EUROGRAPHICS 2026 / B. Masia and J. Thies
(Guest Editors)

COMPUTER GRAPHICS *forum*
Volume 45 (2026), Number 2

Dripping Thin Films for Real-time Digital Painting

Zoé Herson [1] Axel Paris [1] Élie Michel [1]

[1] Adobe

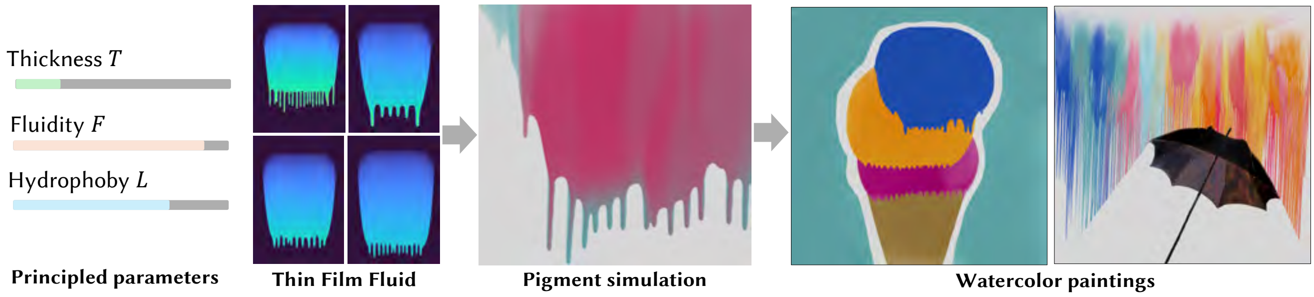


Figure 1: *Building upon prior work in physical modeling and numerical methods, we propose a formulation of a Thin Film model that provides control over the shape of dripping fluid fingers. We analyze its parameter-space and extract principled parameters to control the dripping features. Combined with a pigment simulation (diffusion, advection, and mixing), this expands the range of effects that can be artistically explored in a real-time painting simulation.*

Abstract

We present a real-time method to capture and simulate the dynamic behavior of watercolor painting. We develop a physically accurate, grid-based, real-time fluid simulation based on a reparameterized Thin Film model. The equations are rewritten so as to create principled parameters that finely control the length, thickness, and frequency of dripping. Our close connection with physics allows both theoretical and experimental validation of our method. The resulting system can reproduce dripping, fluid-air interface, and pigment advection and diffusion, all controllable by the user in real-time. Our experiments show that artists can use our system to create interesting and varied digital paintings.

CCS Concepts

• ***Computing methodologies*** → ***Model verification and validation; Real-time simulation;***

1. Introduction

Fluid simulation has a crucial role to play in the experience of digital painting, as it acknowledges the paint for what it is: a liquid. Not only do computational physics simulations successfully reproduce the complex dynamic behavior of watercolor flows, but they may also be seen as an essential part of the creative process itself, as the emergence of realistic effects (including dripping, wet painting techniques, and more) gives more expressivity to the artist when they run in real-time during authoring.

The behavior of fluids has been studied either through Navier-Stokes differential equations [Sto45] or Lattice Boltzmann models [KKK*16]. When adding extra hypotheses about flow and fluid properties, these can be simplified into domain-specific expressions that help their study and leverage numerical models. For painting, a thin film of fluid is applied to a relatively smooth planar canvas. This configuration suggests a preferred class of fluid models called *Thin Film fluid*.

Thin films have been of interest to physicists and industrial engineers for a long time. A noteworthy application close to painting is industrial coating [WR04] where a thin film is applied to either protect or color a surface. In this case, the Thin Film model is used to prevent the emergence of *fingering*, a flow instability driven by gravity, more commonly called dripping [Kon03]. Conversely, it is the modeling of this very effect that makes Thin Film simulation attractive for watercolor painting simulation.

Existing methods focus on rigorous solutions to the general equation using energy-preserving schemes [VRBC18] but this approach introduces discretization error that biases the fluid behavior and thus hinders the emergence of dripping (Sec. 5.2). Dripping only occurs under certain parameter settings, and we believe that understanding such parameter space is an important research topic, akin to other works on Reaction-Diffusion [CLH16, SKJY06] or physically-based rendering [BBP15, DJ18].

Our derivation is anchored in the physical setting of dripping watercolor paint, which in turn allows deriving a set of principled parameters over the dripping effects. More specifically, our contributions are:

- A Thin Film model enhanced with a new term to model the hydrophobic aspect of the medium, leading to a larger diversity of dripping features.
- A new parameterization of the Thin Film equations, coupled with the physics formulation, with principled parameters.
- We show that existing solvers based on energy regularization exhibit too much discretization error in regimes where some dripping behavior can be observed [VRBC18]. We introduce an alternative formulation that mitigates this bias in dripping regions of the parameter space.

We validate our method against experimental results from physics. Targeting watercolor painting applications, we design controls that have a physical interpretation, and show that various effects emerge from the simulation (see Fig. 1 and accompanying video).

2. Previous work

Our method relates to fluid simulation and Thin Film models in Physics and Computer Graphics, as well as digital painting. While some recent work has been addressing 3D fluid simulation with realistic and controllable dripping effects [SMB19, SLW*23, PT24], here we focus on techniques restricted to a planar setting compatible with a real time painting context.

Thin Film in Physics. In the past decades, extensive research has been conducted on Thin Film in Physics. Perhaps surprisingly, Thin Film equations successfully model fluids in many different settings, for instance in complex coating flows [WR04], modeling granular and debris flows in geophysics [Anc07], or even ice sheet models [BHG01]. We focus on works that study dripping liquid and fingering in theoretical and experimental contexts, and refer the reader to reviews for an in-depth analysis [ODB97, CM09]. Kalliadasis et al. [KRQSV11] study the general problem of a thin layer on an inclined plane in different settings. They rely on equations that couple the height of the fluid and its flow rate, which allows to simulate flows at moderate Reynolds numbers. In a painting context with a low turbulence fluid, a simpler height-based formulation is sufficient. Closer to our method is a body of work that studies the *instabilities* in Thin Film models, responsible for the emergence of dripping fingers under the influence of gravity. This phenomenon has been studied either numerically [Kon03, DK01, LJK14], or experimentally [VIC98, JSMB99]. These are not directly applicable to our context as they work in idealized settings under a fixed gravity force. Still, we validate our work by comparing against these experimental results.

Thin Film in Computer Graphics. Carlson et al. [CMVHT02] adapted the Marker-and-cell method [HW65] derived from Navier-Stokes to handle fluids with a high viscosity. By treating solid objects as highly viscous fluids, this provides a framework for physics-based animations. Several studies focused on simulating such viscous liquids in an accurate manner [LBB17], although this leads to very compute-intensive solvers. Building upon the gradient flow model [RV13], Vantzos et al. [VAW*17] first introduced the Thin Film equation in Computer Graphics. They successfully reproduce effects such as droplets, evaporation, and viscous fingering over curved surfaces. They solve a sparse linear system at each step, which in this case cannot be done interactively. Thin Film models may also be applied to the modeling of soap film flows. Combined with a soap concentration field that evolves, Huang et al. [HIK*20] simulate and render soap film on spherical bubbles. Ishida et al. [ISN*20] later extended the model to handle varying surface thickness, allowing for more interesting interference patterns during rendering. Closer to our work, the method described in Vantzos et al. [VRBC18] simplified the original method for a 2D context, allowing for real-time simulation. Their formulation improves stability and ensures mass conservation as well as non-negativity of the fluid height. However, their regularization energy leads to bias that often hinders the creation of dripping.

Digital Painting. Reproducing real paint behavior brings lots of challenges and various research topics. Several papers investigated brush modeling, which may be done using polygonal splats [DKMI13], subdivision surfaces [BWL04], deformable models [CT02], or particles distributed along individual bristles [CKIW15, FRLH23]. After deposition, fluid and pigments may be simulated over the canvas using grid-based [CT05], particle-based [YJC*13], or hybrid approaches [CKIW15, WLT18]. A number of methods simulate the motion of paint around the brush, which is a fine approximation for viscous paint (oil, acrylic) for which paint motion is entirely driven by the brush-canvas interaction. However, this is not enough for watercolor painting where pigments move and mix without restraint inside the fluid. In this spirit, the method by Van Laerhoven et al. [VLVR05] focuses on watery paint, and successfully captures watercolor effects such as capillary absorption and pigment diffusion. Recently, dendritic painting techniques using Reaction-Diffusion equations have been successfully simulated on planar surfaces [COKE20]. Compared to these, we allow additional effects to emerge through the underlying physics of the simulation, for instance by tilting the canvas and letting the paint drip nicely (see accompanying video).

3. Overview

Background in Fluid simulation. The macroscopic dynamics of a fluid are driven by the combination of **(i)** its intrinsic properties (e.g., viscosity, surface tension), which stem from the microscopic molecular chemical interactions, and **(ii)** the flow condition, namely the spatial dimensions and range of velocities over which the fluid evolves. One often uses the Reynolds number Re to characterize different flows, defined as the ratio of inertia forces versus viscous forces:

$$Re = \frac{\text{Inertia forces}}{\text{Viscous forces}} = \frac{\rho\, U\, L}{\mu} \qquad (1)$$

where U is a characteristic speed of the flow, L is a characteristic length, and μ is the dynamic viscosity of the fluid. A flow with a high Reynolds number is considered *turbulent*, whereas a flow with a small number is called *laminar*. We restrict our study to thin layers of slow laminar flows exhibiting complex surface tension behavior at a small scale (painting on a canvas). These problems are usually modeled using Thin Film equations derived from Navier-Stokes. They assume the following hypotheses called *lubrication approximation*: **(1)** the height h of the fluid must be small

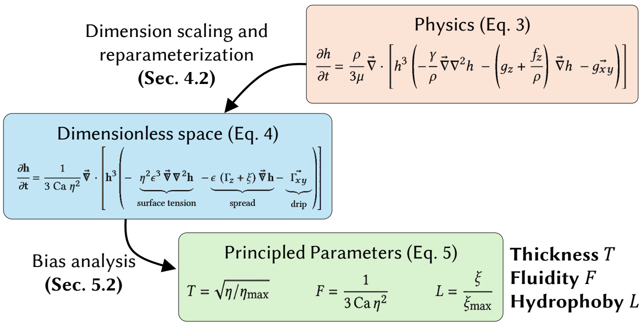

Dimension scaling and reparameterization
(Sec. 4.2)
Physics (Eq. 3)
$\frac{\partial h}{\partial t} = \frac{\rho}{3\mu}\vec{\nabla}\cdot\left[h^3\left(-\frac{\gamma}{\rho}\vec{\nabla}\nabla^2 h - \left(g_z + \frac{f_z}{\rho}\right)\vec{\nabla}h - \vec{g_{xy}}\right)\right]$
Dimensionless space (Eq. 4)
$\frac{\partial \mathbf{h}}{\partial \mathbf{t}} = \frac{1}{3\,\mathrm{Ca}\,\eta^2}\vec{\nabla}\cdot\left[\mathbf{h}^3\left(-\underbrace{\eta^2\epsilon^3\vec{\nabla}\nabla^2\mathbf{h}}_{\text{surface tension}} - \underbrace{\epsilon\,(\Gamma_z+\xi)\vec{\nabla}\mathbf{h}}_{\text{spread}} - \underbrace{\vec{\Gamma_{xy}}}_{\text{drip}}\right)\right]$
Bias analysis
(Sec. 5.2)
Principled Parameters (Eq. 5)
$T = \sqrt{\eta/\eta_{\max}}$
$F = \frac{1}{3\,\mathrm{Ca}\,\eta^2}$
$L = \frac{\xi}{\xi_{\max}}$
Thickness T
Fluidity F
Hydrophoby L

Figure 2: *Starting from the physics formulation of the thin film equation (orange), we introduce a new dimension scaling (blue) based on experimental work, and derive 3 principled parameters (T,F,L) that are more controllable by users (green). In practice, the dimensionless space is the one we implemented, and its direct link to physics is what allows us to validate our method.*

compared to the characteristic dimension of the surface onto which it flows, **(2)** the normal component of the velocity must be negligible compared to the in-plane ones, and **(3)** the fluid must have a low Reynolds number. These hypotheses enable us to alleviate the simulation's compute needs so that it runs in real time, while ensuring that complex behavior can still emerge.

Problem setting. We aim at incorporating realistic fluid behavior in digital painting applications. We adopt a Thin Film model as painting complies well with the lubrication approximation. One interesting phenomenon is the dripping down of a fluid along the canvas (fingering instability). This is caused by an accumulation of matter at the borders. When this accumulation overcomes the cohesive forces of the liquid, elongated features (*fingers*) emerge and keep stretching in the gravity direction. These fingers naturally emerge from real liquids on inclined surfaces with varying patterns that we reproduce.

Outline. We first explain our contributions to the Thin Film model (Sec. 4), in particular how we derive a new set of *principled parameters* starting from the physics formulation of the thin fluid film model (Fig. 2). One crucial aspect of our derivation is that we retain an explicit connection to physics notations. This allows validating against theoretical results as well as analyzing the parameter-space to determine regions where dripping occurs (Sec. 5). Finally, we explain our painting system along with extensions for modeling the medium surface and pigment mixing (Sec. 6). Notations used in the paper are reported in Table 1.

4. Our Thin Film model

4.1. Physical modeling

We derive a formulation of the Thin Film equation adapted to the needs of a painting context, which exposes multiple degrees of freedom to control the dripping behavior. As we remain connected to the original equations from physics, parameters can be set so as to reproduce real experiments which we later use to validate our model and explore the parameter space.

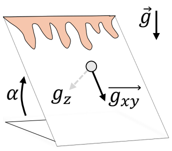
Table 1: *Notations (following Kondic et al. [Kon03]) in the paper with units.*

Quantity	Symbol	Units
Fluid height	h	m
Fluid velocity	$\vec{u} = (\vec{v}, w)$	$m.s^{-1}$
Dynamic viscosity	μ	$N.s.m^{-2}$
Surface tension	γ	$N.m^{-1}$
Mass density	ρ	$kg.m^{-3}$
Capillary number	Ca	-
Capillary length	a	m
Capillary length (dimensionless)	η	-
Gravity vector	$\vec{g} = g\vec{\Gamma}$	$m.s^{-2}$
Custom force	$\vec{f} = \xi\rho g\vec{u_z}$	$kg.m^{-2}.s^{-2}$
Characteristic time	t_c	s
Characteristic length	x_c	m
Characteristic height	h_c	m
Fluid thinness ratio	$\epsilon = h_c/x_c$	-
Time step	Δt	s
Space increment	Δx	m
Pigment quantity	c	-
Diffusion coefficient	D	$m^2.s^{-1}$

Our Thin Film equation (TF) derives from Navier-Stokes (NS) by injecting extra hypotheses from the *lubrication approximation* [Kon03]. Recall the NS equations [Sto45] for incompressible fluids:

$$\rho\left(\frac{\partial\vec{u}}{\partial t} + (\vec{u}\cdot\vec{\nabla})\vec{u}\right) = -\vec{\nabla}p + \mu\nabla^2\vec{u} + \rho\vec{g} + \vec{f} \tag{2}$$

where ρ is the fluid's mass density, $\vec{u}$ the velocity, p the pressure field, μ the dynamic viscosity, and $\vec{g}$ the gravity vector. We introduce a new term with respect to Kondic et al. [Kon03] defined as $\vec{f} = \xi\rho g\vec{u_z}$, that is a custom force along the normal direction of the plane. This acts as a cohesive force on the fluid (the opposite of a spread) which in practice tends to create longer dripping patterns. To make better use of the Thin Film hypotheses, it is common practice to express this NS equation in a frame aligned with the flow plane. In particular, we write the velocity as $\vec{u} = (\vec{v}, w)$, with $\vec{v}$ the in-plane velocity and w the normal component. We also write $\vec{g} = (g_x, g_y, g_z)$, where $\vec{g_{xy}} = (g_x, g_y)$ is the gravity projected onto the flow plane, which is tilted by an angle α (see inset). Following these notations, the lubrication approximation is expressed as:



1. **Aspect ratio**: the height h of the fluid is small compared to the in-plane length-scale W, so gradients of any physical quantities are overall dominated by their normal z component, e.g. $\frac{\partial\vec{u}}{\partial x} \ll \frac{\partial\vec{u}}{\partial z}$.
2. **Planar velocity**: the normal component w of the velocity is negligible, and the in-plane velocity $\vec{v}$ can be approximated by its mean value along the normal axis (z axis).
3. **Reynolds number**: It is assumed that Re is small enough to neglect the inertia terms, i.e., not only are the left and right-

hand terms of Eq. 2 equal, but they are also null. It results that the pressure field p can be derived from the velocity $\vec{u}$.

The average in-plane velocity $\langle\vec{v}\rangle$ is obtained by integrating Eq. 2 along the normal axis z, for which we employ the boundary conditions suggested by Kondic et al. [Kon03], namely **(i)** *no-slip* at the fluid-canvas interface, i.e., $\vec{v}(z=0) = 0$, and **(ii)** *Laplace-Young* boundary condition at the air-fluid interface. The latter states that the surface pressure is $p(z=h) = p_0 - \gamma\kappa$, where p_0 is the air pressure, γ is the surface tension parameter and κ is the curvature of the interface – commonly approximated as $\nabla^2 h$. This leads to the average fluid velocity (see Supplemental Material for the derivation):

$$\langle\vec{v}\rangle = \frac{h^2}{3\mu}\left((\rho g_z + f_z)\vec{\nabla}h + \gamma\vec{\nabla}\nabla^2 h + \rho\vec{g_{xy}}\right) \tag{3}$$

We then inject this mean velocity in the equation of mass conservation $\frac{\partial h}{\partial t} = -\vec{\nabla}\cdot(h\vec{v})$ and finally obtain the following 2D Thin Film equation [ODB97]:

$$\frac{\partial h}{\partial t} = \frac{\rho}{3\mu}\vec{\nabla}\cdot\left[h^3\left(-\frac{\gamma}{\rho}\vec{\nabla}\nabla^2 h - \left(g_z + \frac{f_z}{\rho}\right)\vec{\nabla}h - \vec{g_{xy}}\right)\right] \tag{4}$$

4.2. Dimension scalings and reparameterization

Eq. 4 relates to *physical* quantities, which are important to compare with experimental results and to assign a value to quantities that we can measure. We re-parameterize these quantities for two reasons: **(i)** stability of numerical simulation requires *dimensionless* values that remain within a couple of orders of magnitude around 1, and **(ii)** user control is better when exposed parameters are *principled*, i.e., linear and orthogonal enough with respect to emerging phenomena that users are interested in. While typical scaling such as Nusselt or Shkadov are frequent in the literature [KRQSV11], they were designed for specific theoretical experiments (thermal diffusivity, boundary layer approximations) rather than direct manipulation from users. We thus introduce a new dimension scaling based on the work of Kondic et al. [Kon03]. Their approach aims at studying the instability theoretically, with each dimension (horizontal, vertical, temporal) dependent on each other so that when one changes, everything is rescaled. This standardizes the appearance of the flow, but prevents any visual diversity from emerging. In contrast, our formulation exposes more degrees of freedom (each dimension remains independent) and does not degenerate at null tilt angle ($\alpha = 0$), as these are needed in the context of digital painting. We nonetheless keep an explicit relation to the physical quantities for validation (Sec. 5), something that is difficult to do with existing formulations [VRBC18].

We note in general $\mathbf{q}$ the dimensionless counterpart of a quantity q. We first scale the height of the fluid h by a characteristic height h_c that denotes the order of magnitude of the fluid height in the bulk: $\mathbf{h} = h/h_c$. Similarly, we scale the horizontal dimensions (x,y,z) by a characteristic length x_c that corresponds to the physical size of a few grid cells: $(\mathbf{x},\mathbf{y},\mathbf{z}) = (x/x_c, y/x_c, z/x_c)$. Finally, we write $\mathbf{t} = t/t_c$, where the characteristic time t_c is the frame duration. To remain consistent with the initial hypothesis, we need the ratio $\epsilon = h_c/x_c \ll 1$. These three characteristic values (h_c, x_c, t_c) are then used to scale other physical quantities of the equation. We introduce the dimensionless parameter $\eta = a/h_c$, where $a = \sqrt{\gamma/\rho g}$ is commonly referred to as the *capillary length* and characterizes the balance between capillary forces and gravity (higher η means stronger surface tension). We also use the dimensionless *capillary number* $\mathrm{Ca} = \mu u_c/\gamma$, where $u_c = x_c/t_c$ is the characteristic velocity of the fluid. The capillary number is used in the Physics literature to characterize the balance between viscosity and capillary forces (higher Ca means stronger viscosity). By replacing h, (x,y,z), and t by their dimensionless counterparts and writing $\Gamma = \vec{g}/g$ the unit direction of the gravity with respect to the canvas frame, we derive the scaled equation:

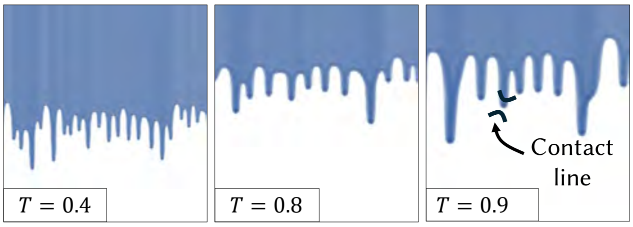


Figure 3: *Our simulation can reproduce a variety of dripping patterns of different frequencies controlled by T. Parameters: $Ca = 0.0005$, $\epsilon = 0.1$, varying number of iterations to reproduce patterns of approximately similar sizes.*

$$\frac{\partial \mathbf{h}}{\partial \mathbf{t}} = \frac{1}{3\,\mathrm{Ca}\,\eta^2}\vec{\nabla}\cdot\left[\mathbf{h}^3\left(-\underbrace{\eta^2\epsilon^3\,\vec{\nabla}\nabla^2\mathbf{h}}_{\text{surface tension}} - \underbrace{\epsilon\,(\Gamma_z + \xi)\,\vec{\nabla}\mathbf{h}}_{\text{spread}} - \underbrace{\vec{\Gamma_{xy}}}_{\text{drip}}\right)\right] \tag{5}$$

where the change of variable introduces the dimensionless spatial derivative $\vec{\nabla} = x_c\vec{\nabla}$. The ratio ϵ is derived from characteristic dimensions x_c and h_c and the gravity direction Γ is exposed as a tilt angle $\alpha \in (0, \frac{\pi}{2})$ and a gravity direction $\beta \in (-\pi, \pi)$. We denote $\xi = f_z/\rho g$ the scaled custom force. Finally, we derive three principled parameters, the *fluidity* F, *drip thickness* T, and *medium hydrophoby* L, from the dimensionless degrees of freedom Ca, η, and ξ:

$$T = \sqrt{\eta/\eta_{\max}} \qquad F = \frac{1}{3\,\mathrm{Ca}\,\eta^2} \qquad L = \frac{\xi}{\xi_{\max}} \tag{6}$$

We experimentally measured $\eta_{\max} = 150$ and $\xi_{\max} = 10$ in the case of $\epsilon = 0.1$. Following Kondic et al. [Kon03], we know that η varies with $\epsilon^{-3/2}$, thus we deduce the general definition $\eta_{max} = \left(\frac{\epsilon}{0.1}\right)^{-3/2} \times 150$. Parameters T, F, and L are the ones exposed to the user and map respectively the apparent thickness of the fingers, the flow speed (a dimensionless inverse viscosity), and a control over the vertical length of the fingers (see Fig. 15 for a visual explanation of each parameter). As both T and F depend on η, in practice we use the following workflow (1) we first pick a value for T and (2) set the fluidity to the maximum value without bias $F_{\max} = \frac{1}{3\,\mathrm{Ca}_{\min}\eta^2}$. We determine experimentally from Fig. 8 that $\mathrm{Ca}_{\min} = 10^{-4}$ for $\epsilon = 0.1$, and generalize to $\mathrm{Ca}_{\min} = \left(\frac{\epsilon}{0.1}\right)^3 10^{-4}$. These correspond to the values of unbiased simulations described in Sec. 5.2. To summarize, our formulation differs from the one of Kondic et al. in two ways: we inject an additional force in the model that acts on the fingers' length (controlled by L), and we release the constraints on the three characteristic values (h_c, x_c, t_c), which increases the diversity of shape or scale that can emerge.

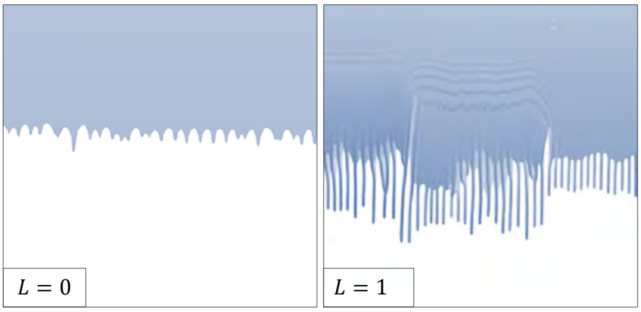


Figure 4: *Influence of the principled hydrophoby parameter L. A value of $L = 1$ accelerates the dripping and creates thin elongated fingers.*

4.3. Numerical scheme and Implementation

Our Thin Film model (Eq. 5) is a fourth order differential equation, which requires some care to solve numerically. We base our solver on the state-of-the-art method from Vantzos et al. [VRBC18], which runs in real-time while preserving mass and ensuring stability. However, we found that their regularization approach introduces bias that hinders the emergence of dripping fingers, so we propose an alternative solving scheme that mitigates the instabilities by clamping the flux rather than minimizing an energy. Their formulation writes:

$$\frac{\partial \underline{u}}{\partial t} + \mathrm{div}\underline{\vec{f}} = 0, \qquad \underline{\vec{f}} = -M(\underline{u})\vec{\nabla}(\underline{W} - \underline{\epsilon}\Delta\underline{u} + \underline{\eta}\underline{u})$$

where underlined symbols correspond to notations from Vantzos et al. [VRBC18]. We can map these notations to our physically grounded parameterization (the mobility function M is detailed below):

$$\underline{u} \leftrightarrow \mathbf{h}, \qquad \underline{\epsilon} \leftrightarrow \frac{\epsilon^3}{\mathrm{Ca}}, \qquad \underline{W} \leftrightarrow -g_{xy} \cdot \begin{pmatrix} x \\ y \end{pmatrix},$$
$$\underline{\vec{f}} \leftrightarrow -\frac{1}{3\mathrm{Ca}\,\eta^2}\vec{\Phi}, \qquad \underline{\eta} \leftrightarrow -\frac{\epsilon(\xi+\Gamma_z)}{\mathrm{Ca}\,\eta^2}$$

Solving scheme. Like Vantzos et al., we use a solving scheme based on staggered grids, i.e., $\frac{\partial f}{\partial x}[x,y]$ is evaluated as $\frac{1}{\Delta x}\Big(f[x+\frac{1}{2},y] - f[x-\frac{1}{2},y]\Big)$, where Δx denotes the spatial discretization step. The discrete update $\Delta\mathbf{h}$ at a pixel x, y is expressed as a sum of four flow terms Φ that correspond to discretizing the top level divergence of Eq. 5 (see Fig. 5):

$$\Delta\mathbf{h}[x,y] = \frac{\partial \mathbf{h}}{\partial t}[x,y]\mathbf{\Delta t} = \frac{\mathbf{\Delta t}}{3\;\mathrm{Ca}\;\eta^2}\vec{\nabla}\cdot\vec{\Phi}[x,y]$$

where

$$\vec{\nabla}\cdot\vec{\Phi}[x,y] = \frac{1}{\Delta x}\Bigg(\Phi_x[x+\frac{1}{2},y] - \Phi_x[x-\frac{1}{2},y] + \Phi_y[x,y+\frac{1}{2}] - \Phi_y[x,y-\frac{1}{2}]\Bigg) \tag{7}$$

Applying the same method recursively leads to the following expression for the first flux (see additional material for other terms):

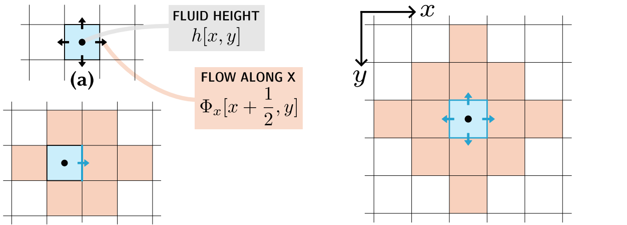
$$\Phi_x[x+\frac{1}{2},y] = h^3[x+\frac{1}{2},y]\Bigg(\frac{\eta^2\epsilon^3}{\Delta x^3}\Big(\nabla^2\mathbf{h}[x+1,y] - \nabla^2\mathbf{h}[x,y]\Big) + \frac{\epsilon(\Gamma_z+\xi)}{\Delta x}(\mathbf{h}[x+1,y] - \mathbf{h}[x,y]) + \Gamma_x\Bigg) \tag{8}$$

where the dimensionless Laplacian $\nabla^2\mathbf{h}[x,y]$ is evaluated using the five-point stencil. The only quantity evaluated at a non-integer index is $h^3[x+\frac{1}{2},y]$. We evaluate it as a function of $M(\mathbf{h}[x,y],\mathbf{h}[x+1,y])$ of its integer neighbors, with M corresponding to the mobility function as introduced by Vantzos, and detailed below.

Mass preservation. The local nature of Eq. 8 makes it a good fit for massively parallel contexts. However, a naive implementation easily leads to numeric instabilities and negative height $\mathbf{h}$. Vantzos et al. [VRBC18] ensure non-negativity by clamping each flow $\vec{\Phi}$ so that $\Delta\mathbf{h} > -\mathbf{h}$. As opposed to clamping $\mathbf{h}+\Delta\mathbf{h}$ to 0 for each pixel individually, clamping fluxes between cells preserves the total mass of fluid as long as two neighbor pixels evaluate the same flow term Φ across the edge that separates them, clamping included. For the latter point, Vantzos et al. suggest the use of a *domino relaxation* pattern, where each simulation step is split into multiple passes that apply the flow across the central edge of non-overlapping pairs of pixels. In practice, we use a random permutation of 4 domino passes rather than the 8 passes they described, as this reduces the computational cost while not introducing significant bias.

Regularization factor and bias. Vantzos et al. dampen the expression of the flow by dividing it by a factor θ defined as (with our notations) $\theta = 1 + 2\Delta t\,\mathbf{h}^3\big(\frac{5\epsilon^3}{\mathrm{Ca}\,\Delta x^4} - \frac{\epsilon(\Gamma_z+\xi)}{\mathrm{Ca}\,\eta^2\Delta x^2}\big)$. This prevents the instabilities inherently introduced by the discretization but trades them for bias. By bias, we mean the numerical difference between a coarse simulation and a finer one with a smaller time step. When bias is high, the two simulations lead to completely different outcomes. Instead of the factor θ, we propose to mitigate the instabilities by clamping the flux: similarly to how we ensure non-negativity of $\mathbf{h}$, we enforce that $\Delta\mathbf{h} < h_{\max} - \mathbf{h}$. This also introduces bias, but we show in Sec. 5.2 that it increases the range of parameters for which dripping fingers appear, something essential for a painting.

Mobility M and boundary conditions. The solving scheme requires the mobility function $M(h_1,h_2)$ to be symmetrical, positive for any $h_1 \neq h_2$, and such that $M(h,h)$ equals $\frac{h^3}{3}$. Vantzos et al. propose two different formulations for this mobility function, both functions equal zero if one of the two cells is dry ($h_1 = 0$ or $h_2 = 0$), which means that a dry cells block acts as a wall that fully blocks the flow. This is an issue in our painting context, as we need a distinction between boundaries that block the flow of paint and dry pixels that the paint may drip on. We experimented with $M(h_1,h_2) = \big(\frac{h_1+h_2}{2}\big)^3$ and $M'(h_1,h_2) = \frac{1}{2}(h_1^3+h_2^3)$, and found the former to generally produce longer drips. In both cases, we have $M(h_1,h_2) \neq 0$ when at least one of h_1 and h_2 is not null, so that overflowing dry cells is possible. To model blocking boundaries, we cancel flow terms that correspond to wall edges.



(b) **(c)** Samples needed to evaluate Δh

Figure 5: *The variation* Δh *is expressed in Eq. 5 as the divergence of a flow, denoted as* $\vec{\Phi} = (\Phi_x, \Phi_y)$. ***(a)*** *Our discretization scheme evaluates this flow on a staggered grid, i.e., on the edges of the regular grid for which values of h are stored, which is denoted by non-integer indices like* $y+\frac{1}{2}$. ***(b)*** *Applying this approach recursively to the expression of* Φ *leads to an expression that samples h at each colored cell (see for instance Eq. 8 for* $\Phi_x[x+\frac{1}{2},y]$*).*

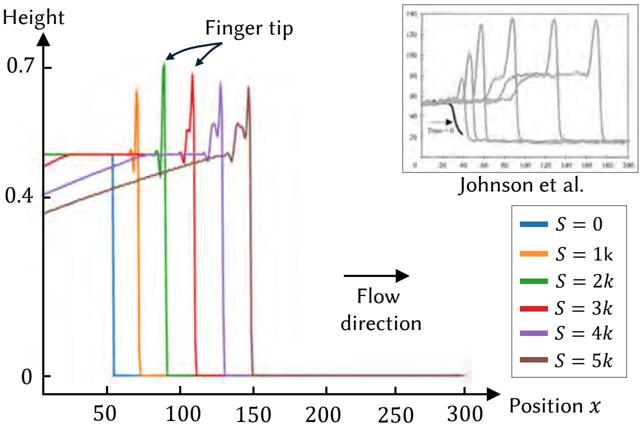


Figure 6: *Height profile of a single slice of fluid flowing down on an inclined surface at different time steps S using our simulation. Similar to real experiments (inset [JSMB99]), we observe an accumulation of matter at the finger tip before dripping.*

Performance. We measured the efficiency of our Thin Film solver and found that performance remains real-time and scaled linearly with regards to the grid discretization, going from 88.0μs (256^2 resolution) to 3.1ms (4096^2 resolution) for a simulation step. Measurements were performed on a NVIDIA RTX 3080 Ti Laptop GPU.

5. Validation and Analysis

The numerical Thin Film model successfully produces dripping fingers as shown in Fig. 3, but this emerging phenomenon is sensitive to the choice of parameters. Here we first validate our formulation against experimental and theoretical results from Physics (Sec. 5.1). Thanks to our connection with physical quantities, and in particular with the scaling of Kondic et al. [Kon03], we perform a bias analysis to identify the range of interest of simulation parameters (Sec. 5.2). We also compare with the formulation from Vantzos et al. [VRBC18] and show that our solver has less bias, which favors dripping behavior.

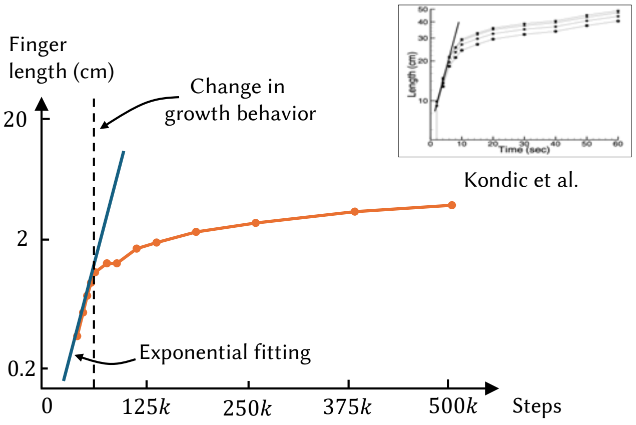


Figure 7: *Evolution of fingers' length measured with our simulation (exponential scale on vertical axis). As predicted by theoretical and experimental results (inset [Kon03]), we see an initial exponential behavior followed by a growth reduction.*

5.1. Validation

Qualitative. Experiments have been conducted on falling thin films and the fingering instability. Johnson et al. [JSMB99] measured the height profile of the fluid on a single y-slice at different time steps, revealing an accumulation of fluid at the tip of the fingers. As reported in Fig. 6, we successfully reproduce this behavior and obtain three key sequential stages as observed by Johnson: accumulation of matter at borders, progressive deformation of the tip, and finally dripping when surface tension is overcome. This is a first good indicator of the physical accuracy of our model.

Quantitative. We also compare our Thin Film method against results from Kondic et al. [Kon03]. They simulate a flowing thin layer of silicon oil and measure the wavelength of the dripping fingers, found to be within $\approx 1.8\ cm$ and $\approx 2.9\ cm$ with an exponential growth phase. To match these experimental results, we set our simulation parameters to match the properties of silicon oil. More precisely, we know its surface tension $\gamma = 21 \cdot 10^{-3}\ N.m^{-1}$, kinematic viscosity $\nu = \mu/\rho = 50 \cdot 10^{-6}\ m^2.s^{-1}$, and mass density $\rho = 960\ kg.m^{-3}$. This allows deducing $\mathrm{Ca} = \rho \nu x_c / \gamma t_c$. The characteristic length and duration are respectively set to $x_c = 1.85mm$ and $t_c = 1.85s$, and we use $\Delta x = 1$ and $\Delta t = 0.1$. These were chosen to have both enough spatial and temporal resolution to properly measure the initial exponential growth phase. Other parameters values are $\mathrm{Ca} = 2.3 \cdot 10^{-3}$, $\eta = 12$, $\epsilon = 0.19$ and $\xi = 0$. Note that we do not match the initial height of fluid from the experiment as it does not fit our main hypothesis for this model, namely that the height is small compared to the horizontal dimensions.

We run the simulation for 500k steps (which corresponds to $\approx$ one day but runs in 70s) at different angles α, and measured the length and spatial frequency of the fingers. Table 2 gathers theoretical wavelengths λ_t obtained through Linear Stability Analysis (LSA), experimental ones λ_{exp}, and measurements with our model λ_{ours}, which fall within a good range against theoretical and experimental values. Fig. 7 shows the evolution of the fingers length with respect to time. Following LSA prediction as well as real ex-

Table 2: *Wavelength between fingers with λ_t the theoretical value from LSA, λ_{exp} the experimental value [Kon03], and λ_{ours} our method.*

α	λ_t (cm)	λ_{exp} (cm)	λ_{ours} (cm)
30°	3.7	2.7	2.9
60°	2.3	2.5	1.9
82°	1.9	1.7	1.8

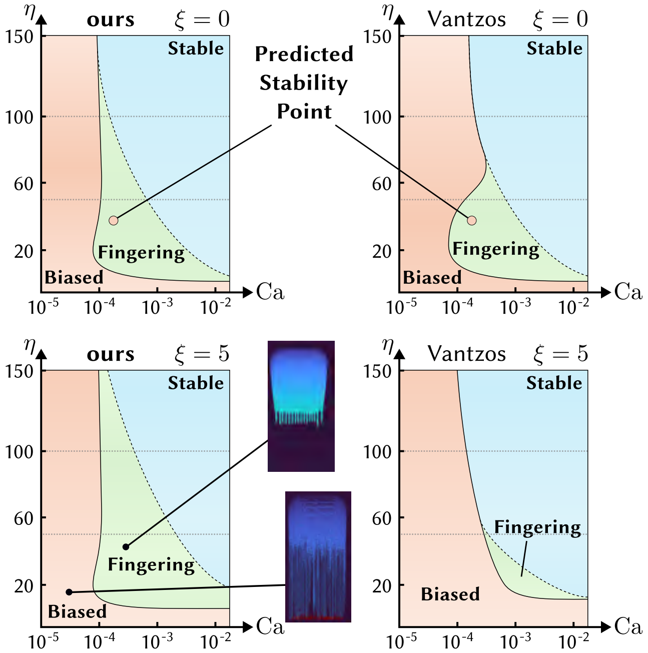


Figure 8: *Starting around the stability point that we predict from Kondic et al. [Kon03], we map the region where bias is introduced and qualitatively locate the range where fingering starts to occur. Bias is either due to a finger size below Δx (small η) or a fluidity too fast for Δt (small Ca). When using the regularization from Vantzos et al. [VRBC18] (right), the biased area takes over the fingering area, whereas our method (left) preserves it better, especially when increasing the hydrophoby ξ (bottom row). Measured for $\Delta x = 1$, with an initial deposit of $h = \frac{1}{2}h_c$ and $h_{\max} = 1.1h_c$.*

periments, the curve shows an initial exponential behavior and the growth then progressively slows down. As we did not match the volume and height of the fluid, the total length differs between experimental and simulated results. This does not impact the growth mechanism or the wavelength of the fingers that we successfully match.

5.2. Bias analysis and mapping of parameters

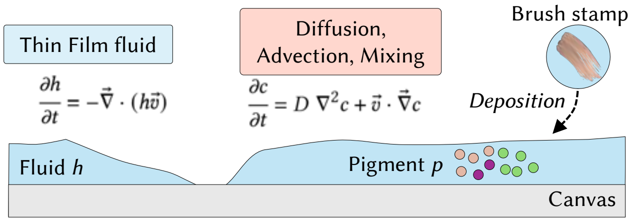
The discrete and local explicit scheme that we employ out of the necessity of simulating paint in real-time has a limited range of validity. We map the region of the parameter space for which the solver behaves correctly by running it at two different time steps $\Delta t = 0.1t_c$ (real-time simulation at 10 steps per frame) and $\Delta t = 0.01t_c$ (slow reference), adapting the number of simulated steps in consequence (30,000 when $\Delta t = 0.1$ and 300,000 when $\Delta t = 0.01$). To compare the resulting simulation states, we use a geometric measure which penalizes changes in the shape of the wet area. We use the Symmetric Difference over Union (see additional material) as it captures well the errors in dripping patterns. This quantitative measure draws the boundary of the *Biased* regions (orange) in Fig. 8 (using a threshold at 0.1). We also qualitatively check for the emergence of dripping fingers (dotted limit between green and blue regions). The stable zone (blue), although valid and robust, leads to simulations with less diverse behaviors (fluid is too viscous). It is from this analysis that we derive the value of η_{max} and the quadratic mapping from T to η, used in Eq. 6.

In order to target the region where dripping fingers effectively emerge, we sample the parameter space around the *Predicted Stability Point*, which we get thanks to our connection with physical quantities in Sec. 5.1. This point corresponds to the constraint that binds the dimensionless parameters described by Kondic et al. [Kon03], namely $\eta^2\epsilon^3 = 1$ and $3\,\mathrm{Ca}\,\eta^2 = 1$. We then sample parameters along a dense grid, thus collecting 108 points per slice of constant ξ. As can be seen on Fig. 8, our solver reduces the bias compared to the state of the art, leading to larger regions where dripping occurs.

6. Paint simulation

Paint is modeled as a fluid heightfield $h : \mathbb{R}^2 \to \mathbb{R}$ and a pigment field $p : \mathbb{R}^2 \to \mathbb{R}^3 \times \mathbb{R}$, which consists of an RGB color and a quantity c in m^3. When painting, the user puts pigments and fluid on the canvas using a 2D brush and the fluid heightfield h evolves using our Thin Film model. A velocity field $\langle \vec{v} \rangle$ (Eq. 3) is derived. Sec. 6.1 explains the motion and diffusion of pigments from this fluid velocity, and Sec. 6.2 describes a phenomenological extension to control the influence of the underlying medium, which we believe is a valuable addition for digital painting (see Fig. 9 for an overview).

Our watercolor simulation combined with dripping allows users to achieve a wide range of effects, featuring dripping and pigment mixing and advection (Fig. 13 and 14). The additional parameter L that we introduce wrt. Kondic et al. [Kon03] gives a more precise control over the fingers length (Fig. 4).



Thin Film fluid
$\frac{\partial h}{\partial t} = -\vec{\nabla} \cdot (h\vec{v})$
Diffusion, Advection, Mixing
$\frac{\partial c}{\partial t} = D\,\nabla^2 c + \vec{v} \cdot \vec{\nabla} c$
Brush stamp
Deposition
Fluid h
Pigment p
Canvas

Figure 9: *Paint is modeled as a thin layer of fluid that carries pigments. It is encoded as a heightfield h governed by a Thin Film model, and a pigment field that is diffused and advected along the fluid velocity.*

6.1. Diffusion and Advection

Both diffusion and advection play a central part in watercolor painting. While advection is solely based on the fluid velocity field, diffusion is responsible for the motion of pigments when the fluid is

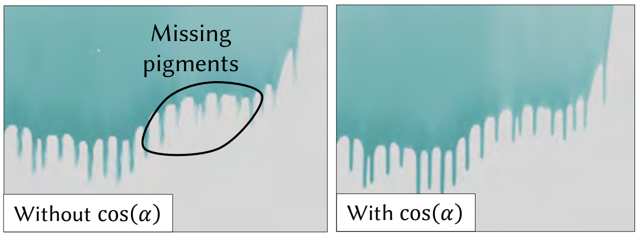


Figure 10: *Influence of the multiplicative diffusion factor* $\cos(\alpha)$ *which cancels diffusion based on the value of the inclination angle of the canvas. In case* $\alpha = \pi/2$*, pigments are moved only using advection which allows them to properly follow the fluid motion.*

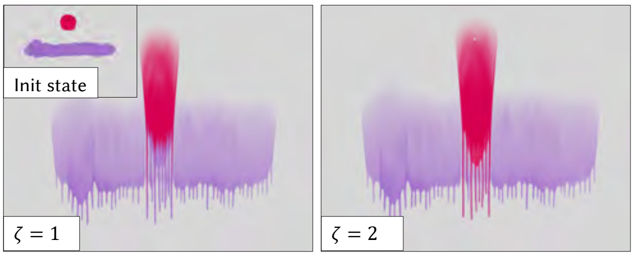


Figure 11: *Influence of the pigment boost* ζ*, which impacts the RGB color mixing by fostering the color of incoming pigments (red) over already present ones (purple), here in case of dripping due to the canvas being tilted.*

not moving. The usual equation to model the mobility of the pigments is the diffusion equation [YJC*13]. We model pigment transport with the following equation:

$$\frac{\partial c}{\partial t} = \underbrace{\cos(\alpha)\, D\, \nabla^2 c}_{\text{Diffusion}} - \underbrace{\vec{v} \cdot \vec{\nabla} c}_{\text{Advection}} \tag{9}$$

with c the pigment quantity and D the diffusion coefficient (set to $5x_c^2/t_c$ in our implementation). The coefficient $\cos(\alpha)$ is an addition from our part which lowers the diffusion effect compared to advection as the canvas is being tilted. This leads to more natural and aesthetic results where pigments properly follow the fluid in case of higher velocities (see Fig. 10).

The advection-diffusion Eq. 9 is meant to deal with particles that are indistinguishable from one another. This is a challenge when modeling pigments with a wide variety of colors. One solution is to consider a palette with a fixed number of base colors [VLVR05] and to simulate all of them independently. This is computationally demanding and leads to a high memory footprint as quantities of all pigments need to be stored for each cell. The key problem occurs when $\partial c/\partial t = 0$, for instance when all neighbor cells have the same pigment and water quantities: the pigment quantity does not change, but the colors C of pigments should nonetheless mix. To address this, we decompose the right-hand side of Eq. 9 into four flow terms $(\varphi_i)_{0 \leq i < 4}$ (for the four neighbors i), of which we distinguish the incoming positive part φ_i^+ from the outgoing negative part φ_i^-. We thus consider the new pigment color of the cell to be a weighted average $C \leftarrow \frac{1}{A}\left[(c + \sum_i \varphi_i^-)C + \sum_i \varphi_i^+ C_i\right]$ with a normalization term $A = c + \sum_i \varphi_i^+ + \varphi_i^-$. Furthermore, when paint is in motion, for instance when it drips due to gravity, we want newly arriving pigments to cover those that are already present. We thus boost the contribution of incoming pigment colors by multiplying all occurrences of φ_i^+ by a parameter ζ. See Fig. 11 and suppl. video for results.

6.2. Local variations of gravity

The underlying medium has a strong influence on the way watercolor flows. We model the canvas relief as a heightfield m, from which we locally rotate the gravity vector at each pixel (see Fig. 12). This approach is more direct than computing a field of gravity potential that would match the expected canvas grain. From the local gravity, the velocity from Eq. 3, that derives from integrating Navier-Stokes along the z axis, becomes (see additional material):

$$\langle \vec{v} \rangle = \frac{h^3}{8\mu} \rho \vec{\nabla} g_z + \frac{h^2}{3\mu}\left((\rho g_z + f_z)\vec{\nabla} h + \gamma \vec{\nabla} \nabla^2 h + \rho \vec{g_{xy}}\right) \tag{10}$$

And as a consequence the scaled Thin Film equation 5 becomes:

$$\frac{\partial \mathbf{h}}{\partial \mathbf{t}} = \frac{1}{3\,\mathrm{Ca}\,\eta^2} \vec{\nabla} \cdot \left[\mathbf{h}^3 \left(X - \frac{3\epsilon}{8} \mathbf{h}\, \vec{\nabla} \Gamma_z\right)\right] \tag{11}$$

where $X = -\eta^2 \epsilon^3\, \vec{\nabla} \nabla^2 \mathbf{h} - \epsilon(\Gamma_z + \xi)\, \vec{\nabla} \mathbf{h} - \vec{\Gamma_{xy}}$ regroups the terms already present in Eq. 5. This allows paint to deposit non-uniformly in small concavities in the canvas, as shown in Fig. 12.

6.3. Comparison with other painting system

As no code is readily available for other painting techniques, we perform a qualitative comparison. The watercolor painting system of Van Laerhoven et al. [VLVR05] considered only a fixed number of base colors for memory and computational reasons, which limits the expressiveness. In contrast, we allow arbitrary RGB colors to be stored in cells, and our advection-diffusion scheme ensures that mixing occurs between two cells even with similar quantities. The main drawback is that more advanced mixing schemes [SJ21] may not be trivial to implement, but we leave this as future work. On top of that, no dripping behavior was showcased. More recent work such as WetBrush [CKIW15] showcases complex brush-canvas behavior by modeling individual bristles as a set of particles. Although they focused on viscous types of paint such as oil and acrylic, we believe our work is integrable with their advanced modeling of the brush. Wetbrush is computationally intensive, but as our thin fluid simulation performance scales well, we believe the two are compatible and would lead to a more complete watercolor painting application.

7. Limitations and Conclusion

Limitations. Our integration of Thin Film modeling in a digital painting context allows for novel artistic interaction, but artists sometimes felt the need to speed up the emergence of fingers beyond the F_{max} limit. This is an inherent limit of the explicit numeric scheme that we adopted, whose regularization term trades dripping patterns for the sake of stability. Regarding painting, our pigment

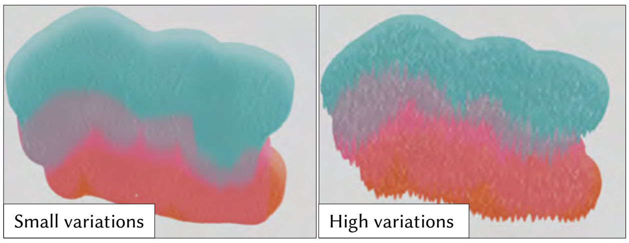


Figure 12: *Influence of local gravity. On top of emphasizing the emergence of fingers, water and pigments deposit in local minima of the canvas, leading to more interesting visual effects.*

mixing remains linear and does not reproduce realistic behavior that artists are used to (i.e., blue and yellow make green).

Future work. We presented a method to simulate the evolution of dripping fingers of fluids for the context of watercolor painting. For further improvements, one could review the physical derivation to enable slightly more turbulent behavior, enabling proper accumulation of fluid in the presence of obstacles. This may involve modeling more than just the static amount of fluid per pixel. Coupling pigment and fluid simulation may allow us to model more phenomena, for instance the coffee ring effect. This is caused by the Marangoni flow, a phenomenon that emerges when a gradient of surface tension is created at a fluid-air interface, and that causes pigment accumulation at the border when the fluid dries.

Acknowledgments. We thank Daichi Ito for the creation of several paintings and feedback during development.

References

[Anc07] ANCEY C.: Plasticity and geophysical flows: A review. *Journal of Non-Newtonian Fluid Mechanics 142*, 1 (2007), 4–35. 2

[BBP15] BARLA P., BELCOUR L., PACANOWSKI R.: In praise of an alternative brdf parametrization. In *Proceedings of the Third Workshop on Material Appearance Modeling: Issues and Acquisition* (2015), Eurographics Association. 1

[BHG01] BARAL D., HUTTER C., GREVE R.: Asymptotic theories of large-scale motion, temperature, and moisture distribution in land-based polythermal ice sheets: A critical review and new developments. *Applied Mechanics Reviews - APPL MECH REV 54* (05 2001). 2

[BWL04] BAXTER W. V., WENDT J., LIN M. C.: IMPaSTo: A realistic, interactive model for paint. In *Proceedings of the International Symposium on Non-Photorealistic Animation and Rendering, NPAR* (2004), ACM, pp. 45–56. 2

[CKIW15] CHEN Z., KIM B., ITO D., WANG H.: Wetbrush: Gpu-based 3d painting simulation at the bristle level. 2, 8

[CLH16] CHI M., LIU W., HSU S.: Image stylization using anisotropic reaction diffusion. *Vis. Comput. 32*, 12 (2016), 1549–1561. 1

[CM09] CRASTER R. V., MATAR O. K.: Dynamics and stability of thin liquid films. *Rev. Mod. Phys. 81* (Aug 2009), 1131–1198. 2

[CMVHT02] CARLSON M., MUCHA P. J., VAN HORN R. B., TURK G.: Melting and flowing. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2002), p. 167–174. 2

[COKE20] CANABAL J. A., OTADUY M. A., KIM B., ECHEVARRIA J.: Simulation of dendritic painting. *Computer Graphics Forum 39*, 2 (2020), 597–606. 2

[CT02] CHU N.-H., TAI C.-L.: An efficient brush model for physically-based 3d painting. In *10th Pacific Conference on Computer Graphics and Applications, 2002. Proceedings.* (2002), pp. 413–421. 2

[CT05] CHU N. S.-H., TAI C.-L.: Moxi: real-time ink dispersion in absorbent paper. 2

[DJ18] DUPUY J., JAKOB W.: An adaptive parameterization for efficient material acquisition and rendering. *Transactions on Graphics (Proceedings of SIGGRAPH Asia) 37*, 6 (2018). 1

[DK01] DIEZ J., KONDIC L.: Contact line instabilities of thin liquid films. *Physical review letters 86* (02 2001), 632–5. 2

[DKMI13] DIVERDI S., KRISHNASWAMY A., MĚCH R., ITO D.: Painting with polygons: A procedural watercolor engine. *IEEE Transactions on Visualization and Computer Graphics 19* (2013), 723–735. 2

[FRLH23] FENDER A. R., ROBERTS T., LUONG T., HOLZ C.: Infinitepaint: Painting in virtual reality with passive haptics using wet brushes and a physical proxy canvas. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (2023), Association for Computing Machinery. 2

[HIK*20] HUANG W., ISERINGHAUSEN J., KNEIPHOF T., QU Z., JIANG C., HULLIN M. B.: Chemomechanical simulation of soap film flow on spherical bubbles. *ACM Trans. Graph. 39*, 4 (2020). 2

[HW65] HARLOW F. H., WELCH J. E.: Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *The Physics of Fluids 8*, 12 (12 1965), 2182–2189. 2

[ISN*20] ISHIDA S., SYNAK P., NARITA F., HACHISUKA T., WOJTAN C.: A model for soap film dynamics with evolving thickness. *ACM Trans. Graph. 39*, 4 (2020). 2

[JSMB99] JOHNSON M. F. G., SCHLUTER R. A., MIKSIS M. J., BANKOFF S. G.: Experimental study of rivulet formation on an inclined plate by fluorescent imaging. *Journal of Fluid Mechanics 394* (1999), 339–354. 2, 6

[KKK*16] KRUEGER T., KUSUMAATMAJA H., KUZMIN A., SHARDT O., SILVA G., VIGGEN E.: *The Lattice Boltzmann Method: Principles and Practice*. Graduate Texts in Physics. Springer, 2016. 1

[Kon03] KONDIC L.: Instabilities in gravity driven flow of thin fluid films. *Siam review 45*, 1 (2003), 95–115. 1, 2, 3, 4, 6, 7

[KRQSV11] KALLIADASIS S., RUYER-QUIL C., SCHEID B., VELARDE M.: Falling liquid films. 2, 4

[LBB17] LARIONOV E., BATTY C., BRIDSON R.: Variational stokes: a unified pressure-viscosity solver for accurate viscous liquids. *ACM Trans. Graph. 36*, 4 (2017). 2

[LJK14] LI Y., JEONG D., KIM J.: Adaptive mesh refinement for simulation of thin film flows. *Meccanica 49*, 1 (Jan. 2014), 239–252. 2

[ODB97] ORON A., DAVIS S. H., BANKOFF S. G.: Long-scale evolution of thin liquid films. *Rev. Mod. Phys. 69* (Jul 1997), 931–980. 2, 4

[PT24] PROBST T., TESCHNER M.: Unified pressure, surface tension and friction for sph fluids. *ACM Trans. Graph. 44*, 1 (2024). 2

[RV13] RUMPF M., VANTZOS O.: Numerical gradient flow discretization of viscous thin films on curved geometries. *Mathematical Models and Methods in Applied Sciences 23*, 05 (2013), 917–947. 2

[SJ21] SOCHOROVÁ V., JAMRIŠKA O.: Practical pigment mixing for digital painting. *ACM Trans. Graph. 40*, 6 (2021). 8

[SKJY06] SANDERSON A., KIRBY R., JOHNSON C., YANG L.: Advanced reaction-diffusion models for texture synthesis. *J. Graphics Tools 11* (01 2006), 47–71. 1

[SLW*23] STOMAKHIN A., LESSER S., WRETBORN J., FLYNN S., NIXON J., ILLINGWORTH N., ROLLET A., BLOM K., MCHALE D.: Pahi: A unified water pipeline and toolset. In *Proceedings of the 2023 Digital Production Symposium* (2023), Association for Computing Machinery. 2

[SMB19] STOMAKHIN A., MOFFAT A., BOYLE G.: A practical guide to thin film and drips simulation. In *ACM SIGGRAPH 2019 Talks* (2019), Association for Computing Machinery. 2

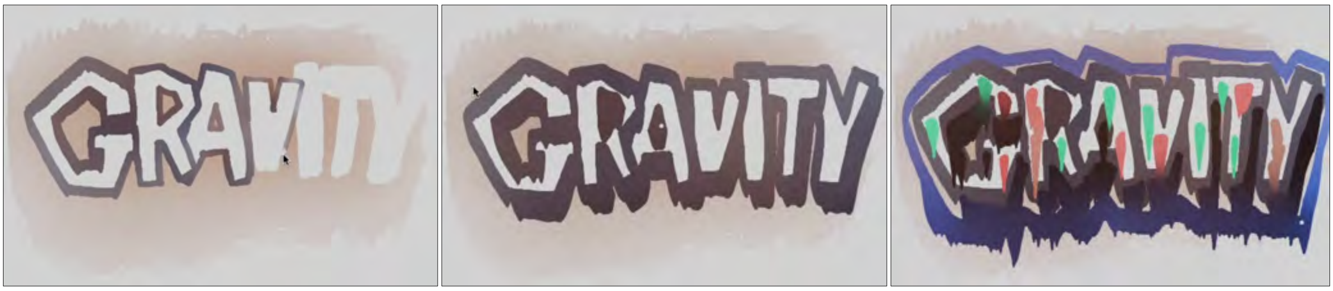


Figure 13: *Three frames of a painting session from an artist using gravity to create dripping around a WordArt (see accompanying video).*

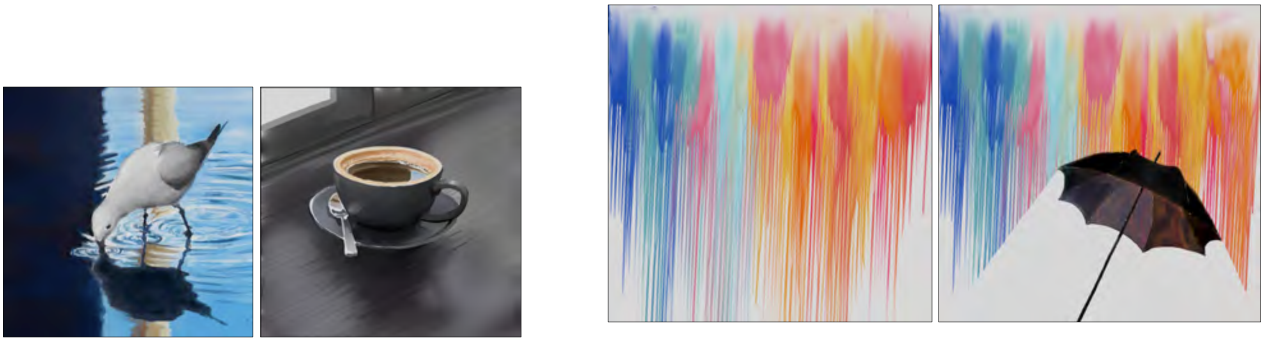


Figure 14: ***Left:*** *Two paintings that showcase our pigment diffusion schemes done by a professional artist over a 1k canvas (see accompanying video for interactive sequences).* ***Right:*** *Two frames of a painting session where the artist extensively used dripping and pigment mixing before creating the umbrella shape.*

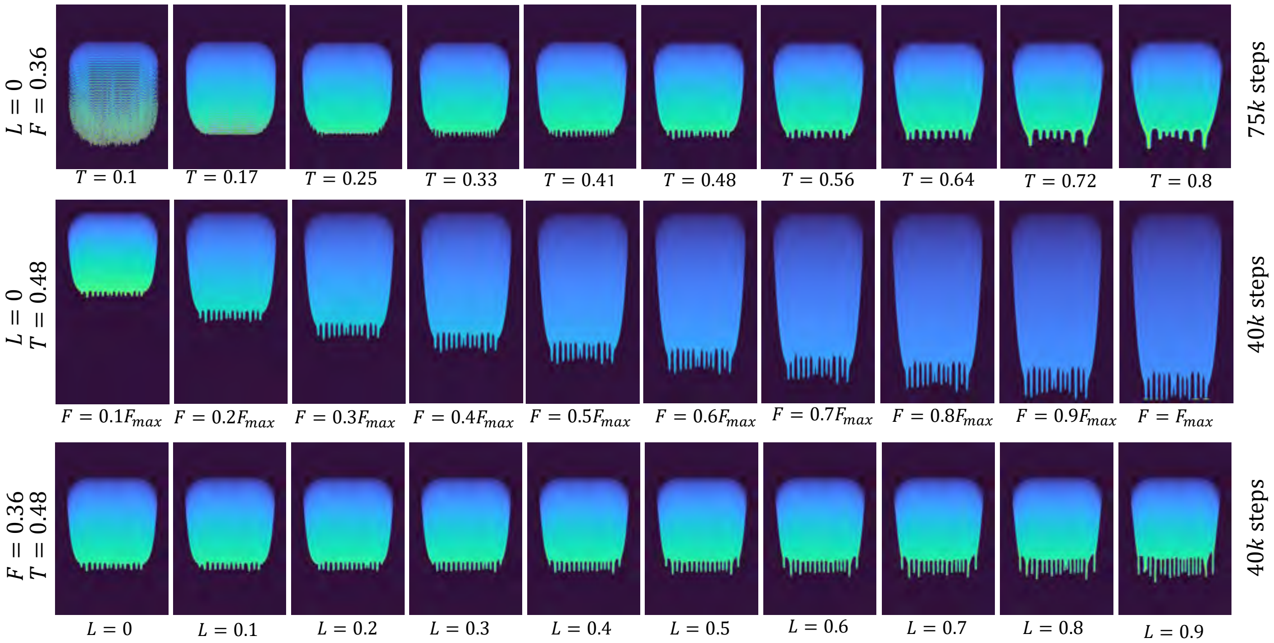


Figure 15: *Our principled parameters F, T, and L provide a decoupled control over the speed of the flow as well as the thickness and length of the dripping fingers. Artifacts occurring at $T = 0.1$ are due to insufficient grid resolution (Δx too large) with regards to the desired thickness and fluidity.*

[Sto45] STOKES G. G.: On the theories of internal friction of fluids in motion and of the equilibrium and motion of elastic solids. *Transactions of the Cambridge Philosophical Society 8* (1845), 287–305. 1, 3

[VAW*17] VANTZOS O., AZENCOT O., WARDEZTKY M., RUMPF M., BEN-CHEN M.: Functional thin films on surfaces. *IEEE Transactions on Visualization and Computer Graphics 23*, 3 (2017), 1179–1192. 2

[VIC98] VERETENNIKOV I., INDEIKINA A., CHANG H.-C.: Front dynamics and fingering of a driven contact line. *Journal of Fluid Mechanics 373* (1998), 81–110. 2

[VLVR05] VAN LAERHOVEN T., VAN REETH F.: Real-time simulation of watery paint. *Computer Animation and Virtual Worlds 16*, 3-4 (2005), 429–439. 2, 8

[VRBC18] VANTZOS O., RAZ S., BEN-CHEN M.: Real-time viscous thin films. *ACM Trans. Graph. 37*, 6 (dec 2018). 1, 2, 4, 5, 6, 7

[WLT18] WANG X., LIU S., TONG Y.: Stain formation on deforming inelastic cloth. *IEEE Transactions on Visualization and Computer Graphics 24*, 12 (2018), 3214–3224. 2

[WR04] WEINSTEIN S. J., RUSCHAK K. J.: Coating flows. *Annual Review of Fluid Mechanics 36*, Volume 36, 2004 (2004), 29–53. 1, 2

[YJC*13] YOU M., JANG T., CHA S., KIM J., NOH J.: Realistic paint simulation based on fluidity, diffusion, and absorption. *Computer Animation and Virtual Worlds 24* (05 2013). 2, 8