

Zoomorphic Design

Noah Duncan¹ * Lap-Fai Yu² Sai-Kit Yeung³ Demetri Terzopoulos¹

¹University of California Los Angeles

²University of Massachusetts Boston

³Singapore University of Technology and Design



(a) Input



(b) Result

Figure 1: A zoomorphic playground created by our approach.

Abstract

Zoomorphic shapes are man-made shapes that possess the form or appearance of an animal. They have desirable aesthetic properties, but are difficult to create using conventional modeling tools. We present a method for creating zoomorphic shapes by merging a man-made shape and an animal shape. To identify a pair of shapes that are suitable for merging, we use an efficient graph kernel based technique. We formulate the merging process as a continuous optimization problem where the two shapes are deformed jointly to minimize an energy function combining several design factors. The modeler can adjust the weighting between these factors to attain high-level control over the final shape produced. A novel technique ensures that the zoomorphic shape does not violate the design restrictions of the man-made shape. We demonstrate the versatility and effectiveness of our approach by generating a wide variety of zoomorphic shapes.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Geometric Algorithms

Keywords: zoomorphic design, shape creation

* Part of the work was done when Noah was visiting SUTD.

ACM Reference Format

Duncan, N., Yu, L., Yeung, S., Terzopoulos, D. 2015. Zoomorphic Design. *ACM Trans. Graph.* 34, 4, Article 95 (August 2015), 13 pages. DOI = 10.1145/2766902 <http://doi.acm.org/10.1145/2766902>.

Copyright Notice

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SIGGRAPH '15 Technical Paper, August 09 – 13, 2015, Los Angeles, CA.
Copyright 2015 ACM 978-1-4503-3331-3/15/08 ... \$15.00.
DOI: <http://doi.acm.org/10.1145/2766902>

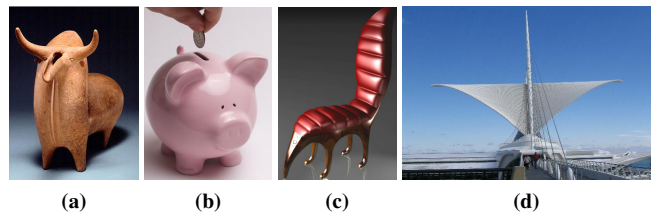


Figure 2: Historic and modern zoomorphic shapes. (a) Bull-shaped vessel circa 1000 BC symbolizing fertility. (b) Piggy bank. (c) Anteater chair by artist Maximo Riera. (d) The Milwaukee Art Museum, designed in the shape of a bird in flight.

“In all things of nature there is something of the marvelous.”

— Aristotle

1 Introduction

For centuries, humanity has attempted to capture the marvels of nature in man-made objects. Such objects range from ancient pottery vessels, to modern day piggy banks, designer chairs, and even buildings (Fig. 2). Man-made shapes that have the form or appearance of an animal are called *zoomorphic*. Since the beginning of recorded history, artists have created zoomorphic shapes by “applying animalistic-inspired qualities to non-animal related objects” [Coates et al. 2009].

Zoomorphic concepts are present in architecture [Aldersey-Williams 2003], furniture [Coates et al. 2009], and product design [Bramston 2008; Lidwell and Manacsa 2011]. Research suggests that children have a natural affinity for animals, which may explain the frequent presence of zoomorphism in children’s toys [Lidwell 2014]. Fig. 1 illustrates how zoomorphic design can create a more appealing children’s playground.

We propose a novel computational approach to tackle the unique challenges involved in creating zoomorphic shapes. Some zoomor-

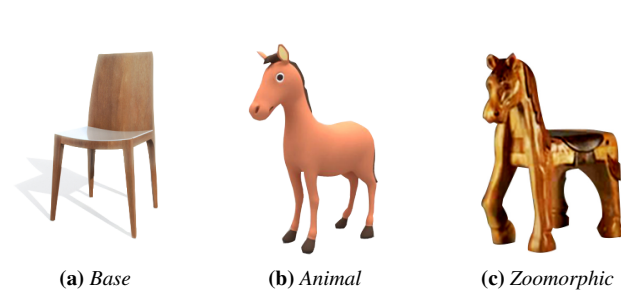


Figure 3: The three shapes in our approach. Note: The objects shown were not created by our approach.

phic designs mimic animals at only an abstract level, such as the design of the Milwaukee Art Museum which is inspired by the shape of a bird in flight (Fig. 2(d)), while others include components that directly mimic the shapes of animal parts. Our approach focuses on the latter category.

Designing a zoomorphic shape entails high-level tradeoffs, such as compromising between faithfulness to the animal form and retaining usefulness. For example, the chair in Fig. 2(c) has proportions similar to an anteater, but the sitting area is narrow and the “snout” may interfere with a sitter’s legs. High-level design goals correspond to low-level geometric operations, such as deformations of the animal-inspired and man-made components in the shape. The low-level operations can be tedious to execute manually. Therefore, our goal is to enable the user to direct the high-level design, while automating the low-level operations. We also hope to inspire the user by suggesting unusual yet viable designs that may not have been considered, such as the pink horse chair in Fig. 1.

Our approach takes two surface meshes as input, which we call the “**base shape**” and the “**animal shape**”. The base shape represents the portions of the zoomorphic shape that are not animal-related. The base shape is generally man-made and represents the ‘functional category’ of the shape we want to create. The animal shape represents the portions of the zoomorphic shape that are animal-related. For the zoomorphic shape in Fig. 3, the base shape is an ordinary chair and the animal shape is a horse. Our approach constructs a zoomorphic shape by merging the two input shapes.

2 Related Work

To our knowledge, no prior work in computer graphics has proposed or developed a computational approach to designing zoomorphic shapes. However, our work is related to existing research on 3D shape modeling, optimization and analysis, mesh composition, and computational design.

3D Shape Modeling. Many methods have been developed to automatically or semi-automatically create novel 3D shapes. Igarashi et al. [1999] introduce a sketching interface for designing 3D freeform models. Schmidt et al. [2010] and Takayama et al. [2011] developed interactive tools for transferring geometry and surface details between models. Funkhouser et al. [2004], Sheffer et al. [2007] and Jain et al. [2012] introduce approaches for generating novel 3D models by combining the components of existing models. Chaudhuri et al. [2010; 2011] develop tools to automatically suggest components that could be attached to an existing model based on the model’s shape or semantic attributes. Our work is similar in that we suggest animal shapes to be added to base shapes. However, their approach does not optimize for design factors in the final shape or take measures to ensure that the design restrictions of the original shape are satisfied. Kalogerakis et al. [2012] introduce a probabilis-

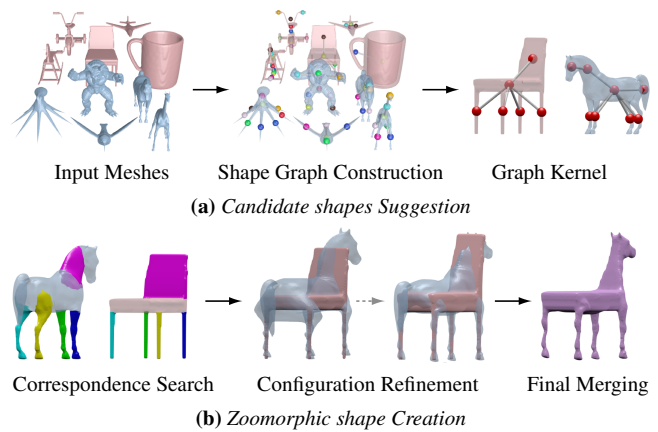


Figure 4: Overview of our approach.

tic model for synthesizing plausible man-made shapes in a category by combining components of existing shapes in the same category. Since we combine shapes from different categories, our criteria for plausibility differ.

3D Shape Optimization. Several works alter the geometry of an existing shape in order to optimize for certain criteria, such as stackability [Li et al. 2012], stability [Prévost et al. 2013], spinnability [Bächer et al. 2014], and aerodynamic characteristics [Umetani et al. 2014]. Zheng et al. [2014] optimize man-made shapes to fit a given humanoid figure better. We optimize two shapes jointly for how well they can be combined to create a zoomorphic shape.

3D Shape Analysis. We focus on a few highly related works in the rich body of literature on 3D shape analysis. Laga et al. [2013] find semantic correspondences between shapes and identify functional regions on a shape by using a graph representation of the shape and a graph kernel score to identify shape regions with similar contexts. We use similar technical ingredients for a different purpose. Our graph kernel score quickly identifies shapes that are likely to result in good optimization. Zhang et al. [2008] conduct a search for a partial correspondence between two shapes, using a deformation energy associated with each correspondence to identify the best correspondence. In our correspondence search, the ultimate goal is not to identify a correspondence, but to perform a coarse exploration of a highly complex energy landscape as the first stage of our optimization process. Shapira et al. [2010] detect analogous parts between objects that may belong to different categories using a hierarchical segmentation based on the shape-diameter function. Our correspondence search may be regarded as a different way of finding analogous parts, with considerations particular to our problem. Our work contributes to structure-aware shape processing [Mitra et al. 2013] by introducing a general approach for ensuring that the addition of new geometry to an object does not violate the object’s design restrictions.

Computational Design. Our work is a novel instance of a recent stream of research that addresses highly open-ended design problems by computer that have traditionally been the domain of artists and designers. Such problems range from computational interior design [Yu et al. 2011; Merrell et al. 2011], to the design of clothes [Umetani et al. 2011], accessories [Igarashi et al. 2012], puzzles [Zhou et al. 2014], mechanical toys [Zhu et al. 2012], and even cities [Aliaga et al. 2008; Vanegas et al. 2012]. The present paper tackles the new problem of computationally designing zoomorphic shapes.

3 Overview

Fig. 4 shows the main components and workflow of our approach to zoomorphic design. The first step is deciding what shapes to use for the base shape and animal shape (Fig. 4(a)). Our method efficiently identifies desirable pairings of a base shape and animal shape from a database, using a graph-kernel based method. It then merges the two shapes by deforming, repositioning, and removing unwanted geometry from them (Fig. 4(b)). We formulate this process as an optimization of several important design factors that include the prominence of the visually-salient regions of the animal shape, the degree of distortion of the base shapes and animal shapes, and the smoothness of the transition between the base shape and animal shapes. We enable the user to adjust the weighting given to each factor, which provides high-level control over the resulting design. The process is guided by a novel technique, called the *Volumetric Design Restriction* (VDR), which ensures that the design restrictions of the base shape are satisfied in the zoomorphic design.

4 Preprocessing

Input Data. The input shapes take the form of triangular meshes. The input shapes are divided into two categories, the animal shapes and the base shapes. We assume that the input shapes are oriented upright and require a segmentation of all shapes into semantically meaningful parts. For shapes that have several similar shapes in the database that have already been segmented, the segmentation can be transferred automatically using the method of [Kalogerakis et al. 2010]. Nothing in our approach precludes a hierarchical segmentation, but we used a simple segmentation to produce all the results shown in this paper.

Annotation. Each base shape is annotated with the volumetric design restriction labels. These labels are used to describe design restrictions of the base shape which should be preserved in the zoomorphic shape (Section 6.1). This is done either manually or automatically using the method described in [Kalogerakis et al. 2010]. Each animal shape is annotated with the visual salience labels, which need not be precise, so the annotation can be done quickly. These labels are used to identify regions of the animal shape which should be visible in the zoomorphic shape (Section 6.3). The manual annotations need only be obtained once per input shape rather than once per synthesis operation, so they are not overly time consuming.

5 Candidate Shapes Suggestion

Given a database containing all the input shapes, pairs of base shapes and animal shapes with high similarity scores are suggested as input candidates to create zoomorphic shapes (Section 7). Our method performs this operation automatically using a graph kernel technique in which the input shapes are represented as graphs.

5.1 Shape Graphs and Graph Kernels

For each shape, which has already been segmented as described in Section 4, we construct a shape graph to capture the structural relationship between its segments. The similarity between different shapes can then be efficiently computed by comparing their respective shape graphs with a graph kernel.

The shape graph is constructed as follows: Each segment corresponds to a node. Each adjacency between segments corresponds to an edge. Each segment has some geometric attributes, which

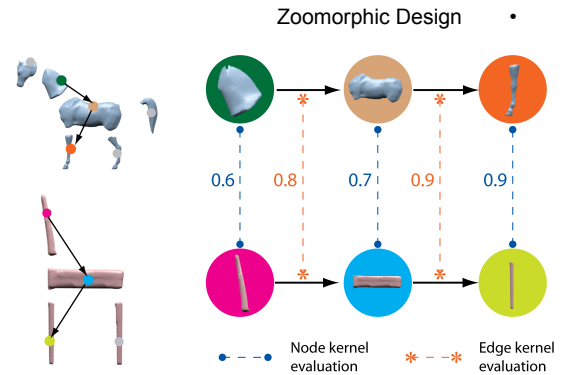


Figure 5: Example of a p -walk ($p = 2$) over the shape graphs of a horse and a chair. The node kernel and edge kernel evaluate the similarity between the corresponding nodes and edges. A larger number indicates a higher similarity.

characterize the segment, such as the part scale or centrality. The attributes of a segment are stored in its corresponding node.¹

Graph kernels are a general tool for measuring the similarity between two graphs [Kashima et al. 2004; Shawe-Taylor and Cristianini 2004]. We use graph walk kernels to compare the similarity between every base shape and animal shape pair, as illustrated by the example in Fig. 5. A graph walk kernel evaluates the similarity of all pairs of p -walks on the shape graphs ($p = 3$ in our experiments), where a p -walk traverses $p + 1$ nodes and p edges. To evaluate the similarity between two walks, we employ a node kernel and an edge kernel to compute the similarity between the corresponding nodes and edges of the walks. Specifically, a node kernel takes two nodes as input and computes a similarity score using the attributes stored at the nodes. Analogously, an edge kernel takes two edges as input and computes a similarity score. A graph walk kernel can be evaluated efficiently by dynamic programming [Shawe-Taylor and Cristianini 2004].

We note that the graph kernel is conservative—it identifies some, but not all of the pairings that will result in a desirable zoomorphic shape. In our approach, the animal shape may deform itself considerably, which the graph kernel does not consider. Therefore, the user is free to ignore the graph kernel’s suggestions and select pairings with low similarity scores. In our results showcase (Figure 17) each base shape shown had a high similarity score (within ten percent of the highest score) with its paired animal shape, with the exception of the go-kart.

6 Problem Formulation

Assuming that a base shape and animal shape pair has been selected, let us denote the base shape as \mathcal{M}_B and the animal shape as \mathcal{M}_A . We first describe two important concepts used throughout our method—the “volumetric design restriction” and the “configuration energy”.

6.1 Volumetric Design Restriction

The base shape, for example, a chair or a mug, usually possesses certain geometric features or structures that are crucial to its design and correspond to high level properties such as sittability or containability. For example, consider the face mug in Fig. 6(a). The naive addition of the face conflicts with the restriction that the mug base shape must hold liquid. For a more complicated example, consider the insect chair in Fig. 6(d). In what poses does the insect’s tail conflict with the design restriction that a person must be able to

¹ See the supplementary document for details of all the attributes used.

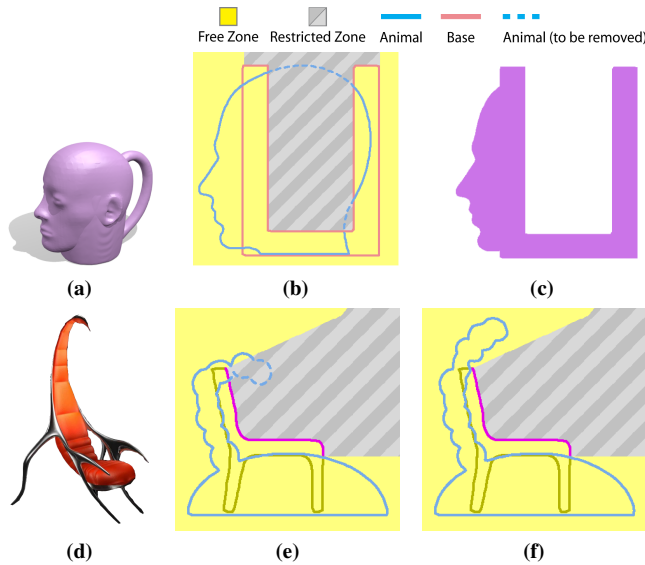


Figure 6: Examples of volumetric design restrictions. (a) A simple merge between the face and mug destroys the liquid-containing ability of the resulting zoomorphic shape. (b) Shapes under the volumetric design restriction. (c) The resulting zoomorphic shape, which is still a container. (d) What poses for the insect’s tail allow a person to sit on the chair? (e) The tail intrudes into a restricted zone and interferes with sitting. (f) A minor adjustment to the tail pose removes it from the zone and allows people to sit.

sit in the chair? The animal shape can merge with the base shape in a variety of ways, so it is important that the merge preserves these crucial properties on the base shape, which we call “design restrictions”. The problem of preserving certain qualities of a man-made object under geometric modification is uniquely challenging in our setting, because of the many ways the animal shape can interfere with these qualities.

The volumetric design restriction (VDR) is a novel concept, which uses a labeling of the base shape surface to specify the volume of space in which the presence of geometry from the animal shape will violate the design restrictions of the base shape (Fig. 6(b,e)). We call this volume of space the *restricted zone*, and the remaining volume the *free zone*. In creating a zoomorphic shape, geometry from the animal shape can lie in the *free zone*, but not in the *restricted zone*.

Labeling the base shape to specify the *restricted* and *free* zones has two advantages over specifying the zones directly. First, the labeling means that as the base shape deforms, the zones deform accordingly, which is necessary since our optimization procedure deforms the base shape. Note that the zones are related to the geometry of the base shape. For example, a container needs to preserve a zone for it to contain water. If the container widens, then this zone should also become wider. Second, once a labeling has been specified for several base shapes in a category, we can train a classifier to transfer the labeling to other shapes in that category [Kalogerakis et al. 2012]. By default, the labels are generated manually. To assist users in the manual labeling task, we developed a basic user interface that displays the changes in the zones interactively as the user modifies the labeling.²

Free and Restricted Zones. In our formulation, the label assigned to a face f on a surface mesh determines how the space around f is partitioned into *free* and *restricted* zones. We motivate our for-

²See the supplementary video for a demo.

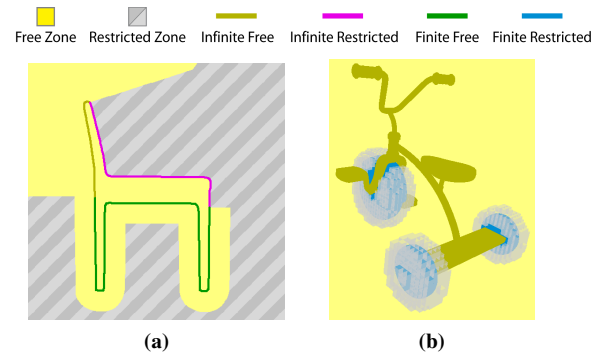


Figure 7: (a) Volumetric design restriction of a chair. Three segments (purple, olive, & green) have different outward zones corresponding to their label types (if all the free zones were filled with material, the resulting shape would still be a chair). (b) Volumetric design restriction of a tricycle. The wheel segments (in blue) are painted with labels of the *Finite Restricted* case, which protect them from being covered by any animal shape geometry (we illustrate only labels of the *Finite Restricted* case in this example).

mal definition of this partitioning by showing four different cases—*Infinite Free*, *Infinite Restricted*, *Finite Free*, *Finite Restricted*, that arise in our problem of how to preserve a design restriction when adding new geometry to an object. Each case corresponds to a different way of partitioning the space.

Infinite Free: Consider adding geometry to the back of the chair in Fig. 7(a). Here we do not place any restrictions on how much animal shape geometry can be added (assuming we ignore the mass of the added geometry). Therefore, *all* the space around the surface is *free*.

Infinite Restricted: Consider the need to preserve a sitting region on the chair. Here, we must preserve the flatness of the surface so that sitting is comfortable and preserve the empty space around the surface so that a human can occupy it. These requirements mean that *all* the space around the surface is *restricted*.

Finite Free: Consider adding geometry to the legs of the chair. In contrast to the *Infinite Restricted* case, we do not care about preserving the flatness of the leg surface. In fact, adding geometry from the animal shape would enhance the leg’s appearance. However, we still want the leg to be roughly cylindrical in shape; e.g., we do not want animal shape geometry that juts far out from the leg. In this case, space within a *finite* distance from the leg is *free* and space beyond that distance is *restricted*.

Finite Restricted: In Fig. 7(b), consider the need for the wheels on the tricycle to spin freely and to have circular symmetry. We cannot allow any contact of the wheel with the animal shape, but in contrast to the *Infinite Restricted* case, there is no need to preserve an empty space for a human to occupy. Here, the partition is the opposite of the *Finite Free* case—a finite space around the surface is *restricted*, anything further out is *free*.

Note that for all cases, the space inside the base shape is *free*, because if a space is already occupied by one shape it doesn’t matter if it is also occupied by another.

Let each face $f_i \in \mathcal{M}_{\text{base}}$ be assigned a label $l_i = (\alpha_i, \beta_i)$, where $\alpha_i \in \{1, -1\}$ denotes a zone type (*free* or *restricted*) and $\beta_i \in \mathbb{R}^+$ denotes a distance threshold from face f_i . Now consider an arbitrary point \mathbf{p} in the space. Let \hat{f} denote the closest face on $\mathcal{M}_{\text{base}}$ to \mathbf{p} . To determine the zone $r(\mathbf{p})$ assigned to \mathbf{p} , we compute the signed distance γ from $\mathcal{M}_{\text{base}}$ to \mathbf{p} . Suppose \hat{f} has label $\hat{l} =$

$(\hat{\alpha}, \hat{\beta})$. Then,

$$r(\mathbf{p}) = \begin{cases} 1 & \text{if } \gamma \leq 0, \\ \hat{\alpha} & \text{if } \gamma > 0 \text{ and } \gamma < \hat{\beta}, \\ -\hat{\alpha} & \text{if } \gamma > 0 \text{ and } \gamma \geq \hat{\beta}, \end{cases} \quad (1)$$

where $r(\mathbf{p}) = -1$ refers to the *restricted* zone and $r(\mathbf{p}) = 1$ refers to the *free* zone. The formula means that point \mathbf{p} belongs to the *free* zone if it is inside the shape; it belongs to zone type $\hat{\alpha}$ if it is outside the shape and within the distance threshold $\hat{\beta}$ from \hat{f} ; otherwise it belongs to the opposite zone type $-\hat{\alpha}$. Each of the four cases mentioned above corresponds to a labeling:

	$\beta < +\infty$	$\beta = +\infty$
$\alpha = -1$	Finite Restricted	Infinite Restricted
$\alpha = 1$	Finite Free	Infinite Free

Functionality. We briefly clarify the relationship of the VDR to functionality. We believe that the volumetric design restriction can preserve some types of functionality such as containability, graspability, or sittability. In general, it can preserve functionalities which are apparent from visually inspecting an object. In [Zheng et al. 2013], another work that deals with preserving functionality in man-made objects, this type of functionality was called “functional plausibility”. However, the VDR cannot deal with more complex functionalities like stability, structural strength or aerodynamics which in general, cannot be determined from visual inspection alone. Our user study (Section 10) showed that the VDR has a major impact on whether our approach generates zoomorphic shapes that are plausible examples of the category of man-made shape they were derived from. Since plausibility is highly related to functionality for man-made shapes this result supports our statement about functionality.

Examples. The volumetric design restriction can be applied to resolve the previously mentioned issues raised in the creation of zoomorphic shapes. Figure 6(c) shows the face mug example. The *restricted* zone removes the geometry of the face that prevents the cup from holding water. Figure 6(f) shows the insect chair example. The *restricted* zone signals the animal shape to alter its pose to preserve the chair’s sittability.

The VDR has the additional advantage that it can be naturally integrated into our optimization framework. We discuss this in Section 7.

6.2 Deformation Models and Configuration

Unique to our problem is that a zoomorphic shape is composed of a base shape and animal shape, which are generally an organic object and a man-made object. We allow both the animal shape and base shape to deform during the optimization process. The animal shape and base shape use different deformation models that are well-suited for organic and man-made objects, respectively.

Animal shape Deformation Model. In general, different animal shapes require different deformation models (Fig. 8 (a,b)). Currently our approach supports two models—Linear Blending Skinning (LBS) and Free-form Deformation (FFD), and it may be extended to support others as well. Animal shapes which use LBS are generally creatures with clearly defined limbs, such as squids or horses. In our model, the control parameters specify only the translations of the LBS handles. The remaining degrees of freedom are found by the method in [Jacobson et al. 2012]. Their method also allows us to specify only a subset of the translations and find the rest automatically. LBS requires that the input mesh comes with a skeleton and weights. These can be found manually or automatically with [Baran and Popović 2007] and [Tagliasacchi et al. 2012]. Animal shapes which use FFD are non-articulated shapes, such as

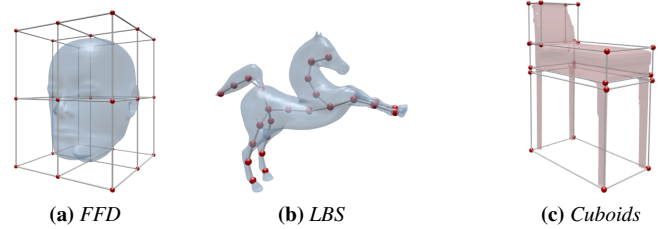


Figure 8: Different deformation models. (a), (b) Animal shape deformation models. FFD is used to deform a face (a) and LBS is used to deform a horse (b). (c) Base shape deformation model. Each cuboid encloses a group of segments that are constrained to share the same transformation.



Figure 9: Different configurations. Each configuration ϕ encodes how the animal shape \mathcal{M}_A and base shape \mathcal{M}_B deform and position themselves to create a zoomorphic shape.

faces, which are not well-described by a skeleton. We enclose the model in a FFD cubic lattice whose density can be specified by the user. Generally a 1×1 or 2×2 lattice offers enough control.

Base shape Deformation Model. We use a model in which each segment in the base shape \mathcal{M}_B can be transformed by a scaling and translation (Fig. 8(c)). The translations are constrained to preserve segment adjacencies. Groups of segments can be constrained to share the same transformation, which is useful for preserving symmetry and functionality, such as ensuring that the legs of a chair have equal length. Scales in different directions can be constrained to be equal, which is useful for ensuring that wheels remain circular. This deformation model is very simple, yet it provides a sufficient amount of freedom for a wide range of man-made objects.

Configuration. We define $\phi = (\phi_a, \phi_b)$, where ϕ_a and ϕ_b are the vectors of configuration parameters of the animal shape and base shape, respectively, to be the “configuration”, which encodes how the animal shape \mathcal{M}_A and base shape \mathcal{M}_B deform and position themselves to create a zoomorphic shape. The meaning of the parameters depend on the chosen deformation models. Fig. 9 shows example configurations for a horse and chair.

6.3 Configuration Energy

We define a configuration energy to measure the desirability of the zoomorphic shape resulting from a given configuration. A configuration that results in a desirable zoomorphic shape should have a low energy. We identify desirable configurations by minimizing the configuration energy:

$$E(\phi, \mathbf{w}) = w_{\text{df}}^a E_{\text{df}}^a(\phi_a) + w_{\text{df}}^b E_{\text{df}}^b(\phi_b) + w_r E_r(\phi) + w_{\text{vs}} E_{\text{vs}}(\phi) + w_g E_g(\phi), \quad (2)$$

where $\mathbf{w} = [w_{\text{df}}^a, w_{\text{df}}^b, w_r, w_{\text{vs}}, w_g]^T$ is a vector of weights. For fully automatic operation, setting all the weights to 1.0 generally produces a reasonable result. However, allowing the user to adjust these weights can lead to interesting changes in the designed zoomorphic shape (see Section 9). We discuss the individual energy terms next:

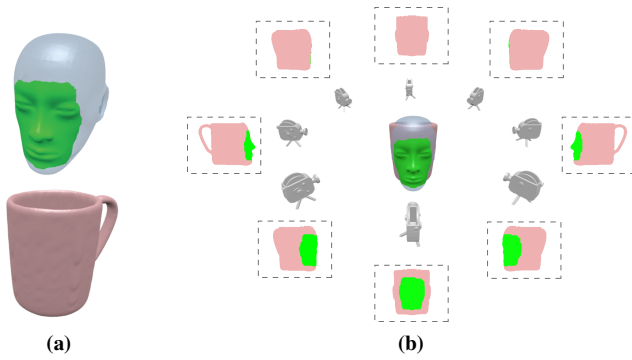


Figure 10: Computing the visual saliency. (a) Input mug and face with visual saliency annotations (green). (b) Images captured by 8 cameras looking at the shapes from different viewpoints.

Animal shape Deformation. We penalize deformation of the animal shape \mathcal{M}_A by defining

$$E_{\text{df}}^{\text{a}}(\phi_a) = \frac{D(\phi_a)}{D_m} + C(\phi_a), \quad (3)$$

where $D(\phi_a)$ is the mesh deformation energy defined in [Sorkine and Alexa 2007] and D_m is a normalization term found by taking the median of the energies encountered during the correspondence search (see Section 7.1). The term $C(\phi_a)$ returns $+\infty$ if the deformation is so high that it is invalid, and 0 otherwise. For animal shapes deformed using LBS, we define $C(\phi_a)$ in terms of the handle positions. Specifically, if any skeleton bone is stretched by more than a threshold, or if the angle between a pair of bones differs from the rest angle by more than a threshold, the deformation is invalid. For animal shapes deformed with FFD, $C(\phi_a) = 0$.

Base shape Deformation. This term penalizes non-uniform scaling of the base shape. We formulate $E_{\text{df}}^{\text{b}}(\phi_b)$ as the sum of squared differences of all pairs of segment scales of the base shape segments. Let $s_{i,u}$ be the segment scale of base shape segment i with respect to axis $u \in \{x, y, z\}$. Then,

$$E_{\text{df}}^{\text{b}}(\phi_b) = \frac{1}{K_{\text{df}}^{\text{b}}} \sum_{i < j} \sum_{u, v} (s_{i,u} - s_{j,v})^2, \quad (4)$$

where K_{df}^{b} is a normalization constant equal to the number of terms in the sum.

Registration. This term encourages the animal shape and base shape to align with each other. We compute the term over a set of uniformly sampled vertices \mathcal{V}_r from the animal shape \mathcal{M}_A . Given a vertex $\mathbf{v} \in \mathcal{V}_r$, define $\mathbf{n}_{\mathbf{v}}$ as the vertex normal and $d(\mathbf{v})$ as the distance function from \mathbf{v} to the base shape \mathcal{M}_B , and denoting the gradient of the distance from \mathbf{v} to the base shape \mathcal{M}_B as $(\nabla d)_{\mathbf{v}}$,

$$E_r(\phi) = 2 - \frac{1}{|\mathcal{V}_r|} \sum_{\mathbf{v}} \exp\left(-\frac{d(\mathbf{v})^2}{\sigma_v^2}\right) - \frac{1}{|\mathcal{V}_r|} \sum_{\mathbf{v}} \exp\left(-\frac{\arccos(\mathbf{n}_{\mathbf{v}} \cdot (\nabla d)_{\mathbf{v}})^2}{\sigma_n^2}\right), \quad (5)$$

where we set σ_v equal to 1/4 the diagonal of the bounding box enclosing both objects and $\sigma_n = \frac{\pi}{4}$.

Visual Saliency. This term encourages the appearance of visually salient regions from the animal shape in the zoomorphic shape. We assume the animal shape \mathcal{M}_A has had its visually salient regions

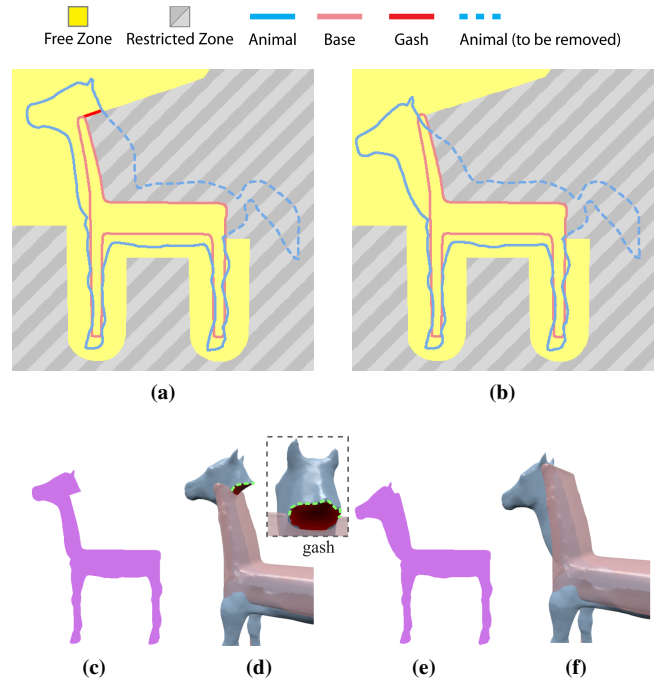


Figure 11: Considering gashes when creating a zoomorphic shape. (a) After removing the portion of the animal shape in the restricted zone, a gash (red) will appear. (b) Our optimizer bends the horse's head slightly down and raises the chair's back to avoid the formation of a gash. (c) Zoomorphic shape with a gash. (d), (e) Our optimizer removes the gash surface (red) by minimizing the length of the intersection (green) between the gash surface and the animal shape's surface. (f) Zoomorphic shape without a gash.

labeled. The labeling can be provided automatically with existing methods [Lee et al. 2005] or manually by the user. Manual labeling offers greater control and need not be very precise. We penalize configurations that occlude or remove the visually salient regions of \mathcal{M}_A . We evaluate the degree of occlusion and removal by rendering the base shape and animal shapes deformed by configuration ϕ across a set of camera views Ω and measuring the area-weighted proportion of salient faces visible in each view. We do not render the parts of \mathcal{M}_A that exist in *restricted* zones, as those parts of \mathcal{M}_A will be absent in the synthesized zoomorphic shape. Let \mathcal{V}_{vs} be the set of visually salient faces in the animal shape and let $\mathcal{V}_{\text{vs}}^{\omega} \subseteq \mathcal{V}_{\text{vs}}$ be the set of visually salient faces visible from camera view $\omega \in \Omega$. Our visual saliency term measures the proportion of the visually salient regions which are visible:

$$E_{\text{vs}}(\phi) = -\frac{1}{\mathcal{A}(\mathcal{V}_{\text{vs}}) |\Omega|} \sum_{\omega \in \Omega} \mathcal{A}(\mathcal{V}_{\text{vs}}^{\omega}), \quad (6)$$

where $\mathcal{A}()$ computes the total area of all the faces. Fig. 10 depicts how we place the cameras. We uniformly arrange eight cameras in a circular-disc manner on a horizontal plane level to the shapes, with the shapes situated at the center. The cameras are at the minimum distance from the shapes that allow them to see the entirety of the shapes. Each visually salient face is rendered in a different color to detect if it is visible. Thus, the cameras capture eight images which are used to evaluate $E_{\text{vs}}(\phi)$.

Gash. In creating the zoomorphic shape, any geometry from the animal shape that intrudes into the *restricted* zones needs to be removed. This removal will create “gashes” on the animal shape at the boundary between the *restricted* and *free* zones. On one side of

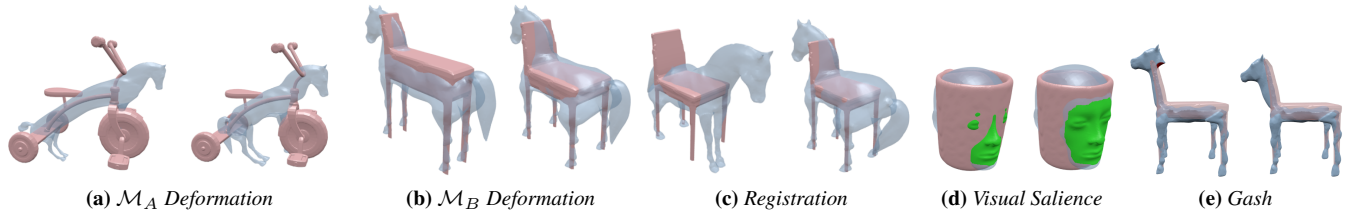


Figure 12: Effects of omitting an energy term. Each subfigure shows the result without (left) and with (right) an energy term. (a) Without the animal shape deformation term (left), the horse’s body is stretched out excessively to match the tricycle’s frame. (b) Without the base shape deformation term (left), the chair’s base is squeezed to match the horse’s body. (c) Without the registration term (left), the horse is not well-aligned with the chair. (d) Without the visual saliency term (left), the face’s details are not revealed in the merge. (e) Without the gash term (left), the resulting zoomorphic shape shows a gash.

the boundary, the animal shape is preserved, while on the other side it is removed. Fig. 11(a-b) shows a 2D example of a gash created on a horse when merging with a chair, and the resulting zoomorphic shapes in 3D. In general, these gashes are aesthetically undesirable. The issue is resolved if the gashes occur inside the base shape, because this conceals them from view. Therefore, our gash term penalizes only the visible gashes (Fig. 11(b)). We explicitly define when a gash is visible in our gash energy formulation. In 3D, a gash is a surface. This surface extends into the interior of the animal shape. To calculate the area of this surface, we would need a volumetric representation of the animal shape, which would be costly to work with. Instead, we identify the intersection of the gash surface with the animal shape’s surface, which will be a curve whose length is efficiently computable. We can identify these curves by examining the vertices of the animal shape \mathcal{M}_A . We search for pairs of adjacent vertices (\mathbf{m}, \mathbf{n}) , where \mathbf{m} is in a *free zone* and \mathbf{n} is in a *restricted zone*, and both vertices are outside the base shape. These pairs of vertices are at the locations where the gashes on \mathcal{M}_A occur in the final merge, and are not concealed by the base shape from view, ie: the gash is visible (if \mathbf{m} were inside the base shape then the gash would not be visible). For each such pair, we mark \mathbf{m} as a *gash vertex*. Let \mathcal{V}_g be the set of all gash vertices. Summing the lengths of all edges connecting gash vertices gives us the length of all intersections, which we use as the gash term:

$$E_g(\phi) = \frac{1}{K_g} \sum_{\substack{\mathbf{v}' \in \mathcal{N}(\mathbf{v}) \\ \mathbf{v}, \mathbf{v}' \in \mathcal{V}_g}} \|\mathbf{v} - \mathbf{v}'\|, \quad (7)$$

where $\mathcal{N}(\mathbf{v})$ is the set of vertices adjacent to \mathbf{v} , and K_g is a normalization constant equal to the largest geodesic distance between two vertices in \mathcal{M}_A .

Fig. 12 illustrates the negative effects of dropping any of the terms of the configuration energy function.

7 Zoomorphic Shape Creation

Given a base shape and animal shape pair, we need to search for a desirable way to arrange them before merging them to create a zoomorphic shape. We do this by minimizing the energy in Section 6.3 in a coarse-to-fine manner. In the coarse stage, we find a good correspondence between the two shapes. This coarsely aligns the animal shape with the base shape and gives us an initial configuration. In the fine stage, we refine the initial configuration with continuous optimization. This stage expands the degrees of freedom of the optimization to resolve situations that require subtle adjustments. Finally, unwanted geometry is removed from the two shapes and the resulting shapes are merged by a union operation in the volumetric space to create the zoomorphic shape.

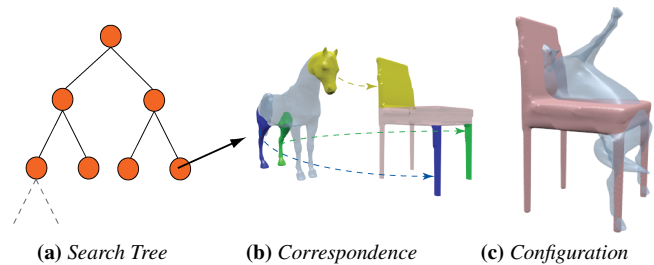


Figure 13: (a) Each node of the search tree refers to a correspondence. (b) A correspondence between a horse and a chair; corresponding segments have the same color. (c) The configuration induced by this correspondence has a high energy due to the large deformation.

7.1 Correspondence Search

A correspondence is a set of pairs of corresponding segments from the animal shape \mathcal{M}_A and base shape \mathcal{M}_B . We define a correspondence as $c = \{(a_i, b_i), i = 1, \dots, N\}$, where segment a_i in the animal shape \mathcal{M}_A has a corresponding segment b_i in the base shape \mathcal{M}_B , and N is the total number of pairs of corresponding segments.

The goal of the correspondence search is to coarsely explore the configuration energy landscape, by explicitly associating each correspondence with a configuration. Good correspondences are associated with low-energy configurations, which can be used as initialization points for the fine-scale energy minimization. We justify our search by the observation that many zoomorphic shapes have an “implicit correspondence”, where parts of the animal shape coincide spatially with corresponding parts of the base shape. Hence, explicitly searching for a good correspondence provides us with a reasonable strategy for globally exploring the configuration energy landscape. Note that the configurations at this stage need not be optimal. We merely need to identify a rough pose of \mathcal{M}_A that can be refined later (Section 7.2).

We use a combinatorial tree search similar to [Zhang et al. 2008]. As Fig. 13 illustrates, each node of the tree stores a correspondence between \mathcal{M}_A and \mathcal{M}_B . A node is expanded into its child nodes, where each child node has a correspondence c_{child} equal to its parent node’s correspondence c_{parent} plus a new pair of corresponding segments (a_k, b_k) . That is, $c_{\text{child}} = c_{\text{parent}} \cup \{(a_k, b_k)\}$.

For notational convenience, let us define $\phi(c)$ as the configuration associated with correspondence c . Then $\phi(c)$ should deform \mathcal{M}_A such that each segment a_i of \mathcal{M}_A is aligned with its corresponding segment b_i of \mathcal{M}_B . To find $\phi(c)$, we first apply local transfor-

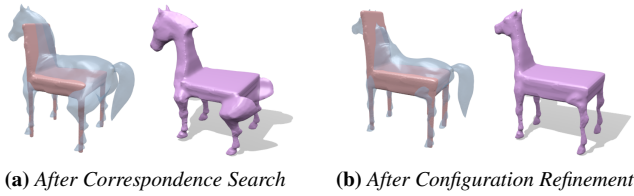


Figure 14: Configurations and zoomorphic shapes generated after correspondence search and after configuration refinement.

mations that align each animal shape segment a_i in c to its corresponding base shape segment b_i . Since in general, c is a partial correspondence, we still need to find a deformation for the animal shape segments that were not in c . We find this deformation by minimizing the ARAP energy [Sorkine and Alexa 2007] over the entire animal shape with the locally transformed segments constrained to be fixed. Note that the local transformations are completely determined by their segment pair a_i, b_i , so they can be computed in advance for each possible segment pair. In our implementation we use non-rigid ICP to compute the local transformations, but other methods may be used.

Our search starts from the root of the tree. It computes the configuration energy of each visited node. If a node stores a correspondence that is associated with an infinite energy (this situation occurs when the animal shape deformation is infeasible), we prune its subtree. Our search also only considers correspondences that satisfy bilateral symmetry.

The output of this stage is a set of correspondences and their associated configurations. We select the configuration $\hat{\phi}$ with the lowest energy and the associated correspondence \hat{c} for the next stage.

7.2 Configuration Refinement

In this step, we refine the rough configuration given from the correspondence search. We optimize the full set of control parameters ϕ_a and ϕ_b of both \mathcal{M}_A and \mathcal{M}_B using $\hat{\phi} = (\hat{\phi}_a, \hat{\phi}_b)$ as the initialization point. The registration term $E_r(\phi)$, gash term $E_g(\phi)$ and visual salience term $E_{vs}(\phi)$ are non-differentiable, and the full energy landscape usually has many local minima. Therefore, we minimize the configuration energy using Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [Lozano 2006]. We terminate once the configuration energy has not decreased by more than 10% in the last 25 iterations.

The modifications made to the configuration in this stage are small in absolute terms, but they usually improve the final zoomorphic shape’s appearance significantly. In particular, this stage has enough degrees of freedom to make the delicate adjustments necessary to lower the gash term $E_g(\phi)$, which has a large impact on the zoomorphic shape’s appearance. Fig. 14 shows the configuration and shape of the Horse Chair after correspondence search and after configuration refinement.

7.3 Removals and Merging

Once the input shapes are in their final configuration, we remove certain regions from both shapes and then merge them to form a zoomorphic shape. To do the removals, we first convert the shapes to their volumetric representations.

For the animal shape \mathcal{M}_A , we remove any of its geometry that lies in the restricted zones given by the VDR. The base shape removals

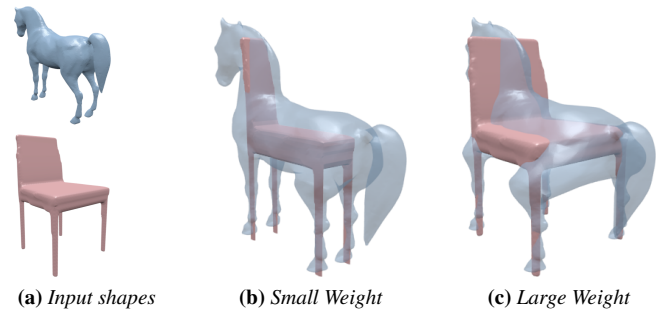


Figure 15: Effects of weight w_{af}^b in base shape control. (a) Input shapes. (b) A small weight causes the chair (base shape) to deform strongly to match with the horse. (c) A large weight causes the horse to deform strongly to match with the chair.

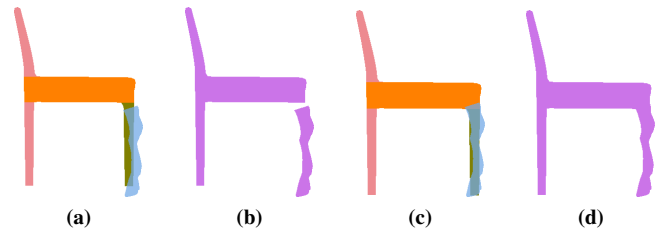


Figure 16: Explanation of the overlap criterion for removing base shape segments. The chair leg is adjacent to the chair base. (a) The horse leg does not overlap with the chair base. (b) If we remove the chair leg and generate the zoomorphic shape, the horse leg is disconnected. (c, d) When the horse leg does overlap with the chair base, the zoomorphic shape is contiguous.

are motivated by the observation, that in many cases, the appearance of the zoomorphic shape is improved if a segment of the base shape does not appear in the zoomorphic shape, because it has been “replaced” by a corresponding segment in the animal shape. For example, when designing the Horse Chair, we want the original chair legs to be removed, so that they can be completely replaced by the horse’s legs. Our formal procedure is to remove a segment b_i on \mathcal{M}_B if all the following conditions are met:

1. Removing b_i will not reveal a gash on \mathcal{M}_A .
2. There should be a segment a_i from \mathcal{M}_A that replaces the b_i to be removed (e.g., a horse’s leg replacing a chair’s leg). This is done by checking if b_i was assigned a corresponding segment a_i from \mathcal{M}_A in the correspondence search.
3. The corresponding segment a_i should also overlap with each of b_i ’s adjacent segments (e.g., a horse’s leg replacing a chair’s leg should also overlap with the chair’s base). We detect this overlap using the volumetric representation of the segments. Fig. 16 shows an example which motivates this requirement.

After performing removals, we take the union of the volumetric representations as the final output. To make the transition between the shapes appear more natural, we apply several iterations of implicit smoothing to the volumetric representation of the zoomorphic shape at the locations where the base shape and animal shape intersect. Finally, the volumetric representation is converted to a manifold mesh using a standard method such as Marching Cubes.

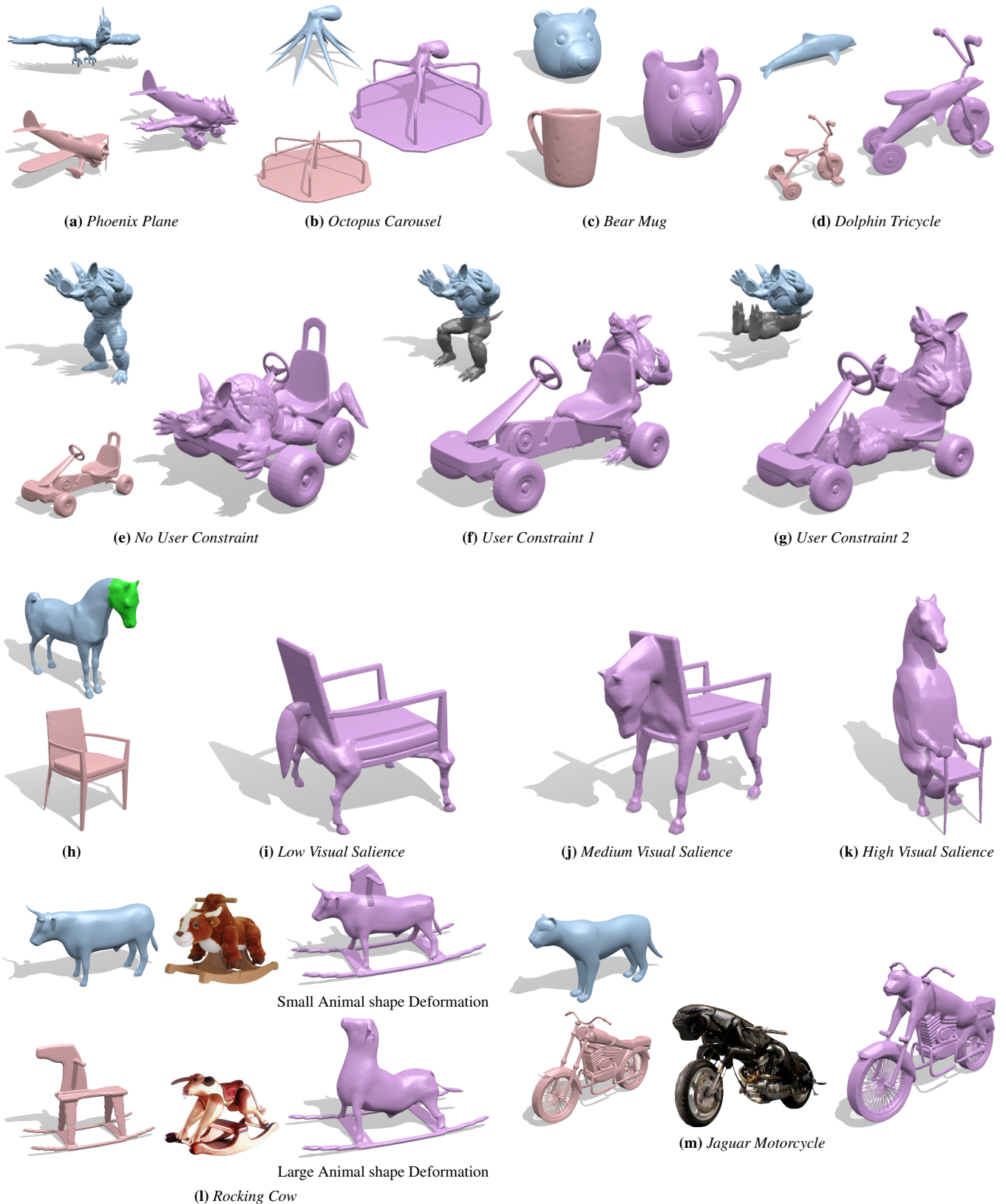


Figure 17: Zoomorphic designs created by our system. (a)–(d) Different zoomorphic shapes. (e)–(g) Merging an armadillo with a go-kart. (e) With no user constraint, the go-kart changes its proportions to accommodate the width of the armadillo’s body. (f) With the armadillo legs constrained (gray) to a sitting pose, the armadillo merges with the seat. (g) With the armadillo legs constrained to stretch out, the armadillo aligns its legs with the frame. (h)–(k) Changing the visual saliency weight leads to different designs for a Horse Armchair. Visual saliency annotations for the horse are shown in green. (l) Rocking Cow with small and large deformations. (m) Jaguar Motorcycle.

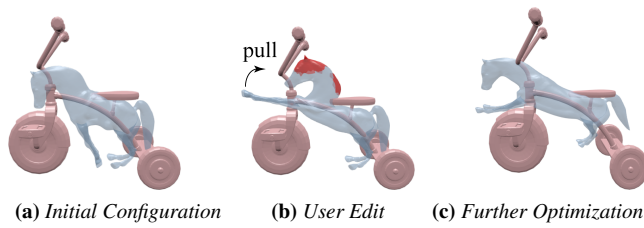


Figure 18: Our optimizer refines the configuration upon user edit. (a) Initial configuration. (b) During user edit, the frontal legs are pulled up by the user, causing intrusion (red) into the restricted zone according to the volumetric design restriction. (c) The optimizer further optimizes the configuration to avoid the intrusion.

8 User Control and Enhancements

One of the goals of our approach is to allow the user to direct the high-level design of zoomorphic shapes. We provide several ways for the user to do so.

Correspondence Search. The user can explore the design space by adjusting the weight vector \mathbf{w} used to select $\hat{\phi}$. Figs. 17(i)–(k) shows different Horse Chairs created using different visual salience weights. Fig. 17(l) shows two Rocking Cows created with different animal shape deformation weights.

Base shape Control. After the correspondence search returns an initial configuration $\hat{\phi}$ and a correspondence \hat{c} , we allow the user to refine the configuration by adjusting the trade-off between base shape and animal shape deformation. The user visualizes this trade-off interactively. Our user interface includes a slider that adjusts the base shape deformation weight w_{dr}^b . After the weight changes, we alter the configuration $\hat{\phi}$ to minimize the configuration energy using gradient descent. We optimize only with respect to the base shape control parameters $\hat{\phi}_b$; i.e., the scales of the cuboids. We deform \mathcal{M}_A accordingly whenever \mathcal{M}_B is deformed, such that its segments still align with the corresponding segments in \mathcal{M}_B given by \hat{c} . This deformation is obtained by applying the affine transformations from the base shape deformation model (Section 6.2) to their corresponding segments in the animal shape. We show the effects of changing w_{dr}^b in Fig. 15. Since we only alter the base shape part scales and scale the animal shape accordingly in this stage, the registration, visual salience, and gash terms will not change significantly from their values in the original configuration $\hat{\phi}$. Therefore we make our optimization more efficient by recomputing only the animal and base shape deformation energy terms.³

User-Guided Refinement. Inspired by the user interface in [Prévost et al. 2013], we allow the user to stop the optimization process, change the pose of the animal shape, and then resume the optimization. Fig. 18 shows an example of altering a horse’s pose.

Segment Removal. For more flexibility, we allow the user to manually specify any segment to be removed in the creation of the zoomorphic shape. For example, in creating the horse chair shown in Fig. 14, the user specified that the segment of the chair’s back should be removed. The effect of removing a segment can be easily previewed in our user interface.

9 Results and Discussion

We have applied our zoomorphic design technique to a variety of models taken from free 3D model repositories on the internet⁴, the Princeton Shape Benchmark [Shilane et al. 2004] and the Shape COSEG Dataset.⁵ Figs. 17, 18, and 19 show the results.

Figs. 17(a)–(d) shows a variety of zoomorphic designs created using our system. The volumetric design restriction ensures that the airplane’s cockpit and wheels are preserved, that the octopus leaves room to occupy the carousel, that the Bear Mug can contain fluid, and that the dolphin’s fin doesn’t prevent sitting on the tricycle.

Pose Constraints. We allow the user to rigidly constrain parts of the animal shape and optimize the non-constrained parts of the animal shape and base shape. We show examples of this process in Figs. 17(e)–(g). Different constraints lead to different proportions in the base shape. In each case, the VDR ensures that there is room for riders to place their feet on the front of the go-kart. For the non-constrained example, the presence of the armadillo’s hands causes the front of the go-kart to widen considerably.

Changing Weights. Figs. 17(i)–(k) shows the effects of using different visual salience weights to produce interesting yet reasonable changes in the design. As the visual salience weight increases, the horse’s head grows increasingly prominent in the zoomorphic shape. In the least salient configuration, the horse’s head is completely removed. In the next most salient configuration, it is visible, but blocked from some viewpoints by the back of the chair. In the most salient configuration, it remains visible from all view angles. Here, the VDR ensured that the horse’s head is high enough that someone can sit underneath it. Fig. 17(l) shows the effects of changing the animal shape deformation weight. Using a large deformation weight for the cow results in a small deformation. This creates a Rocking Cow in which the cow remains close to its initial pose, and the handles from the base shape are used. In contrast, using a small deformation weight for the cow causes the cow to sharply bend its head such that its horns replace the base shape handles.

User Guidance. Zoomorphic shapes are often enhanced when the animal shape is placed in a pose of semantic significance, which our approach cannot recognize automatically. Fortunately, the fine scale optimization step can naturally incorporate user guidance which can provide such semantics. In Fig. 18, the initial solution found by our optimizer is altered by the user, who wants the horse’s legs stretched out in a leaping pose. The manual adjustment puts the horse’s head in a pose that creates gashes, increasing the energy. This prompts our optimizer to find a configuration close to the user’s adjustment, but with lower energy because the gashes are removed. The gashes arose because the horse’s head blocked the rider’s view and therefore intruded into the restricted zone of the VDR.

Note that most of our results required a non-trivial set of changes in order to create aesthetically pleasing zoomorphic shapes that satisfied the design restrictions of the base shape. For example, to create the Horse Tricycle in Fig. 18(a), we need to ensure that the horse’s body merges naturally with the seat and main support, while not blocking the view of the rider or their ability to reach the pedals. In satisfying these objectives, the optimization could not deform the horse unnaturally. To get a good visual salience energy, it also needed to place the horse’s head in a prominent location. The optimization process slightly raises the seat to match with the horse body while bending the neck and legs to avoid blocking the rider.

³See the supplementary video for a demo.

⁴3DVIA, Archive 3D and Autodesk 123D

⁵<http://web.siat.ac.cn/~yunhai/ssl/ssd.htm>



Figure 19: (a) A simple extension of our approach allows creating a zoomorphic shape using multiple animal shapes. A Dragon Vase is created by merging a vase with several dragons. (b) Our approach is not limited to animal shapes. In this example, a shoe and a chair are used to create a Shoe Chair.



Figure 20: A corner of a “zoomorphic restaurant” furnished with a Manatee Chair, an Octopus Lamp, a Dolphin Bottle, and Fish Plates.

Other Results. Our approach can be easily applied to create zoomorphic designs similar to real-world designs. Fig. 17(l) shows our design of Rocking Cows and Fig. 17(m) shows our Jaguar Motorcycle, along with their real-world counterparts. In designing a zoomorphic shape, it is also possible to merge multiple animal shapes with a base shape. Fig. 19(a) shows the design of a Dragon Vase by merging three dragons with a vase. Fig. 19(b) shows an example of merging a non-animal shape (a shoe) with a base shape (a chair), which results in a Shoe Chair. Fig. 20 shows an application of our approach to virtual scene modeling. We create a “zoomorphic restaurant” furnished with a Manatee Chair, an Octopus Lamp, a Dolphin Bottle, and Fish Plates.

Performance We tested an unoptimized implementation of our approach on a 2.4 GHz laptop. By far, the most time-consuming tasks are the correspondence search and configuration refinement, which on average take about 1.5 minutes and 1.0 minute, respectively. Our approach can provide immediate feedback in some situations where it would be highly desirable, such as seeing how the results of the correspondence search change after changing the weights used to select the best correspondence, and seeing how the input shapes deform after adjusting the slider in the base shape control.

10 Evaluation

We conducted informal studies to evaluate our results. The settings of our studies are similar to those in other recent creative 3D modeling efforts [Kalogerakis et al. 2012; Alhashim et al. 2014; Zheng et al. 2013]. Volunteers, who were university students from different majors, were recruited to participate in our studies. Each participant was shown a sequence of images via a web browser. Each

participant was required to answer a question about each image. There were two tests in our studies.

The first test aimed at determining whether the results are zoomorphic—the main theme of our work. At the beginning, the participant was given the definition of zoomorphism⁶ and was shown several images of zoomorphic objects. Next, the participant was randomly presented a sequence of 20 images. Ten images showed a base shape and the other 10 images showed a zoomorphic shape (a result). For each image, the participant had to decide whether or not the shape shown was zoomorphic, or indicate they could not decide.

The second test aimed at determining whether the zoomorphic shapes are plausible examples of the type of base shape from which they were derived. The participant was presented a random sequence of 20 images, each of which showed a zoomorphic shape (a result) created with or without the volumetric design restriction. For each image, the participant had to answer the question “Is this a plausible *category*?”, where *category* refers to the object category of the base shape from which the zoomorphic shape in question was created. For example, for the Horse Chair, the participant was asked “Is this a plausible chair?”. The answer could be either “yes”, “no”, or “undecided”.

We collected 880 responses from 44 participants for our first test. 89.55% of the results were regarded as zoomorphic, while 86.14% of the base shapes were regarded as non-zoomorphic. From informal interviews after the studies, we found that some of the results were not chosen as zoomorphic because the animal quality was subtle. The dolphin tricycle in Fig. 17(d) is an example; some participants were unaware that a dolphin has been merged with the tricycle’s frame.

We collected 820 responses from 41 participants for the second test. 85.85% of the results created with the volumetric design restriction were regarded as visually plausible. Participants pointed out a few problems in the results that make them seem implausible despite the VDR—the Horse Chair with high visual salience may not be structurally stable; the sitting area for the Jaguar Motorcycle may be too small. On the other hand, only 36.59% of the results created without the volumetric design restriction were regarded as visually plausible. These results support our belief that the VDR can preserve in the zoomorphic shape certain essential features from the base shape that would otherwise be lost.

We emphasize that our studies are meant only as an informal preliminary evaluation rather than a scientific validation. They do provide us with useful insights about the design of zoomorphic shapes.⁷

11 Summary

We introduced the problem of zoomorphic shape creation to computer graphics and offered a novel approach to solve it. This problem has the unique challenge of combining very different shapes while preserving key properties and structures. Our approach creates a zoomorphic shape by altering and then merging a man-made and animal shape. We developed a novel technique to ensure that the design restrictions of the man-made shape are still satisfied in the zoomorphic shape. We incorporated this technique into an optimization process that jointly deforms the two shapes to improve the appearance of the resulting zoomorphic shape, according to high level preferences given by the user.

⁶From Wikipedia: “the shaping of something in animal form or terms”.

⁷Our supplementary document provides further details and a list of all the shapes used.

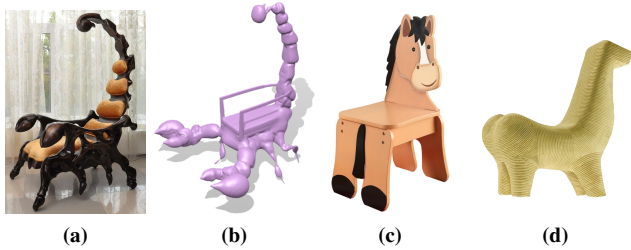


Figure 21: *Limitations.* Our system can only place the scorpion arms at the sides (a) while an artist places them to coincide with the chair arms (a). (c)–(d) Zoomorphic shapes that could not be created by our system, suggesting alternative models based on texture synthesis guided by geometry (c) or shape abstraction (d).

11.1 Limitations and Future Work

Our approach considers the animal shape as a coherent entity, which constrains its parts to locations that will not induce a large global deformation. However, some real world designers relax this constraint, which allows them to create viable zoomorphic shapes that our approach cannot. For example, the designer of the scorpion chair in Fig. 21(a) chose to place the scorpion arms above and behind the scorpion head such that they overlap with the chair arms. The designer effectively disconnected the arms from the rest of the object, prioritizing semantic correspondence over physical validity. A real scorpion cannot deform in this manner, yet the Scorpion Chair is visually provocative. A scorpion chair generated using our approach cannot place the arms in this manner, as it would incur an extremely high deformation energy. Instead we place the scorpion arms at the sides of the chair (Fig. 21(b)), which may be undesirable since the arms take up much space. We believe our approach could be extended naturally to handle this type of deformation by allowing parts of the animal shape to disconnect themselves should the deformation energy become too large.

There are other styles of zoomorphic art that our approach is not designed to handle. Figs. 21(c)–(d) show two examples. The development of approaches to handling these styles offers interesting directions for future work in zoomorphic design. We believe that we have struck in the present paper a reasonable balance between the conciseness of the approach and the variety of results it can produce.

Our volumetric design restriction cannot preserve the functionality of the base shape in all cases. We do not guarantee that the zoomorphic shape will be aerodynamic, physically stable, or capable of bearing a given weight. An interesting direction for future work is to integrate our optimization framework with other approaches that handle these more complex types of functionality (see [Prévost et al. 2013] and [Umetani et al. 2014]). Nevertheless, our approach produces physically realizable results in some cases. Fig. 22 shows a fabricated version of the Horse Chair from Fig. 14.



Figure 22: A 3D-printed horse chair

12 Acknowledgements

We are grateful to the anonymous reviewers for their constructive comments. We also thank Michael S. Brown for narrating the video; William Chi-Fu Lai for providing help and advice on rendering; and Ibraheem Alhashim and Alec Jacobson for providing source code for ARAP mesh deformation. Lap-

Fai Yu is supported by the University of Massachusetts Boston StartUp Grant P20150000029280 and by the Joseph P. Healey Research Grant Program provided by the Office of the Vice Provost for Research and Strategic Initiatives & Dean of Graduate Studies of the University of Massachusetts Boston. Sai-Kit Yeung is supported by SUTD-ZJU Collaboration Research Grant 2012 SUTDZJU/RES/03/2012, SUTD-MIT International Design Center Grant IDG31300106, and Singapore MOE Academic Research Fund MOE2013-T2-1-159. Part of the work was done when Noah was visiting SUTD and supported by Singapore MOE Academic Research Fund MOE2013-T2-1-159. We acknowledge the support of the SUTD Digital Manufacturing and Design (DManD) Centre which is supported by the Singapore National Research Foundation.

References

- ALDERSEY-WILLIAMS, H. 2003. *Zoomorphic: New Animal Architecture*. HarperDes.
- ALHASHIM, I., LI, H., XU, K., CAO, J., MA, R., AND ZHANG, H. 2014. Topology-varying 3D shape creation via structural blending. *ACM Trans. Graph.* (July).
- ALIAGA, D. G., VANEGAS, C. A., AND BENEŠ, B. 2008. Interactive example-based urban layout synthesis. In *ACM Trans. Graph.*, vol. 27, ACM, 160.
- BÄCHER, M., WHITING, E., BICKEL, B., AND SORKINE-HORNUNG, O. 2014. Spin-it: Optimizing moment of inertia for spinnable objects. *ACM Trans. Graph.* 33, 4, 96.
- BARAN, I., AND POPOVIĆ, J. 2007. Automatic rigging and animation of 3D characters. In *ACM Trans. Graph.*, vol. 26, ACM, 72.
- BRAMSTON, D. 2008. *Basics Product Design 01: Idea Searching*. AVA Publishing.
- CHAUDHURI, S., AND KOLTUN, V. 2010. Data-driven suggestions for creativity support in 3D modeling. *ACM Trans. Graph.* 29, 6 (Dec.), 183:1–183:10.
- CHAUDHURI, S., KALOGERAKIS, E., GUIBAS, L., AND KOLTUN, V. 2011. Probabilistic reasoning for assembly-based 3D modeling. In *ACM Trans. Graph.*, vol. 30, 35.
- COATES, M., BROOKER, G., AND STONE, S. 2009. *The Visual Dictionary of Interior Architecture and Design*. Fairchild Books.
- FUNKHOUSER, T., KAZHDAN, M., SHILANE, P., MIN, P., KIEFER, W., TAL, A., RUSINKIEWICZ, S., AND DOBKIN, D. 2004. Modeling by example. In *ACM Transactions on Graphics (TOG)*, vol. 23, ACM, 652–663.
- IGARASHI, T., MATSUOKA, S., AND TANAKA, H. 1999. Teddy: A sketching interface for 3D freeform design. In *ACM Trans. Graph.*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, SIGGRAPH '99, 409–416.
- IGARASHI, Y., IGARASHI, T., AND MITANI, J. 2012. Beady: Interactive beadwork design and construction. *ACM Trans. Graph.* 31, 4, 49.
- JACOBSON, A., BARAN, I., KAVAN, L., POPOVIĆ, J., AND SORKINE, O. 2012. Fast automatic skinning transformations. *ACM Trans. Graph.* 31, 4, 77.
- JAIN, A., THORMÄHLEN, T., RITSCHER, T., AND SEIDEL, H.-P. 2012. Exploring shape variations by 3d-model decomposition and part-based recombination. In *Computer Graphics Forum*, vol. 31, Wiley Online Library, 631–640.

- KALOGERAKIS, E., HERTZMANN, A., AND SINGH, K. 2010. Learning 3D mesh segmentation and labeling. *ACM Trans. Graph.* 29, 4 (July), 102:1–102:12.
- KALOGERAKIS, E., CHAUDHURI, S., KOLLER, D., AND KOLTUN, V. 2012. A probabilistic model for component-based shape synthesis. *ACM Trans. Graph.* 31, 4, 55.
- KASHIMA, H., TSUDA, K., AND INOKUCHI, A. 2004. Kernels for graphs. In *Kernel methods in computational biology*, 155–170.
- LAGA, H., MORTARA, M., AND SPAGNUOLO, M. 2013. Geometry and context for semantic correspondences and functionality recognition in man-made 3D shapes. *ACM Trans. Graph.* 32, 5, 150.
- LEE, C. H., VARSHNEY, A., AND JACOBS, D. W. 2005. Mesh saliency. In *ACM Trans. Graph.*, vol. 24, ACM, 659–666.
- LI, H., ALHASHIM, I., ZHANG, H., SHAMIR, A., AND COHEN-OR, D. 2012. Stackabilization. *ACM Trans. Graph.* 31, 6, 158.
- LIDWELL, W., AND MANACSA, G. 2011. *Deconstructing Product Design: Exploring the Form, Function, Usability, Sustainability, and Commercial Success of 100 Amazing Products*. Rockport Publishers.
- LIDWELL, W., 2014. Empowering children through eyewear design, Aug.
- LOZANO, J. A. 2006. *Towards a new evolutionary computation: Advances on estimation of distribution algorithms*, vol. 192. Springer.
- MERRELL, P., SCHKUFZA, E., LI, Z., AGRAWALA, M., AND KOLTUN, V. 2011. Interactive furniture layout using interior design guidelines. In *ACM Trans. Graph.*, vol. 30, ACM, 87.
- MITRA, N., WAND, M., ZHANG, H. R., COHEN-OR, D., KIM, V., AND HUANG, Q.-X. 2013. Structure-aware shape processing. In *SIGGRAPH Asia 2013 Courses*, ACM, 1.
- PRÉVOST, R., WHITING, E., LEFEBVRE, S., AND SORKINE-HORNUNG, O. 2013. Make it stand: Balancing shapes for 3D fabrication. *ACM Trans. Graph.* 32, 4.
- SCHMIDT, R., AND SINGH, K. 2010. Meshmixer: An interface for rapid mesh composition. In *ACM SIGGRAPH 2010 Talks*, ACM, 6.
- SHAPIRA, L., SHALOM, S., SHAMIR, A., COHEN-OR, D., AND ZHANG, H. 2010. Contextual part analogies in 3D objects. *International Journal of Computer Vision* 89, 2-3, 309–326.
- SHAWE-TAYLOR, J., AND CRISTIANINI, N. 2004. *Kernel methods for pattern analysis*. Cambridge University Press.
- SHEFFER, V. K. D. J. A. 2007. Shuffler: Modeling with interchangeable parts. *Visual Computer journal*.
- SHILANE, P., MIN, P., KAZHDAN, M., AND FUNKHOUSER, T. 2004. The princeton shape benchmark. In *Shape modeling applications, 2004. Proceedings*, IEEE, 167–178.
- SORKINE, O., AND ALEXA, M. 2007. As-rigid-as-possible surface modeling. In *Symposium on Geometry processing*, vol. 4.
- TAGLIASACCHI, A., ALHASHIM, I., OLSON, M., AND ZHANG, H. 2012. Mean curvature skeletons. In *Computer Graphics Forum*, vol. 31, Wiley Online Library, 1735–1744.
- TAKAYAMA, K., SCHMIDT, R., SINGH, K., IGARASHI, T., BOUBEKEUR, T., AND SORKINE, O. 2011. Geobrush: Interactive mesh geometry cloning. *Computer Graphics Forum* 30, 2, 613–622.
- UMETANI, N., KAUFMAN, D. M., IGARASHI, T., AND GRINSPUN, E. 2011. Sensitive couture for interactive garment editing and modeling. *ACM Trans. Graph.* 30, 4.
- UMETANI, N., KOYAMA, Y., SCHMIDT, R., AND IGARASHI, T. 2014. Pteromys: Interactive design and optimization of free-formed free-flight model airplanes. *ACM Trans. Graph.* 33, 4, 65.
- VANEGAS, C. A., GARCIA-DORADO, I., ALIAGA, D. G., BENES, B., AND WADDELL, P. 2012. Inverse design of urban procedural models. *ACM Trans. Graph.* 31, 6 (Nov.).
- YU, L.-F., YEUNG, S. K., TANG, C.-K., TERZOPOULOS, D., CHAN, T. F., AND OSHER, S. 2011. Make it home: Automatic optimization of furniture arrangement. *ACM Trans. Graph.* 30, 4, 86.
- ZHANG, H., SHEFFER, A., COHEN-OR, D., ZHOU, Q., VAN KAICK, O., AND TAGLIASACCHI, A. 2008. Deformation-driven shape correspondence. In *Computer Graphics Forum*, vol. 27, Wiley Online Library, 1431–1439.
- ZHENG, Y., COHEN-OR, D., AND MITRA, N. J. 2013. Smart variations: Functional substructures for part compatibility. *Computer Graphics Forum (Eurographics)* 32, 2pt2, 195–204.
- ZHENG, Y., DORSEY, J., AND MITRA, N. J. 2014. Ergonomic-driven geometric exploration and reshaping. *CoRR abs/1402.5440*.
- ZHOU, Y., SUEDA, S., MATUSIK, W., AND SHAMIR, A. 2014. Boxelization: Folding 3D objects into boxes. *ACM Trans. Graph.* 33, 4, 71.
- ZHU, L., XU, W., SNYDER, J., LIU, Y., WANG, G., AND GUO, B. 2012. Motion-guided mechanical toy modeling. *ACM Trans. Graph.* 31, 6, 127.