# Computational Hydrographic Printing

Yizhong Zhang[*]    Chunji Yin[*]    Changxi Zheng[†]    Kun Zhou[* ‡]

[*] State Key Lab of CAD&CG, Zhejiang University    [†] Columbia University

## Abstract

Hydrographic printing is a well-known technique in industry for transferring color inks on a thin film to the surface of a manufactured 3D object. It enables high-quality coloring of object surfaces and works with a wide range of materials, but suffers from the inability to accurately register color texture to complex surface geometries. Thus, it is hardly usable by ordinary users with customized shapes and textures.

We present *computational hydrographic printing*, a new method that inherits the versatility of traditional hydrographic printing, while also enabling precise alignment of surface textures to possibly complex 3D surfaces. In particular, we propose the first computational model for simulating hydrographic printing process. This simulation enables us to compute a color image to feed into our hydrographic system for precise texture registration. We then build a physical hydrographic system upon off-the-shelf hardware, integrating virtual simulation, object calibration and controlled immersion. To overcome the difficulty of handling complex surfaces, we further extend our method to enable multiple immersions, each with a different object orientation, so the combined colors of individual immersions form a desired texture on the object surface. We validate the accuracy of our computational model through physical experiments, and demonstrate the efficacy and robustness of our system using a variety of objects with complex surface textures.

**CR Categories:** I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Color, shading, shadowing, and texture

**Keywords:** 3D printing, hydrographics, water transfer printing, physical simulation, viscous sheets, texture mapping

## 1 Introduction

Recent advances in direct digital fabrication — or "3D printing" — promise fast prototyping of objects with almost arbitrary shapes. In parallel, numerous computational methods have been devised for fabricating shapes with desired geometric (e.g., [Chen et al. 2014]), physical (e.g., [Bickel et al. 2010; Stava et al. 2012; Skouras et al. 2013]) and reflectance properties (e.g., [Weyrich et al. 2009; Lan et al. 2013]). Perhaps equally important as the shape fabrication are methods that color the surface of a fabricated piece

[*]{yizhongzhang,yinchunji}@zju.edu.cn, kunzhou@acm.org

[†]cxz@cs.columbia.edu

[‡]Corresponding author

**Figure 1: Colorful Bunny.** *We present a new computational approach for hydrographically printing arbitrary color textures on complex 3D surfaces and aligning the textures precisely. The input of our system includes the desired surface texture (upper left) of a physical shape (lower left). Through our hydrographic system, we color the surface of the 3D shape using the desired texture (right).*

with a specified texture and work with complex surface geometry made of a wide range of materials. Such a method, unfortunately, has not been explored, and thus is the focus of this paper.

Currently, most of the 3D printed shapes at home are monochromatic, although high-end 3D printers offer the ability to fabricate colored models. These printers often support only a few colors and limited materials (e.g., plastic and sandstone), and their maintenance and printing cost can be unaffordable for many personal uses. In industrial production, there exist many ways for decorating 3D surfaces. These methods, including chemical- and electro-plating [Schlesinger and Paunovic 2011], decals [Hopper 2004], enamels [Darty 2004], and even manual painting, require heavy-duty devices, high operational cost, and skilled personnel; and they can still be limited by complex surface geometries. All these factors render them almost impossible for coloring customized 3D surfaces for personal uses.

In this paper, we propose *computational hydrographic printing*, a computational augmentation of hydrographic printing methods for physically decorating 3D surfaces with user-customized color textures. Our method works for complex surface geometries and a wide range of materials including plastic, wood and porcelain. The whole system is easy to set up for personal uses, and enjoys a low operational cost (less than 40 US cents per printing).

**Traditional hydrographic printing.** Hydrographic printing (also named hydroprinting or water transfer printing) is a well-known technology for transferring color images to a 3D surface [Wikipedia 2014]. It starts by printing a pixel image on a polyvinyl alcohol (PVA) film (Figure 3) using a conventional inkjet printer. This PVA color film is then put on top of water to carry color inks floating there. After that, an activator chemical is sprayed on the film, softening the color film to make it easy to stretch and also activating a binding agent. Lastly, a substrate object is slowly dipped into the water through the floating film. After

**Figure 2: Traditional method vs our method. (left)** *Traditional hydrographic printing is mainly used for coloring 3D shapes (inset) with repeated color patterns, because it lacks a method to (i) map a surface texture to a printable 2D image and (ii) precisely register the color texture on surface.* **(right)** *Our method solves these problems, so user-specified color textures can be hydrographically printed.*



**Figure 3: PVA film with a printed color pattern. (left)** *The PVA film is attached on an A4 paper, so it can be fed into a conventional inkjet printer to print prescribed color patterns.* **(right)** *the film is put on water and ready for transferring colors to an object surface.*

touching the object, the film becomes stretched, wraps around the object surface and adheres there. During this process, the color ink printed on the PVA film moves with the film and eventually gets transferred to the surface (see supplementary video). Applicable to a wide range of materials, hydrographic printing is able to produce high-quality colors on curved surfaces (Figure 2).

However, when used for coloring a complex surface with a user-specified texture, hydrographic printing suffers from fundamental limitations. First, it is extremely hard to precisely align a color film to the object surface, because the object stretches the color film while the film floats on water in the process of immersion. In addition, objects with complex surface shapes can severely distort the color film or even tear it apart. As a result, current hydrographic printing is limited to transferring repeatable color patterns (e.g., marble and camouflage patterns) for which precise registration of the color pattern is of little importance.

**Our proposal: a computational approach.** We address these limitations by developing a computational model for hydrographic printing. We model the color film floating on water as a thin viscous sheet subject to boundary conditions determined by the object immersion process. This model allows us to predict the stretch and distortion of color films using numerical simulation, and in turn to create a map between the colors on the film and the surface locations where they are transferred to. With this map, we compute a color image for printing on the PVA film which, after the hydrographic immersion, forms the desired color texture on the object surface.

The computation of color images printed on PVA films can be useful in practice only when the setup of a virtual simulation is identical to that in a physical hydrographic process. To this end, we designed a calibrated system, wherein the object immersion is precisely controlled by a mechanical apparatus, and the object orientation and its dipping location are measured by a 3D vision system (Figure 8). Built upon off-the-shelf hardware, this system can be easily set up by ordinary users. Together with our virtual simulation model, it offers high-quality texture alignment in hydrographic printing and thus enables almost any color texture to be transferred on an object surface.

We further avoid severe film distortion and the danger of tearing a film by introducing a multi-immersion scheme: the object can be dipped multiple times each with a different orientation and a film printed with a different color pattern. The computation of color patterns allows the transferred colors from individual immersions to be combined into a desired surface texture.

**Contributions.** In summary, our key contribution is the first computational hydrographic printing method for coloring 3D objects

with user-specified and registered surface textures. In particular, we highlight the following technical contributions:

- We develop the first computational model for simulating the distortion of color films in a hydrographic printing process.

- We design a controlled hydrographic printing system, integrating together the virtual simulation, system calibration, and controlled immersion.

- We further extend our method to enable multiple immersions, making our system more robust and effective when handling complex surface shapes.

## 2   Related Work

**3D Printing and appearance fabrication.** Under the umbrella of 3D fabrication design, numerous methods have been developed for fabricating shapes with desired physical properties, such as stress distributions [Stava et al. 2012], articulated characters [Bächer et al. 2012; Calì et al. 2012], shape balance [Prévost et al. 2013], motion sequences [Coros et al. 2013; Ceylan et al. 2013; Thomaszewski et al. 2014], and deformation behaviors [Bickel et al. 2009; Skouras et al. 2013]. But none of these methods focuses on fabricating surface colors.

Other methods have been proposed for fabricating materials with desired appearance properties, such as surface reflectance [Weyrich et al. 2009; Matusik et al. 2009; Lan et al. 2013], subsurface scattering [Dong et al. 2010; Hašan et al. 2010], and reflectance functions [Malzbender et al. 2012]. These methods fabricate the desired appearance by designing heterogeneous material distributions, and thus are often limited to a certain type of material. In contrast, our goal is to precisely decorate 3D objects with user-specified color textures. Our method is "nonintrusive", making no change of substrate objects. It is also applicable for a wide range of materials.

**Coloring object surfaces.** As briefly discussed in the introduction, there exist various methods for producing colored objects, and each has its own merits. 3D color printers offer high-quality reproduction of surface colors, but are limited by usable colors and materials. Their expensive cost is also a hurdle for ordinary users. Vinyl decals [Hopper 2004] can transfer arbitrary color patterns to objects, but are limited to simple geometries such as developable surfaces. Both plating and enameling [Schlesinger and Paunovic 2011; Darty 2004] require complex devices and are applicable only to limited types of materials such as metal. Most importantly, they are all costly and hardly accessible by ordinary users. Hydrographic printing has the advantages of supporting both complex textures and various materials including metal, plastic, and hard woods. It has been extensively used to decorate furniture, electronic products, and automotive accessories.
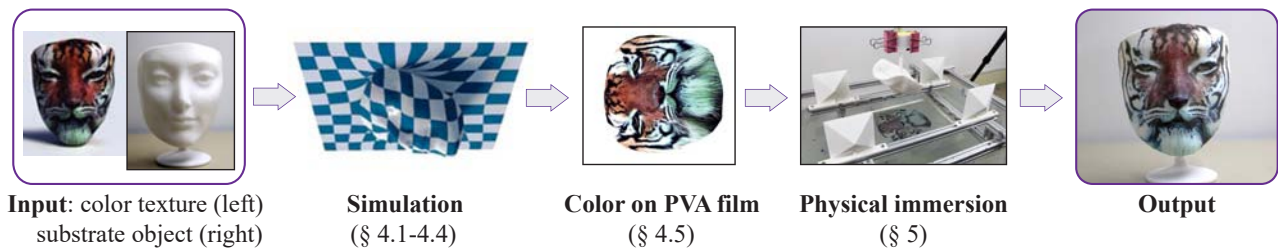
**Figure 4: System pipeline.** *Our system takes as input a physical substrate object and its desired surface texture. After attaching the object into a gripper and before the immersion, we measure its orientation and dipping location (§5.1), which is used to set up a virtual simulation. After simulating the color film distortion in hydrographic printing, we compute a color image and print it on a PVA film (§4). Finally, we immerse the object into the water through the floating color film and transfer a desired color pattern to the object surface.*

Despite its wide application, to the best of our knowledge there is no computational model to date for simulating the dipping process. Thus, the traditional hydrographic printing is "uncontrollable" in the sense that the surface location where the color ink is transferred to is unpredictable. These limitations render it virtually impossible to generate registered 3D surface textures such as those shown in our examples.

**Viscous sheet simulation.** We build our hydrographic model on viscous fluid models [Carlson et al. 2002; Rasmussen et al. 2004; Bargteil et al. 2007; Wojtan and Turk 2008; Wicke et al. 2010]. These models simulate viscous fluids based on the theory of continuum mechanics. In particular, our model is related to Stokes flow [White and Corfield 1991], which is an approximation of the Navier-Stokes equation in the regime of low Reynolds numbers, as well as the viscous sheet model [Ribe 2002; Batty et al. 2012]. We tailor these models for a specific simulation of color films' distortion: by constructing a proper boundary condition defined on the object's waterlines, we are able to simplify the deformation of color films into a purely 2D process (§4), which in turn enables fast and accurate prediction of films' distortion in the process of hydrographic printing.

**Texture mapping.** Texture mapping is a widely used tool in production for (*virtually*) decorating 3D models. Numerous methods have been proposed for calculating a map between a planar image and a 3D surface, aiming to minimize angle or area distortion [Sander et al. 2001; Lévy et al. 2002; Desbrun et al. 2002] as well as to satisfy user-specified feature correspondence [Kraevoy et al. 2003; Zhou et al. 2005]. We refer interested readers to the comprehensive survey of [Sheffer et al. 2006] for more elaborate discussion. The goal in this paper is to produce a *physical* realization of a 3D model with user-customized color textures. To this end, we employ a viscous sheet model to simulate the distortion of a color film during object immersion, and compute a map between the film and the object surface.

## 3   Overview

We begin by summarizing the procedures of our hydrographic printing system. Figure 4 depicts the pipeline, in which the key components include a virtual simulation method for predicting the color film distortion and computing a pixel image printed on the film prior to the hydrographic printing, as well as a 3D vision system for precisely measuring and controlling the dipping process. Briefly, the steps of our method are:

1. **System setup.** For precisely controlling the printing process, we first set up a calibrated system, in which the object is held firmly by a gripper and can be immersed into the water with a constant dipping velocity (Figure 8 and §5.1). We further

develop a 3D vision system to measure the object's orientation and dipping location (§5.2) for computationally predicting color film distortion during the immersion.

2. **Virtual simulation.** Before the start of physical hydrographic printing, we simulate it virtually. In particular, we wish to predict how the PVA color film is distorted during the printing before adhering to the object. To this end, we develop an efficient and accurate simulation model (§4.1–4.4).

3. **Computation of printed color patterns.** Through the simulation, we establish a map between points on the color film and their locations when transfered to the object surface. This map in turn allows us to compute a pixel image printed on the color film according to the object's surface textures (§4.5).

4. **Physical color transfer.** Finally, after printing the computed pixel image on the color film, we place it on water, fix its location, and start the physical hydrographic printing process to transfer colors to the object surface. Furthermore, to robustly print colors on a complex surface geometry, we extend our method to enable multiple immersions using different object orientations (§6).

## 4   Simulation of Hydrographic Printing

A key step of our hydrographic printing system is the computation of a pixel image to print on a PVA film such that after transferred to the object's surface, the colors form a desired texture. The input of this step from the user includes the 3D geometry of an object and the desired surface color field. In our system, we attach the object on a mechanical gripper and dip it into the water vertically with a fixed orientation (Figure 8). Both the object orientation and dipping location are measured by our 3D vision system described in §5 and thus are known in this step of computation.

The central goal of the simulation step is to establish a map that maps points on the color film to points on the object surface when the points are transferred to the object surface (Figure 5). Later, provided the desired colors on the object surface, we use the inverse of this map to determine the color patterns that need to be printed on a color film. To obtain the map, we seek a physical model to simulate the printing process and track the transfer of points of the color film.

We also note that, while the color film stretches and wraps the object surface during the printing process, one can not establish the map by simply unfolding the object surface geometrically. This is because the stretch of the film depends on the dipping process and is non-uniform. In fact, as shown in our results (§7), the stretch is heterogeneous and nonlinear, necessitating a dedicated
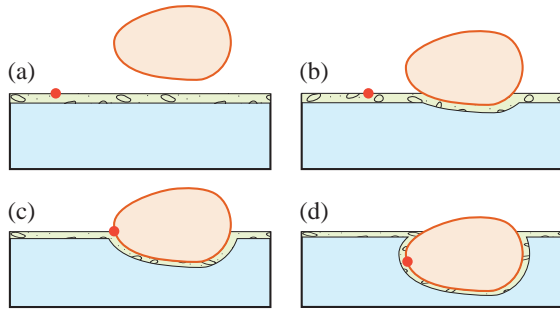
**Figure 5: Hydrographic dipping.** **(a)** *An object starts to move downward; the red particle is a point on the color film on top of the water.* **(b)** *The object starts to sink into the water. Part of the color film adheres to the object; and the rest part gets stretched, so the red particle moves.* **(c)** *The object continues to sink. The red particle arrives object surface and sticks there.* **(d)** *Since adhering, the red particle is fixed on the surface and moves with the object.*

physical simulation.

### 4.1 Color Films as Viscous Sheets

At the beginning of the hydrographic printing, a PVA film is placed on top of water, carrying color inks. Then, an activator (made by a plasticizer dissolved in a volatile solvent) is sprayed on the film. Under the chemical reaction with the activator, the film and ink quickly become soft and adhesive, forming a *thin and viscous* sheet on the water. When an object is lowered into the water and touches the viscous sheet, the ink along with polyvinyl alcohol wraps around the object geometry and adheres tightly to the surface. In common practice, the boundary of the film is attached with the water vat and fixed there. After a part of the viscous sheet adheres to the object's surface, that part is immersed into the water together with the object, and the rest of the viscous sheet remains on the water but gains velocity and starts to stretch (Figure 5).

**2D model.** These observations suggest that we can simulate the film's distortion incurred by dipping an object using a thin viscous sheet model. Furthermore, since the object is dipped into water at a constant and slow speed—for instance, in our system, the object's vertical speed is 5mm/s, it is reasonable to assume that (i) the water underneath the viscous film is quasistatic; (ii) the fluid velocity in the viscous layer is invariant along the vertical direction; and (iii) the viscous sheet only moves horizontally on the water until it reaches the object and immediately adheres to the surface (Figure 5(c)). Under these assumptions, we further simplify the viscous sheet model to simulate it as a 2D plane region rather than a 3D volume. In this 2D region, the film stretches and moves toward the boundary, and adheres to the object surface on the boundary. Moreover, we ignore friction forces on the interface between water and the viscous sheet, because the ink material is hydrophobic, so there is an absence of intermolecular attraction in-between.

While we can simplify the simulation using a 2D model, special care needs to be taken to specify its boundary condition. On the boundary where the film is attached to the vat, the velocity vanishes. On the other boundary where the film touches the dipping object at a point $x$, the velocity direction is along the normal direction of the boundary projected on the water surface, and the velocity magnitude agrees with the surface's velocity at the same point $x$ on the object's surface, because the film is continuously extended to object surface and attached there. Formally,



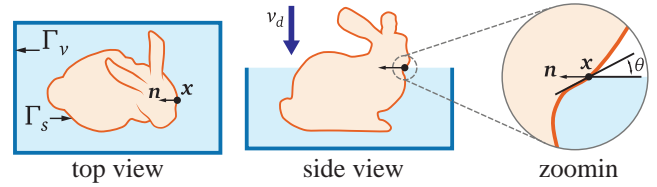**Figure 6: Boundary conditions**

let $u(x), x \in \mathbb{R}^2$ denote the *2D* velocity field on the viscous film. In our system, we lower the object vertically. Thus, we derive the boundary condition of $u(x)$ and obtain

$$u(x) = \frac{1 - \cos\theta}{\sin\theta} v_d n, \quad x \in \Gamma_s,$$
$$u(x) = 0, \quad\quad\quad\quad x \in \Gamma_v, \quad\quad (1)$$

where $\Gamma_s$ and $\Gamma_v$ denote the film boundaries touching the object and the vat, respectively; $v_d$ is the constant vertical dipping velocity; $\theta$ is the angle between the surface tangential and horizontal waterline; and $n$ is the normal direction on the 2D film pointing outside of the film region. Moreover, we note that while $\Gamma_v$ is fixed throughout the hydrographic printing process, $\Gamma_s$ changes, depending on the current immersion depth of the object.

With the boundary condition established, the remaining question is to simulate the film distortion inside of the boundary. To this end, we propose the following two approaches with different performance-accuracy trade-offs. In §4.3, we combine them together to exploit the advantages of both approaches when we initialize the simulated film thickness.

### 4.2 Stokes Flow

As stated, in a hydrographic printing process, the color film becomes viscous after reacting with the activator chemical. Meanwhile, the substrate object is immersed slowly. Thus, the viscous film has a low Reynolds number [1] [Katz and Plotkin 2001], suggesting that we can simulate the viscous film in a special dynamical regime, referred to as *Stokes flow* [White and Corfield 1991]. In 2D, Stokes flow is described by a simple Laplace's equation,

$$\nabla^2 u(x) = 0, \ x \in \Omega, \ s.t. \text{ boundary condition (1)}, \quad (2)$$

where $\Omega$ denotes the region of viscous film on the water; it changes as we immerse the object. In our numerical simulation described in §4.4, we discretize the time domain, and solve this Laplace's equation at each timestep. We note that a viscous film can be stretched on the horizontal plane while getting compressed vertically to preserve its total volume. Therefore, when only simulating the 2D motion of a film in Equation (2), we ignore the divergence-free constraint that a standard fluid simulator normally needs.

By leveraging full-fledged software (e.g., [Lukarski 2013]) for performing the numerical solves of the Laplace's equation, simulating Stokes flow is fast and easy to implement. As will be demonstrated in §7.1, Stokes flow is predictive when the color film has a relatively small distortion, which typically occurs at the early stage of the immersion process. However, when the object is immersed deeply in the water, the viscous film becomes largely distorted. Therein, the nonlinear distortion becomes dominant,

---

[1]Reynolds number, defined as $Re = UL/\mu$, is a dimensionless parameter in the Navier-Stokes equation for a viscous fluid. Here $U$ is the object relative velocity; $L$ is the characteristic linear dimension of the object; and $\mu$ is the kinematic viscosity.

and thus the linear Stokes flow model is insufficient. We therefore simulate the film distortion using a simple but nonlinear viscous sheet model.

### 4.3 Nonlinear Viscous Sheets

Since a PVA film behaves like a Newtonian viscous fluid after casting a chemical activator solution [Wright et al. 2013], we build our nonlinear viscous sheet model on the method of Batty et al. [2012], and refer to their paper for the background of viscous sheets. Following the discrete Stokes-Rayleigh analogy [Rayleigh 1896] used in their method, we simulate an elastic 2D shell at each timestep, followed by a *viscous update* to set the current undeformed strain to the elastic strain. In particular, elastic forces, $f_{int}(x)$, follow from taking the variations of elastic energy with respect to the material coordinates. These forces then deform the viscous sheet in a dynamical equation,

$$\rho u + f_{int}(x) = 0. \tag{3}$$

Different from the method of Batty et al. [2012], we consider specific physical characteristics of hydrographic printing, and tailor the viscous sheet model for efficient simulation. First, when computing the elastic strain energy, we ignore the thin-shell bending energy and consider only the membrane energy. This is because the viscous film in hydrographic printing moves on the horizontal plane on the quasistatic water; it is flat everywhere in the simulated region, and thus has no bending energy at all. The membrane energy is an integral of the membrane energy density, $W_m(x)$, $x \in \Omega$, over the entire film on the water. As developed in [Gingold et al. 2004], the membrane energy density is defined in terms of strain tensors:

$$W_m(x) = \frac{Yh(x)}{2(1-\nu^2)}((1-\nu)\mathbf{tr}(\epsilon^2) + \nu\mathbf{tr}(\epsilon)^2), \tag{4}$$

where $h(x)$ is the film thickness at position $x$; $\mathbf{tr}$ is the trace operator of a tensor; $Y$ is Young's modulus, and $\nu$ is Poisson's ratio. Simulating incompressible viscous fluids gives $\nu = 0.5$; the value of $Y$ is estimated through experiments (described in §7). The membrane strain $\epsilon$ in (4) is the Green strain tensor for measuring the 2D nonlinear geometric deformation of the film (see the definition in [Gingold et al. 2004]). With this energy definition, the (nonlinear) internal membrane force is written as an integral over the entire viscous sheet area $\Omega$,

$$f_{int}(x) = \nabla_x \left( \int_{\Omega} W_m(x) dS \right). \tag{5}$$

**Nonuniform film thickness.** Setting the film thickness $h(x)$ is nontrivial. One may simply assume the film thickness $h$ is a constant everywhere at each simulation timestep, and dynamically update the film thickness at the end of each timestep based on volume conservation (details described later in this subsection). However, as we print color images on the film, the color inks sprayed by the printer in fact increase the thickness of a film. The increment is nonuniform because of an arbitrary color pattern. Since the PVA film is very thin (about 0.5mm), the thickness increment due to color inks can make the film locally less stretchable, and eventually lead to a visible difference from the simulated results (Figure 7).

Unfortunately, prior to a simulation, we have no knowledge of the exact image to be printed on the film. We therefore set up the nonuniform film thickness using two steps: first, we assume a constant film thickness everywhere on the film and simulate the hydrographic printing process. This simulation enables us

**Algorithm 1** A timestep of simulating nonlinear viscous sheets

1: move dipping object downward with a constant velocity
2: update waterline $\Gamma_s$ (see Figure 6)
3: triangulate $\Omega$ with boundary $\Gamma_v$ and $\Gamma_s$
4: interpolate velocity field on new mesh using previous mesh
5: update boundary condition (1)
6: implicit Euler step of time integration
7: update film thickness: $h \leftarrow V_r/A_r$
8: interpolate particle velocities (§4.5)
9: advect particles

to compute an image to be printed on the film using the texture tracking algorithm presented in §4.5. Using this image, we further estimate the amount of color inks distributed on the film, and in turn compute the film thickness increment to initialize another simulation for predicting the film distortion. The two-step film thickness estimation can be performed iteratively to further adjust the thickness, although we found that in our examples only one iteration is sufficient to produce an accurate prediction (Figure 7). In this two-step estimation, we use the Stokes flow model in the first simulation to exploit its fast performance for setting the film thickness, and use the nonlinear viscous sheet model in the later simulation to leverage its accuracy.

We estimate the film thickness increment by computing the amount of ink that is sprayed on the film. When an image is printed on the film, it is discretized as a pixel image. We therefore convert each pixel color to the CMYK color space, which measures the amount of four printer inks (cyan, magenta, yellow and black) used in color printing using percentage values in the range $[0, 1]$. We sum the values of all the four channels together (4.0 at the most) as a measure, $t$, of the total amount of inks. The ink thickness is proportional to the measure, $t$. In practice, we found that a relationship $\Delta h = 0.05t$ mm leads to plausible results.

During the simulation, we dynamically change the film thickness, taking into account its local stretch to preserve the total volume of the film. At the end of each timestep, we update the film thickness using $h(x) = h_0(x)A_0(x)/A_r(x)$, where $h_0(x)$ is the initial film thickness at $x$, and $A_0(x)$ and $A_r(x)$ are respectively the initial and current local area at $x$, estimated using the local triangle mesh area on the discretized film (see §4.4).

### 4.4 Numerical Simulation

We use the finite element method to simulate both the Stokes flow and nonlinear viscous sheets. We represent the viscous film on the water using a triangle mesh. The velocity field $u(x)$ is stored at the positions of triangle vertices. Inside a triangle, its value is interpolated using first-order finite element basis functions (i.e., barycentric interpolation).

**Mesh adaptation for time-varying boundary.** In the process of object immersion, certain parts of the film boundary meet the object surface at the waterline. Since the object is lowered with constant velocity, we are able to compute the object's immersion depth at each timestep and thus update the waterline. We then remesh the film region on the water before integrating the dynamics of the current timestep. Since the waterline changes smoothly, we remesh the 2D region incrementally (using the meshing library, Triangle [Shewchuk 1996]). In our implementation, we use edge length of 1mm for $\Gamma_s$, a lower bound of 30° for triangle interior angle and an upper bound of 1cm² for triangle area. When simulating Stokes flow (§4.2), we construct the finite element linear system of Laplace's equation using the updated mesh. When simulating nonlinear viscous sheets (§4.3), one more step

**Figure 7: Influence of film thickness.** *We compare the texture alignment with and without nonuniform film thickness considered in the simulation.* **(left)** *We use the two-step simulation introduced in §4.3 to compute the printed image for hydrographic printing.* **(right)** *We assume a constant film thickness in the simulation for computing the printed image pattern.*

is needed: we perform barycentric interpolation to get the velocity values at updated mesh vertices using the velocities stored on the previous mesh, followed by a time integration of the dynamical equation (3). Algorithm 1 outlines the procedures of a single timestep for simulating nonlinear viscous sheets, mostly following the method [Batty et al. 2012].

**When to use which.** So far, we have presented two approaches for simulating hydrographic films. Simulating Stokes flow is faster than the nonlinear viscous sheet simulation, but is limited to predicting small film distortions. For instance, the simulation of dipping a sphere in Figure 12 takes 7 seconds using Stokes flow but 5 minutes using nonlinear viscous sheet. In our experiment, we found that if the maximum local stretch is less than 0.3 during the hydrographic printing, Stokes flow is sufficient to produce accurate results. Beyond that, we need to resort to the nonlinear viscous sheet model as presented in §4.3. On the other hand, as described, Stokes flow is suitable for estimating the inhomogeneous film thickness before starting the simulation of film distortion. Thanks to its low computational cost, it introduces little overhead but can greatly improve the accuracy of the nonlinear viscous sheet simulation (Figure 7). Additionally, in §6, we extend our system to use multiple immersions for printing a single texture. In that case, we use the Stokes flow to construct a surface texture decomposition for individual hydrographic printing. We defer the details until that section.

## 4.5 Texture Tracking

With the simulation methods depicted, we now create a map between the points on a color film and the object surface locations where the points are transfered to. Eventually, we will print on the color film a pixel image. The resolution of the image is limited by the printer, since it is unnecessary to have a resolution higher than what the printer can resolve. We therefore use the highest image resolution that the printer can resolve. To determine the color value at each pixel of the printed image, we assign a virtual particle at the center of each pixel on the color film and advect all particles with the velocity field $u(x, t)$ until they touch the object surface. We compute the color of the touching point using the surface texture mapping at that point, and the color is in turn assigned to the corresponding pixel of the image.

At each timestep, the advection velocity at each particle is interpolated in the triangle it is located in using finite element basis. For a particle originated at $x$ on the color film, we track its motion until it meets the object at $\tilde{x}$ on the surface. Then we create a map $\phi(x) : x \mapsto \tilde{x}$. After submerging the entire object, we collect
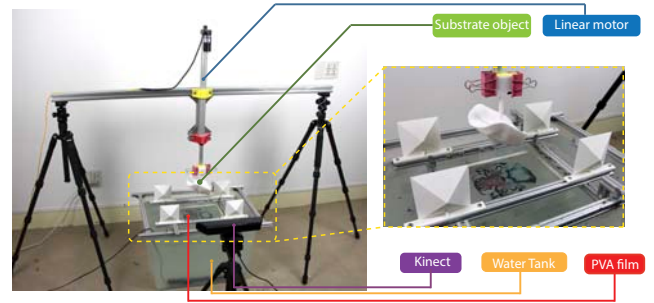
**Figure 8: System setup.** *Our hydrographic printing system consists of a vat filled with water to hold the color film, a linear motor to drive a mechanical gripper that holds the substrate object, and a Kinect for measuring the object's orientation and dipping location.*

all the particle maps and create an inverse map $\phi^{-1}(\tilde{x}) : \tilde{x} \mapsto x$, which allows us to map the desired texture color at object surface $\tilde{x}$ to a pixel color at $\phi^{-1}(\tilde{x})$ on the film. Assembling these maps together, we obtain a pixel image that needs to be printed.

## 5 Physical Realization

We now present our system to physically execute a controlled hydrographic printing. Our goal here is to build a system (§5.1) that can perform controlled object immersion consistent with the setup in the virtual simulation. To integrate our simulation method developed in §4, we build a 3D vision system (§5.2) to measure the object's orientation an dipping location to initialize the simulation.

### 5.1 System Setup

We build a hydrographic printing system that is able to precisely control the object orientation and dipping location. The entire system is built upon off-the-shelf hardware and can be easily set up by ordinary users. As shown in Figure 8, the object is held by a mechanical gripper connected at one end of a vertically placed aluminum rod. Driven by a linear DC motor, the rod can move upward or downward at a constant speed. In practice, we connect the motor to a voltage controller, which allows us to control the dipping velocity in a range from 0mm/s to 10mm/s. In all our examples, we set the velocity to be 5mm/s when immersing objects.

Below the rod is a vat filled with water. We place the printed color film on the water. To prevent the film from floating around, we place four sliding bars attached to the four sides of the film, respectively. We mark the locations of sliding bars to fix the films at the same position for every printing. We purchased the PVA films for color printing. Each film has a precise A4 paper size, so it can be supplied into a conventional inkjet printer to print any color pattern (Figure 3).

### 5.2 3D Acquisition

When we attach an object to the gripper for color transfer, it is difficult to precisely adjust its location and orientation. Yet both are needed to initialize a simulation. We bypass this problem by first attaching the object on the gripper and then measuring its location and orientation using a 3D acquisition method. With the measured object orientation and dipping location, we are able to run the simulation and in turn print a color film. After that, we start the immersion process.
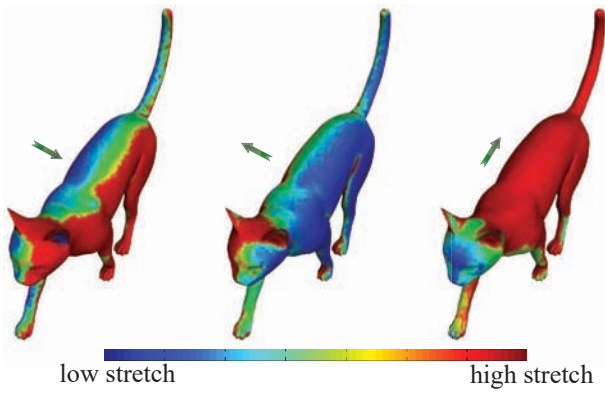
**Figure 9: Color film stretch.** *We visualize the color film stretch when the film adheres on the object surface. The colormap encodes the stretch coefficients (i.e., the largest singular value of the Jacobian matrix). The three plots shows respectively the distributions of film stretch when the cat is dipped along three different orientations (indicated by the green arrows).*

To measure the object location and orientation, we first reconstruct a point cloud of the 3D scene using a Microsoft Kinect and the KinectFusion method [Izadi et al. 2011]. To help locate the position of the color film on top of water, we place four pyramid shaped 3D markers on the sliding bar (Figure 8). Since we know the exact 3D models of the markers, we align them with the Kinect reconstructed 3D scene using the Iterative Closest Point (ICP) algorithm [Besl and McKay 1992]. The four markers indicate the location of the color film, establishing a frame of reference. We also use the ICP algorithm to align the 3D model of the object, which is provided by the user. The registered object 3D model provide us with its exact location and orientation with respect to the PVA film. In practice, the measured values result in an accuracy simulation with an error less than 1.5mm (§7.1).

**Pipeline summary.** Lastly, we summarize every step of our hydrographic printing system (Figure 4). The user provides the 3D geometry of an object that is to be colored, a desired color field on the surface, as well as a physical object identical to the given 3D geometry. We attach the object to the gripper and measure its location and orientation. This enables us to set up a virtual simulation to predict the color film distortion and in turn determine a pixel color image to print on the PVA film (recall §4.5). Finally, we print the film, place it on the water, and start the physical hydrographic printing.

## 6  Extension: Multiple Immersions

The hydrographic printing system that we have presented is able to produce high-quality surface color for many objects with a large variety of shapes (demonstrated in §7). The printing quality, however, deteriorates when the color film is stretched largely in the process of dipping. This is because when a color film is stretched extensively, its physical parameters such as the viscosity also change, making our virtual simulation initialized with fixed parameter values much less predictive. As the object gets immersed deeper into the water, the film stretch becomes larger and may even break. If an object has complex geometry or large surface area, it is almost unavoidable to stretch the color film extensively when we dip the object with a fixed orientation.

Another difficulty arises when part of the object surface is almost in parallel to water surface when submerged into water. In that case, the $\theta$ value in the simulation boundary condition (1) is



**Figure 10: Multi-immersion images.** *Each row corresponds to an orientation shown in Figure 9.* **(left)** *The complete images for printing using a single immersion;* **(middle)** *the blending mask for seamlessly combining multiple immersions;* **(right)** *the final images used for the multi-immersion printing.*

nearly 180°, leading to a large boundary velocity computed in (1). Consequently, the film and its underlying water move quickly, introduce large errors in the simulation.

**Dipping with multiple orientations.** We circumvent these difficulties by dipping the object multiple times each with a different orientation. This is workable because the color film itself is transparent if no ink is printed (by setting the corresponding pixel values to be completely white); transferring a transparent film to the object surface would not affect its appearance, and if a surface point is colored multiple times, its color blends the ink of every individual printing.

**Surface texture decomposition.** In light of this, we allow the user to set multiple orientations for dipping an object. Accordingly, we decompose the desired surface texture into multiple regions. During the physical hydrographic printing, every immersion colors a single region of the surface texture; and their combination results in a complete surface texture. Therefore, we need to compute a color image to print on a PVA film for each orientation. Given a surface point, we assign its color with one particular film that, when coloring that point, has the least distortion among all color films used in the immersions. To specify each orientation, the user attaches the object to the gripper in a desired orientation. Without physically dipping the object, our 3D vision system as described in §5.2 measures the exact object orientation and location.

Concretely, given an orientation indexed by $i$, we simulate the hydrographic printing using Stokes flow presented in §4 to perform fast stretch estimation. By tracking particles on the film during the simulation (recall §4.5), we obtain a map $\phi_i(x)$ that maps a 2D position on the film to a 3D position on object surface. The Jacobian of this map describes how a local region on the film transforms to the corresponding area on the surface. Its maximum singular value reflects the film's largest local "stretch" at $x$. In practice, we compute the Jacobian for each surface triangle, using the method described in [Sander et al. 2001]: given a triangle with 3 vertices, $q_1, q_2, q_3 \in \mathbb{R}^3$, suppose their corresponding 2D locations on a color film are $p_1, p_2, p_3 \in \mathbb{R}^2$, respectively. There exists a unique affine map $S(p) : p \mapsto q$ for positions on the triangle,

$$S(p) = \frac{1}{\langle p_1, p_2, p_3 \rangle} \left( \langle p, p_2, p_3 \rangle q_1 + \langle p_3, p_1, p \rangle q_2 + \langle p_2, p, p_1 \rangle q_3 \right),$$
$$(6)$$

where $\langle \cdot, \cdot, \cdot \rangle$ denotes the triangle areas formed by the three points. The Jacobian of this affine map is a constant $3 \times 2$ matrix, whose largest singular value reflects the largest stretch in the triangle. We refer to [Sander et al. 2001] for detailed derivations. After computing the largest singular value for all surface triangles, the stretch at every vertex using the orientation $i$ is estimated using an averaged stretch of all triangles incident to the vertex. We repeat this process for each orientation $i$. Figure 9 illustrates colormapped vertex stretch measures under three different orientations.

Lastly, we associate with each surface vertex an orientation that leads to the smallest stretch measure at that vertex. This process splits surface vertices, and thus decomposes the surface texture into multiple regions. Each surface region is hydrographically printed using an individual immersion.

**Computing individual images on films.** The previous step of surface texture decomposition indicates where to print colors on a film in each individual hydrographic printing, but we compute the exact color to print on a film by running additional viscous sheet simulations after the surface texture is fully decomposed. In this step, the user attaches the object to the gripper multiple times again, and every time, the object is held with the same orientation as used for texture decomposition in the previous step. However, different from the previous step, after the object is attached and measured with the 3D acquisition system, we run the simulation, and compute the exact color image to be printed, taking into account the image decomposition (Figure 10). We then perform the physical hydrographic printing immediately. In this way, the film color mapping is predicted using the object locations and orientations precisely agreeing with the physical experiment. Since the previous step only labels which part of the film needs to be printed, it would not affect the print quality significantly as long as the object orientations align closely with the ones used for texture decomposition.

**Boundary blending.** Multiple immersion has the advantage of using less stretched color films to color the object surface. However, even a small calibration error in the printing process can lead to color gaps or seams at the boundaries of texture regions. We address this problem by simply fading out texture regions at boundaries. Given an orientation $i$, it has a region, $\Theta_i$, of vertices assigned to color using that orientation. A blending weight $w_{i,j} = 1$ is assigned to every vertex $j$ in this region. For every vertex $j$ outside of this region, we compute the blending weight $w_{i,j}$ using a simple tent function,

$$w_{i,j} = 1 - \frac{d_{i,j}}{H}, \text{for } d_{i,j} \leq H, \quad \text{and } w_{i,j} = 0, \text{otherwise}, \qquad (7)$$

where $d_{i,j}$ is the geodesic distance of vertex $j$ to the nearest boundary of $\Theta_i$; and $H$ is a scalar to control the fading distance ($5mm$ in our experiments). We finally normalize the blending weights at every vertex $j$ of the mesh: $w_{i,j} = w_{i,j}/\sum_i w_{i,j}$. Let the desired color value at vertex $j$ be $c_j$. The color that is to be transferred to vertex $j$ using the current orientation $i$ is computed as the interpolation between $c_j$ and white color using $w_{i,j}$ (i.e., $w_{i,j}c_j + (1 - w_{i,j})W$, where $W = (255, 255, 255)$).

# 7 Results

In this section, we first describe the experiments we conducted to validate our proposed method (§7.1). We then apply our method to color a variety of 3D surfaces with customized textures, using both the single hydrographic immersion (§7.2) and the multi-immersion methods (§7.3). Our simulation was run on a Intel i5-4430 CPU, and the particle texture tracking was run on a Nvidia



**Figure 11: Repeatability.** *We color a single facial model multiple times, each with the same chessboard pattern and hydrographic printing setup. From left to right, the surfaces are hydrographically printed once, twice and three times, respectively.*

GTX 780 GPU. All the examples take up to 5 minutes to simulate a single immersion.

**Parameters.** There are three physical parameters in the viscous sheet model: namely, density, thickness and Young's modulus. In our examples, density $\rho = 1300 \text{kg/m}^3$ and thickness $h = 0.5\text{mm}$ are provided by the hydrographic kit manufacturer. Young's modulus ($Y = 1.82\text{MPa}$) is measured through a physical experiment: we print a chessboard texture to a sphere, we then run multiple simulations with different Young's modulus values, and choose the value that produces the best prediction. We only conduct this physical experiment once and reuse the parameter values in all our examples.

## 7.1 Validation Experiments

We verify our method by answering three questions: (i) can our motor-controlled hydrographic printing system repeat the same experiment? Once the experiment is repeatable, then we ask (ii) how accurately can our simulation method predict the hydrographic printing results? In addition, we verify (iii) whether the nonlinear viscous sheet model is more predictive than the Stokes flow model.

**Repeatability.** A critical disadvantage of traditional hydrographic printing technique is the lack of repeatability. Thus, we verify that our well-controlled system can repeat the hydrographic printing precisely. To this end, we use a chessboard texture repeatedly transferred to a single object surface. Every time, we print the same chessboard pattern on a PVA film, and conduct the same hydrographic printing using a single object. Since hydrographic printing has been a standard process, we strictly follow the instructions: spray 1ml of activator on an A4-sized film and wait for 20 seconds before dipping. If our system can repeat a hydrographic printing precisely, then the color pattern on the object surface after multiple immersions should remain the same. Figure 11 shows that our system indeed has repeatability. We use three identical surfaces hydrographically printed using the same film once, twice and three times, respectively. As a result, all three surfaces gain the same surface textures. And throughout multiple repeated immersions of a single surface, the chessboard pattern is always mapped to the same locations of that surface, with only its color getting darker because of the ink accumulation.

**Accuracy.** We validate the accuracy of our method using two substrate objects: a square board with a size of 12cm×12cm×1cm (Figure 12 (top)) and a sphere with a radius of 8cm (Figure 12 (bottom)). We further assign small dimples regularly distributed over the surface. Those dimples serve as geometric markers to
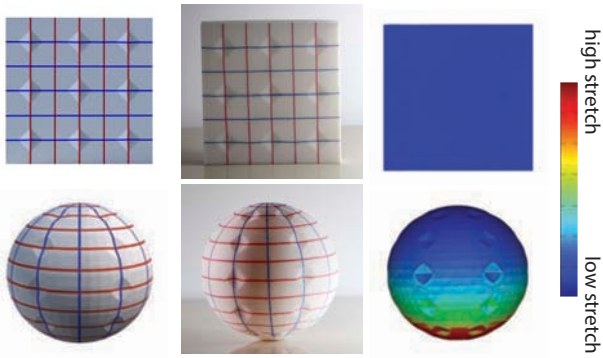
**Figure 12: Accuracy evaluation.** *We evaluate the computational accuracy by aligning cross-line texture with surfaces marked using small dimples:* **(left)** *the desired textures,* **(middle)** *the resulting textures transferred on object surfaces, and* **(right)** *the colormapped film stretches.*

indicate a desired texture registration; they are so shallow that they have negligible influence on the distortion of the film. The textures of the board are grid lines, while the texture of the sphere is longitudes and latitudes (Figure 12). Our goal is to align the texture using our method such that the crosspoints of the lines are at the centers of those dimples. Thereby, we can quantitatively evaluate the deviation of the experiment results from the computational prediction by measuring the distance between the texture crosspoints on the surface and the corresponding center positions of the dimples.

The board is dipped into the water at an angle of 30 degrees. The maximum error is within 1.5mm. The sphere is dipped with its north pole pointing downward. The maximum error on the northern hemisphere is within 2mm. However, near its south pole the error is much larger (about 5mm). This is because after the water surface passes the sphere's equator, the film gets stretched largely, and near the south pole the relative angle between the water surface and the object surface approaches to 180°, leading to an ill-posed boundary condition for our simulation (recall Equation (1), when $\theta \approx 180°$).

**Viscous sheet versus Stokes flow.** We compare the accuracy of the Stokes flow and the viscous sheet model for hydrographic prediction. We use a U-shaped 3D model dipped upside down so that the two "feet" of the model touch the color film first and the horizontal bottom part gets immersed last (Figure 13). We choose this model because the film stretch increases along the two "feet" of the model, and the stretched color pattern is finally transferred to the horizontal middle part of the model, which allows us to easily examine the film distortion. As shown in Figure 13, the nonlinear viscous sheet model, compared with the Stokes flow, is able to predict much larger film distortion.
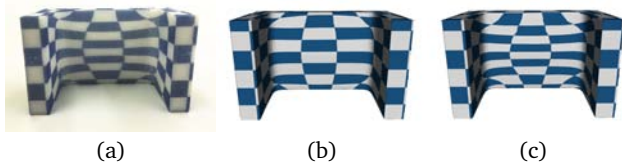


(a)　　　　　(b)　　　　　(c)

**Figure 13: Stokes flow versus viscous sheet.** *We compare both models using a chessboard pattern transferred on a U-shaped 3D model:* **(a)** *the photo of hydrographically printed 3D model,* **(b)** *the predicted color pattern using Stokes flow, and* **(c)** *the predicted color pattern using the nonlinear viscous sheet model.*



**Figure 14: Elephant mug.** *We purchased a porcelain mug* **(left)***, and decorate it with an elephant texture using our method* **(right)***.*



**Figure 15: Globes.** *We make globes by transferring an earth texture to spheres. From left to right, the second and fourth models are the original spheres made by plastic (3D printed) and wood, respectively. The first and third are the ones with hydrographically printed textures on both materials.*

### 7.2 Printing with Single Immersion

For many simple objects, a single hydrographic immersion is able to produce plausible results. Figure 2 illustrates two examples using our computational hydrographic printing method. Compared to the traditional hydrographic printing, our method allows a color texture to align accurately on the object surface.

**Printing on different materials.** The standard hydrographic printing is able to decorate 3D objects made of a wide range of materials, and our method inherits this versatility. In Figure 14, we show a porcelain mug decorated with an elephant texture using our system. Since this mug was purchased, we first acquired its 3D geometry using a 3D scanner; an artist then designed an elephant texture to map on the mug. We provided the acquired 3D model and the color texture into our system to compute a color image on the PVA film, and performed the hydrographic printing.

In addition, we demonstrate the hydrographic printing of an earth texture on both a (3D printed) plastic sphere and a wood sphere (Figure 15). Our method is able to produce nicely looking surface textures. Note that although the color of the substrate object (e.g., the color of wood) does not affect the hydrographic printing process, it does affect the resulting appearance. This issue can be addressed by compensating the substrate color when computing pixel images or by printing a white base color layer before printing the surface texture. We leave these experiments as a future exploration.

### 7.3 Multiple Immersion

Lastly, we decorate complex 3D models using our multi-immersion method (Figure 16). The human head model and the zebra model are both hydrographically printed twice along two

**Figure 16: Multi-immersion printing.** *For such complex models as zebra* **(left)***, cat* **(middle)** *and the head* **(right)***, we perform multi-immersion printing as introduced in §6.*

orientations, while the cat and bunny (in Figure 1) are printed three times. We refer to the supplementary video for the entire hydrographic printing process.

## 8 Discussion and Future Work

We have introduced the first computational hydrographic printing method for physically decorating 3D objects with user-customized surface textures. We developed a computational model to predict the distortion of color films in a hydragraphic printing process. We extend this method to precisely combine multiple printing processes together to generate color textures on complex surfaces. We further build a hydrographic system using off-the-shelf hardware, providing a powerful, practical tool for coloring complex 3D objects with user-specified and registered surface textures. We performed a series of physical experiments to validate our system and demonstrate its printing quality.

Our method has several limitations. First, if an object is highly concave and has significant self-occlusions, it is difficult to cover its entire surface with color films even using multiple immersions (Figure 17). This is a common problem for all hydrographic printing techniques. Second, while our boundary blending scheme in multiple immersions always generates smooth transition between texture regions, it may cause color deviation from the original texture in blending regions. The appearance of color blending is not a simple addition of ink. One unit of ink painted on the surface looks different from half unit of ink painted twice. This is also related to how an inkjet printer maps the color to the amount of ink sprayed on the paper. To achieve more accurate color blending, it is necessary to have a model to measure and calibrate the color blending deviation. Third, the color film distortion during the hydrographic printing also affects the appearance of the color. The more stretched a film becomes, the lighter the color is when transferred to the surface (Figure 17). Ideally, we need to compensate the color changes because of the film stretch by printing more ink to the regions that would be largely stretched. But that would in turn affect the simulation results because effectively the film becomes thicker with more ink printed. One possible future solution is to use an iterative algorithm, with a cost of increased simulation time. Fourth, the film thickness estimation in our nonlinear viscous sheet model is an approximation. It is nontrivial to quantitatively measure the approximation error and its effect on the simulation result. In the future, it is possible to completely avoid this approximation and use a color layer of uniform thickness by using CMYKW-based inkjet printers. Lastly, our current system relies on the user to manually choose the object orientations for dipping. It would be interesting to develop a fully automatic approach that computes the optimal object orientations for hydrographic printing.

**Figure 17: Failure case.** *When we color the bottom surface of a cat bowl, the PVA film has been extremely stretched (or even broken) before adhering to the surface, resulting in deteriorated printing quality:* **(left)** *the desired surface texture and* **(right)** *our hydrographic printing result.*

## Acknowledgements

## References

BÄCHER, M., BICKEL, B., JAMES, D. L., AND PFISTER, H. 2012. Fabricating articulated characters from skinned meshes. *ACM Trans. Graph. 31*, 4, 47.

BARGTEIL, A. W., WOJTAN, C., HODGINS, J. K., AND TURK, G. 2007. A finite element method for animating large viscoplastic flow. *ACM Trans. Graph. 26*, 3 (July).

BATTY, C., URIBE, A., AUDOLY, B., AND GRINSPUN, E. 2012. Discrete viscous sheets. *ACM Trans. Graph. 31*, 4 (July).

BESL, P. J., AND MCKAY, N. D. 1992. Method for registration of 3-d shapes. In *Robotics-DL tentative*, International Society for Optics and Photonics, 586–606.

BICKEL, B., BÄCHER, M., OTADUY, M. A., MATUSIK, W., PFISTER, H., AND GROSS, M. 2009. Capture and modeling of non-linear heterogeneous soft tissue. *ACM Trans. Graph. 28*, 3 (July).

BICKEL, B., BÄCHER, M., OTADUY, M. A., LEE, H. R., PFISTER, H., GROSS, M., AND MATUSIK, W. 2010. Design and fabrication

of materials with desired deformation behavior. *ACM Trans. Graph. 29*, 4, 63.

CALÌ, J., CALIAN, D. A., AMATI, C., KLEINBERGER, R., STEED, A., KAUTZ, J., AND WEYRICH, T. 2012. 3d-printing of non-assembly, articulated models. *ACM Trans. Graph. 31*, 6, 130.

CARLSON, M., MUCHA, P. J., VAN HORN, III, R. B., AND TURK, G. 2002. Melting and flowing. In *Proceedings of SCA02*, 167–174.

CEYLAN, D., LI, W., MITRA, N. J., AGRAWALA, M., AND PAULY, M. 2013. Designing and fabricating mechanical automata from mocap sequences. *ACM Trans. Graph. 32*, 6, 186.

CHEN, X., ZHENG, C., XU, W., AND ZHOU, K. 2014. An asymptotic numerical method for inverse elastic shape design. *ACM Trans. Graph. 33*, 4, 95.

COROS, S., THOMASZEWSKI, B., NORIS, G., SUEDA, S., FORBERG, M., SUMNER, R. W., MATUSIK, W., AND BICKEL, B. 2013. Computational design of mechanical characters. *ACM Trans. Graph. 32*.

DARTY, L. 2004. *The art of enameling: techniques, projects, inspiration*. Lark Books.

DESBRUN, M., MEYER, M., AND ALLIEZ, P. 2002. Intrinsic parameterizations of surface meshes. *Comput. Graph. Forum 21*.

DONG, Y., WANG, J., PELLACINI, F., TONG, X., AND GUO, B. 2010. Fabricating spatially-varying subsurface scattering. *ACM Trans. Graph. 29*, 3, 62.

GINGOLD, Y., SECORD, A., HAN, J. Y., GRINSPUN, E., AND ZORIN, D. 2004. A discrete model for inelastic deformation of thin shells. In *Proceedings of SCA04*.

HAŠAN, M., FUCHS, M., MATUSIK, W., PFISTER, H., AND RUSINKIEWICZ, S. 2010. Physical reproduction of materials with specified subsurface scattering. *ACM Trans. Graph. 29*, 3.

HOPPER, R. 2004. *Making Marks: Discovering the Ceramic Surface*. Krause Publications Craft.

IZADI, S., KIM, D., HILLIGES, O., MOLYNEAUX, D., NEWCOMBE, R., KOHLI, P., SHOTTON, J., HODGES, S., FREEMAN, D., DAVISON, A., ET AL. 2011. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of UIST 11*, ACM, 559–568.

KATZ, J., AND PLOTKIN, A. 2001. *Low-speed aerodynamics*, vol. 13. Cambridge University Press.

KRAEVOY, V., SHEFFER, A., AND GOTSMAN, C. 2003. Matchmaker: constructing constrained texture maps. *ACM Trans. Graph. 22*, 3, 326–3338.

LAN, Y., DONG, Y., PELLACINI, F., AND TONG, X. 2013. Bi-scale appearance fabrication. *ACM Trans. Graph. 32*, 4, 145.

LÉVY, B., PETITJEAN, S., RAY, N., AND MAILLOT, J. 2002. Least squares conformal maps for automatic texture atlas generation. *ACM Trans. Graph. 21*, 3, 362–371.

LUKARSKI, D. 2013. Paralution - library for iterative sparse methods on multi-core cpu and gpu devices. In *NVIDIA GPU Technology Theater*, SC 13.

MALZBENDER, T., SAMADANI, R., SCHER, S., CRUME, A., DUNN, D., AND DAVIS, J. 2012. Printing reflectance functions. *ACM Trans. Graph. 31*, 3, 20.

MATUSIK, W., AJDIN, B., GU, J., LAWRENCE, J., LENSCH, H. P. A., PELLACINI, F., AND RUSINKIEWICZ, S. 2009. Printing spatially-varying reflectance. *ACM Trans. Graph. 28*, 5, 128.

PRÉVOST, R., WHITING, E., LEFEBVRE, S., AND SORKINE-HORNUNG, O. 2013. Make it stand: balancing shapes for 3d fabrication. *ACM Trans. Graph. 32*, 4, 81.

RASMUSSEN, N., ENRIGHT, D., NGUYEN, D., MARINO, S., SUMNER, N., GEIGER, W., HOON, S., AND FEDKIW, R. 2004. Directable photorealistic liquids. In *Proceedings of SCA04*, 193–202.

RAYLEIGH, J. W. S. B. 1896. *The theory of sound*, vol. 2. Macmillan.

RIBE, N. 2002. A general theory for the dynamics of thin viscous sheets. *Journal of Fluid Mechanics 457*, 255–283.

SANDER, P. V., SNYDER, J., GORTLER, S. J., AND HOPPE, H. 2001. Texture mapping progressive meshes. In *Proceedings of SIGGRAPH 01*, ACM, New York, NY, USA, 409–416.

SCHLESINGER, M., AND PAUNOVIC, M. 2011. *Modern electroplating*, vol. 55. John Wiley & Sons.

SHEFFER, A., PRAUN, E., AND ROSE, K. 2006. Mesh parameterization methods and their applications. *Foundations and Trends in Computer Graphics and Vision 2*, 2.

SHEWCHUK, J. R. 1996. Triangle: Engineering a 2d quality mesh generator and delaunay triangulator. In *Applied computational geometry towards geometric engineering*. Springer, 203–222.

SKOURAS, M., THOMASZEWSKI, B., COROS, S., BICKEL, B., AND GROSS, M. 2013. Computational design of actuated deformable characters. *ACM Trans. Graph. 32*, 4, 82.

STAVA, O., VANEK, J., BENES, B., CARR, N., AND MECH, R. 2012. Stress relief: Improving structural strength of 3d printable objects. *ACM Trans. Graph. 31*, 4, 48.

THOMASZEWSKI, B., COROS, S., GAUGE, D., MEGARO, V., GRINSPUN, E., AND GROSS, M. 2014. Computational design of linkage-based characters. *ACM Trans. Graph. 33*, 4 (July), 64:1–64:9.

WEYRICH, T., PEERS, P., MATUSIK, W., AND RUSINKIEWICZ, S. 2009. Fabricating microgeometry for custom surface reflectance. *ACM Trans. Graph. 28*, 3, 32.

WHITE, F. M., AND CORFIELD, I. 1991. *Viscous fluid flow*, vol. 3. McGraw-Hill New York.

WICKE, M., RITCHIE, D., KLINGNER, B. M., BURKE, S., SHEWCHUK, J. R., AND O'BRIEN, J. F. 2010. Dynamic local remeshing for elastoplastic simulation. *ACM Trans. Graph. 29*, 4 (July), 49:1–49:11.

WIKIPEDIA, 2014. Hydrographics (printing). http://en.wikipedia.org/wiki/Hydrographics_(printing).

WOJTAN, C., AND TURK, G. 2008. Fast viscoelastic behavior with thin features. *ACM Trans. Graph. 27*, 3 (Aug.), 47:1–47:8.

WRIGHT, E. J., ANDREWS, G. P., MCCOY, C. P., AND JONES, D. S. 2013. The effect of dilute solution properties on poly (vinyl alcohol) films. *Journal of the mechanical behavior of biomedical materials 28*, 222–231.

ZHOU, K., WANG, X., TONG, Y., DESBRUN, M., GUO, B., AND SHUM, H.-Y. 2005. Texturemontage: Seamless texturing of arbitrary surfaces from multiple images. *ACM Trans. Graph. 24*, 3.