My work in geometry processing makes sense of existing geometric data and provides interfaces to put that data to further use.

Barriers stand between geometric data and the people who want to analyze and understand that data. Potential consumers and content creators cannot access or edit geometry because of poor human-computer interfaces. Meanwhile, some data never reaches its intended users because processing breaks down due to lack of robustness to noise.

**My long-term goal is to dismantle the barriers between humans and geometry.**

I attack from both fronts: I bring ideas from differential geometry and finite-element analysis to model geometric problems more intuitively and more robustly. Meanwhile, I pursue user interfaces to reduce human effort and increase creative or scientific exploration of geometric data. To elaborate on these recurring themes, allow me to share four parallel and complementary branches of investigation.

**Fabricating User Interfaces**

I am interested in solving problems that offer creative or scientific exploration to a human user. The advanced optimizations and careful methodologies common to geometry processing are in vain if the user’s direct interface to the computer is too cumbersome or finicky.

Advances in rapid fabrication will enable new user interfaces for direct manipulation of three-dimensional geometry.

The accelerating capabilities and diminishing costs of 3D printing provide an opportunity to shift physical human-computer interfaces from general purpose, mass produced devices such as the mouse and keyboard to devices customized for specific tasks and users.

To take advantage of this opportunity, I developed a physical input device to control virtual characters [1]. The user constructs a customized skeleton device matching the virtual character from reconfigurable 3D-printed joints with internal sensors (see Figure 1). The device replaces virtual widgets with a direct manipulation interface instantaneously mapping a user’s physical gestures to the virtual character’s limbs. To encourage further research, I released the designs freely as open source hardware.

Customized devices can also help connect people to physical geometry [2]. Currently, we are investigating computer-aided design of stencils for painting on arbitrarily shaped objects. Traditional paper stencils help artists and novices paint on flat surfaces, but flat stencils fail on curved surfaces with bumpy details. Our work will use 3D printing to generalize stencils for painting on any surface that can be scanned using modern computer vision techniques.

**Future directions: interfaces customized for each use** The costs of 3D printing continue to diminish. Meanwhile, availability, versatility and speed increase. To take advantage of this progress, I will customize interfaces not just for individual users, but even individual uses. I will develop devices whose designs respond to the very task they help perform.

The first application of this philosophy will be to design interfaces for modeling novel 3D shapes from scratch, complementing our device for animating existing shapes.

Possibilities also emerge for ubiquitous interfaces, where any ambient object transforms via augmentation into a listening input device. For example, a few sensors casually placed on an office swivel chair could instantly begin measuring rotations to control a virtual 3D scene. For domain-specific tasks, comfortable standard tools will become valid inputs to virtual worlds:
a surgeon’s knife or an engineer’s wrench. Formerly inert or inanimate objects will reveal new usefulness by joining the Internet of Things.

The current state of fabricated user interfaces is just the tip of the iceberg because so many underlying ingredients are left unsolved. The research of following sections—while sprouting their own motivations—feed directly into the realization of my vision for better user interfaces.

### Real-Time Shape Articulation

Originally motivated by animation for film and video games, the impact of real-time shape articulation reaches well beyond the entertainment industries [3], most recently fueling fabrication and virtual/augmented reality research. Over the past decade, we have experienced great improvements in highly accurate but slow shape deformation methods. Poor runtime is a barrier preventing exploration of a shape’s potential rearrangements.

Understanding and exploiting how humans expect a shape to deform could yield fast yet intuitive deformation algorithms with comfortable interfaces.

The shape articulation algorithms in my thesis [4] offer unprecedented performance and new interfaces. With a few high-level commands—*turn the head like this*, or *bend the fender like that*—a user’s changes immediately propagate across the shape in a plausible and intuitive way (see Figure 2). Prior to my work, an artist would need to edit one small detail of the shape and then the next and so on. I automate much of this labor without compromising quality or user experience. Users may control the entire shape via simplified interfaces at speeds 100x faster than the previous state-of-the-art methods.

At first, deforming a high-resolution shape seems to be a high-dimensional problem. This is discouraging for human users who naturally prefer a small set of parameters. However, the subspace of plausible deformations is actually quite small and manageable. For example, fingers move according to their limited joints and the relative motion of the hand and arm. For a given shape, my mathematical model identifies a small, intuitive subspace of deformations [5]. This process decouples deformation complexity from the shape’s representation. This also reduces computational complexity, so that deforming a high-resolution shape costs as much as deforming a low-resolution one (see Figure 3). The performance gains are measured in orders of magnitude [6], enabling for the first time high-resolution shape articulation for frame-rate-sensitive virtual/augmented reality applications.

By replacing the tedious manual labor required to set up a shape deformation system, I was able to develop new virtual user interfaces. In Figure 2, an internal skeleton opens the alligator’s jaw, an enclosing cage engorges its belly, and loose points bend its flexible tail. Prior to my work, extreme but useful deformations—such as the twisting of an arm (see Figure 4) or exaggerated stretching—produced degenerate geometry. By deconstructing the most popular model for real-time shape deformation, I identify how to handle such extreme deformations [7].

Shape articulation research reduces human effort for computer animation tasks, but it also achieves a broader impact as a subroutine for more complex tasks: interactive alignment of medical images, real-time simulation of elastic objects, and intelligent “fabrication-aware” modeling for 3D printing (see Figure 5). The mathematical machinery I develop is also useful in automated contexts where the user is replaced with a larger optimization or machine learning algorithm. Interactive pose detection in computer vision is improved by optimizing over the small set of plausible shape deformations rather than the full, unwieldy space.

I explored the application of my continuous theory for fast, reduced deformation to other forms of geometric data. Splines and subdivision surfaces are used by the graphic design and computer-assisted design communities. Previously, fast freeform deformation techniques were not available for this data. I demonstrate techniques to operate natively on this data: i.e. without the losses of converting the data to another format [8].
Advances in real-time shape deformation research places us at a unique vantage point to scope out the exciting future ahead. I discuss this in detail in an invited essay for the “Dissertation Impact” series of IEEE Computer Graphics & Applications [9].

Future directions: a real-time deformable world I embark to discover a compact representation for the deformation of any arbitrary input object. This is inherently challenging due to the endless variety of materials, shapes and interactions. My strategy is to first understand and reproduce the rules guiding a physical shape’s deformation and our human expectation of that behavior. Then I will exploit this knowledge to determine how and when the rules can be broken or simplified.

As a concrete next step, I consider the problem of real-time response to contacts. Real-time shape deformation is considerably harder if we prohibit shapes from overlapping themselves or each other. A poking finger should make an impression on a belly not slip inside it. A sliding chair should collide against a desk rather than skate through it.

My future research will uncover methods for handling contacts and collisions in real time. I will focus on two complementary scenarios.

Safe contact handling requires that at any moment surfaces strictly do not overlap. Existing solutions that guarantee safe handling are far from real time. Dynamic and adaptive updates to the real-time methods above could be the missing innovation. Safe contact handling during virtual design ensures that user expectations are met when objects are fabricated in the real world, where overlaps are physically impossible.

I also propose exchanging this strictness for a weaker constraint that we meet only the human perception of correct behavior. Often the strict geometric quantity—do two shapes overlap—is not an ideal translation of the original human goal, e.g. simulate a realistic coronary bypass surgery at haptic rates. What is required to induce perception of safe contacts while maintaining high frame rates? By answering this, users could freely explore the creative space un-inhibited by lag. I intend to conduct large-scale perceptual studies to guide this endeavor.

Higher-Order Partial Differential Equations

Partial differential equations (PDEs) elegantly model many geometry processing problems, but the community has largely focused on simpler, lower-order equations. PDEs involving higher-order derivatives are more challenging to control, but by definition more powerful. The most enticing reward is their promise of smoother solutions, which are desired in applications such as: modeling the smooth chassis of a car, smoothly interpolating scattered weather data, or filling holes in scans of cultural heritage artifacts. However, technical barriers prevent the widespread use of higher-order PDEs: they are more difficult to solve numerically, and it is difficult to make their boundary conditions intuitive to users.

Simplifying the use of higher-order partial differential equations could unlock new human understanding and exploration of geometric data.

My past and current research approaches this problem by first considering continuous equations assuming smooth geometry. A major challenge is to provide simple mathematical models that (1) directly plug into existing discretizations and (2) still achieve smooth convergence and retain the expressiveness of the original formulation.

As a first step, I proved the convergence of existing state-of-the-art methods that employ higher-order PDEs for surface modeling [10]. Previous methods allowed for control of position along the surface boundary, but not control of tangents or curvature which are desired for more expressive edits. By formally deriving a formulation based on mixed finite elements, we enabled intuitive control of tangent and curvatures at the boundary. This allows, for example, precise user control over creases in a surface (see Figure 6).
A theme throughout my research is determining how to translate human desires from natural language into mathematical constraints. For example, we consider the problem of inflating a flat cartoon into a smooth 3D surface, so that modern graphics engines can add beautiful effects [11]. Using a sketching interface, a user can comfortably indicate the correct depth ordering of components and sharp boundaries between objects. Hidden from the user, we interpret these sketches as inequality and tangent constraints on the PDE used to generate the 3D inflated surface (see Figure 7). Manipulating the underlying PDE exposes different mathematical quantities—some more translatable into user interfaces than others. Naturally, we prefer simpler interfaces. By enforcing this simplicity during design of the PDE, we may also simplify the algebraic structure, enhancing solver performance [12].

An important aspect is to treat the input shape as a mathematically generic domain. Specific instances may then consider a patch of the plane, a solid in space, or a curved surface, but the underlying machinery remains. For example, treating an image as a surface in 5D we developed a method to align images despite large changes in viewpoint and pose. Users edit one image and see their changes applied immediately to many other images [13]. This collaboration with Disney Research resulted in a US patent application.

The real-time shape deformation methods of the previous section employ higher-order PDEs to propagate a user’s edits smoothly across the shape. Constraints imposed on the PDE prevent unnatural responses of previous methods: if the user gestures to the right, no part of the shape moves left. This work was recently reprinted as a “Research Highlight” in Communications of the ACM [14].

Other problems require more intricate constraints. For example, typical data smoothing will blur away salient structures while simultaneously creating new, spurious ones. My formulation guarantees the preservation of existing structure and prevent filtering artifacts [15]. In Figure 8, domain experts get a clearer understanding of Hurricane Isabel’s structure after controlled filtering [16]. This paradigm of structure-preserving constraints has impact beyond scientific data visualization to de-noising for medical imaging (see Figure 9), image segmentation for computer vision, and smooth color interpolation for graphic design.

**Future directions: human-PDE interaction** People are generally good at reasoning about values: heights, positions, temperatures. With practice, we learn to reason about first derivatives, too: inclines, velocities, etc. But with each increase in the order of derivative, we lose intuition and slip into abstraction. My previous work shows that the power of higher-order PDEs is ultimately governed by higher-order boundary conditions and constraints set by a human user. I intend to investigate the interaction between human users and the PDEs the control, using the field of human-computer interaction as a model. I will both observe the ways human users interact with PDEs and also design novel modalities of interaction.

Consider a seemingly innocuous user interaction with a PDE: assigning a value at an isolated point. For example, specifying colors at few points in an image and solving a PDE to interpolate them. A common solution of posing this problem as a low-order Poisson equation turns out to be mathematically invalid. Solutions are not unique and all are discontinuous, contrary to the user’s expectation. Preliminary experiments indicate better behavior for certain higher-order PDEs. As a next step toward better human-PDE interaction, I will categorize problems where point constraints are well-posed. For important problems where they are not, I will search for satisfactory alternatives with comfortable user control.

**Robust Geometry Processing**

Numerical methods for geometry processing hinge on the ability to represent a shape discretely. For 3D geometric data, there exists a wide disparity between shape descriptions that are good enough for visualization and those that admit computation (see Figure 9). Poor robustness to arbitrary input geometry presents a barrier preventing adoption of powerful geometry processing techniques.
By plugging the robustness leaks in the geometry processing pipeline, users could unleash advanced methods on their "messy" geometric data.

My approach is to adapt traditional methods to work even in the presence of messy data. For example, a clean surface induces a function called the winding number at every point in space. This number is either zero if the point is outside the surface or one if inside. The winding number counts how many times a surface winds around a given point. Most unprocessed surface descriptions contain self-overlaps, open holes, thin sheets and disconnected floating parts, which invalidate conventional smooth geometry theory. Instead, I rearrange the winding number as a surface integral that is robust to these problems. The resulting generalized winding number varies smoothly corresponding to how much a point is inside the input surface [17]. Prior to this work, volumetric computation on many shapes was impossible due to the inability to determine inside from outside (see Figure 10).

Surface descriptions often contain self-overlaps, but I exploit the fact that these overlaps often occur in a structured way. For example, two legs may accidentally overlap during animation. Our algorithm untangles self-overlapping geometry in order to create a well-posed representation of its inner volume [18]. My research allows users to continue to process their geometric data despite artifacts arising from previous processing.

Assuming clean geometry often makes sense at first so that research can flourish to maturity. Eventually practical applications will demand robustness. For example, research in multigrid solvers for PDEs classically assume regular if not rectangular domains. While multigrid achieves optimal performance, prior methods break down on high-resolution geometry with high-frequency details. Instead, we leverage the powerful contact-handling machinery from physically based simulation to construct a specialized multiresolution hierarchy of an arbitrary surface. Each coarser surface strictly contains or nests the finer surfaces, allowing pure linear interpolation between layers rather than relying on error-prone extrapolation [19] (see Figure 11). Beyond improving multigrid solvers, these nested surfaces improve domain reduction for physical simulation or real-time shape deformation.

Future directions: big dirty data Today geometric data is coming in from more and more sources, but it is less and less reasonable to expect that this data is pristine.

It is not sustainable to continue to restrict each step in the geometry processing pipeline to two-pass methods that pre-process and then operate on the cleaned up data. I intend to work on geometric processing for big dirty data.

To this end, I am currently developing an algorithm for efficient and exact boolean operations (a.k.a. constructive solid geometry). A robust implementation to this well known problem has remained surprisingly elusive. Given this troubled history, it will be interesting to examine solutions as an application of formal verification. This would also lead to reasoning about robustness for ill-defined inputs in general, a fundamental barrier permeating the entire geometry processing pipeline.

Commitments

I am dedicated to the scientific reproducibility of experiments and results. I release open source software for all my research endeavors, culminating in two popular libraries that I continue to maintain: the geometry processing toolbox for MATLAB, gptoolbox; and the C++ library implementing over 100 ACM SIGGRAPH and Symposium on Geometry Processing (SGP) papers, libigl. Libigl won the 2015 SGP Software Award and is in use by many academic and industry institutions worldwide.

My research is committed to bringing people to a better geometric understanding of problems relevant to their lives. My past and current research not only provides initial positive answers, but also establishes an exciting position to scope out the most promising future directions.
References


