Computational Thermoforming

Christian Schüller[1] Daniele Panozzo[1] Anselm Grundhöfer[2] Henning Zimmer[2] Evgeni Sorkine[1] Olga Sorkine-Hornung[1]

[1]ETH Zurich [2]Disney Research

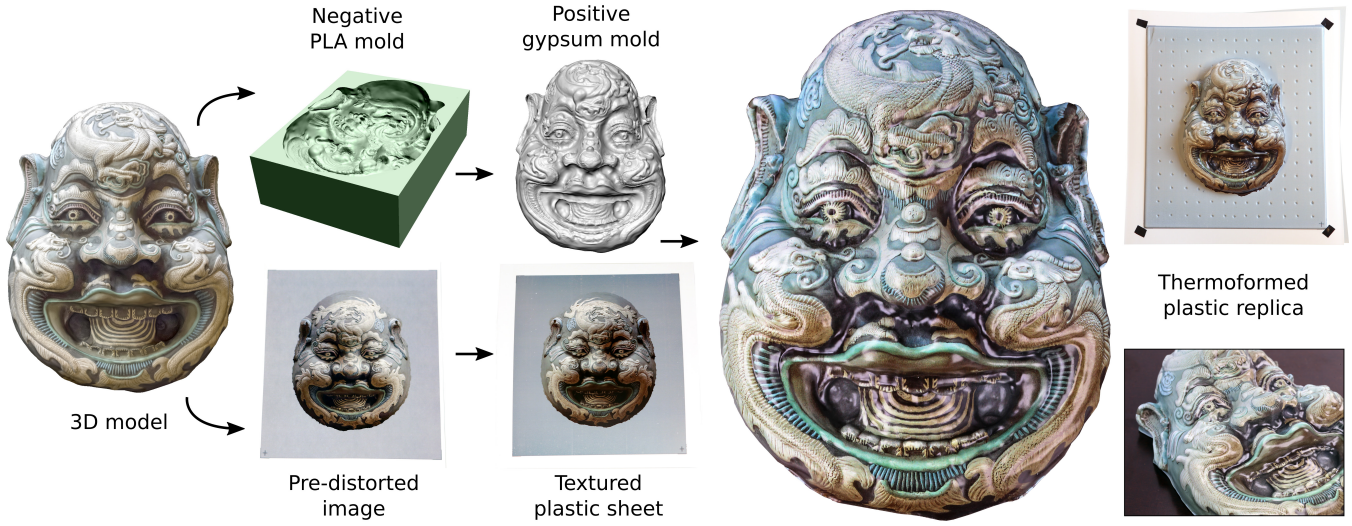


Figure 1: *Our pipeline for producing plastic replicas of textured digital 3D models by computational thermoforming.*

Abstract

We propose a method to fabricate textured 3D models using thermoforming. Differently from industrial techniques, which target mass production of a specific shape, we propose a combined hardware and software solution to manufacture customized, unique objects. Our method simulates the forming process and converts the texture of a given digital 3D model into a pre-distorted image that we transfer onto a plastic sheet. During thermoforming, the sheet deforms to create a faithful physical replica of the digital model. Our hardware setup uses off-the-shelf components and can be calibrated with an automatic algorithm that extracts the simulation parameters from a single calibration object produced by the same process.

Keywords: thermoforming, vacuum forming, digital fabrication, texture transfer, viscoplastic sheets, physical simulation

Concepts: •**Computing methodologies** → *Shape modeling;*

1 Introduction

Automatically creating faithful physical replicas of digital 3D models is one of the major challenges in digital fabrication. Many fabrication techniques have been proposed to accurately reproduce the geometry of a 3D model, but very few methods can produce objects with a colored surface.

We propose a hardware and software solution to produce highly detailed textured objects using thermoforming [Klein 2009]. Thermoforming, also called vacuum forming, is an industrial process used to fabricate a large part of the objects we use daily, such as food packaging, disposable plates, blister packaging, plastic toys and interior paneling. The thermoforming process deforms a plastic sheet, forcing it to assume the shape of the desired mold. First, the sheet is heated until it transitions to a viscous state, and then vacuum is created below it, so that the sheet tightly adheres to the mold. Interestingly, the technique can produce highly detailed colored objects if an image is printed onto the plastic sheet prior to deformation. This method is currently limited to industrial applications, since printing on a thick plastic sheet requires a flatbed printer and heat-resistant inks, and the cost and effort of producing a mold is usually justifiable only if it is used to produce a considerable amount of thermoformed objects with the same shape.

Our proposal uses the same principle, but it is tailored to small-scale production and is accessible to universities, fabrication labs and hobbyists. On the algorithmic side, we propose a software simulation that creates the necessary pre-distorted texture image to be printed on the plastic, thereby ensuring that once the sheet is deformed, each pixel of the texture lands in its correct location on the 3D shape. The material model and the parameters for the simulation are automatically extracted by scanning and analyzing a single calibration object made with our forming pipeline. On the hardware side, we propose an effective method to produce a gypsum mold using a 3D printer with polylactic acid (PLA) filament, and a simple way to print texture on a plastic sheet using a standard color laser printer and transfer paper. The individual hardware components in our pipeline can be easily substituted thanks to our simple calibration procedure.

SIGGRAPH 2016 Posters, July 24-28, 2016, Anaheim, CA
ISBN: 978-1-4503-4279-7/16/07
DOI: http://dx.doi.org/10.1145/2897824.2925914

We validate our method with objective experiments that densely measure the fabrication errors, and with qualitative examples that demonstrate the variety of objects that can be fabricated with our technique. We provide comparisons with textured objects produced by hydrographic transfer and color 3D printing, where our technique provides superior quality while being considerably cheaper and faster.

We expect our contribution to have a strong impact both in digital fabrication, where it allows inexpensive production of highly detailed physical replicas of digital objects, and in industrial applications, where multiple designs can be easily tested before starting the mass production of thermoformed products.

The main contributions of this paper can be summarized as follows:

1. An algorithm to convert a textured digital 3D model into a 3D-printable mold, plus the image to be printed on the plastic sheet before thermoforming.
2. A calibration algorithm that estimates the material parameters from a set of photographs taken with a reflex camera.
3. A low-cost thermoforming procedure that employs off-the-shelf hardware to transfer the texture and to fabricate the mold.

2 Related work

Manufacturing textured objects is an important long-standing problem that has been tackled with many technologies in the last decades. We provide an overview of the state-of-the-art methods proposed both in academia and in industry.

Color 3D printing. Powder 3D printers can produce textured objects by mixing colored binders during the printing process [3dsystems 2016]. Similarly, paper printers can produce objects by glueing printed sheets of paper together [Ltd. 2016]. Both technologies require expensive dedicated 3D printers and long printing time [Rouiller et al. 2013; Vidimče et al. 2013; Hergel and Lefebvre 2014; Reiner et al. 2014; Cuttlefish 2015]. Our proposal has much lower hardware requirements, and the fabricated objects enjoy a superior surface quality (Figure 7).

Hydrographics. Texture can be transferred onto an object after its fabrication using water transfer printing, or hydrographics [Zhang et al. 2015; Panozzo et al. 2015]. The texture image is printed on a polymer sheet that dissolves in water, leaving the ink floating on its surface. The object is then dipped into the water to transfer the ink onto it. A limitation of this technique is that flat regions parallel to the water surface cannot be properly colored due to the formation of air bubbles. Our method shares some similarities with hydrographics, but thanks to the use of thermal transfer paper instead of water, the colors are more vivid and flat parts are not problematic. The surface finish is also superior (Figure 7). However, our method is limited to plastic materials and cannot be used to fabricate colored wood or glass objects.

Projection. Time-varying texture can be applied to objects that undergo rigid transformations or non-rigid animation known in advance using projectors [Lincoln et al. 2009; Raskar et al. 2001; Bermano et al. 2013]. However, this technique only temporarily produces the colored appearance, whereas we aim to permanently color the objects.

Assisted painting. A few recent techniques propose to paint a flat canvas using a robot [Lindemeier et al. 2013] or computer controlled spraying [Shilkrot et al. 2015; Prévost et al. 2015]. These techniques are currently restricted to flat surfaces, and it would be challenging to extend them to paint on a complex shape such as our result in Figure 1.

Industrial thermoforming. Several commercial software solutions [Accuform 2016; Rheoware 2016; ESI 2016] can simulate the thermoforming process by using advanced non-linear FEM models for plastic sheets [Nied et al. 1990; Kouba et al. 1992; Koziey et al. 1997]. Differently from us, they are particularly interested in the thickness of the sheet after deformation, which determines the robustness and material properties of the fabricated good. They rely on a large set of parameters that are difficult to find without physical material tests [Mieghem et al. 2015]; the tests are expensive and must be performed on each set of machines and materials used, making them feasible only for industrial production.

While the methods above could be used for creating textured models, their running time and set-up cost is not required for our application. We demonstrate that a simplified simulation that runs in minutes instead of hours is sufficient to obtain a high accuracy while avoiding the necessity of performing material tests. We estimate the parameters of the simulation using a grid search on the small set of parameters of our simplified model. The same approach would be infeasible with an elaborate material model and simulation approach since it would require an unrealistic amount of computation.

Another interesting industrial approach to textured thermoforming [Thermo3D 2016] requires thermoforming the desired shape with a special calibration texture and then 3D scanning it. Correspondences between the flat texture and the formed 3D surface are then extracted, and the resulting mapping is employed to apply arbitrary images and text onto each subsequent copy of the same 3D object. This approach is limited to simple geometries that are easy to scan. Our calibration procedure is based on this idea, and we propose an algorithm that uses optical flow to obtain an accurate and dense mapping between the flat sheet and the calibration object's surface. Having this mapping, we estimate the parameters of our simulation. In contrast to [Thermo3D 2016], we only need to perform this process once for each type of hardware setup (not per shape) during the initial calibration, using a simple shape that can be easily 3D scanned. After having estimated the simulation parameters from it, our thermoforming approach scales to extremely detailed objects (Figure 1).

3 Method

In this work, we use inexpensive off-the-shelf hardware and we demonstrate that it is sufficient to produce high-quality thermoformed objects. We start by detailing our hardware procedure, which combines thermal color transfer with thermoforming; we then explain the algorithm to generate the distorted texture image to be printed on the plastic sheet.

3.1 Hardware setup

Thermoforming is a manufacturing process where a plastic sheet is heated to a forming temperature, deformed to a specific shape in a mold, and trimmed to create a usable product. The procedure for the small manual thermoforming machines, which are commonly available in any fabrication lab, is divided into four steps:

1. *Preparation*: the mold is anchored to the vertically movable platform and lowered into the chamber of the forming machine. The plastic sheet is placed on top of the chamber to seal it and is clamped by a metal frame.
2. *Heating*: the plastic sheet is uniformly heated to forming temperature, which is above the glass transition of the plastic.
3. *Mold raise*: the mold is raised and pushed into the plastic sheet.

4. *Vacuum*: a vacuum is created between the mold and the sheet, resulting in forces that pull the heated plastic to the mold.

The deformed plastic is optionally trimmed to remove the border. The above technique can produce colored objects by printing an image onto the plastic sheet before thermoforming it. The printing requires the use of dedicated, expensive flatbed-printers and a special heat resistant ink [FUJIFILM 2016] to withstand the high temperature and stretch of the surface in the vacuum forming process, limiting it's applicability to an industrial setting.

Texture transfer. We found that with a special thermal transfer paper [transferpaper 2015], it is possible to inexpensively print and thermoform high-resolution images on plastic sheets with a visual quality comparable to the industrial approach. The procedure is simple and similar to the printing of custom graphics on T-shirts. First an image is printed on the transfer paper using a standard office laser printer. Then it is transfered onto plastic with a common thermal press, gluing the toner particles onto the sheet's surface. The resulting prints have a vivid color and are robust to heat and stretch deformation in the thermoforming process (Figure 2).

Mold fabrication. We use a combination of 3D printing and casting to fabricate heat-resistant molds. Using boolean operations, we produce a negative copy of the 3D shape we would like to replicate (Figure 3). The negative mold is then 3D printed using a PLA printer and filled with a gypsum water mix. The extraction of the cast would be extremely difficult for complex shapes with concavities, but, incidentally, PLA is the perfect material for this application. We can heat the PLA using a heat gun until it melts and then extract the gypsum mold without any risk of damaging it (Figure 3).

Off-the-shelf hardware. This novel combination of texture transfer and mold fabrication drastically reduces the cost and the time needed to thermoform colored objects without compromising the quality of the results. We used this technique to produce all results in the paper and, to the best of our knowledge, this approach has never been used before in thermoforming applications.

3.2 Simulation

The thermoforming process induces a complex deformation of the plastic sheet to adapt its geometry to the mold. We propose an algorithm to simulate this deformation and to invert it, effectively converting a textured digital 3D model into the 2D image to be printed on the plastic sheet prior to deformation.

Assumptions. To simplify the simulation, we make the following assumptions:

1. The adhesion force between the plastic and the mold is infinite [Nied et al. 1990]: the parts of the sheet that touch the mold move rigidly with it.
2. The temperature of the plastic sheet is uniform, and the thermoforming process is fast, making the effect of the plastic cooling negligible.
3. The change in pressure due to the activation of the vacuum pump is instantaneous.

These assumptions allow us to design an efficient algorithm for the entire simulation, enabling to find the optimal parameters for a specific hardware setup using a grid search (Section 3.3). The efficiency of our method is also convenient while preparing new designs: it only takes us minutes to compute and visualize a digital preview of the thermoforming result of a model with complex geometry.

We use the thin sheet model proposed in [Batty et al. 2012] with an additional plasticity model and a simplified handling of contacts.

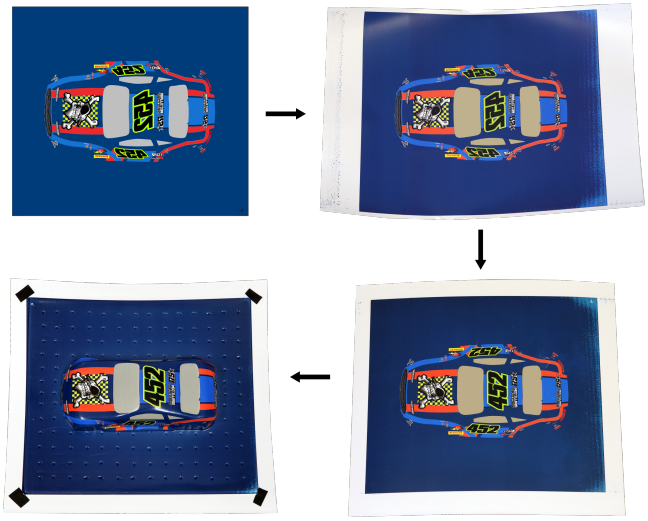


Figure 2: *Original image, printed image, transferred on plastic, thermoformed.*

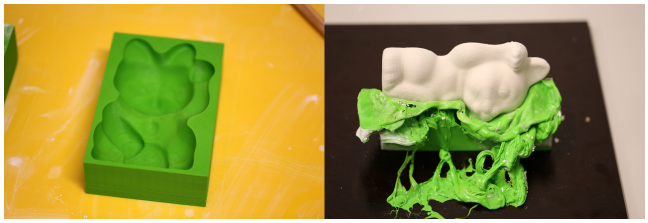


Figure 3: *Negative mold, half-melted mold revealing the positive gypsum cast.*

For the sake of completeness, in the following we report all details of our simulation, and we provide MAPLE scripts to generate C++ code for the gradient and Hessian of the energy in the additional material.

Discrete viscous sheets [Batty et al. 2012]. The plastic sheet (membrane) is represented as a triangle mesh with a scalar field on faces, whose values represent the thickness of the sheet. To denote quantities in the undeformed (reference) state, we use a bar over the respective letter. To model the stretching of the membrane we use the hyperelastic St. Venant-Kirchhoff material model. For a single triangle the Green strain is defined as

$$\varepsilon = \frac{1}{16A^2}\sum_{i=1}^{3}(l_i^2 - \bar{l}_i^2)(t_j \otimes t_k + t_k \otimes t_j), \qquad (1)$$

where A is the triangle area, l_i and $\bar{l}_i$ are deformed and reference lengths of edge i, respectively. $\otimes$ denotes the outer product of two vectors, and t_i is the outward normal to the edge i in the plane of the undeformed triangle. The strain-energy potential over the surface is:

$$E_s = \frac{1}{2}\sum_t A_t \frac{Y h_t}{(1-v^2)}\left((1-v)Tr\left((\varepsilon_t)^2\right) + v(Tr(\varepsilon_t))^2\right),$$

where Y is Young's modulus, v is Poisson's ratio, and h_t is the thickness value of triangle t.

The membrane model does not capture the bending of the membrane, which is modeled separately by a term that depends on the dihedral

angle θ_e between each pair of faces sharing an edge e:

$$E_b = \sum_e \frac{Y h_e^3}{12(1-v^2)} \frac{3\bar{l}_e^2}{A_s} (\theta_e - \bar{\theta}_e)^2,$$

where h_e is the mean thickness of the two incident triangles on e and A_s their summed area.

Viscosity is modeled using forces derived from a discrete dissipative potential:

$$E_{\text{visc}} = (o/\triangle t)\, E(p_{k+1}, p_k),$$

where E is an elastic potential expressed in terms of the deformed and a reference material configuration. In this case these are the configurations between two consecutive integration steps p_{k+1} and p_k, where $\triangle t$ is the step size and the scalar o controls the viscosity. For the stretching energy we have $E_s(\varepsilon_{k+1}, \varepsilon_k)$, where ε_{k+1} and ε_k are the corresponding strains. Equivalently, for the bending energy we use $E_b(\theta_{e,k+1}, \theta_{e,k})$. Viscous forces are then computed by differentiation of E_{visc} with respect to the end-of-step configuration p_{k+1}.

Incompressibility is enforced by updating the thickness values h of all triangles after each integration step to be $h = V/A$, where A is the current triangle area and V it's constant volume.

The vertex positions are updated at each step using a first order implicit Euler (backward Euler) integration; we refer to [Witkin and Baraff 1997] for a detailed description. This concludes the summary of the model proposed in [Batty et al. 2012]. In the following we discuss the extensions necessary to adapt it to simulate a thermoforming process.

Plasticity. We experimented with the fully viscous model proposed in [Batty et al. 2012] and realized that it is problematic during the heating phase of the thermoforming process: with pure viscosity, the sheet flows down the forming chamber instead of only slightly bending. We found that the behavior of heated plastic can be approximated well with an additive model [Hill 1998; Simo and Hughes 1998]:

$$\varepsilon = \varepsilon^e + \varepsilon^p,$$

where the strain ε is divided into elastic and plastic parts. Following [Müller and Gross 2004; O'Brien et al. 2002], we similarly update the plastic strain after every time step only if its norm exceeds the yield strain c_{yield} (which is a property of the material):

$$\text{if } \|\varepsilon\|_2 > c_{\text{yield}}, \;\; \varepsilon^p_{k+1} = \varepsilon^p_k + \triangle t \cdot c_{\text{creep}} \cdot \varepsilon^e,$$

where c_{creep} is a material parameter that controls the plastic flow velocity. Note that the update of the elastic strain cannot be bigger than $\triangle t \cdot c_{\text{creep}} \leq 1$.

Plastic rest pose. We update the rest pose of the mesh to directly account for the plastic deformation of the sheet.

By fixing an orthogonal reference system for each triangle whose third axis is parallel to the triangle normal, the symmetric Green strain tensor (Eq. (1)) has the following form:

$$\varepsilon = \begin{bmatrix} e_1 & e_2 & 0 \\ e_2 & e_3 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

where there are only 3 independent coefficients. Note that in Eq. (1), the strain is expressed as a linear combination of the difference of the squared edge lengths $s_i = l_i^2 - \bar{l}_i^2$. We can thus rewrite this relation in matrix form as:

$$\mathbf{T}\mathbf{s} = \mathbf{e},$$

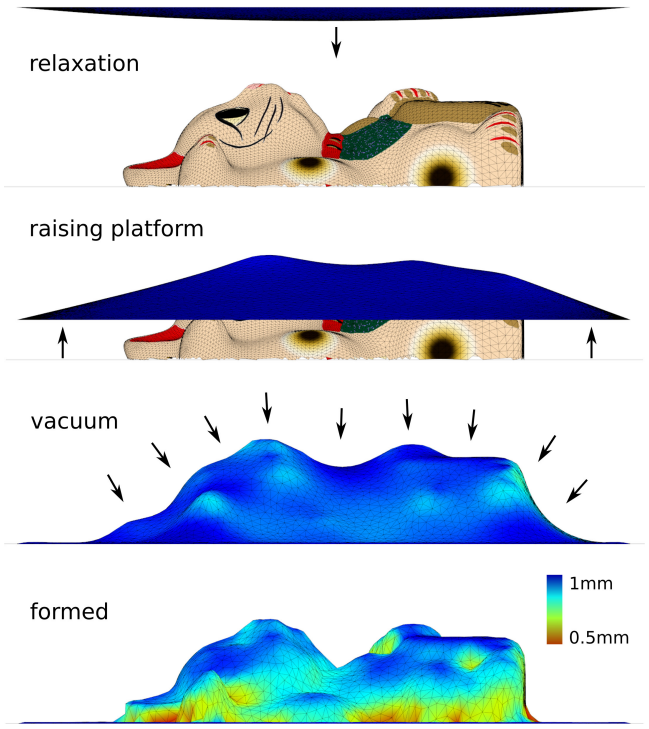


Figure 4: *The simulation is divided in three phases: from top to bottom, relaxation, raising the mold, activating the vacuum. The color of the sheet visualizes the thickness of the simulated plastic sheet.*

where $\mathbf{s} = [s_1, s_2, s_3]$ and $\mathbf{e} = [e_1, e_2, e_3]$. Solving this linear system (and removing the constant $\bar{l}_i^2$) gives us the squared edge lengths of the rest pose. To reconstruct the mesh, we consider each triangle independently, and we reconstruct its geometric embedding using the closed-form formula in the Appendix.

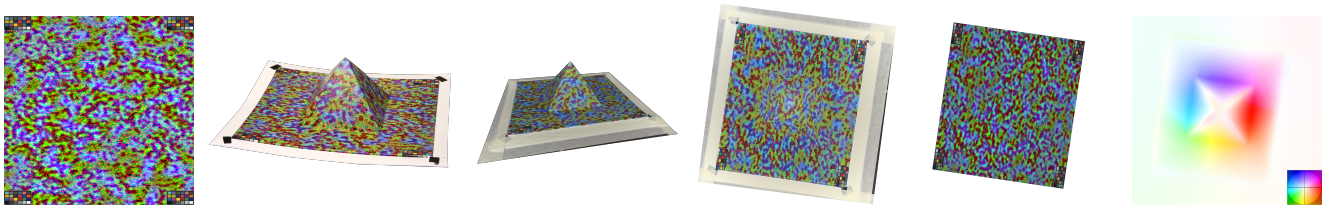
Since the membrane is very thin, we simplify the bending plasticity formulation by always updating the rest pose dihedral angles with the current ones, following [Batty et al. 2012].

External forces. There are two kinds of external forces that act on the mesh vertices: The first is gravity, which is modeled as a constant force f_g in the negative z direction. The vacuum pressure induces forces proportional to the vertex Voronoi areas, oriented in the opposite direction of the surface normal. Gravity is active during the entire simulation, while the vacuum acts only in the last stage.

Contacts. Since the adhesion forces between the plastic and the mold are dominant [Nied et al. 1990], we glue the sheet to the mold upon contact. We only check for collisions between the sheet vertices and mold triangles and use hard positional constraints to move them rigidly with the mold. To account for the thickness of the sheet, we interpolate the triangle thickness values to the vertices and perform the collision detection on two offset surfaces, using EMBREE [Wald et al. 2014] to check for vertex-triangle collisions.

Simulation stages. Equipped with this simulation model, we run our simulation in three phases, illustrated in Figure 4:

1. *Relaxation* (1 second). In this phase, the plastic sheet is fixed on its border and the only external force is gravity.



(a) texture pattern (b) formed calibration shape (c) 3D reconstruction (d) reconstructed 2D texture (e) aligned & color corrected (a) (f) displacements between (d) and (e)

Figure 5: *Calibration process. We print a specifically designed texture pattern (a) on a plastic sheet and perform thermoforming with a calibration shape (b). The result is 3D scanned (c) and a 2D texture is computed (d). After alignment and color correction of the texture pattern (e) we estimate the displacements in the material (f). The inset in (f) illustrates the color coding of the displacement vectors.*

2. *Raising the mold* (about 1 second). The mold is raised with a speed of $0.1\,{}^{\mathrm{m}}/_{\mathrm{s}}$. During this phase, the mold touches the sheet, raising and deforming it.
3. *Vacuum* (until convergence). The vacuum forces are activated and the sheet is pulled toward the mold.

The simulation is interrupted when all vertices touch either the mold or the base plate or when the simulation time exceeds 5 seconds.

3.3 Calibration

Different types of plastic and thermoforming hardware setups produce different results, requiring different simulation parameters to accurately model their behavior. Classically, these parameters are computed from the specifications of the thermoforming hardware (temperature, vacuum pressure, speed of the moving platform, etc.) and the material used, and are acquired via physical material tests. To avoid this difficult and error prone procedure and enable the usage of various hardware setups, we propose an automatic procedure that relies only on fabricating and 3D scanning a single calibration object.

Calibration object and texture pattern. Our calibration object is a pyramid, chosen because it is simple to fabricate and scan (Figure 5). Other shapes could also be used, the only requirement being that they should be easy to 3D scan after the thermoforming. As we measure the thermoforming deformations from a 3D reconstruction of the textured object, we designed an RGB texture pattern (Figure 5 (a)) that has dense features that are easy to automatically detect. The pattern is then transferred to the plastic sheet and thermoformed (Figure 5 (b)). We obtain this pattern by computing Gaussian noise of different resolutions in each of the RGB color channels. This results in an image with well distinguishable features at various frequencies distributed over the RGB color channels, allowing us to reliably detect deformations of different magnitudes. This pattern can be seen as an extension of the greyscale wavelet pattern of [Atcheson et al. 2008]; we attach a high-resolution copy in the supplemental material. Additionally, we print standard color checker charts [X-Rite 2016] on the pattern to enable color correction of the captured images to compensate for imperfections in the camera hardware and capture setup.

3D reconstruction. We capture about 100 high-resolution images of the thermoformed calibration object using a Canon 6D camera and feed them into an off-the-shelf multi-view reconstruction system [Agisoft 2016] to compute a 3D model of the object (Figure 5 (c)). To increase the quality and robustness of the deformation estimation under uneven lighting conditions, we also perform color correction on the texture using the color checkers embedded in the pattern.

Preprocessing. The reconstructed 3D model is then flattened onto the UV domain using the as-rigid-as-possible parameterization algorithm [Liu et al. 2008] which results in the 2D texture shown in (Figure 5 (d)). The input texture pattern (Figure 5 (a)) is roughly aligned to it using a homography transformation, which maps the 4 corners of the input pattern to the 4 corners of the parameterized model (manually selected). An aligned and color correction input pattern is shown in (Figure 5 (e)).

Deformation estimation. After preprocessing, we estimate the deformation by computing a dense displacement field between the reconstructed 2D texture and the input texture. We use a well-established optical flow method [Brox et al. 2004] for computing the displacement field, which can, thanks to our special pattern, robustly reconstruct the flow even for large and complex deformations. To speed up the computation, we run the optical flow solver on images of size 2K×2K pixels, which we found to be sufficient to obtain an accurate deformation estimation. Computing a single flow field at this resolution takes about 32 s on a single workstation (Intel Xeon E5-1680 v3, 64 GB RAM) using a CPU implementation. We use a fixed set of parameters for all results: smoothness weight $\alpha = 20$, gradient weight $\gamma = 5$ and pyramid steepness $\eta = 0.95$. Please refer to the original paper [Brox et al. 2004] for an explanation of these parameters. The final deformation is estimated from the optical flow field by applying the inverse mapping of the UV coordinates and the homography transformation used for the initial pattern alignment.

While it might be tempting to directly use this dense map to compute the deformed pattern to print, entirely sidestepping the need for the simulation and parameter fitting, this is only possible for simple geometries that can be easily scanned with high accuracy. Besides the additional production costs, it is also much more time-consuming to fabricate and scan the object with the calibration texture than to run our simulation, which only takes 5 minutes.

Parameters and grid search. Our simulation depends on the following parameters: Young's modulus (Y), creep (c_{creep}), yield strain (c_{yield}), viscosity (v), Poisson's ratio, dimensions and density of the sheet, vacuum pressure and elevation speed of the mold. Since our model is a simplified approximation of the true physical vacuum forming process, the parameters from material tables do not necessarily minimize the alignment error. Therefore, we optimize for them using material tables to define reasonable ranges.

We experimentally observed that only the first 4 parameters need to be optimized to obtain an accurate simulation, while the others can be copied from a material table (Poisson's ratio: 0.35, sheet density: $1330\,{}^{\mathrm{kg}}/_{\mathrm{m}^3}$), easily measured like the thickness (1 mm) and size ($24 \times 26\,\mathrm{cm}^2$) of the sheet and elevation speed ($0.1\,{}^{\mathrm{m}}/_{\mathrm{s}}$), or found in a specification sheet (vacuum pressure: 80 kPa).

We restrict these four parameters to lie in plausible ranges ($Y \in [5 \cdot 10^5, 5 \cdot 10^6], c_{\mathrm{creep}} \in [0, 1000], c_{\mathrm{yield}} \in [0, 0.1], v \in [0, 10^{-4}]$)

and we search in this restricted space using a grid-search approach. We sample 625 points (5 per dimension) and pick the ones with the lowest average error with respect to the ground truth. The error is measured as the average of the Euclidean distance between our simulation and the acquired ground truth. To account for registration errors, we optimize for a small translation (up to 2 mm) in the mold by uniformly sampling the space of translations and picking the best candidate. The parameters we found with this procedure for our hardware setup are $Y = 2.75 \cdot 10^6$, $c_{\text{creep}} = 500$, $c_{\text{yield}} = 0.1$ and $v = 2.5 \cdot 10^{-5}$.

3.4 Computational thermoforming

After introducing the hardware, the simulation and the calibration details, we now present our complete thermoforming pipeline, which converts a textured digital 3D model into a high-resolution plastic replica (Figure 1).

1. *Simulation.* The 3D model is used as the mold in the simulation (Section 3.2), which is run using the parameters obtained from the automatic calibration procedure (Section 3.3). The simulation starts with a flat triangle mesh model $M = \{V, F\}$ of the plastic sheet and produces a new set of vertex positions V' that correspond to the sheet after thermoforming.
2. *Projection.* After simulation, the input 3D model is projected onto the simulated mesh of the plastic sheet $M' = \{V', F\}$ using ray casting. For each vertex of the 3D model, we compute its barycentric coordinates in M' and then use the same coordinates to find the corresponding point in M. The image to print on the plastic is obtained by rendering the now flattened 3D model, using these new locations in M.
3. *Mold creation.* The 3D model is subtracted (in the Boolean sense) from a box to create a negative model of the mold. The negative is 3D printed and used to fabricate the gypsum mold (Section 3.1).
4. *Texture transfer.* The image is printed and transferred onto a plastic sheet using thermic transfer paper (Section 3.1).
5. *Thermoforming.* The plastic sheet is thermoformed, producing the textured replica.

4 Results

We ran our simulation algorithm on a dual processor workstation (Xeon CPU E5-2650 v2, 64 GB RAM) and used PARDISO [Schenk et al. 2007; Schenk et al. 2008; Kuzmin et al. 2013] to solve the involved linear systems. We discretize the plastic sheet using a mesh with 10K triangles. The time needed for the simulation is mostly independent of the geometry used and is around 5 minutes. The offline calibration procedure takes about 40 hours, mostly not involving any user interaction: 1 hour to fabricate the calibration object, 30 minutes to take the photographs, 2 hours for 3D reconstruction, 1 minute for optical flow and 36 hours for the parameter grid search.

Quantitative evaluation. We thermoformed a hemisphere and the cat model and used our calibration pipeline to measure the deformation introduced by the thermoforming. We then compared it with the result of our simulation, using the parameters we previously estimated on our pyramid calibration object. We obtained an average displacement error of 1.5 mm and 1.6 mm, respectively, which is close to the best fitting error that we got on the pyramid for the calibration (1.3 mm). A visual comparison of the errors and their corresponding histograms are shown in Figure 6. The distribution of the errors shown in the third row suggests that there is some non-uniformity in the actual thermoforming process. A possible source could be the imperfection of the heating system, which we assumed to be uniform and did not include in our model.

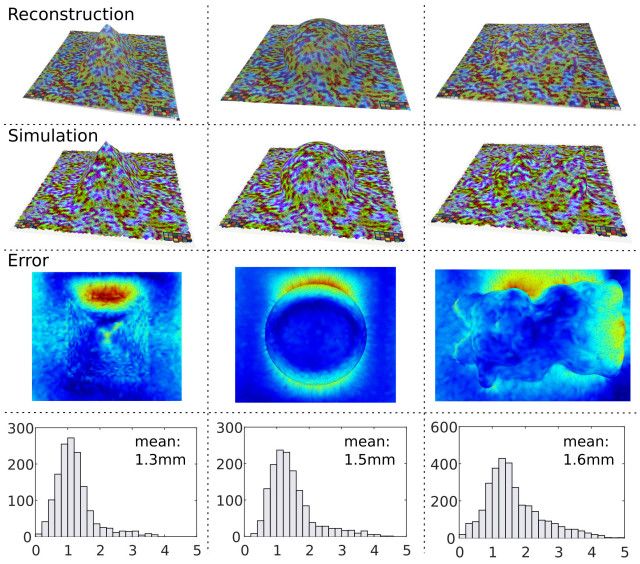


Figure 6: *Left, from top to bottom: the 3D reconstruction of the calibration shape, our simulation result, the visualization of the Euclidean distance error between the reconstruction and our simulation (dark blue: small error, dark red: higher error). The corresponding histogram in the bottom shows the distribution of alignment errors in millimeters. Note that this shape has been used for calibration and therefore has the smallest error. In the middle and on the left, we show our validation, computing the error on a hemisphere and on the cat shape using the same parameters. The error distributions in the histograms are very similar, suggesting that our simulation accurately reproduces the thermoforming process.*

Figure 7: *Our replica of the cat model (right) has a superior surface quality to that created with a ZCorp 650 powder printer [3dsystems 2016] (left) or hydrographic transfer [Panozzo et al. 2015] (middle).*

Comparisons. We show a comparison between our fabrication technique and two competing methods in Figure 7. Our result is not affected by the flat regions that cause artifacts in the hydrographics technique proposed in [Panozzo et al. 2015], and it has a superior resolution in respect to powder-based printing techniques.

Fabricated examples. We fabricated various models to test our method and potential applications (Figure 8). Mimicking the plastic food replicas commonly used by restaurants in Japan, we fabricated two loaves of bread of different sizes (Figure 9). Since the objects produced with our technique are lightweight and very robust, the

Figure 8: *An overview of the examples produced with our method.*

Figure 9: *Plastic food samples can be fabricated with our technique, avoiding hours of manual painting.*

Figure 10: *A replica of a miniature modeling stump. Note how the texture aligns with the model's geometric features.*

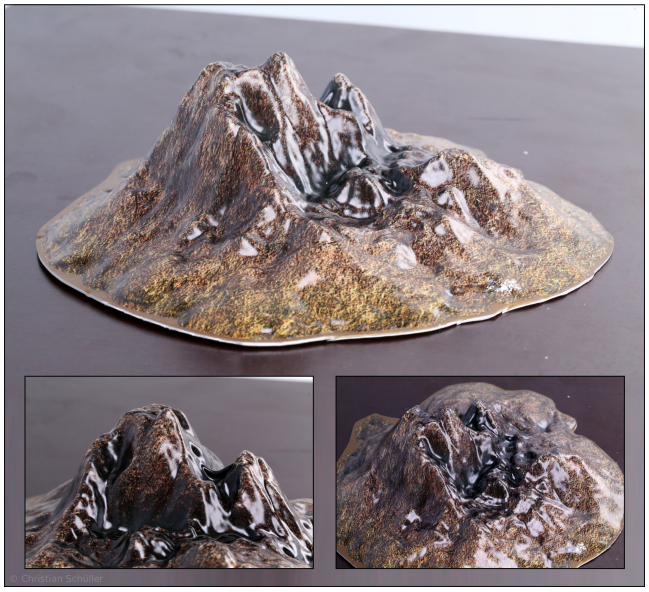


Figure 11: *A scaled replica of a mountain. The lightweight and robust material is well suited for application in model building.*

method is ideally suited to produce scenery pieces for model building, such as a mountain miniature (Figure 11) or a stump (Figure 10). By thermoforming transparent plastic, it is possible to obtain replicas of objects that contain transparent parts. In Figure 12, we fabricate the shell of a radio controlled car, leaving the windows transparent. This technique could be particularly useful for creating customized product packaging, as we demonstrate in Figure 13. Extremely detailed objects can also be fabricated with our technique, such as the Chinese mask in Figure 14. This object has many detailed features, which are accurately preserved in the physical replica.

Limitations and future work. The main limitation of our work lies in the registration between the mold and the printed plastic sheet before thermoforming. We currently use visual markers and perform the alignment by hand, which results in an alignment error of up to 2 mm. This could be avoided by using a customized thermoforming machine, but it would be an interesting challenge to tackle this problem using a lower cost approach that does not require special hardware. Currently, our solution only supports single-layered plastic sheets and cannot be used to produce closed objects. Depending on the complexity of the geometry, it might also be difficult to remove the gypsum mold.

An interesting avenue for future work would be the automatic design of decomposable molds, enabling thermoforming-based fabrication of objects with large concavities.

5 Concluding remarks

We proposed a new digital fabrication method to manufacture objects with a high resolution texture using thermoforming. Our solution relies on common hardware available in many digital fabrication labs and produces objects with a surface quality greatly superior to competing techniques. We believe computational thermoforming will have a significant impact in the fabrication community thanks to its low cost, low hardware requirements, high fabrication speed and quality, and that it has the potential to be a valuable tool for industries to quickly experiment with different thermoformed product designs.

Figure 12: *An RC car shell fabricated with our method using transparent plastic.*

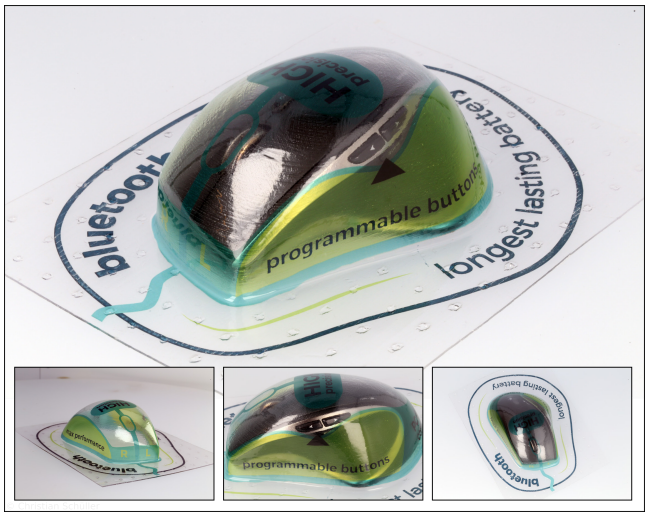


Figure 13: *A customized mouse packaging produced with our method.*

6 Acknowledgements

The authors thank the anonymous reviewers for their helpful comments and suggestions. We are grateful to Kaan Yücer, Vaclav Hnizda and Brian McWilliams for the help with the video and to Alessia Marra and Michael Rabinovich for the support with the 3D models. We would like to thank Bernhard Thomaszewski, Olga Diamanti, Peter Kaufmann and Wenzel Jakob for the insightful discussions and Mike Battersby of Fujifilm for his support and help with the plastic printing tests. Furthermore, we would like to thank Mirko Meboldt and the staff of the Raplab D-Arch ETHZ for the free access and help with the thermoforming machines and equipment. This work was supported in part by the ERC grant iModel (StG-2012-306877) and by a gift from Adobe Research.

Figure 14: *Our technique can faithfully replicate extremely detailed models.*

References

3DSYSTEMS, 2016. ProJet CJP 660Pro. http://www.3dsystems.com/3d-printers/professional/projet-660pro. Accessed: 2016-01-18.

ACCUFORM, 2016. T-sim. http://www.t-sim.com. Accessed: 2016-01-18.

AGISOFT, 2016. Agisoft photoscan standard edition 1.2.2. http://www.agisoft.com/. Accessed: 2015-12-18.

ATCHESON, B., IHRKE, I., HEIDRICH, W., TEVS, A., BRADLEY, D., MAGNOR, M. A., AND SEIDEL, H. 2008. Time-resolved 3D capture of non-stationary gas flows. *ACM Trans. Graph. 27*, 5.

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

BERMANO, A., BRÜSCHWEILER, P., GRUNDHÖFER, A., IWAI, D., BICKEL, B., AND GROSS, M. 2013. Augmenting physical avatars using projector-based illumination. *ACM Trans. Graph. 32*, 6, 189:1–189:10.

BROX, T., BRUHN, A., PAPENBERG, N., AND WEICKERT, J. 2004. High accuracy optical flow estimation based on a theory for warping. In *Computer Vision - ECCV 2004, 8th European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004. Proceedings, Part IV*, Springer, T. Pajdla and J. Matas, Eds., vol. 3024 of *Lecture Notes in Computer Science*, 25–36.

CUTTLEFISH, 2015. Cuttlefish: 3D printing pipeline. https://www.cuttlefish.de/. Accessed: 2015-03-01.

ESI, 2016. ESI Pam-form. https://www.esi-group.com/software-solutions/virtual-manufacturing/composites/solutions-plastics-trims. Accessed: 2016-01-18.

FUJIFILM, 2016. Thermoforming ink. http://www.fujifilm.com.au/powerofinkjet/applications/thermoforming. Accessed: 2016-04-19.

HERGEL, J., AND LEFEBVRE, S. 2014. Clean color: Improving multi-filament 3D prints. *Computer Graphics Forum 33*, 2.

HILL, R. 1998. *The Mathematical Theory of Plasticity*. Oxford classic texts in the physical sciences. Clarendon Press.

KLEIN, P. 2009. *Fundamentals of Plastics Thermoforming*. Synthesis lectures on materials engineeering. Morgan & Claypool.

KOUBA, K., BARTOS, O., AND VLACHOPOULOS, J. 1992. Computer simulation of thermoforming in complex shapes. *Polymer Engineering and Science 32*, 10, 699–704.

KOZIEY, B., GHAFUR, M., VLACHOPOULOS, J., AND MIRZA, F. 1997. Computer simulation of thermoforming. In *Composite Sheet Forming*, D. Bhattacharyya, Ed., vol. 11 of *Composite Materials Series*. Elsevier, ch. 3, 75 – 89.

KUZMIN, A., LUISIER, M., AND SCHENK, O. 2013. Fast methods for computing selected elements of the greens function in massively parallel nanoelectronic device simulations. In *Euro-Par 2013 Parallel Processing*, F. Wolf, B. Mohr, and D. Mey, Eds., vol. 8097 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 533–544.

LINCOLN, P., WELCH, G., NASHEL, A., ILIE, A., STATE, A., AND FUCHS, H. 2009. Animatronic shader lamps avatars. In *Proc. ISMAR*, 27–33.

LINDEMEIER, T., PIRK, S., AND DEUSSEN, O. 2013. Image stylization with a painting machine using semantic hints. *Computers & Graphics 37*, 5.

LIU, L., ZHANG, L., XU, Y., GOTSMAN, C., AND GORTLER, S. J. 2008. A local/global approach to mesh parameterization. In *Proceedings of the Symposium on Geometry Processing*, SGP '08, 1495–1504.

LTD., M. T., 2016. Mcor Technologies Ltd. http://mcortechnologies.com/. Accessed: 2016-01-18.

MIEGHEM, B. V., DESPLENTERE, F., BAEL, A. V., AND IVENS, J. 2015. Improvements in thermoforming simulation by use of 3D digital image correlation. *Express Polymer Letters 9*, 2.

MÜLLER, M., AND GROSS, M. 2004. Interactive virtual materials. In *Proc. Graphics Interface*, 239–246.

NIED, H. F., TAYLOR, C. A., AND DELORENZI, H. G. 1990. Three-dimensional finite element simulation of thermoforming. *Polymer Engineering and Science 30*, 20, 1314–1322.

O'BRIEN, J. F., BARGTEIL, A. W., AND HODGINS, J. K. 2002. Graphical modeling and animation of ductile fracture. *ACM Trans. Graph. 21*, 3, 291–294.

PANOZZO, D., DIAMANTI, O., PARIS, S., TARINI, M., SORKINE, E., AND SORKINE-HORNUNG, O. 2015. Texture mapping real-world objects with hydrographics. *Comput. Graph. Forum (Proc. Symp. Geometry Processing) 34*, 5, 65–75.

PRÉVOST, R., JACOBSON, A., JAROSZ, W., AND SORKINE-HORNUNG, O. 2015. Large-scale painting of photographs by interactive optimization. *Computers & Graphics*. In press.

RASKAR, R., WELCH, G., LOW, K.-L., AND BANDYOPADHYAY, D. 2001. Shader lamps: Animating real objects with image-based illumination. In *Proc. EG Workshop on Rendering*.

REINER, T., CARR, N., MĚCH, R., ŠT'AVA, O., DACHSBACHER, C., AND MILLER, G. 2014. Dual-color mixing for fused deposition modeling printers. *Computer Graphics Forum 33*, 2.

RHEOWARE, 2016. Rheoware simulation. http://www.blowmolding-thermoforming-simulation.com. Accessed: 2016-01-18.

ROUILLER, O., BICKEL, B., KAUTZ, J., MATUSIK, W., AND ALEXA, M. 2013. 3D-printing spatially varying BRDFs. *IEEE Computer Graphics and Applications 33*, 6.

SCHENK, O., WCHTER, A., AND HAGEMANN, M. 2007. Matching-based preprocessing algorithms to the solution of saddle-point problems in large-scale nonconvex interior-point optimization. *Computational Optimization and Applications 36*, 2-3, 321–341.

SCHENK, O., BOLLHÖFER, M., AND RÖMER, R. A. 2008. On large-scale diagonalization techniques for the anderson model of localization. *SIAM Rev. 50*, 1 (Feb.), 91–112.

SHILKROT, R., MAES, P., PARADISO, J. A., AND ZORAN, A. 2015. Augmented airbrush for computer aided painting (CAP). *ACM Trans. Graph. 34*, 2.

SIMO, J. C., AND HUGHES, T. J. R. 1998. *Computational inelasticity*. Interdisciplinary applied mathematics. Springer.

THERMO3D, Q., 2016. Quadraxis Thermo3D. http://quadraxis.com/site/?page_id=45. Accessed: 2016-01-18.

TRANSFERPAPER, 2015. Themagictouch cpm 6.2 - hard surface transfer paper. http://www.themagictouch.com/cpm.html. Accessed: 2016-13-04.

VIDIMČE, K., WANG, S.-P., RAGAN-KELLEY, J., AND MATUSIK, W. 2013. OpenFab: A programmable pipeline for multi-material fabrication. *ACM Trans. Graph. 32*, 4.

WALD, I., WOOP, S., BENTHIN, C., JOHNSON, G. S., AND ERNST, M. 2014. Embree: A kernel framework for efficient CPU ray tracing. *ACM Trans. Graph. 33*, 4, 143:1–143:8.

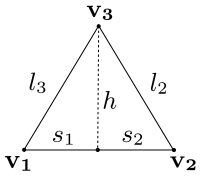
WITKIN, A., AND BARAFF, D., 1997. Physically based modeling: Principles and practice, siggraph course notes.

X-RITE, 2016. X-rite. http://xritephoto.com/. Accessed: 2016-01-18.

ZHANG, Y., YIN, C., ZHENG, C., AND ZHOU, K. 2015. Computational hydrographic printing. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2015) 34*, 4 (Aug.).

Appendix

Given the three edge lengths of an arbitrary triangle, we reconstruct its vertex positions up to a rigid transformation. W.l.o.g., we place



the vertices $\mathbf{v_1}$ and $\mathbf{v_2}$ at 2D coordinates $(0, 0)$ and $(l_1, 0)$, respectively. Using Pythagoras triangle theorem we get:

$$s_1^2 = l_3^2 - h^2, \;\; s_2^2 = l_2^2 - h^2, \;\; l_1 = s_1 + s_2.$$

Solving for s_1 we get $s_1 = (l_1^2 - l_2^2 + l_3^2)/(2l_1)$, and then

$$\mathbf{v_3} = \left(s_1, (l_3^2 - s_1^2)^{1/2}\right).$$