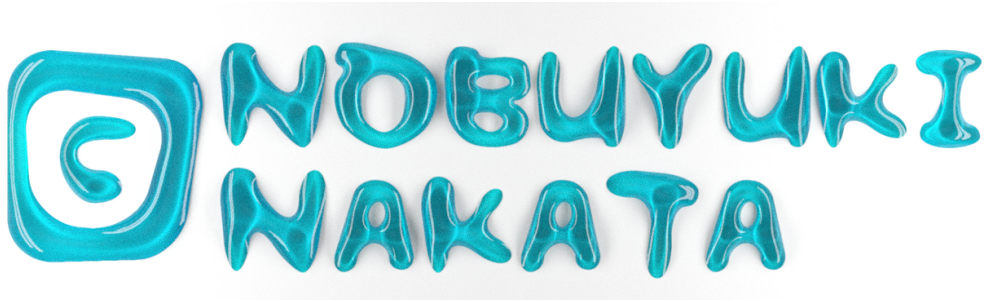

Figure 4.10: A 3D droplet font from 2D droplet font (Chocolate Syrup font).

our method independently to generate the droplet shape. The total volume of the droplet only was fixed and we did not apply any other special treatment to maintain the coherence between frames. We are aware that our method takes only static physical properties into account, and dynamic properties, such as inertia, are ignored. Nevertheless, the generated animation is quite pleasing.

In the real world, we can also observe the phenomenon wherein a small dry

Figure 4.11: Droplets on a curved surface.

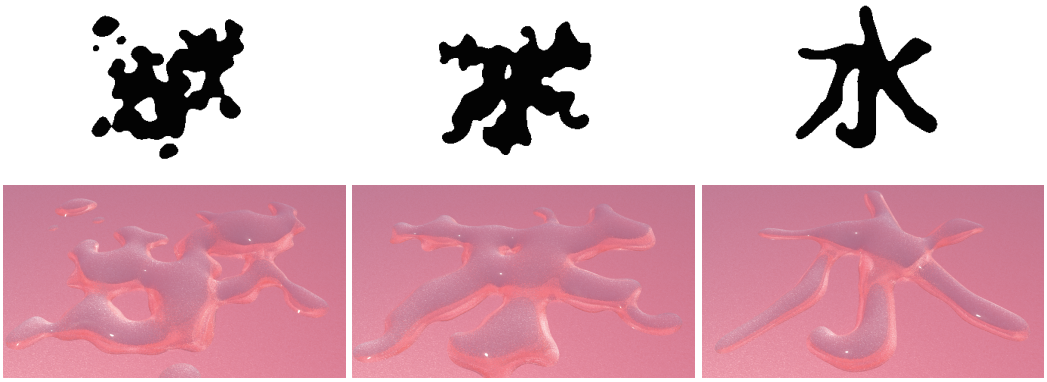

Figure 4.12: Animated droplets.

region surrounded by a liquid suddenly disappears (Figure 4.1). Our simple method can also reproduce this phenomenon well.

4.2.5 Limitations

Our simple method of course has some limitations. We assumed the droplet surface as a single valued function, and approximated the surface area computation using the Taylor expansion. These assumptions pose limitations on the droplet shapes that our method can generate. For example, if a large droplet is flowing

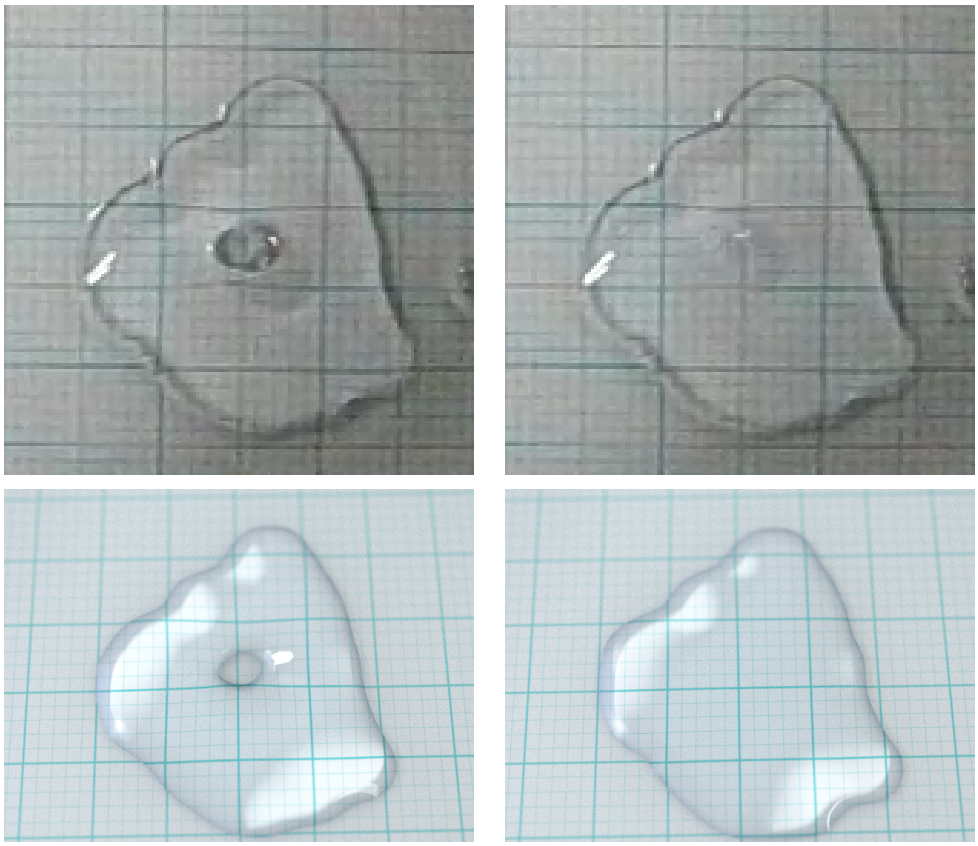

Table 4.1: The phenomenon wherein a dry region surrounded by a liquid suddenly disappears; in the real world (top row) and from our results (bottom row).

down a window, its surface cannot to be represented as a single valued surface. For our quasi-static applications, we found that our method is still useful, however. For static and small droplets, as shown in Figure 4.13, they stay on the surface even when the surface is near vertical, and the droplet surfaces can still be viewed as single valued functions if we take a planar surface close to the object surface as the xy plane in our method. For such a case, we can even generate droplet shapes assuming a nearly horizontal object surface, and then map the droplet shapes onto the vertical object surfaces, like texture mapping.

If a physically invalid situation is specified, the generated shape could also be

Figure 4.13: Droplets on a nearly vertical surface (photograph).

invalid. For example, if we specify a large droplet in a circle shape on a tilted surface as shown in Figure 4.14, the tail of the droplet could get into the object surface. In reality, the liquid will also be pulled by the gravity, so that such a circle shape cannot stay in its position. We believe we can also handle this case if we

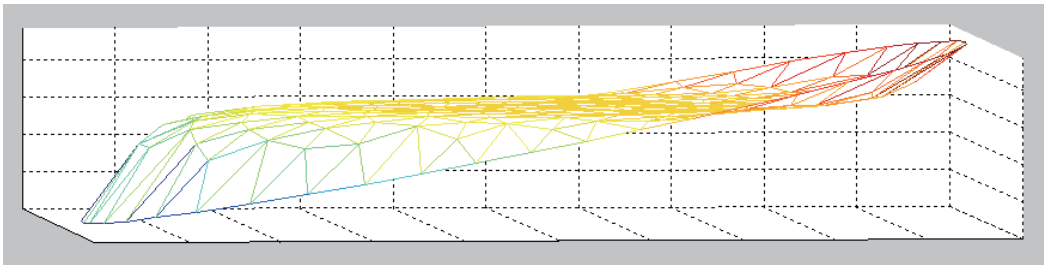

Figure 4.14: 3D droplet shape in a physically invalid situation. The color of the edges shows the height value.

require $u \geq h$.

Chapter 5

Conclusion and Future Work

We focused on surface tension effect on both of droplet motion and droplet shape. By considering the effect on droplet motion, the proposed method in Chapter 3 reproduced the behaviour of a group of water droplets on a hydrophobic windshield. We modeled each of large droplets as a mass point and took into account dynamic hydrophobicity by employing the contact angle hysteresis which causes appropriate adherence for each droplet. We also compared the accelerations of simulated droplets with those of measured real water droplets from literature of surface finishing engineering analysis. By introducing a near boundary layer where the wind is reasonably weakened, our result matched the measured one and reproduced realistic behaviours of the droplets.

For a huge number of tiny water droplets which do not move in our model, we introduced a normal map applied to the windshield. By using the image-based droplets, the Lotus effect was effectively reproduced.

For practical number of large droplets, our method runs in real-time and can be easily adopted as an effect for video games and vehicle simulators. The performance is degraded when the large droplets are not blown off and accumulated on the windshield because the motion simulation is done on a per large droplet basis.

By considering surface tension effect on droplet shape, the proposed method in Chapter 4 reproduced the droplet shape of natural-looking from the adhered region of a droplet. In this method, the generation of the droplet shape is decomposed into the design of the droplet footprint and the computation of the 3D droplet shape from the footprint. Given user specified 2D footprints of the droplets, our method automatically computes the 3D droplet shapes. Our method faithfully captures the physical requirements by considering an energy optimization problem, which minimizes the sum of the surface and gravitational energies of the droplets, satisfying volume conservation. We have demonstrated that we can generate not only natural droplet shapes but also artificial and artistic shapes, including animations.

For future work, we would like to take account of wind pressure and surface tension for droplet deformation in our animation method. Additionally we would like to extend our shape generation method to model the shapes of spikes in ferrofluids by also considering the magnetic energy.

References

[1] Ayako Hashimoto, Munetoshi Sakai, Jeong-Hwang Song, Naoya Yoshida, Shunsuke Suzuki, Yoshikazu Kameshima, and Akira Nakajima. Direct Observation of Water Droplet Motion on a Hydrophobic Self-assembled Monolayer Surface under Airflow. *Journal of the Surface Finishing Society of Japan*, Vol. 59, No. 12, pp. 907–912, 2008.

[2] JS Marshall and W. McK Palmer. The distribution of raindrops with size. *Journal of Meteorology*, Vol. 5, pp. 165–166, 1948.

[3] Young-Jung Yu, Ho-Youl Jung, and Hwan-Gue Cho. A new water droplet model using metaball in the gravitational field. *Computers & Graphics*, Vol. 23, No. 2, pp. 213–222, 1999.

[4] Kazufumi Kaneda, Yasuhiko Zuyama, Hideo Yamashita, and Tomoyuki Nishita. Animation of water droplet flow on curved surfaces. *Proceedings of Pacific Graphics*, pp. 50–65, 1996.

[5] Kazufumi Kaneda, Shinya Ikeda, and Hideo Yamashita. Animation of water droplets moving down a surface. *The Journal of Visualization and Computer Animation*, Vol. 10, No. 1, pp. 15–26, 1999.

[6] Natalya Tatarchuk. Artist-directable real-time rain rendering in city environments. In *ACM SIGGRAPH 2006 Courses*, 2006.

[7] Patrick Fournier, Arash Habibi, and Pierre Poulin. Simulating the flow of liquid droplets. In *Graphics Interface*, pp. 133–142. Citeseer, 1998.

[8] Huamin Wang, P.J. Mucha, and Greg Turk. Water drops on surfaces. *ACM Transactions on Graphics (TOG)*, Vol. 24, No. 3, pp. 921–929, 2005.

[9] Nils Thürey, Chris Wojtan, Markus Gross, and Greg Turk. A multiscale approach to mesh-based surface tension flows. *ACM Transactions on Graphics (TOG)*, Vol. 29, No. 4, p. 48, 2010.

[10] Yizhong Zhang, Huamin Wang, Shuai Wang, Yiying Tong, and Kun Zhou. A Deformable Surface Model for Real-Time Water Drop Animation. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 18, No. 8, pp. 1281–1289, August 2012.

[11] Janne T. Hirvi and Tapani A. Pakkanen. Nanodroplet impact and sliding on structured polymer surfaces. *Surface Science*, Vol. 602, No. 10, pp. 1810–1818, May 2008.

[12] Munetoshi Sakai, Jeong-Hwan Song, Naoya Yoshida, Shunsuke Suzuki, Yoshikazu Kameshima, and Akira Nakajima. Relationship between sliding acceleration of water droplets and dynamic contact angles on hydrophobic surfaces. *Surface Science*, Vol. 600, No. 16, pp. L204–L208, August 2006.

[13] Denis Richard and David Quéré. Viscous drops rolling on a tilted non-wettable solid. *EPL (Europhysics Letters)*, Vol. 48, No. November, p. 286, 1999.

[14] Shunsuke Suzuki, Akira Nakajima, Yuu Sakurada, Munetoshi Sakai, Naoya Yoshida, Ayako Hashimoto, Yoshikazu Kameshima, and Kiyoshi Okada. Mass Dependence of Rolling/Slipping Ratio in Sliding Acceleration of Wa-

ter Droplets on a Smooth Fluoroalkylsilane Coating. *Shikizai kyokaishi*, pp. 3–8, 2009.

[15] Adrien Treuille, Antoine McNamara, Zoran Popović, and Jos Stam. Keyframe control of smoke simulations. *ACM Trans. Graph. (Proc. of SIGGRAPH 2003)*, Vol. 22, No. 3, pp. 716–723, 2003.

[16] Antoine McNamara, Adrien Treuille, Zoran Popović, and Jos Stam. Fluid control using the adjoint method. *ACM Trans. Graph. (Proc. of SIGGRAPH 2004)*, Vol. 23, No. 3, pp. 449–456, 2004.

[17] Raanan Fattal and Dani Lischinski. Target-driven smoke animation. *ACM Trans. Graph. (Proc. of SIGGRAPH 2004)*, Vol. 23, No. 3, pp. 441–448, 2004.

[18] Lin Shi and Yizhou Yu. Taming liquids for rapidly changing targets. In *Proc. of SCA '05*, SCA '05, pp. 229–236, New York, NY, USA, 2005. ACM.

[19] Yoshinori Dobashi, Katsutoshi Kusumoto, Tomoyuki Nishita, and Tsuyoshi Yamamoto. Feedback control of cumuliform cloud formation based on computational fluid dynamics. *ACM Trans. Graph. (Proc. of SIGGRAPH 2008)*, Vol. 27, No. 3, pp. 94:1–94:8, 2008.

[20] Sören Pirk, Ondrej Stava, Julian Kratt, Michel Abdul Massih Said, Boris Neubert, Radomír Měch, Bedrich Benes, and Oliver Deussen. Plastic trees: interactive self-adapting botanical tree models. *ACM Trans. Graph. (Proc. of SIGGRAPH 2012)*, Vol. 31, No. 4, pp. 50:1–50:10, 2012.

[21] Kazufumi Kaneda, Takushi Kagawa, and Hideo Yamashita. Animation of Water Droplets on a Glass Plate. In Proc. of Computer Animation ' 93, pp. 177–189, 1993.

[22] Alain Carre and Martin E. R. Shanahan. Drop Motion on an Inclined Plane and Evaluation of Hydrophobia Treatments to Glass. *The Journal of Adhesion*, Vol. 49, No. 3-4, pp. 177–185, 1995.

[23] José Bico, François Besselievre, and Marc Fermigier. Windswept droplets. In *Bulletin of the American Physical Society*. APS, 2005.