

January 2015

Embodied Interactions for Spatial Design Ideation: Symbolic, Geometric, and Tangible Approaches

FNU VINAYAK

Purdue University

Follow this and additional works at: https://docs.lib.purdue.edu/open_access_dissertations

Recommended Citation

VINAYAK, FNU, "Embodied Interactions for Spatial Design Ideation: Symbolic, Geometric, and Tangible Approaches" (2015). *Open Access Dissertations*. 1325.

https://docs.lib.purdue.edu/open_access_dissertations/1325

This document has been made available through Purdue e-Pubs, a service of the Purdue University Libraries. Please contact epubs@purdue.edu for additional information.

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By FNU Vinayak

Entitled

Embodied Interactions for Spatial Design Ideation: Symbolic, Geometric, and Tangible Approaches

For the degree of Doctor of Philosophy

Is approved by the final examining committee:

Karthik Ramani

Chair

David Ebert

Jitesh Panchal

Niklas Elmqvist

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University's "Policy of Integrity in Research" and the use of copyright material.

Approved by Major Professor(s): Karthik Ramani

Approved by: Jay P. Gore

Head of the Departmental Graduate Program

11/30/2015

Date

EMBODIED INTERACTIONS FOR SPATIAL DESIGN IDEATION:
SYMBOLIC, GEOMETRIC, AND TANGIBLE APPROACHES

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Vinayak

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

December 2015

Purdue University

West Lafayette, Indiana

Dedicated to my family.

ACKNOWLEDGMENTS

*If you don't fail at least 90 percent of the time,
you're not aiming high enough.*

– Alan Kay

There is little possibility of intellectual growth without the necessary and sufficient amount of challenge. The credit for helping me internalize this fact through experience goes to my advisor Dr. Karthik Ramani. His constant demand for the seemingly impossible is what has motivated me to strive for higher standards and has led me to a deeper understanding of the vision of design democratization. I am grateful to him for guiding me in developing the art of embedding design principles within design research itself.

I have had the good fortune of learning from great mentors at Purdue University. Of these, I would first like to extend my gratitude to late Dr. Shreeram Abhyankar for introducing me to Algebraic Geometry - a subject which I could never have approached without his energetic and insightful teaching. I am very grateful to Dr. Niklas Elmquist for introducing me to the colorful areas of information visualization and visual analytics. His insights in human-computer interaction (HCI) have been of critical importance in my research. Be it the regarding the future of the design community or the preparation for an academic life, my conversations with Dr. Jitesh Panchal have been most insightful. I thank him sincerely for sharing his insights with me. Finally, I would like to thank Dr. David Ebert for providing his valuable input to this thesis.

I personally believe that no HCI research can exist without its most fundamental component: users! So, I would like to thank all the people who gave up some of their valuable time towards helping me ask (and to some extent answer) the research questions that I have posed during the course of this work.

I wish to thank my colleagues and close friends Senthil Chandrasegaran, Devarajan Ramanujan, Cecil Piya, Ayan Sinha, Sujin Jang, and William Benjamin. They have en-

riched my experience at C-Design Lab beyond all measures. My past collaborators and dear friends, Dr. Sundar Murugappan and Andrew (Woodie) Polk played significant roles in shaping my research through their valuable inputs. A very special thanks to my friend and colleague Bill “*The Voice of C-Design*” Bernstein. His voice was the one that gave life to the video descriptions accompanying most of my publications at C-Design Lab.

The unconditional support of my parents and my brother is what has ultimately transformed my efforts into this thesis. I thank them, beyond measure and words, for standing by my side in all my efforts. As I write this acknowledgment, I cannot help getting distracted by a tail, wagging vehemently, seeking attention. Thanks to Dasha who, in the last few months, has become an integral part and the official stress-buster of my family.

Finally, I thank Kalyani – my wife, my best friend, and my daily dose of sanity. I have no idea what I would do without you!

TABLE OF CONTENTS

	Page
LIST OF TABLES	x
LIST OF FIGURES	xi
ABSTRACT	xvii
1. INTRODUCTION	1
1.1 The Context of Spatial Design Ideation	2
1.2 The Motivation for Spatiality	3
1.3 Embodiment & Embodied Interactions	5
1.4 Overview	5
1.4.1 Conceptual Organization	6
1.4.2 Summary of Contributions	8
1.4.2.1 Symbolic Approach	8
1.4.2.2 Geometric Approach	9
1.4.2.3 Tangible Approach	10
2. SYMMETRIC BI-MANUAL INTERACTIONS FOR MID-AIR CREATION OF ROTATIONALLY SYMMETRIC SHAPES	11
2.1 Overview	12
2.2 Implementation	14
2.3 Results	15
2.4 Conclusions	18
3. GESTURE-BASED MID-AIR INTERACTIONS FOR DESIGN IDEATION . .	19
3.1 Related Work	20
3.1.1 Mid-Air Shape Modeling	20
3.1.2 Gesture Recognition & Hand Skeletal Tracking	20
3.1.3 A Brief Review of Geometric Modeling for Design	22
3.2 Overview	23
3.3 Shape-Gesture-Context Interplay	24
3.3.1 Gesture	25
3.3.2 Context	25
3.3.3 Generalized Cylinder (GC)	26
3.3.4 Sectional and Skeletal Deformations	27
3.3.5 Intelligent Generalized Cylinder	30
3.4 Implementation Approach for SGC-I	32
3.5 IGC Creation	35

	Page
3.6 IGC Modification	37
3.6.1 Skeletal Deformation	39
3.6.2 Sectional Deformation	44
3.6.3 Sectional Scaling	45
3.7 IGC Manipulation	46
3.8 Implementation	47
3.9 Results	47
3.10 Conclusions	49
4. GEOMETRIC APPROACH FOR MID-AIR DEFORMATION THROUGH PROX- IMAL ATTRACTION	50
4.1 Related Work	51
4.1.1 Instrumented Controllers	51
4.1.2 Gesture-Based Input	52
4.2 Problem	52
4.2.1 Overview	53
4.3 Proximal Attraction	55
4.3.1 Pot Definition	55
4.3.2 Deformation Method	57
4.3.3 Attraction	60
4.3.4 Pushing vs. Pulling:	61
4.3.5 Accidental Deformation:	62
4.4 Guidelines for Controllability	65
4.4.1 Sampling and Resolution	65
4.4.2 Deformation Parameters ($\mathcal{D}(\alpha, \beta)$)	65
4.4.3 Attraction Parameters ($\mathcal{P}(\varepsilon_C, \varepsilon_P, \tilde{\lambda}, \tilde{\lambda}^P, A, \lambda(t), T)$)	66
4.5 Implementation	66
4.5.1 System Description	66
4.5.2 Interface	69
4.5.2.1 Leap Motion Controller:	69
4.5.2.2 SoftKinetic:	69
4.6 Results	71
4.6.1 Hand and Physical Tools	71
4.6.2 Virtual Tools	74
4.6.3 Efficiency	74
4.7 User Evaluation with Virtual Tools	76
4.8 Discussion	77
4.8.1 Three-Dimensionality	77
4.8.2 Limitations	77
4.9 Conclusions	78
5. HAND GRASP & MOTION FOR MID-AIR SHAPE DEFORMATION	80
5.1 Related Works	81

	Page
5.1.1 Mid-Air Gestures	81
5.1.2 Mid-Air Virtual Pottery	81
5.1.3 Hand Grasp	82
5.2 Overview	83
5.2.1 Intent & Controllability	83
5.2.2 Rationale for Pottery	83
5.2.3 Evaluation Approach	84
5.3 Hand as a Point: Clutching by Proximal Attraction	85
5.3.1 Technique	85
5.3.2 Preliminary Evaluation	88
5.3.2.1 Apparatus	88
5.3.2.2 Implementation	88
5.3.2.3 Participants & Procedure	89
5.3.3 Results	90
5.3.4 Takeaways	91
5.4 Hand as a PCL: Shaping by Proximal Attraction	91
5.4.1 Technique	91
5.4.1.1 Pushing vs. Pulling	92
5.4.1.2 Initialization Time	93
5.4.2 Experiment	93
5.4.2.1 Apparatus	94
5.4.2.2 Implementation & Interface	94
5.4.2.3 Participants	95
5.4.2.4 Procedure	95
5.4.3 Metric for Quality of User Response	96
5.4.4 Results	98
5.4.4.1 User Performance (T1)	98
5.4.4.2 Hand Usage (T1)	99
5.4.4.3 Hand Usage (T2)	101
5.4.4.4 Reaching, Grasping, & Deformation Strategies	101
5.4.4.5 Intent & Controllability (Q1)	102
5.4.4.6 User Experience (Q2)	104
5.4.5 Takeaways	104
5.5 Hand as a PCL: Grasp + Motion	106
5.5.1 Technique	106
5.5.2 Experiment	108
5.5.2.1 Participants	108
5.5.2.2 Procedure	109
5.5.3 Results	110
5.5.3.1 User Performance (T1)	110
5.5.3.2 Hand Usage (T1)	111
5.5.3.3 Hand Usage (T2)	114

	Page
5.5.3.4 Geometric Characterization of Tools	114
5.5.3.5 Tool Usage (T3)	115
5.5.3.6 Intent & Controllability (Q1)	116
5.5.3.7 User Experience (Q2)	116
5.5.4 Limitations	118
5.6 Discussion	119
5.6.1 Spectrum of Expressiveness	119
5.6.2 Definition of Intent:	119
5.6.3 Generalization	119
5.6.4 Precise & Selective Reachability	121
5.7 Conclusions	122
6. SPATIAL DESIGN IDEATION USING A SMARTPHONE AS A HAND-HELD REFERENCE PLANE	123
6.1 Related Works	124
6.1.1 Mobile Devices for 3D Manipulation	124
6.1.2 Design with Mobile Devices	125
6.2 MobiSweep	126
6.2.1 System Setup	126
6.2.2 Design Rationale	126
6.2.3 Gesture Definition	128
6.2.4 Modeling States	130
6.2.5 Modeling Work-Flow	131
6.2.5.1 Shape Creation	131
6.2.5.2 Shape Modification	132
6.2.5.3 Shape Manipulation	133
6.3 Implementation	133
6.3.1 Hardware & Software	133
6.3.2 Algorithms	135
6.3.2.1 Sweep Surface Generation	135
6.3.2.2 Section Modification	135
6.4 User Evaluation	136
6.4.1 Participants	136
6.4.2 Procedure	137
6.4.3 Results	139
6.4.3.1 Interactions	139
6.4.3.2 Creative Support	140
6.4.3.3 Utility	141
6.4.3.4 Limitations	142
6.5 Discussion	142
6.6 Conclusions	143
7. CONCLUSION	144

	Page
7.1 Summary	144
7.2 Design Implications	145
7.3 Future Directions	147
7.4 Closing Statement	149
LIST OF REFERENCES	150
VITA	163

LIST OF TABLES

Table	Page
3.1 Operation parameter definition for modeling operations.	34
4.1 The times (millisec.) taken for the steps of our algorithm are shown for different combinations of pot resolution and PCL sampling resolution.	75
5.1 Behavioral observations in our preliminary evaluation.	89

LIST OF FIGURES

Figure	Page
1.1 Conceptual organization of the thesis.	7
2.1 Overview of <i>HandyPotter</i> : shape modeling metaphor inspired by pottery (left) and interpretation of hand locations for shape creation (right).	11
2.2 Constrained and free form generalized cylinders.	12
2.3 Computational pipeline for <i>HandyPotter</i>	14
2.4 Virtual working volume for sweep creation.	15
2.5 A snapshot of the interface for <i>HandyPotter</i> showing the user creating a shape using two-handed motion.	16
2.6 Fully constrained, partially constrained and free-form generalized cylinders using <i>HandyPotter</i>	17
2.7 Pot-like shapes modeled using <i>HandyPotter</i>	17
3.1 An overview of Shape-Gesture-Context Interplay.	19
3.2 Components of the proposed shape exploration paradigm.	24
3.3 Section parameters (a) orientation (b) handle and deflection (c) handle-vertex distance.	28
3.4 Selected hand postures (a) Release (b) Grab (c) Point.	28
3.5 Physical interpretation of the SGC-I framework for IGC modeling.	32
3.6 Mapping of hand motion is illustrated for the creation of IGC through scaling, rotation, translation of a given section of a swept surface.	34
3.7 Fully constrained, partially constrained and free-form IGC, orientational and translational snapping in swept surfaces. Note that this particular subset of shapes was shown in <i>HandyPotter</i> (Chapter 2).	34
3.8 Skeleton deformation (a) method (b) circular bending and (c) rectification for $\theta = \pi/3$ and $0.75 \leq d_{bt} \leq 0.95$	37
3.9 Recursive bending of IGC.	38
3.10 Deformation functions (a) long-range plot (b) close-up for range $[0, 1]$	41

Figure	Page
3.11 Recursive multi-handle deformation of a half-unit circle based on gaussian (a-d), cauchy (e-h) and laplacian of cauchy (i-l) functions.	42
3.12 Deformation of a cylinder original shape (left) to horizontally constrained 2D deformation (middle-left) and recursive 3D deformations (middle right and right).	43
3.13 Recursive scaling of a cylinder.	44
3.14 Pipeline showing the flow of information from the user to the proposed system.	45
3.15 User interface of the prototype.	46
3.16 Shapes modeled using Shape-It-Up.	48
4.1 Definition of a pot and the deformation using proximal persistence.	53
4.2 Deformation strategy of a curve γ (row 1) involves initializing a seed displacement (row 2), application of the Laplace operator to obtain a smooth deformation curve δ (row 3), and finally the computation of deformed curve $\gamma + \delta$	57
4.3 The selective application of Laplace operator is based on the window operator α which defines the neighborhood around some given vertex i in a profile.	58
4.4 Recursive attraction of a handle q towards a manipulator p is shown. The parameter $\lambda \in [0, 1]$ defines the rate of convergence and is akin to the <i>smoothing constant</i> in single exponential smoothing.	58
4.5 Algorithm for the application of proximal attraction to a discrete profile curve of the pot for one manipulator.	61
4.6 An illustration of proximal attraction for hand PCL-based deformation is shown.	63
4.7 The intent for push and pull can be disambiguated by the identifying the shape of a manipulating object even when the overall motion is similar (top row). We apply outward attraction on the profile in terms of a smooth function decreasing with respect to the distance between that manipulators and the handle (bottom row).	64
4.8 The overall setup consists of a computer and a depth camera. We implemented our approach using two such cameras, SoftKinetic DS325 and the Leap Motion controller.	67
4.9 The depth sensing module for SoftKinetic involves the segmentation of hand's depth image from the full depth image using the DepthSense SDK. This is followed by optional image processing steps for skeletal and boundary images, and subsequently the segmented depth image is converted to a PCL using camera parameters.	68

Figure	Page
4.10 Our interface consists of active partitions of the total interaction space where each partition is associated with a virtual camera. Cameras C0, C1, and C2 correspond to the pottery wheel, tool-set 1, and tool-set 2 respectively. User's motion towards the left or right extremity results in camera transition and a subsequent re-mapping of the users physical interaction space to the active partition. The selection of tools is based on a dwell-time strategy.	70
4.11 Attraction parameters identified with sufficiently-responsive deformation were applied for skeletal (left), boundary (middle), and hybrid (right) PCL's of the hand. While the boundary and hybrid PCL's are generally more controllable, the skeletal PCL shows better controllability for local-pulling tasks.	71
4.12 Three use cases are shown for using day-to-day physical objects (left - kitchen knife, middle - mobile phone, and right - CD-case) as shaping tools.	72
4.13 A variety of virtual tools are shown with their respective deformations on the pots.	73
4.14 The results of the pot composition are shown Pots were created by participants using our full hand PCL-based interface.	74
4.15 Pots that were created by young participants using our Leap-based interface.	76
5.1 Algorithm for one-point pot deformation is illustrated for proximal-attraction. The pot is gradually deformed by attracting the profile towards the hand (represented by a point). Subsequently, each section is re-scaled to obtain the deformed pot surface.	86
5.2 Two strategies are shown for clutching and deforming a pot using hand as a single point. In the first approach (a) grab and release gestures. The second (b) is the proximal-attraction approach.	86
5.3 An example of common behavior is shown wherein users shaped their hands to express their intent for deformation.	87
5.4 Algorithm for pot deformation is illustrated for proximal-attraction. The profile is deformed based on the proximity of the points on a given hand PCL. Subsequently, each section is re-scaled to obtain the deformed pot surface.	91
5.5 The apparatus (a) consists of the user, a computer and a depth camera. The user sees a PCL of their hand deforming a rotating pot (b).	94
5.6 Eight pre-defined pots were shown to participants in the quiz. These are: (a, b) thin convex and thin concave, (c,d) fat convex and concave, (e, f) round and flat, and (g, h) flat at center and ends.	95

Figure	Page
5.7 Curvature cross-correlation is used as measure of quality of user created pots. Given the observed and target profiles (a), we compute curvature signatures (b), and subsequently compute maximum correlation between the signatures (c). Notice the sensitivity of the curvature signature (b) for a seemingly high visual similarity. In the example shown, the profile of the observed pot are shifted upwards by 5 sections with respect to the target pot.	97
5.8 User performance is shown for the each quiz problem as a bag-plot. The x-axis is time in the range [0, 14] minutes and the y-axis is the curvature cross-correlation in the range [0, 1]. The dark and light blue regions show the bag and fence regions, respectively. The white circle is the Tukey depth median and the points marked with red circles are the outliers. The insets show the actual pot profiles (black lines) created by the users in comparison to the target shapes (beige region) of the Quiz. The coordinates of the tukey median (C) and the spread (Sp) are provided for each target shape.	97
5.9 Common user patterns are shown in terms of grasp and motion performed by users for each target shape (in decreasing order of occurrence along columns). The hand images represent the grasp and the arrows (red) show the motion of the hand. The most successful strategies are indicated by blue boxes for each target shape.	100
5.10 Two examples are shown of common deformation strategies are shown through which users created (a) thin concave and (b) flat-round features.	100
5.11 User response to are shown for proximal-attraction. The main issue in terms of controllability (a) was the slow response and difficulty of pushing in comparison to pulling.	103
5.12 Algorithm for grasp+motion technique is illustrated. The main steps involve computation of axial KDE for hand PCL, detection of intent for smoothing, differentiation between pulling and pushing, and deformation of the pot. In this example, we show the details of the pulling deformation (row 2).	105
5.13 a. KDE functions are shown for a pulling (left) and pushing (right) intents, b. Computation of attraction rate using the angle of grasp is illustrated.	107
5.14 The thin convex and concave features were modified according to the capability provided by the <i>grasp+motion</i> technique.	108
5.15 User created pot profiles (black curves) are shown relative to the target shapes (light brown cross sections). The top and bottom rows shows the results for proximal-attraction and grasp+motion approaches respectively. Visual inspection evidently shows improvements in the creation of flat, round and smooth features. More significant improvements were observed in the creation of fat convex features in comparison to proximal-attraction.	110

Figure	Page
5.16 A comparison between proximal-attraction (top row) and grasp+motion (bottom row) is shown in terms of (a) the time taken by users to shape a target profile, (b) the quality of users' responses in terms of curvature cross-correlation of profiles, and (c) the distribution of users with respect to the number of trials per target profile.	112
5.17 User performance is shown for the each quiz problem as a bag-plot. The x-axis is time in the range [0, 14] minutes and the y-axis is the curvature cross-correlation in the range [0, 1]. The dark and light blue regions show the bag and fence regions, respectively. The white circle is the Tukey depth median and the points marked with red circles are the outliers. The insets show the actual pot profiles (black lines) created by the users in comparison to the target shapes (beige region) of the Quiz. The coordinates of the depth median (C) and the spread (Sp) are provided for each target shape.	113
5.18 The characterization of tool geometry is visualized for five different physical objects. The objects were chosen to represent concave, convex, flat, and round contacts for deformation.	113
5.19 Examples of tool usage are shown.	115
5.20 User response to are shown for grasp+motion. While the robustness to accidental deformations was perceived to be negligible (a), many users still perceived pulling to be difficult. Users agreed regarding the usefulness of tools but were not in general agreement about preferring them over hands.	117
5.21 Asymmetric deformation can be applied to a pot in two steps. When the pot is rotating, we apply the <i>axial</i> KDE (top row) of the hand PCL for deforming the profile of the pot. Subsequently, users can stop rotating the pot and deform the pot locally using the <i>polar</i> KDE (bottom row).	120
5.22 The computation of two-dimensional KDE in the parametric space of a cylindrical surface leads to the computation of grasp and motion for an arbitrary orientation of the hand PCL with respect to the surface. This allows for arbitrary ddeformation of the surface. Recomputing and segmenting the deformed surface using the method of Bærentzen et al. [146] provides a generalized deformation approach using our KDE based approach.	120
6.1 The idea behind <i>MobiSweep</i> is to utilize the spatial relation between the action of sweeping with the creation of the swept surface.	124
6.2 Setup for <i>MobiSweep</i> comprises of a visual display of the virtual environment and a smartphone that acts as a reference plane in the virtual environment.	127

Figure	Page
6.3 There are six gestures (row 1) that enable the interactions across three modeling states: Configure (row 2), Author (row 3), and Manipulate (row 4). 0F, 1F, 2F, and 3F denote 0, 1, 2, and 3 finger gestures respectively.	129
6.4 For shape creation (a), the user selects the Add Shape menu item and creates a sweep surface using one or three finger offsetting gesture. For manipulating a shape (b), the user first hovers on a desired sweep surface and selects the shape using the <i>double-tap</i> gesture. In the shape modification example (c) the user modifies the initial section by sketching, creates a sweep surface, and modifies the final section by sketching, scaling, and panning.	134
6.5 Algorithms for (a) sweep generation and (b) section sketching.	136
6.6 Design concepts generated by the users are shown (kettles and jars are shown in the top three rows and lamps in the bottom two rows). Each box represents concepts generated by one user.	138
6.7 User feedback for interaction ratings in the context of the work-flow states.	139
6.8 User feedback for creativity support in <i>MobiSweep</i>	141

ABSTRACT

Vinayak. Ph.D., Purdue University, December 2015. Embodied Interactions for Spatial Design Ideation: Symbolic, Geometric, and Tangible Approaches. Major Professor: Karthik Ramani, School of Mechanical Engineering.

Computer interfaces are evolving from mere aids for number crunching into active partners in creative processes such as art and design. This is, to a great extent, the result of mass availability of new interaction technology such as depth sensing, sensor integration in mobile devices, and increasing computational power. We are now witnessing the emergence of maker culture that can elevate art and design beyond the purview of enterprises and professionals such as trained engineers and artists. Materializing this transformation is not trivial; everyone has ideas but only a select few can bring them to reality. The challenge is the *recognition* and the subsequent *interpretation* of human actions into *design intent*.

Taking inspiration and guidance from embodied interactions, our focus in this work is to design, develop, and evaluate interfaces and interaction techniques to support idea generation for the design of three-dimensional (3D) shapes. Grounded in the phenomenological tradition of philosophy, embodied interactions represent a general framework to understand the transformation of *action* into *meaning* – in our case, design intent. We present concrete instances of this framework by investigating symbolic, geometric, and tangible approaches for transforming human action into design intent.

We begin with the symbolic transformation of action to intent by introducing *shape-gesture-context interplay* (SGC-I), a framework that enables the expression of design intent in the virtual environment through a prescribed set of hand gestures combined with arm motion. Here, the main idea is to automatically deduce the nature and extent of geometric constraints by interpreting human gestures and motions *in the context of a given modeling operation*.

We explore the geometric transformation of action to intent by introducing *proximal attraction*, a dwell-time approach that interprets a shape deformation as a progressive convergence of the shape of the manipulated object (such as a lump of clay) to that of the manipulating object (such as a hand or a tool). We demonstrate our approach using a modeling metaphor inspired by pottery, which offers an unambiguous interaction context via a well-defined and intuitive relationship between the use of hands and the shaping of pots. We implement and evaluate a novel algorithm that uses kernel-density estimation to characterize the contact between the hand and a 3D shape. In contrast to the symbolic nature of *SGC-I*, the geometric approach does not compute any finite set of gestures or hand skeleton. Instead, it implicitly extracts the grasp and motion from the raw point-cloud (PCL) of the user’s hand for deforming the shape of a pot in 3D space. This feature of our method directly allows a user to shape pots by using physical artifacts as tools adding tangibility to the process of shape ideation.

In the final stage of this work, we introduce a novel interaction metaphor for using a hand-held smartphone as a medium for spatial design ideation. The core goal here is to enable direct externalization of spatial design concepts by embedding the geometric representation of the artifact within the physicality and tangibility of the creation process itself. To achieve this goal, we re-purpose a smartphone as a hand-held *reference plane* for creating, modifying, and manipulating 3D sweep surfaces. We implement *MobiSweep*, a prototype application to explore a new design space of constrained spatial interactions that combine direct orientation control with indirect position control via well-established multi-touch gestures. *MobiSweep* leverages kinesthetically aware interactions for the creation of a sweep surface by utilizing the spatial relationship between the physical action of sweeping and the creation of the resulting swept surface. The design concepts generated by users, in conjunction with their feedback, demonstrate the potential of such interactions in enabling spatial ideation.

1. INTRODUCTION

For years radios had been operated by means of pressing buttons and turning dials; then as the technology became more sophisticated the controls were made touch-sensitive—you merely had to brush the panels with your fingers; now all you had to do was wave your hand in the general direction of the components and hope. It saved a lot of muscular expenditure, of course, but meant that you had to sit infuriatingly still if you wanted to keep listening to the same program. – *Douglas Adams, Hitchhiker’s Guide to the Galaxy 1979*

Computer support for design, architecture, and art has evolved in much the same way in the real world as the “radio” which Mr. Adams imagined in his work of science fiction. The invention of the mouse [1] fundamentally changed how digital content was created, manipulated, and accessed. Novel interactions were developed to provide a graphical-user-interface (GUI) using the windows-icons-menus-pointers (WIMP) paradigm. It was possible for a designer to create digital geometric models by using geometrically coherent interaction metaphors such as *click-and-drag* or *click-and-move*. Innovations in touch-enabled technologies took a significant step forward wherein a user could create geometric content by scribbling on a tablet, much in the same way as one would sketch on a piece of paper. The recent success of Microsoft Kinect in the gaming industry demonstrated the significance of using spatial human motion as a method for expression of the user’s intent towards more engaging virtual experiences.

There seems to be common element behind these success stories – each of these developments took a radical step towards integrating human cognitive capabilities and motor skills within the interactive workflow. In this thesis, our aim is to explore how physical human movement can be controllably transformed into the intent for designing three-dimensional artifacts. Our broader approach, to this end, is to view the process of idea generation in early phase design through the lens of embodied interactions.

1.1 The Context of Spatial Design Ideation

Early-phase ideation is fundamental to product and industrial design processes. As Hsu and Liu [2] note: “*even the highest standard of detailed design cannot compensate for a poor design concept formulated at the conceptual design phase*”. Idea generation and detailed design are fundamentally different in terms of the utility that they provide to the design process. Ideation involves *divergent thinking* for quick externalization of a wide variety of ideas, as opposed to detailed design which entails *convergent thinking*. From a cognitive standpoint, these thinking modes are understood as lateral and vertical transformations respectively [3]. Thus, the primary goal of the ideation process is to help the designer understand the design problem by exploring the problem space through quick externalization of ideas [4, 5]. This exploratory nature of ideation demands an uninhibited flow between what a designer is *thinking* and what the designer is *doing* to communicate the thought.

Horváth [6] states: “*conceptual design can be seen as a cognitive process in which, ideation, externalization, synthesis and manipulation of mental entities, called design concepts, takes place in symbiosis in a short-term evolutionary process*”. Historically, sketching has played the most significant role to help externalize visual ideas; many creative ideas began with sketches. Designers still predominantly prefer sketching to express their ideas [2, 7]. To this end, current research on computer-aided ideation tools is also targeted towards sketch-based brainstorming [8, 9]. It is, however, natural to assume that the exploration of 3D product forms would require the designer to imagine concepts that are spatial. Consequently, externalizing a 3D concept as a collection of segmented 2D projections is a task that requires significant skill and training. This is perhaps why sketching is perceived as a challenging medium by novice designers while communicating 3D forms [10]. While 2D artifacts like sketches and drawings are better off being created with 2D interfaces, we find that the creation of 3D shapes using 2D interfaces limits the capability of designers to experiment at conceptual and artistic levels [11]. There has been significant effort made towards creating sketching systems, for instance *EverybodyLovesSketch* [12], that

aim at reducing the barrier to entry for users untrained in sketching. Even these systems are mainly focused towards the creation of detailed 3D sketches rather than quick design conceptualization.

We seek to bridge the gap between the designer’s imagination of a concept and the communication of the imagined concept. To do so, we focus primarily on the context of computer-supported design ideation, i.e. the generation of ideas in the form of virtual artifacts towards designing a product form. In particular, we are interested in spatial design ideation – a term that, in this thesis, will assume two different meanings. In its first form (*spatial design* ideation), it refers to the generation of ideas in the form of three-dimensional shapes. In its second form (*spatial design ideation*), it refers to the spatial nature of the interaction between the designer and the computer towards the generation of ideas in the design process.

1.2 The Motivation for Spatiality

The primary motivation behind this work directly comes from the statement made by Sproull [13]: “*Only the determined model three dimensional objects and they rarely invent a shape at a computer, but only record a shape so that analysis and modeling can proceed. The grand challenges in three dimensional modeling are to make simple modeling easy and complex modeling accessible to far more people.*”.

Scali et. al [14] note, “*highly digitized industrial processes tend to overlook the role of the designer in the process itself*”. We find that computer support for ideation has received very little attention in existing literature. Tools for 3D design are not suitable for ideation since they do not embody the notion of *controlled vagueness* [15] that is central to the process of idea generation. In particular, the interfaces of computer-aided design (CAD) software lack the spontaneity crucial to quick externalization of visual ideas [16]. Further, the instructions and training are primarily dedicated towards learning the usage of a modeling tool rather than focusing on the process of design itself. As a result, CAD tools end up supporting the creation of sophisticated artifacts *once the designer has learned the*

usage of the modeling tool. The amount of time spent in merely familiarizing oneself with the tool digresses the designer's attention from the design activity. Having said that, merely catering to designers, i.e. individuals who have a professional stake in the activity of design, is only a part of Sproull's vision. Facilitating the process of design for anyone that wishes to externalize visual ideas is the bigger challenge. It is in response to this challenge that we invoke the need for spatiality, not just in terms of *what* is created at the end of a design process, but more so *how* it is created during the process.

As Michael Polanyi states [17, p.56]: "*The body is the ultimate instrument of all our external knowledge, whether intellectual or practical experience [is] always in terms of the world to which we are attending from our body.*" This is the fundamental insight that we intend to embed in the design of interfaces for idea generation. Specifically, we identify that proprioceptive and kinesthetic control come naturally to humans in any physical interaction and are the key ingredients towards the flow of the design processes that involve human action and perception [18, 19]. In contrast to this, we find that most current geometric design processes are compartmentalized into disconnected sub-processes that do not allow the user to invoke their internal conceptual models. For example, sketch-based design approaches force the user to think in a projected (2D) version of the actual modeling (3D) workspace.

Klemmer et al. [20] state: "*One of the most powerful human capabilities relevant to designers is the intimate incorporation of an artifact into bodily practice to the point where people perceive that artifact as an extension of themselves; they act through it rather than on it*". Systems such as *Spatial Sketch* [21] and *Proto-TAI* [22] are examples of embodied approaches towards the creation of physical artifacts via bodily movement. This is what motivates our interest in spatial interactions wherein the idea is to reduce the barrier to entry in design by integrating our innate human ability and dexterity in virtual shape conceptualization [23]. Drawing from these works, we argue that enabling the direct externalization of spatial design concepts can be effectively achieved by embedding the geometric representation of the artifact within the physicality of the creation process itself.

1.3 Embodiment & Embodied Interactions

The notion of embodied interactions is an abstract one and requires explanation as to why it matters in spatial design ideation. The concept of embodied interactions was proposed by Paul Dourish [17] as a philosophical underpinning for approaching problems in human-computer interactions (HCI). One of the central ideas here was to formulate an integrated research framework inspired by tangible and social computing. To this end, Dourish described embodiment as a “*denotation of participative status rather than mere physical reality*”.

The central premise on which this thesis is built is that the very definition of embodied interactions, as a manifestation of phenomenological tradition, provides a natural basis for viewing design practice as bodily practice that allows one to communicate internal knowledge to the external world through the interactive media.

Within cognitive science, embodied cognition examines the ways in which our interactions with the physical world shape our cognitive experiences from a body-centric point of view [18, 20]. More specifically, embodied cognition holds that our cognitive processes are “*deeply rooted in the body’s interactions with the world*” [24]. This is in stark contrast to decades of research in cognitive science wherein the mind was viewed as a sort of central but detached information processing unit where motor-sensory functions were more-or-less secondary inputs and outputs to a main system [25]. Although there are many tenets of this body-centric view, the primary conclusion relevant to this thesis is that spatial interactions can shape, clarify, and reinforce our cognitive processes, leading to quick learning by novice users. We posit that the key to making digital design accessible to the novice untrained user is to provide a framework that takes advantage of the user’s prior knowledge of the real world and embeds that knowledge into the design tool itself.

1.4 Overview

So far, we have argued in favor of spatiality of interactions for the purpose of facilitating ideation of shapes. It is evident from past research that the proposition of spatial

interactions for designing virtual artifacts is not, in itself, novel. However, the reason why spatial design ideation is still a relevant and important research topic is that it still poses a myriad variety of challenges.

In the context of creative endeavors such as design, Mr. Adams' metaphorical radio is indeed here and is in fact hard to operate. There are two fundamental causes responsible for this difficulty. As Mr. Adams so eloquently puts it: "*wave your hand in the general direction of the components and hope*". This is essentially the problem of ambiguity in recognizing the intent. He further adds: "*you had to sit infuriatingly still if you wanted to keep listening to the same program*". This is the problem of lack of controllability. Though these issues must be resolved in any effort made towards computer-supported ideation, they become much more pronounced when considering spatial interactions.

Converting user input into a meaningful shape modeling task involves (a) acquisition, segmentation and processing of hand data, (b) extracting a virtual representation of the hand, (c) mapping the gestures to a shape modeling tasks, (d) modeling an appropriate response of the shape as intended by the user's input. Simply deciding when to start and stop a modeling task is an issue, unlike interacting with a tablet or a mouse, where touch or a button press can unambiguously trigger events. The challenge lies in the recognition and subsequently the interpretation of human action into meaningful design intent of the user so as to allow fluid and robust interactions for shape ideation. Below, we will describe a conceptual organization of this thesis with regards to how we will attempt this challenge by borrowing from the principles of embodied interactions.

1.4.1 Conceptual Organization

There are two fundamental components of embodiment that are of direct and critical relevance to this thesis. These are *action* and *meaning*. We are particularly interested in two derivative concepts, *intentionality* and *coupling*. In the framework of embodied interactions, intentionality helps define how an action is related to meaning and coupling

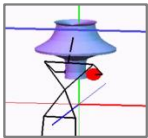

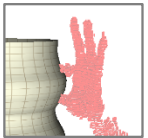
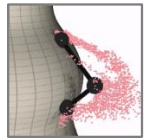

Approach	Prototype Application	Technical Overview
Symbolic	  <i>HandyPotter</i> <i>Shape-It-Up</i>	Input: Hand Gesture Modeling Operations: Creation, Deformation Shape Representation: Intelligent Generalized Cylinders Modeling Context: General Shape Modeling, Pottery Evaluation Method: Field study
Geometric	  <i>Prox. Attraction</i> <i>Grasp+Motion</i>	Input: Point Cloud Modeling Operation: Deformation Shape Representation: Simple Homogeneous Gen. Cyl. Modeling Context: Pottery Evaluation Methods: Quantitative, User Experience
Tangible	 <i>MobiSweep</i>	Input: Multi-touch + Tilt Modeling Operation: Creation, Deformation, Manipulation Shape Representation: Cubic-hermite sweep surfaces Modeling Context: 3D Composition Evaluation Methods: Usability study, Creativity study

Figure 1.1. Conceptual organization of the thesis.

can be understood as the chain of events that makes this relationship between action and meaning effective.

In our context, *design intent* is closely linked to *intentionality* that essentially sets up the relationship between what the user is doing and what they think is the intended outcome of their action. To explore the expression of this design intent, the approach we take is to investigate three different types of couplings between human action and the meaning that emerges from the application of these couplings. Specifically, the three couplings manifest in the form of symbolic, geometric, and tangible approaches for ideation (Figure 1.1). Below we summarize the contributions made in this work through the investigation of these approaches for design ideation.

1.4.2 Summary of Contributions

The notions of embodied cognition and embodied interactions form the philosophical underpinnings of this work. These notions are abstract and only help in defining the broader value of this work. However, in order to investigate how these concepts can be applied to real interaction scenarios, we take a concrete application-oriented approach wherein we attempt to understand spatial interactions for ideation through the design, development, and evaluation of computer interfaces. As described below, the core contributions of this work emerge from the evaluation and analysis of these interfaces that embody the symbolic, geometric, and tangible approaches for ideation.

1.4.2.1 Symbolic Approach

- As the first step towards mid-air shape design, we demonstrate direct creation of 3D shapes through *HandyPotter* (Chapter 2), a prototype application for the creation and modification of rotationally symmetric shapes. The core contribution here is the association between symmetric bi-manual movement of the hands and the creation of rotationally symmetric cylindrical shapes.
- We introduce *shape-gesture-context interplay* (Chapter 3), a framework which unifies the shape exploration process with the gestural intent of the designer. This framework enables the automatic deduction of the nature and extent of geometric constraints by interpreting human gestures and motions. Through this we develop a fundamental theoretical framework wherein the representation of a shape can be tied up seamlessly to the interactions possible on the shape.
- We demonstrate free-form modifications of shapes with gestures through two applications *Shape-It-Up*. Here, we develop the idea of *intelligent generalized cylinders* (IGC) as a shape representation which inherently integrates the contextual interactions induced by the postures and motion of the hands. We demonstrate the creation of IGCs in the context of natural human shape expression. We use the representa-

tions to enable quick creation of a variety of constrained and free-form shapes while retaining the aesthetic characteristics of the shapes.

1.4.2.2 Geometric Approach

- We introduce *proximal attraction* (Chapter 4), a method that directly uses the geometric information encoded in the modeling context and the manipulating object such as a hand or a tool. We develop a virtual pottery application as a concrete demonstration of proximal attraction. Our method uses a simple point-cloud (PCL) based representation of the hands without the need for gesture recognition or hand skeleton tracking. Thus, a user does not need to learn, know or remember any gestures to interact with our system; the user can directly interact to shape pots. Our approach allows for several types of user inputs namely, (a) hands, (b) physical objects as tools and (b) virtual tools.
- We design a novel algorithm to determine user intent for shaping through proximal attraction. Our algorithm, which is a combination of *adaptive exponential smoothing* with *selective Laplacian smoothing*, is particularly robust to unintentional or accidental user inputs. We extensively study the key parameters of our algorithm and provide guidelines to design pottery based interactions for controllability.
- As an extension of proximal attraction, we develop a method that uses the kernel-density estimate (KDE) of the hand's PCL to extract the grasp and motion for deforming the shape of a pot in 3D space (Chapter 5). We present the complete evolution of our algorithm in three stages of iterative design . At the end of each stage, we describe a user evaluation that informs the algorithm development of the subsequent stage. Second, we evaluate our KDE based approach in comparison to proximal attraction. Our evaluations help reveal two core aspects of mid-air interactions for shape deformation, namely, intent & controllability. We characterize user behavior in pottery design in terms of (a) common hand & finger movement patterns for cre-

ating common geometric features, (b) user perception of intent, and (c) engagement, utility, and ease of learning provided by our approach.

1.4.2.3 Tangible Approach

- We explore quick 3D shape composition during early-phase spatial design ideation (Chapter 6). Our approach is to re-purpose a smartphone as a hand-held *reference plane* for creating, modifying, and manipulating 3D sweep surfaces. We implement *MobiSweep*, a prototype application to explore a new design space of constrained spatial interactions that combine direct orientation control with indirect position control via well-established multi-touch gestures. *MobiSweep* leverages kinesthetically aware interactions for the creation of a sweep surface without explicit position tracking. The design concepts generated by users, in conjunction with their feedback, demonstrate the potential of such interactions in enabling spatial ideation.

2. SYMMETRIC BI-MANUAL INTERACTIONS FOR MID-AIR CREATION OF ROTATIONALLY SYMMETRIC SHAPES

In this chapter, we begin with the idea that the expression of rotationally symmetric 3D shapes can be captured effectively using the motion of the hands in 3D space. In doing so, we take the first step towards the development of a symbolic approach for spatial interactions for the creation, modification, and manipulation of 3D shapes. The main goal of this chapter is to demonstrate high-speed creation of shapes which are otherwise difficult to model using existing commercial CAD tools. To this end, we will demonstrate a *pottery-based* shape modeling metaphor for creation of generalized cylinders through bi-manual human motion.

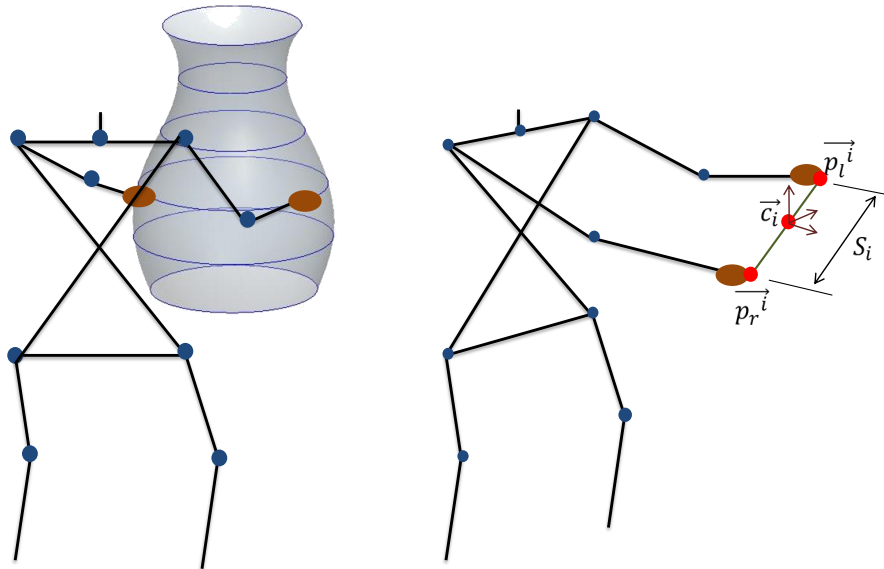


Figure 2.1. Overview of *HandyPotter*: shape modeling metaphor inspired by pottery (left) and interpretation of hand locations for shape creation (right).

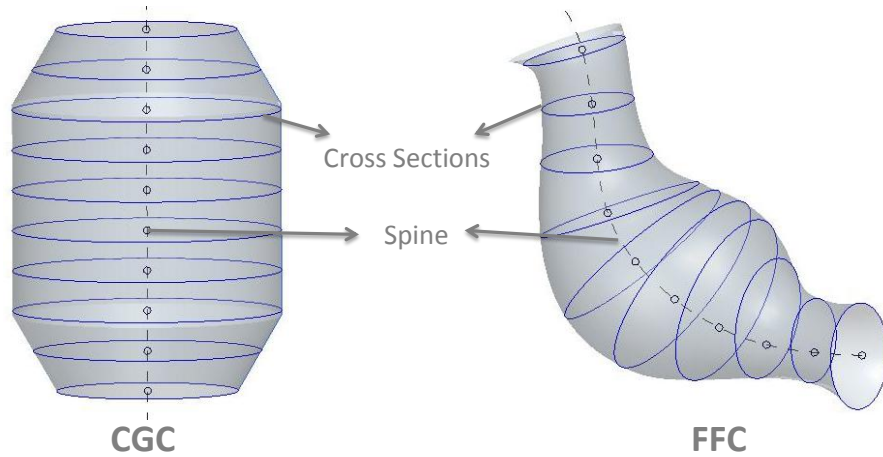


Figure 2.2. Constrained and free form generalized cylinders.

2.1 Overview

Motion is the temporal variation of the spatial configurations of the hand and the arm, i.e. it deals with the dynamic articulations of the hand and arm under kinematic constraints. In the context of this chapter, we constrain the definition of a gesture as the interpretation of hand and arm movements to convey a certain meaningful shape. A typical sweep operation, for instance, can be succinctly defined as a physical process of sweeping a curved piece of wire in 3D space. To this end, we use the *pottery* metaphor for the creation and modeling of sweep surfaces wherein the user moves both hands defining a side profile of the swept surface (Figure 2.1). Our approach, thus, results in an intuitive association of human motion with generalized cylinders. We use this association to enable quick creation of a variety of constrained and free-form shapes while retaining the aesthetic characteristics of the resulting shapes. We categorically show that a wide variety of shapes can be created in a matter of a few seconds.

In HandyPotter, we are particularly interested in the creation of generalized cylinders (GC's). Given a 2D cross-sectional curve and a 3D skeleton curve, [26] defines a GC as the sweep surface of the cross-sectional curve moving along the skeleton curve. The cross-sectional curve may change its shape dynamically. However, in most conventional

methods, the cross-sectional plane is restricted to be orthogonal to the tangent direction of the skeleton curve. GC's have been well-studied in literature since they have the combined potential of being able to represent fairly complex and free-form shapes and still retaining their parametric nature. In a discrete setting, we define a GC as a surface created by a sequence of sections represented as planar poly-lines along a 3D trajectory or *spine* represented as 3D poly-lines. A given GC can be described by the position, orientation and size of each of its cross-sections. We use the locations of the hands, \vec{p}_r^i (right) and \vec{p}_l^i (left) at a given i^{th} instance to evaluate these three parameters. Given a cross-section, the distance between the hands, the orientation of the line-segment joining the two hands and its mid-point, specify the size, orientation and position of the cross-section respectively (Figure 2.1). Thus, the temporal variations of the locations of the two hands in 3D space completely define the evolution of the GC for a given cross-sectional shape. Given a planar cross-section C which tightly fits within a bounding square of unit-length, we represent a GC, G , as a set of scaling factors, s_i , their corresponding center locations, \vec{c}_i and orientations represented as 3×3 rotation matrix θ_i (Equation (2.1)).

$$G = \{C_i | C_i = s_i \theta_i C + \vec{c}_i, 1 \leq i \leq n\} \quad (2.1)$$

$$s_i = \|\vec{p}_r^i - \vec{p}_l^i\|_2$$

$$\vec{c}_i = \frac{\vec{p}_r^i + \vec{p}_l^i}{2}$$

$$\theta_i = Rot(\alpha_i, \beta_i)$$

Here we confine to closed poly-line sections and open poly-line spines and we consider the creation of *constrained generalized cylinder* (CGC) and *free-form generalized cylinder* (FFC) (Figure 2.2). We envisage a CGC as a surface being created by motion of two hands along a *straight spine* wherein a cross-section varies in size, retaining a constant orientation and shape. On the other hand, creating an FFC is enabled by additional capabilities of defining the orientation of each section dynamically and *curvilinear spines*. Thus, from a representational standpoint, the only distinction between CGC and FFC is the level of

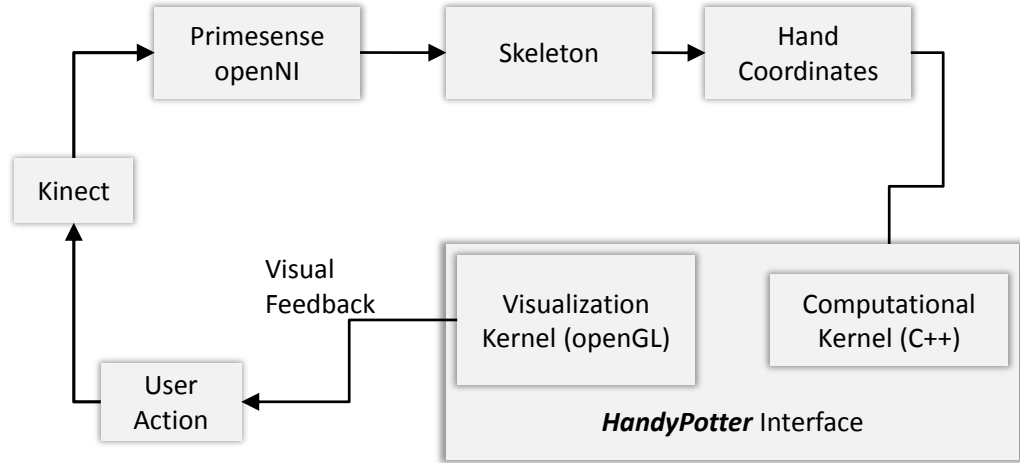


Figure 2.3. Computational pipeline for *HandyPotter*.

controllability of s_i , \vec{c}_i and θ_i . We implement this controllability by defining *snapping* of these parameters within prescribed limits with respect to the extent of the motions of the user.

2.2 Implementation

To demonstrate the shape exploration framework with the pottery metaphor, we developed a modeling tool to support the creation of GC's. In our current prototypical implementation we assume the the shape of the cross-section to be given a-priori. We particularly experimented with circular, square and hexagonal cross-sections to create shape models. The *HandyPotter* prototype was developed in C++ using OpenGL rendering. We use an off-the-shelf Kinect camera in conjunction with the skeletal tracking capability provided by the openNI library to track the hand locations of the user. The pipeline for the implementation is shown in Figure 2.3. The interface developed in this tool is very simple in that the user only sees the global frame of reference and a working volume wherein the user can create the GC's. We define a working volume at a specified distance from the global coordinate frame (Figure 2.4). When both the user's hands are within this working

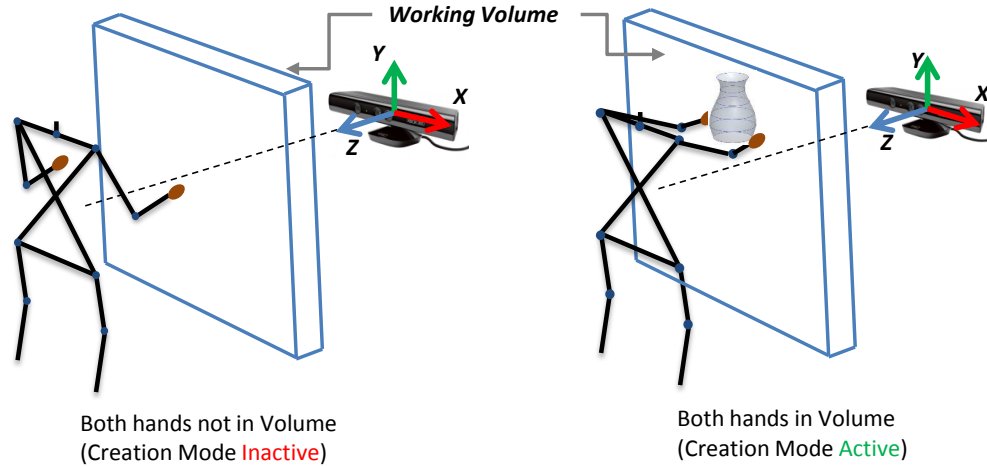


Figure 2.4. Virtual working volume for sweep creation.

volume, the modeling tool starts creating the generalized cylinders described in the previous section. Taking into account the spatial extent within which a user would typically feel comfortable working, we define the working volume at a distance of 1.5 m away from the depth sensor and we define the thickness of the working volume to be 30 cm which is about double the size of the palm. Figure 2.5 shows a typical interactive session using the interface developed.

2.3 Results

Figure 2.6 shows general shapes corresponding to different constraints for the creation of GC's. We show three varieties of GC's, (a) fully constrained, wherein the scale of the GC changes with the hand motions while the spine is kept vertical and orientational variations of cross-sections is not allowed, (b) partially constrained wherein the spine can take a free-form on 3D space while the orientation is still constrained and (c) a completely free-form mode wherein the spine, orientation and the section scales can be varied as per the wish of the user. We were able to create these shapes within 2 to 3 seconds which is a significantly low modeling time. However, a better understanding of 3D interfaces for shape modeling

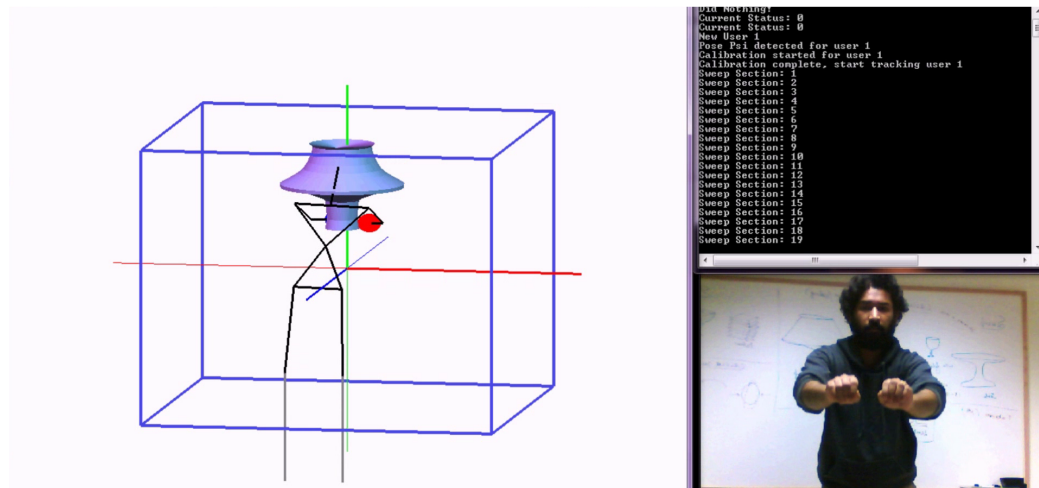


Figure 2.5. A snapshot of the interface for *HandyPotter* showing the user creating a shape using two-handed motion.

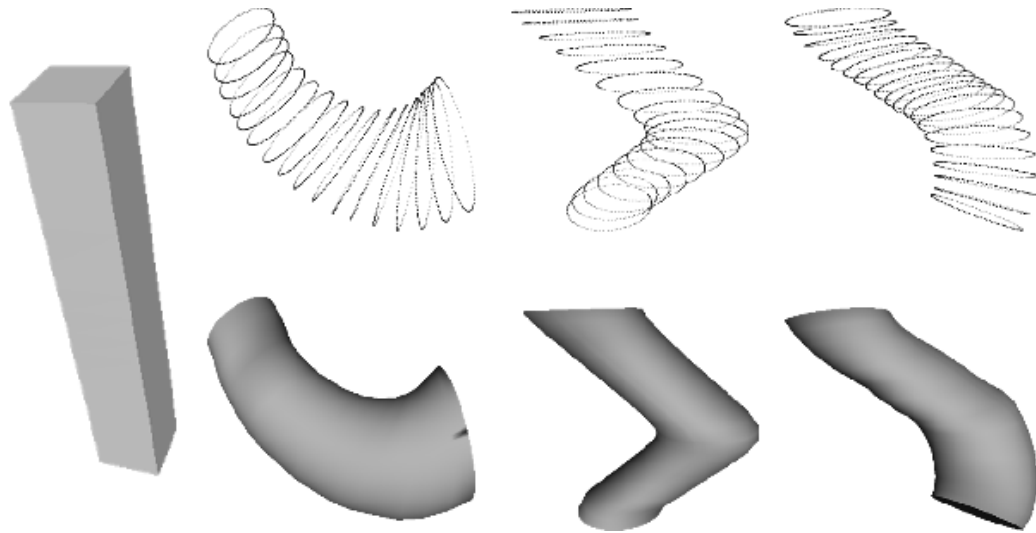


Figure 2.6. Fully constrained, partially constrained and free-form generalized cylinders using *HandyPotter*.

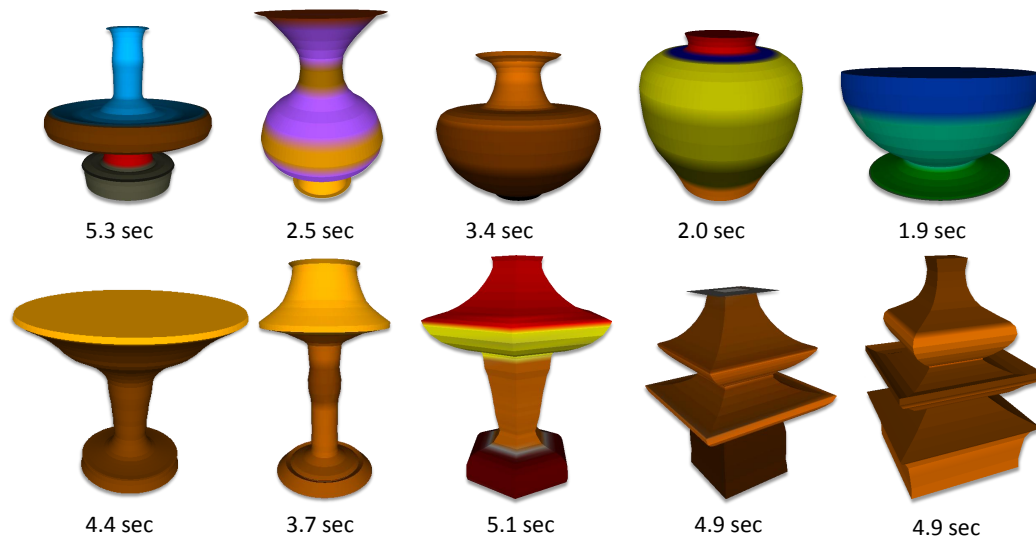


Figure 2.7. Pot-like shapes modeled using *HandyPotter*.

warrants a detailed user-study involving the performance of our system and its comparison with traditional CAD tools.

Figure 2.7 shows some examples of shapes and the corresponding time taken to model each shape. Let us recall that our primary goal was to enable the designer to externalize

rough design prototypes for the purposes of exploration in the early phases of design. Thus, it can be appreciated that the description of shapes in the examples shown, involves only the interpretation of what the designers want to create through their own body motion.

2.4 Conclusions

In this chapter, we presented the idea of pottery-based shape modeling metaphor which enables rapid creation of generalized cylinders purely through arm motion. Here, our goal was mainly to explore, through a minimalistic interface, how action could be transformed into design intent. Thus, *HandyPotter* is a rudimentary effort leading to the symbolic approach for spatial design ideation. It merely offers a glimpse of the spatial interactions that can be enabled by depth cameras for 3D shape exploration. In the following chapter we will generalize the idea behind *HandyPotter* towards developing a general framework for gesture-based interactions for idea generation.

3. GESTURE-BASED MID-AIR INTERACTIONS FOR DESIGN IDEATION

Theories on the use of hand gestures in CAD, have been developed by Horváth [27]. To this end, Horváth also investigated hand motion language for shape conceptualization [6]. Pavlovic et al. [28] illustrates the components of the human gesture interpretation process namely, video input, analysis and recognition supported by a mathematical representation of gestures and finally gesture description for further actions. A review of VR-based assembly and prototyping [29] states that “*the ultimate goal is to provide an invisible interface that allows the user to interact with the virtual environment as they would with the real world*”. However, VR-based technologies are typically more suitable for post-design phases and are less affordable in terms of cost and setup-time. Although we inspire our approaches along the same lines, we do it primarily in the context of the early design phases where iteration of design prototypes necessitates the provision of an *affordable and non-intrusive environment* which can support exploratory thinking amongst designers.

In this chapter, we begin with the first goal of this thesis: investigating and formalizing the symbolic approach for quick generation of 3D shapes. To do so, we present a framework dubbed *shape-gesture-context interplay* (SGC-I), that for the expression of 3D shapes through the naturalistic integration of human hand gestures. Our goal is to develop

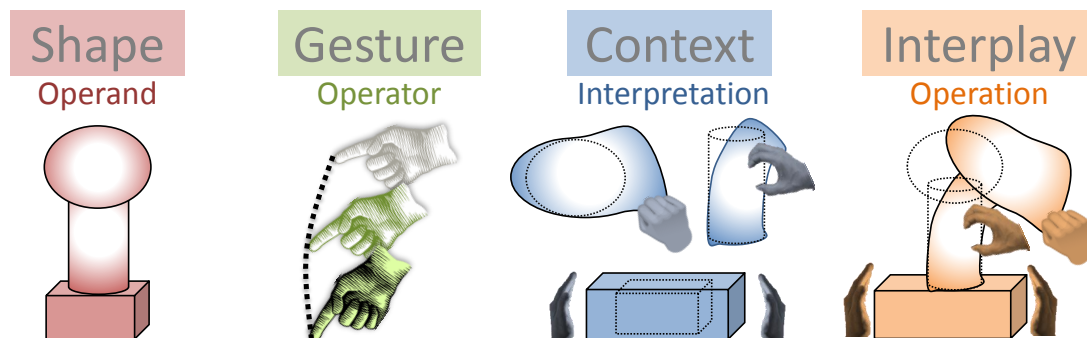


Figure 3.1. An overview of Shape-Gesture-Context Interplay.

a symbolic approach for transforming action into intent. interface . We demonstrate a concrete instance of SGC-I with a prototype interface, *Shape-It-Up*, to facilitate creation, modification and manipulation of shapes through mid-air interactions.

3.1 Related Work

3.1.1 Mid-Air Shape Modeling

Segen and Kumar [30] showed examples of computer-aided design (CAD) with their *Gesture VR* system, which used non-intrusive methods based on computer vision for general virtual reality (VR) applications. Schkolne et al. [31] demonstrated the modeling of organic shapes using a glove-based wearable device. Llamas et al. [32] describe *Twister*, a method for producing intuitive deformations of surfaces using bimanual interactions with two magnetic 6 degree-of-freedom (DOF) hand trackers to determine location, orientation, and pose of hand. Keefe et al. [33] presented a tangent-preserving method for a controllable illustration of 3D lines using the *PHANToM* haptic device in conjunction with a 6 DOF tracker for bi-manual input. More recently, augmented reality interfaces for 3D digital content creation were shown by [34]. Wang et al. [35] presented *6D Hands* to demonstrate computer-aided design (CAD) using marker-less hand tracking. Holz and Wilson [36] proposed *Data miming* as an approach towards descriptive shape modeling wherein a voxel representations of a user's hand motion is used to deduce the shape which the user is describing.

3.1.2 Gesture Recognition & Hand Skeletal Tracking

The pose of the human hand can be described as the spatial configuration of each finger and the palm, i.e. the kinematic parameters (joint angles) of the hand. The existing algorithms are by and large based on 2D images obtained from single or multiple cameras. Erol et al. [37] classify computer-vision based pose estimation algorithms in two broad categories (a) single frame algorithms and (b) model-based tracking algorithms. There

is however an overlap of approaches followed in these categories. Feature-based algorithms involve extraction of relevant features of the hand like silhouettes and boundary contours [38–42]. The feature extraction is typically followed by a global search amongst pose samples, template matching or inverse-kinematics. Other methods [43] use deformable geometric models of the hand to estimate poses. Model-based learning [44, 45] and feature-tracking [46, 47] based algorithms involve the tracking of the hand and subsequent estimation of the joint angles using single or multiple hypothesis. The commonly used approaches are based on Bayesian filtering [48], particle filters and grid based filters. With the advent of low-cost depth cameras very recently, only a couple of papers algorithms have been proposed for hand-pose estimation. Work in [49] used compressed 3D shape descriptors using the depth data from Zcam for 3D hand pose recovery. Oikonomidis et al. [50] showed a robust hand pose estimation based on particle swarm optimization on the GPU using depth images from the Kinect.

The kinematic representation of hands has been employed in many of the model-based approaches mentioned above [38, 39, 44–47, 50]. Different degrees-of-freedom (DOF) models have been used in literature based on the methods applied for the pose estimation algorithm. Thus, these hand representations are more application-specific. Hand representation and kinematics has been widely studied in robotics [51], ergonomics, and digital human modeling literature. Cobos et al. [52] presented an efficient 24-DOF kinematic model of the hand suited for simulation of manipulation tasks. Like other joints of the human body, joints on the human hands are coupled with each other such that their motion is constrained by the motion of the neighboring joints. Work presented in [53] illustrates the joint constraints in the kinematic hand model and quantify these constraints in terms of the relationship between joint angles. A detailed analysis given by van Nierop et al. [54], discusses a natural human hand model with 12 DOF from a bio-mechanical perspective. Besides kinematics models, Ingram et al. [55] show interesting statistical analysis of the joint velocity analysis. Hand representation and kinematics has also been widely studied in ergonomics [53] and digital human modeling literature [56, 57]. A concise summary of

grasp taxonomies is given by Feix et al. [58]. Work in by Rusak et al. [59] demonstrates interactive grasping simulation of product concepts in a VR environments.

3.1.3 A Brief Review of Geometric Modeling for Design

The representation of parametric shapes has been studied extensively in CAD literature. A comprehensive description of these representations namely, (a) decomposition models, (b) constructive models and (c) boundary models, can be found in [60]. Boundary representations (commonly called B-rep's), especially polyhedral models are known to be directly useful for graphical applications [60]. Parametric surface modeling makes use of Bezier, B-Splines and NURBS for modeling complex shapes [61]. Shape representations based on contours and mesh-regions have been explored in recent literature [62], for search, classification, reconstruction and processing operations like segmentation. Lipman et al. [63] presented a differential coordinate representation for surface editing.

Shape deformation methods can be grouped into two broad categories, space based and surface based. In surface-based methods [64, 65] the initial and target positions for a subset of surface points are used to compute the positions of all remaining points. Cage methods create a cell complex around an original shape and perform the deformation on the cell complex with the deformed position of all points calculated with respect to each cell [66–68]. Skeletal methods [69, 70] constrain the deformation of a surface so that properties of the skeleton are preserved. Feature based methods [67, 71–73] use either automatic or user chosen features typically represented as a contour or area where the deformation is constrained. Space based methods [73, 74, 74, 75, 75–78] modify a shape by deforming the embedded space in some fashion.

Generalized cylinders (GC) have been extensively studied in CAD and shape modeling literature [26, 79, 80] and several perspectives have been discussed towards their representation. Chang et al. [26] defines a GC as the sweep surface of the cross-sectional curve moving along the skeletal curve. The cross-sectional curve may change its shape along the skeleton. However, in most existing methods, the cross-sectional plane is typically de-

finned by the normal and bi-normal on the skeletal curve. Work presented in [79] describes sweep surfaces as parallel, rotational, spined and synchronized based on the primitives and the sweeping rules. Special representations towards interactive deformation of GC's were presented in [26, 81]. More recently, [80] presented a direction map based representation of GC's which was particularly congenial for blending amongst the swept cross-sections. Deformation of generalized sweeps have also been investigated extensively in [82, 83], towards applications in human deformation.

Works by Igarashi et al. [84] and Nealon et al. [85] illustrate strategies for creating 3D shapes from 2D profile sketches. Work presented in [86–88] shows methods for data-driven exploratory design to motivate suggestion-based modeling by leveraging on our current prowess in shape-search. Literature in synthetic modeling [89–92] demonstrates the capability of creating large complex 3D scenes from simple building blocks.

3.2 Overview

The SGC-I framework enables a set of generic and systematic tools wherein the shape is considered as the *operand*, gestures as the *operators*, contexts as activity *interpreters* and interplay as the whole operation (Figure 3.1). This framework enables the automatic deduction of the nature and extent of geometric constraints by interpreting human gestures and motions.

We categorize the shape exploration process into three distinct components, namely, (a) shape creation, (b) shape modification and (c) shape manipulation (Figure 3.2). By shape creation, we mean the use of hand gestures to create a shape in an empty working volume. Shape modification refers to interactions with shapes with the intention of changing the geometric characteristics of the shape. Shape manipulation refers to the rigid-body transformations for translating, rotating and scaling 3D shapes. The following sub-sections give a detailed description of our technical approach towards the shape exploration process. Work in [6] presented the conceptual and technical descriptions of hand-based interaction systems. In our work, we define hand gestures as a combination of the posture of the

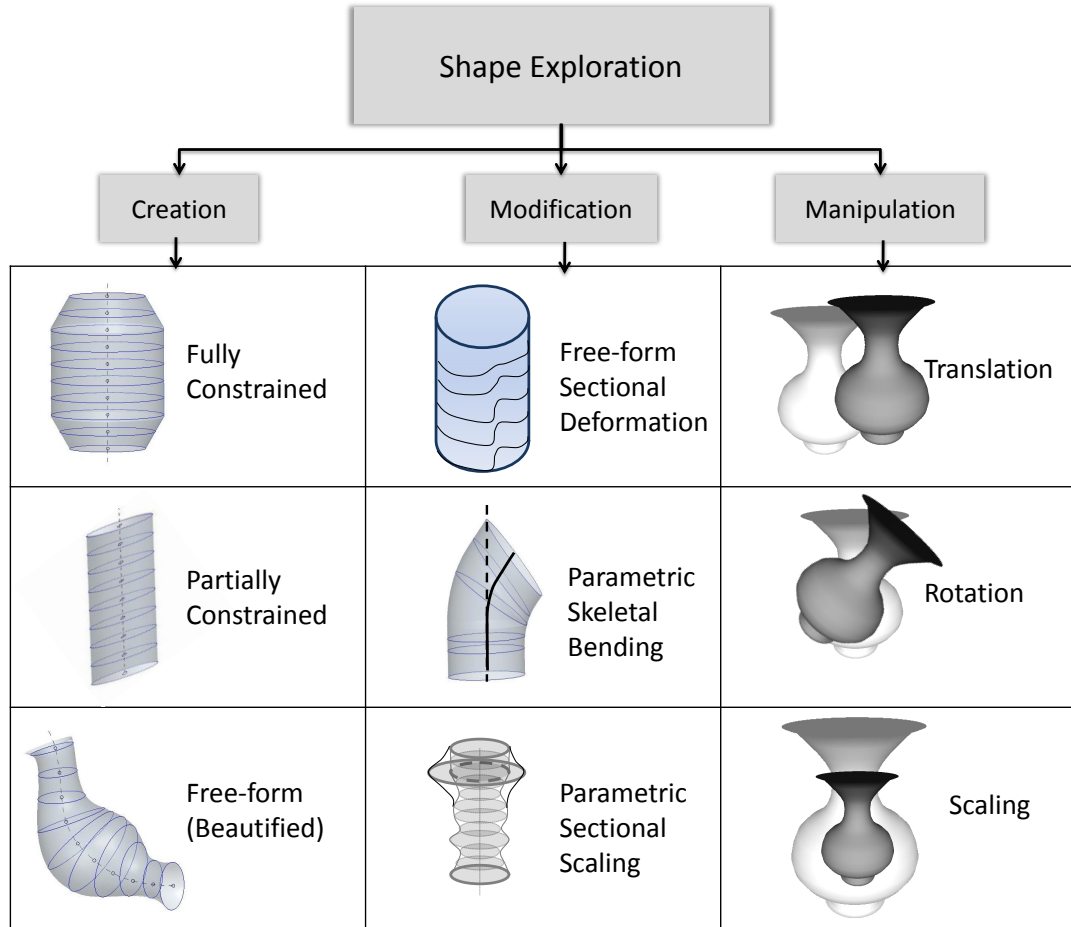


Figure 3.2. Components of the proposed shape exploration paradigm.

hands and the 3D motion of the arms. Most importantly, we encapsulate the three shape exploration components in a unified mathematical framework, SGC-I and demonstrate its strength for mid-air applications.

3.3 Shape-Gesture-Context Interplay

In this section, we present the mathematical representations for the components of the SGC-I paradigm, namely *shape*, *gesture* and *context*. We represent shapes as generalized cylinders (GC) in a way such that naturalistic modifications can be enabled for hand-based interactions. The representation of gestures is sub-divided into postures and motion.

3.3.1 Gesture

Given a finite set of k postures, we define a hand posture as an positive integer $\sigma \in \{0, 1, \dots, k - 1\}$. Additionally, we represent the motions of a hand as the instantaneous locations $\vec{\omega}$ of the center of the hand. We define a gesture Γ as a function of time t (Equation (3.1)) which is represented by an ordered pair of the postures $\sigma(t)$ (Equation (3.2)) and locations $\omega(t)$ (Equation (3.3)) of the left and right hands respectively.

$$\Gamma(t) = (\sigma(t), \vec{\omega}(t)) \quad (3.1)$$

$$\sigma(t) = (\sigma_l(t), \sigma_r(t)) \quad (3.2)$$

$$\sigma_l(t), \sigma_r(t) \in \{0, 1, \dots, k - 1\}$$

$$\omega(t) = (\vec{\omega}_l(t), \vec{\omega}_r(t)) \quad (3.3)$$

$$\vec{\omega}_l(t), \vec{\omega}_r(t) \in \mathbb{R}^3$$

In the given set of gestures, we declare $\sigma = 0$ as the *NULL* posture. In terms of interactions, this gesture plays the role of discontinuing an ongoing interaction in progress. In the future sections we will describe the representation advantage of this imposition.

3.3.2 Context

Before describing the formalization of contexts, we introduce two parameters, $\varepsilon_1 \in \{0, 1, \dots, k - 1\}$ and $\varepsilon_2 \in \mathbb{R}$, which we call the *operation parameters*. These parameters aid in the extract of designer's intent through the interpretation of gestures. The parameter ε_1 represents the posture classifier and ε_2 represents a threshold whose meaning changes

according to the context in which it is used. We define the context Υ_Γ through the boolean association (Equation (3.4)) of a set of modeling operations $\{\Sigma_1, \dots, \Sigma_r\}$, with a given a set of gestures $\{\Gamma_1, \dots, \Gamma_r\}$ by using a combination of the Kronecker delta ($\delta(a, b)$) and heaviside functions ($u(x)$). Here, d is a distance function defined in \mathbb{R}^3 .

$$\Upsilon_\Gamma(\varepsilon_1, \varepsilon_2; \Sigma) = \begin{cases} \Sigma & \text{if } \delta(\sigma, \varepsilon_1) = 1 \text{ \& } u(d - \varepsilon_2) = 1 \\ -\Sigma & \text{otherwise} \end{cases} \quad (3.4)$$

By appropriate selection of the operation parameters, we will be able to select amongst modeling operations like creation, modification and manipulation according the postures and locations of the hands.

3.3.3 Generalized Cylinder (GC)

While many representations have been discussed for GC's, these representations have been mainly envisaged in the typical interaction scenarios wherein the user-input is primarily two-dimensional. Our definition of the GC is inspired from the physical action of holding a sectional curve (say a wire-loop) and sweeping it in 3D space along a path.

A 3D curve approximated by a discrete poly-line of *resolution* (number of edges) “ n ” can be represented as a $3 \times n$ real matrix $C = [\vec{v}_1 \vec{v}_2 \dots \vec{v}_n]$ wherein each column $\vec{v}_i = [x_i \ y_i \ z_i]^T$ represents a point in \mathbb{R}^3 and the columns are *ordered* to define the edges of C . Any two curves defined in the manner described can be added together and multiplied with a scalar. In a geometric sense, the addition can be interpreted as a commutative deformation of either of the two curves with respect to the other. Interestingly, the addition of a curve C_2 with identical columns to another general curve C_1 would result in the translation of C_1 . Thus, in this representation, translation is a special case of deformation of a curve. Similarly, scalar multiplication would correspond to the scaling of the curve. For the purpose of generality, we do not distinguish between closed and open curves in this rep-

resentation. In our implementations, we explicitly specify the closure as per requirements without vertex duplication.

Given a set of 3D curves $\{C_1, C_2, \dots, C_m\}$ all with resolution n , the surface “ S ” of a GC is given by Equation (3.5). Here, $s_i \in \mathbb{R}$, $R_i \in M_{3 \times 3}(\mathbb{R})$ and $T_i \in M_{3 \times n}(\mathbb{R})$ represent the scaling, rotation and translation of the i^{th} cross-section C_i respectively. Here, $\Delta_i^C, \Delta_i^T \in M_{3 \times n}(\mathbb{R})$ represent the deformation of cross-section C_i and the skeleton T_i respectively. The order of cross-sections defines the development of the surface in a discrete sense. In the current section we assume Δ_i^C and Δ_i^T to be a *zero* matrix signifying that the cross-sectional curve is transformed as is and the trajectory is static.

$$S = \{K_i | K_i = s_i R_i (C_i + \Delta_i^C) + (T_i + \Delta_i^T), 1 \leq i \leq m\} \quad (3.5)$$

We define a right-handed global coordinate frame such that the y -axis is vertical and the $z - x$ plane is horizontal. If α_i and β_i be the angles of elevation (rotation about z -axis) and azimuth (rotation about y -axis) (Figure 3.3(a)) then the rotation $R_i = R(\alpha_i, \beta_i)$ be the rotation matrix. Similarly, the translation of the curve from the origin along a vector $\vec{o}_i = [o_{ix} \ o_{iy} \ o_{iz}]^T \in \mathbb{R}^3$ is given by Equation (3.6). As mentioned earlier, it is easily observable that the translation can be considered as a special deformation in $M_{3 \times 3}(\mathbb{R})$ given by the product of the diagonal matrix $D(\vec{o}_i) \in M_{3 \times 3}(\mathbb{R})$ and the *unit* matrix $U_{3 \times n} \in M_{3 \times n}(\mathbb{R})$.

$$T_i = D(\vec{o}_i) U_{3 \times n} = \begin{bmatrix} o_{ix} & 0 & 0 \\ 0 & o_{iy} & 0 \\ 0 & 0 & o_{iz} \end{bmatrix} \begin{bmatrix} 1 & 1 & \dots & \dots & 1 \\ 1 & 1 & \dots & \dots & 1 \\ 1 & 1 & \dots & \dots & 1 \end{bmatrix} \quad (3.6)$$

3.3.4 Sectional and Skeletal Deformations

Geometrically, the deformation of a curve can be seen as a set of translations on each point on the curve. Also, this interpretation holds irrespective whether a curve is closed

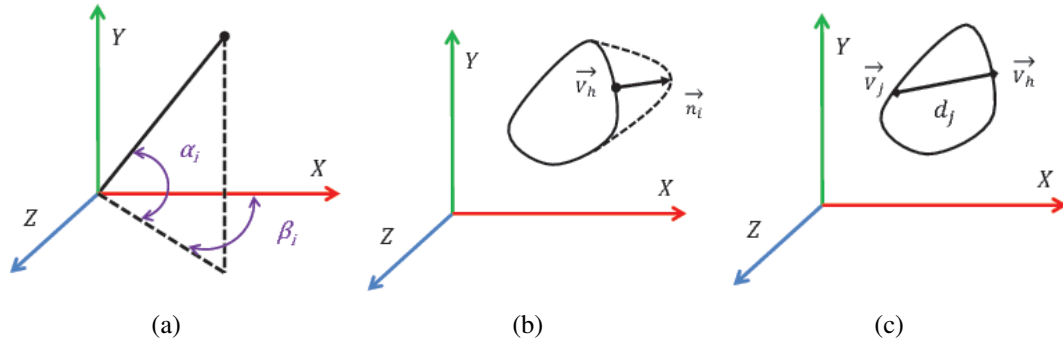


Figure 3.3. Section parameters (a) orientation (b) handle and deflection (c) handle-vertex distance.

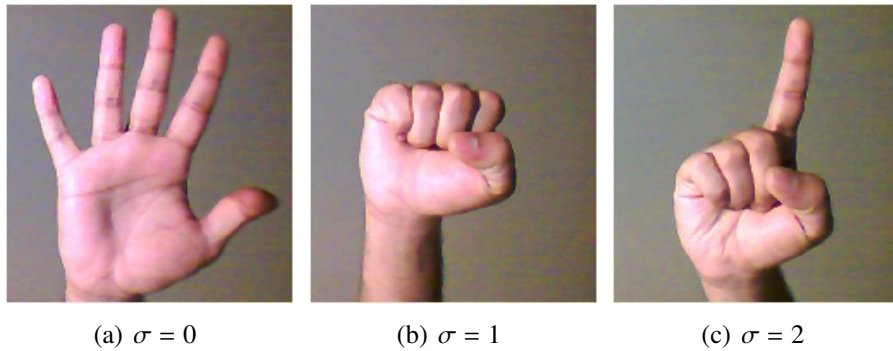


Figure 3.4. Selected hand postures (a) Release (b) Grab (c) Point.

or open. Thus, for a discrete curve of resolution n , the deformation can be described by a set of n translations, one for each point on the curve. We represent the deformation of a curve C using a simple matrix addition of the Δ curve. We present Δ as a *curve* simply because it has the same representational properties as the curve C , i.e. it has the same resolution as C and the columns of Δ are ordered. More precisely, there is a unique one-to-one correspondence between C and Δ . From interaction point-view, deformation requires a point on the curve which is being deflected (pulled or pushed) in a certain direction. In case of sectional deformation, we call this point the *handle* $\vec{v}_h \in C_i$ and the deflection is defined as a vector $\vec{\eta}_i = [\eta_{ix}, \eta_{iy}, \eta_{iz}]^T \in \mathbb{R}^3$ (Figure 3.3(b)). The remaining task is to transfer the deflection to the other points on the curve. For sectional deformations, we achieve this by defining a *deformation-transfer* function $f(d_j)$ which scales the deflection on the handle and applies to the j^{th} point \vec{v}_j on the curve based on the euclidean distance d_j between the handle and j^{th} point (Figure 3.3(c)). Thus, Δ_i can be formulated as follows:

$$\Delta_i^C = D(\vec{\eta}_i) U_{3 \times n} D(f(\vec{d})) \quad (3.7)$$

$$= \begin{bmatrix} \eta_{ix} & 0 & 0 \\ 0 & \eta_{iy} & 0 \\ 0 & 0 & \eta_{iz} \end{bmatrix} U_{3 \times n} \begin{bmatrix} f(d_1) & \cdots & \cdots & 0 \\ 0 & f(d_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & f(d_n) \end{bmatrix}$$

$$\vec{d} = [d_1, d_1, \dots, d_n]^T \quad (3.8)$$

$$d_j = \|\vec{v}_j - \vec{v}_h\|_2$$

In Equation (3.7), $D(\vec{\eta}) \in M_{3 \times 3}(\mathbb{R})$ is the diagonal matrix with components of deflection in the diagonal and $D(f(\vec{d})) \in M_{n \times n}(\mathbb{R})$ is a diagonal matrix with the deformation-transfer function evaluations on the diagonal.

Skeletal deformations could be treated in the same way as sectional ones. However, we present a novel technique which is more suitable for bending of skeletal curves that is not similar to the sectional deformation described above. Since our method is specific in its construct and application, we will describe it in one of the following sections. In the current section we will maintain the general description of deformation as the addition of a curve Δ_i^T to the curve T_i .

3.3.5 Intelligent Generalized Cylinder

The currently known representations of GC's would typically look similar to the one described in this chapter till this stage. Now we introduce the idea of the *intelligent generalized cylinder* (IGC). Shape modeling with sweep representations typically involves constraining the primitives defining the sweep surface. For instance, the representations given in [79] involve constraints of parallelism of section planes for parallel sweeps, intersection of section planes for rotational sweeps and tangency of sections with respect to the trajectory for spined sweeps. The section and skeleton were two primitives in these representations. In our scheme, we have four distinct primitives using which a GC can be developed. These are (a) sectional scale (s_i^*), (b) sectional rotation parameters (α_i^*, β_i^*) (c) sectional translation parameter (\vec{o}_i^*, T_i^*) and (d) sectional deformation (Δ_i^*). The challenge in our scenario is to model the IGC in such a way that the behavior of the GC's during creation, modification and manipulation is naturally associated with the gestures of the designer. Another important issue under consideration is that the jitter in the observed locations of the hands either due to noise in the input data or due to the inherent properties of the control of hand movements should not affect the shape modeling process adversely. The noises from the depth camera can be reduced by smoothing or computer-vision based post-processing of the human skeletal data. We currently smooth the coordinates of each of the joint locations using exponential smoothing. On the other hand, the control of hand movements and their effects on modeling could be a subject of research in its own right. Although studies in neuroscience [93] on human movement sciences can be augmented

into modeling interactions, they are not within the scope of this work. With this in consideration, we develop the IGC representation in a manner which is amenable to future developments involving refined behavioral patterns of human movement.

Before describing the IGC representation, we introduce a third parameter, $\varepsilon_3 \in \mathbb{R}$, which we call the *intelligence parameter*. This parameter, like ε_2 introduced earlier, represents a threshold whose meaning changes according to the context in which it is used. The definition of IGC involves two main modifications in the representation described in Equation (3.5). Firstly, we redefine the four primitives stated above as functions of the hand locations (Equation (3.9)). In the context of SGC-I, the functions F , G_1 , G_2 and H are intended to be based on the locations of the hands and will be defined in later sections. Secondly, we replace the unit matrix $U_{3 \times n}$ with a *heaviside* matrix $\Omega_{3 \times n}$ given by Equation (3.10). The function $u(d - \varepsilon_3)$ is the *heaviside* function.

$$\begin{aligned}
 s_i^* &= F(\omega(t_i), s_{i-1}^*) \\
 \alpha_i^* &= G_1(\omega(t_i), \omega(t_{i-1}), \alpha_{i-1}^*) \\
 \beta_i^* &= G_2(\omega(t_i), \omega(t_{i-1}), \beta_{i-1}^*) \\
 R_i^* &= R(\alpha_i^*, \beta_i^*) \\
 \vec{o}_i^* &= H(\omega(t_i)) \\
 T_i^* &= D(\vec{o}_i^*)\Omega_{3 \times n} \\
 \Delta_i^{C*} &= D(\vec{\eta}_i)\Omega_{3 \times n}D(f(\vec{d})) \\
 \Delta_i^{T*} &= J(\omega(t_i))
 \end{aligned} \tag{3.9}$$

$$\Omega_{3 \times n} = D(u(d - \varepsilon_3))U_{3 \times n} \tag{3.10a}$$

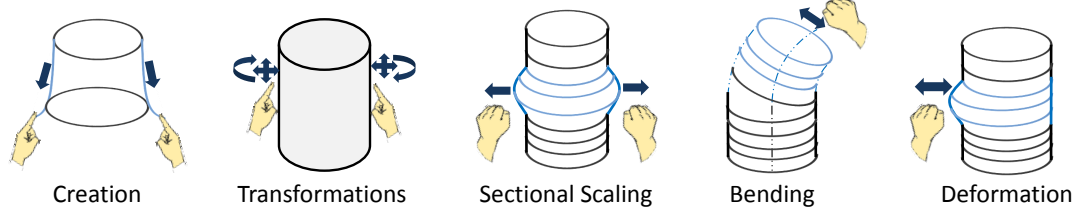


Figure 3.5. Physical interpretation of the SGC-I framework for IGC modeling.

$$D(u(d - \varepsilon_3)) = \begin{bmatrix} u(d_x - \varepsilon_{3x}) & 0 & 0 \\ 0 & u(d_y - \varepsilon_{3y}) & 0 \\ 0 & 0 & u(d_z - \varepsilon_{3z}) \end{bmatrix} \quad (3.10b)$$

The introduction of the heaviside function is the key factor which converts a GC to an IGC. By choosing appropriate values for the intelligence parameter we will be able to define the geometric constraints during the creation and modification operations using the hand locations. Thus, the surface S of the IGC is represented in terms of the hands as:

$$S^* = \{K_i^* | K_i^* = s_i^* R_i^* (C_i + \Delta_i^{C*}) + (T_i^* + \Delta_i^{T*}), 1 \leq i \leq m\} \quad (3.11)$$

Manipulation, being a global transformation, is trivially described as $s_i^G R^G(\alpha_i, \beta_i) S^* + T_i^G$, where s_i^G , $R^G(\alpha_i, \beta_i)$ and T_i^G are the global scale, rotation and translation respectively.

This section described the mathematical framework for SGC-I with IGC as the representation of the shapes that we have considered. In the following sections, we will explicitly provide functions for concretely describing the gestures and contexts for IGC's with examples and results.

3.4 Implementation Approach for SGC-I

In this section, we describe the gestures, operations and contexts which we have implemented in this work towards the creative modeling of IGC's. We consider three hand

postures, namely *release* ($\sigma = 0$), *grab* ($\sigma = 1$) and *point* ($\sigma = 2$) as shown in Figure 3.4. As mentioned earlier, the *release* posture is the *NULL* posture in this set since it naturally creates an affordance for leaving contact with objects. Further, we combine the selected gestures with hand locations to define a set of five contexts leading to five modeling operations $\{\Sigma_1, \dots, \Sigma_5\}$. A physical interpretation of these contexts and the corresponding operations are shown in Figure 3.5. The definition of functions leading to these operations can be prescribed by simply choosing appropriate operation parameters for the right (suffix *r*) and left (suffix *l*) hands (Table 3.1). We assume that the IGC creation process (Σ_1) takes place along the global *z*-axis and the distance function is given by the euclidean distance of the hands with respect to the torso (\vec{p}_{tor}). Thus, we associate a virtual slab which moves with the designer parallel to the vertical (*x* – *y*) plane.

In a naturalistic setup, bending typically takes place by holding a sufficiently slender shape with both hands. We simplify this process to a one hand operation by declaring the object stationary. Thus, we impose a special constraint on the bending operation that a skeleton can be bent only by grabbing the center of the top or the bottom section of an IGC. The distance functions in all other operations are defined between a general point the surface S^* of the IGC and the hand locations (Equation (3.12)). Here the symbols \wedge , \vee and $\underline{\vee}$ mean logical conjunction, logical disjunction and exclusive disjunction respectively.

$$d(\Sigma_i) = \begin{cases} \|\vec{\omega}_{r\wedge l}(t_i) - \vec{p}_{tor}\|_2 & \text{if } i = 1 \\ \arg \min_j (\|\vec{\omega}_{r\vee l}(t_i) - \vec{v}_j^{S^*}\|_2) & \text{if } i = 2 \\ \|\vec{\omega}_{r\vee l}(t_i) - \vec{o}_{1\vee m}^{S^*}\|_2 & \text{if } i = 4 \\ \arg \min_j (\|\vec{\omega}_{r\wedge l}(t_i) - \vec{v}_j^{S^*}\|_2) & \text{otherwise} \end{cases} \quad (3.12)$$

$$\vec{v}_j^{S^*} = j^{th} \text{ vertex of } S^*$$

$$m = \text{number of sections}$$

Table 3.1. Operation parameter definition for modeling operations.

Operation:	Create (Σ_1)	Deform (Σ_2)	Scale (Σ_3)	Bend (Σ_4)	Manipulate(Σ_5)
ε_{1r}	2	1(0)	1	1(0)	2
ε_{1l}	2	0(1)	1	0(1)	2

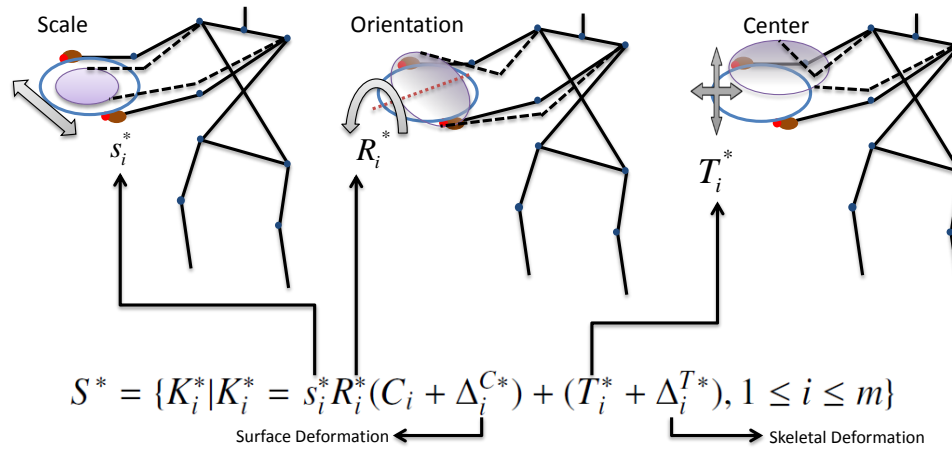
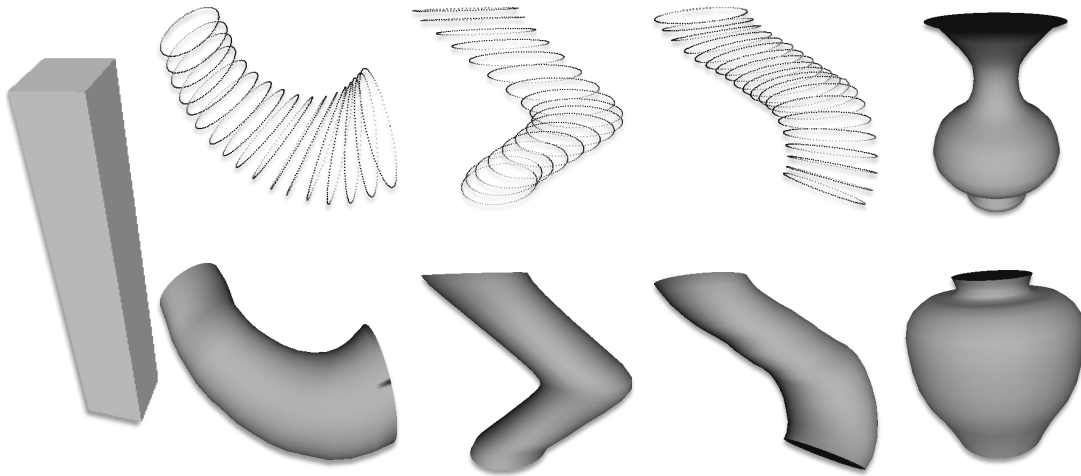


Figure 3.6. Mapping of hand motion is illustrated for the creation of IGC through scaling, rotation, translation of a given section of a swept surface.

Figure 3.7. Fully constrained, partially constrained and free-form IGC, orientational and translational snapping in swept surfaces. Note that this particular subset of shapes was shown in *HandyPotter* (Chapter 2).

3.5 IGC Creation

For simplicity and symmetry, we initialize a template cross-section, C , as a circle of radius 0.5 centered at the origin (Equation (3.13)) and lying on the horizontal ($z-x$) plane. In a discrete setting, this translates to a closed regular polygon with sufficiently large resolution. The advantage of this definition is its inherent capability to represent sharp polygonal cross-sections only by setting a low resolution.

$$C_i = \begin{bmatrix} 0.5 \cos(\theta_1) & \cdots & 0.5 \cos(\theta_m) \\ 0 & \cdots & 0 \\ 0.5 \sin(\theta_1) & \cdots & 0.5 \sin(\theta_m) \end{bmatrix} U_{3 \times n} \quad (3.13)$$

$$\forall 1 \leq i \leq m$$

$$\theta_j = \frac{2j\pi}{n}, 0 \leq j \leq n-1$$

Given a cross-section, the distance between the hands, the orientation of the line joining the two hands and the mid-point of the line joining the two hands specify the size, orientation and position of the cross-section respectively. Thus, the temporal variations of the locations of the two hands in 3D space completely define the evolution of the IGC. We use the locations of the hands, $\vec{w}_l(t_i)$ (left) and $\vec{w}_r(t_i)$ (right) at a given i^{th} instance t_i to evaluate the scaling, rotation and translation (Equation (3.14)). During the creation of an IGC, our goal is not only to allow the designer to create shape freely, but also to implicitly retain the aesthetics of the shape being created on-the-fly. This is where our intelligence parameters and heaviside function based representation play a key role. Using ε_{3s} we control the amount of variation in scale in order for it to take effect (Equation (3.14a)). If the hands are almost at the same distance from one frame to another, we use the previous scale value. Similarly, the choice of $\varepsilon_{3\alpha}$ and $\varepsilon_{3\beta}$ (Equations (3.14c) and (3.14d)) allows for defining the orientation of the current section to be either equal to the previous one or orthogonal to the trajectory.

$$s_i^* = (1 - u(s - \varepsilon_{3s}))s_{i-1}^* + u(s - \varepsilon_{3s})s \quad (3.14a)$$

$$s = \|\vec{\omega}_r(t_i) - \vec{\omega}_l(t_i)\|_2$$

$$\vec{o}_i^* = \frac{\vec{\omega}_r(t_i) + \vec{\omega}_l(t_i)}{2} \quad (3.14b)$$

$$\alpha_i^* = (1 - e)\alpha_{i-1}^* + e\alpha \quad (3.14c)$$

$$e = \max\{u(A_\alpha - \varepsilon_{3\angle}), u(d_\alpha - \varepsilon_{3\alpha})\}$$

$$A_\alpha = |\alpha - \text{elevation}(\overrightarrow{o_i^* o_{i-1}^*})|$$

$$d_\alpha = |\alpha - \alpha_{i-1}^*|$$

$$\alpha = \arctan \left(\frac{\omega_{ry} - \omega_{ly}}{\sqrt{\omega_{rx} - \omega_{lx}^2 + \omega_{rz} - \omega_{lz}^2}} \right)$$

$$\beta_i^* = (1 - e)\beta_{i-1}^* + e\beta \quad (3.14d)$$

$$e = \max\{u(A_\beta - \varepsilon_{3\angle}), u(d_\beta - \varepsilon_{3\beta})\}$$

$$A_\beta = |\beta - \text{azimuth}(\overrightarrow{o_i^* o_{i-1}^*})|$$

$$d_\beta = |\beta - \beta_{i-1}^*|$$

$$\beta = \arctan 2 \left(\omega_{rx} - \omega_{lx}, \sqrt{\omega_{rz}^2 - \omega_{lz}^2} \right)$$

Similar to the strategy given above, the parameters ε_{3x} , ε_{3y} and ε_{3z} in Equation (3.10) can be used to *snap* the skeletal trajectory of the IGC based on the motion of the hands. In our current implementation, we have used a common threshold of $0.5mm$ for the scale and translation parameter values for IGC creation based on the size of objects modeled which is typically of the order of $50cm$ to $100cm$. Figure 3.7 shows shapes for a variety of

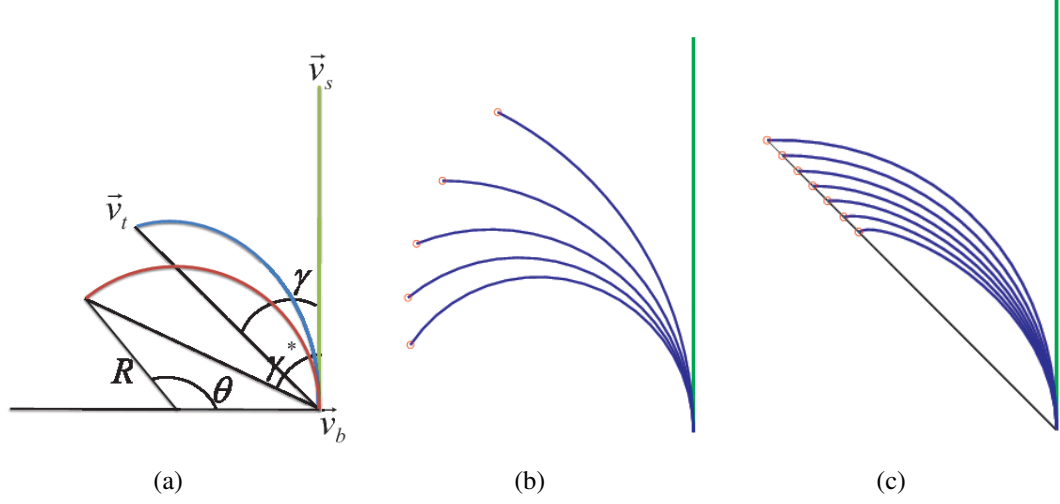


Figure 3.8. Skeleton deformation (a) method (b) circular bending and (c) rectification for $\theta = \pi/3$ and $0.75 \leq d_{bt} \leq 0.95$.

intelligent constraints for the creation of IGC's. We show three varieties of IGC's wherein, (a) the scale of the GC changes with the hand motions while the skeleton is kept vertical and orientability of the sections is not allowed, (b) the skeleton can take a free-form on 3D space while the orientation is still constrained and (c) a completely free-form mode wherein the skeleton, orientation and the section scales can be varied as per the wish of the user.

3.6 IGC Modification

We describe three operations for modifying IGC shapes, namely (a) *skeletal deformation* (bending), (b) *sectional deformation* and (c) *sectional scaling*. With respect to the IGC representation presented earlier, the first two modeling operations translate to the addition of a deformation curve as given in Equation (3.11.) The third operation is similar to the description given in Equation (3.14a).

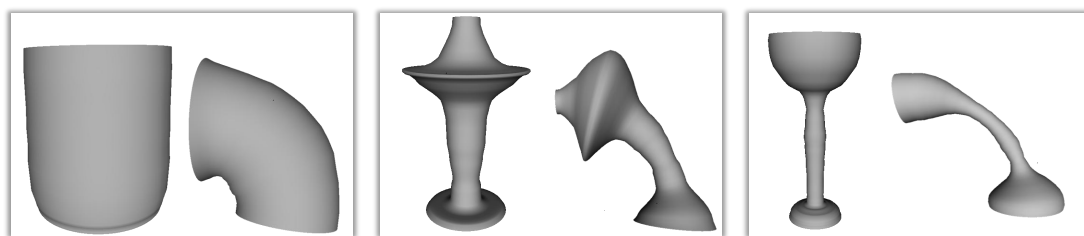


Figure 3.9. Recursive bending of IGC.

3.6.1 Skeletal Deformation

Skeletal deformation entails the bending of the skeletal curve T_i^* of the IGC. We propose a bending scheme which attempts to preserve the total length of the skeletal curve. All we need to achieve this bending is a bending curve $\Delta_i^{T^*}$ which can be added to T_i^* . For simplicity, we assume that the skeleton is vertical in its initial configuration and the bending happens only on the plane defined by the base, source and target points. First, we rotate this plane using a transformation R_B such that all the three points lie on the vertical ($x - y$) plane and the base point is at the origin. Assuming one end of the skeleton to be fixed at the *base point* ($\vec{v}_b = R_B \vec{o}_1^{S^*}$), the problem is to find a set of rotations of each of the line segments or *links* of the skeleton such that the location of the last point moves from the *source point* ($\vec{v}_s = R_B \vec{o}_m^{S^*}$) to the location of the user's hand which we call the *target point* ($\vec{v}_t = R_B \vec{v}_t^*$). This can be seen as an inverse kinematics (IK) problem for a serial manipulator and has been very well-studied in literature [94, 95]. However, we follow a different approach based on the circular bending of a skeleton followed by a rectification method for the final bending.

First, we compute the distance, $d_{bt} = \|\vec{v}_t - \vec{v}_b\|_2$, and the angle $\gamma = \angle(\vec{v}_t \vec{v}_b, \vec{v}_s \vec{v}_b)$. Then we bend the skeleton to a circular arc taking d_{bt} as a chord with its arc length is equal to the length of the skeleton (Figure 3.8(a)). Obtaining the circular bend requires the determination of the radius R of the circle and the angle corresponding to the chord-length satisfying the constraint preserving the length of the skeleton (Equation (3.15)).

$$R = \|\vec{v}_s \vec{v}_b\| / \theta \quad (3.15)$$

$$\theta = \arg \min_{\theta \in [0, 2\pi]} \left(\left| \frac{\sin(\theta/2)}{\theta} - \frac{d_{bt}}{2\|\vec{v}_s \vec{v}_b\|} \right| \right)$$

In Figure 3.8(a) the *bending angle* γ^* is different than the desired angle γ . Thus, in the final step, we minimize the angular error ($|\gamma - \gamma^*|$) by piecewise compensation for this error using the hyperbolic tangent function. Equation (3.16) gives the final coordinates, \vec{q}_i^* , of

the the i^{th} point \vec{q}_i on the bent skeleton. Figures 3.8(c) and 3.8(c) show a set of skeletal deformations for a bending angle of $\gamma = \pi/3$ and varying target points.

$$\vec{q}_i^* = \begin{bmatrix} \cos(\phi_i) & \sin(\phi_i) \\ -\sin(\phi_i) & \cos(\phi_i) \end{bmatrix} \vec{q}_i \quad (3.16a)$$

$$\phi_i = \sum_{j=1}^{i-1} \{\phi_j + w_j(\gamma - \gamma^*)\} \quad (3.16b)$$

$$\phi_0 = 0$$

$$w_i = \frac{\tanh(1/i)}{\sum_{i=1}^m \tanh(1/i)} \quad (3.16c)$$

m = Number of sections in the surface

The deformation curve is then be described in terms of the planar deformation of the skeleton as:

$$\Delta_i^{T*} = R_B^{-1} \begin{bmatrix} q_{1x}^* - q_{1x} & \cdots & q_{mx}^* - q_{mx} \\ q_{1y}^* - q_{1y} & \cdots & q_{my}^* - q_{my} \\ 0 & \cdots & 0 \end{bmatrix} \quad (3.17)$$

Figure 3.9 shows the effect of our bending scheme of three different IGC's. The bending of lamp sections in Figure 3.9(b) is undesirable since the shape of the important features have changed. In such case, a more localized method of bending based on the partial slenderness of a given IGC shape would improve the realism of these deformations.

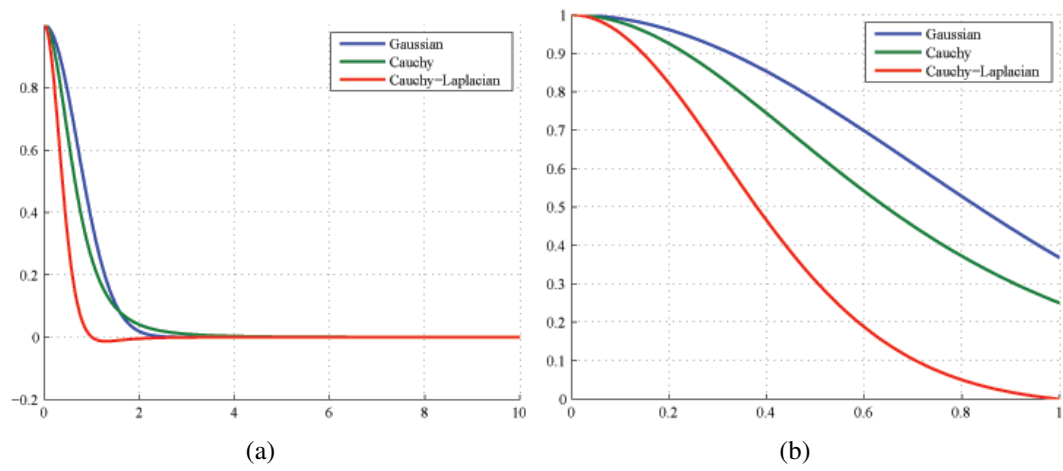


Figure 3.10. Deformation functions (a) long-range plot (b) close-up for range $[0, 1]$.

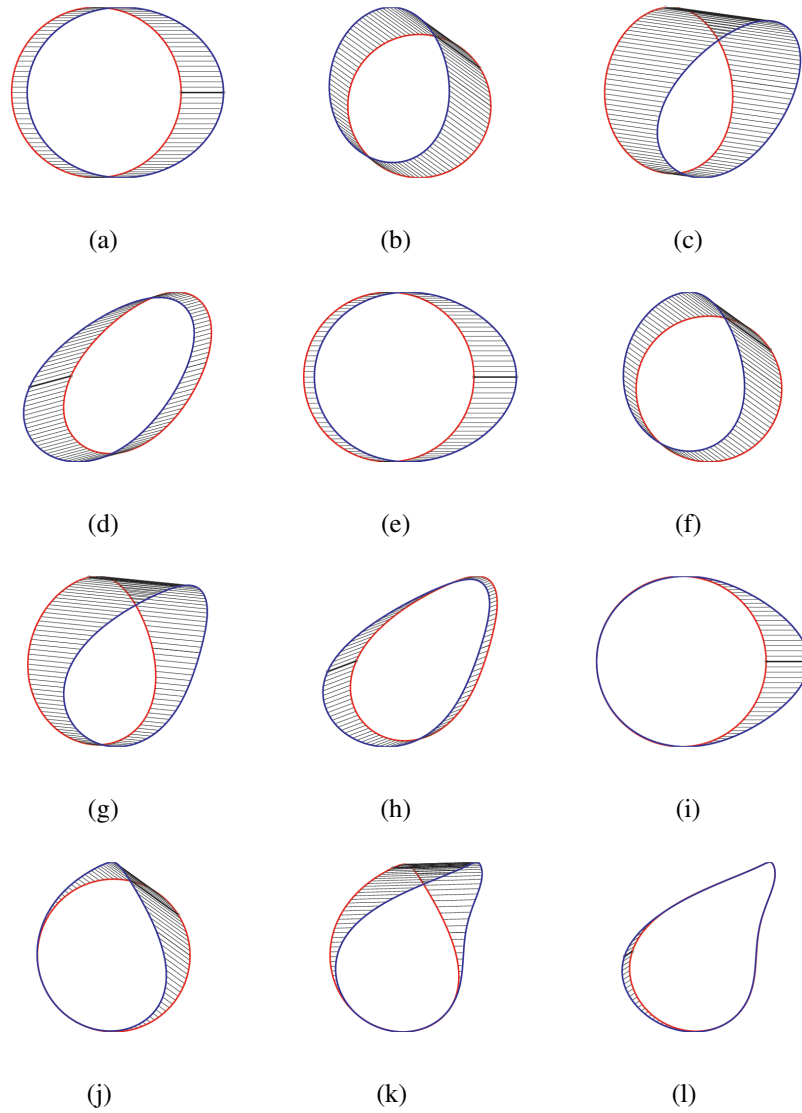


Figure 3.11. Recursive multi-handle deformation of a half-unit circle based on gaussian (a-d), cauchy (e-h) and laplacian of cauchy (i-l) functions.

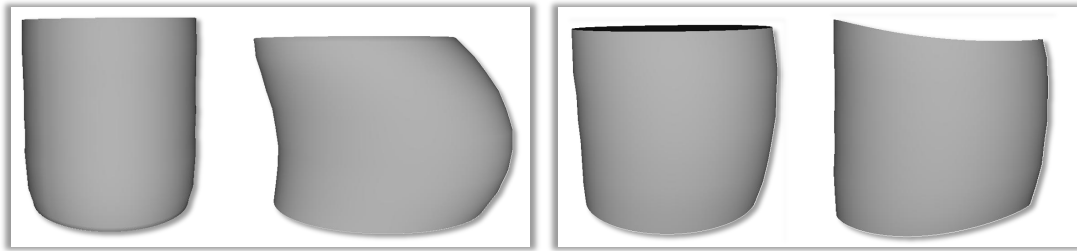


Figure 3.12. Deformation of a cylinder original shape (left) to horizontally constrained 2D deformation (middle-left) and recursive 3D deformations (middle right and right).

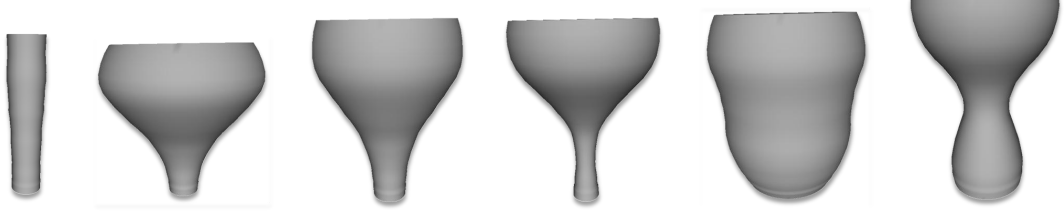


Figure 3.13. Recursive scaling of a cylinder.

3.6.2 Sectional Deformation

The general theoretical details of sectional deformation were described in the mathematical framework. The main detail of the sectional deformation method which was not discussed earlier was the *deformation-transfer* function $f(\vec{d})$ in Equation (3.7). In this section we discuss three functions namely the gaussian function ($f_1(d) = e^{-d^2}$), cauchy's function ($f_2(d) = 1/(1 + r^2)$) and the laplacian of cauchy's function ($f_3(d) = (r^2 - 1)/(r^2 + 1)^4$). Figure 3.10 shows the plots of these functions. Figure 3.11 shows a comparison of the application of these functions on a circle of radius 0.5. We conducted experiments to observe the effect of these functions on the deformation of a circle with single as well as multiple handles. Based on the capacity of localized deformations and retainment of symmetry upon recursive deformation, we found the laplacian of the cauchy's function to be the best choice for our sectional deformation function.

The deformation of an IGC is split into two parts, first being the deformation of the *active* section C_a with center \vec{o}_a , containing the handle vertex and the second being the transfer of deformation to the remaining sections. This is achieved by scaling the deformation matrix $D(\vec{\eta}_i)$ with $f_3(d)$. In this case, d is the distance along the skeletal curve. Thus, the deformation matrix for the j^{th} section ($1 \leq j \leq m$) is given by $\|\vec{o}_j^{S*} - \vec{o}_a^{S*}\|_2 D(\vec{\eta}_i)$. Figure 3.12 shows the deformations of a cylinder with a variety of constraints using repeated application of $f_3(d)$ on an initial IGC.

3.6.3 Sectional Scaling

The general idea of sectional scaling for deformation is akin to the scaling for creation, wherein the location of the hands define the deformation of scale on a certain cross-section based on proximity. This deformation of scale is transferred to the remaining sections using the same strategy of deformation transfer function described in the previous section. If i and j be the active sections due to proximity of left and right hands to the surface S^* then the scale deformation is given by $\delta(i, j)(\|\vec{\omega}_r(t_i) - \vec{\omega}_l(t_i)\|_2 - s_i)$. Note here that the Kronecker delta used here assures that i and j are the same sections, i.e. both the hands must share proximity to the same section which then becomes the active section. For

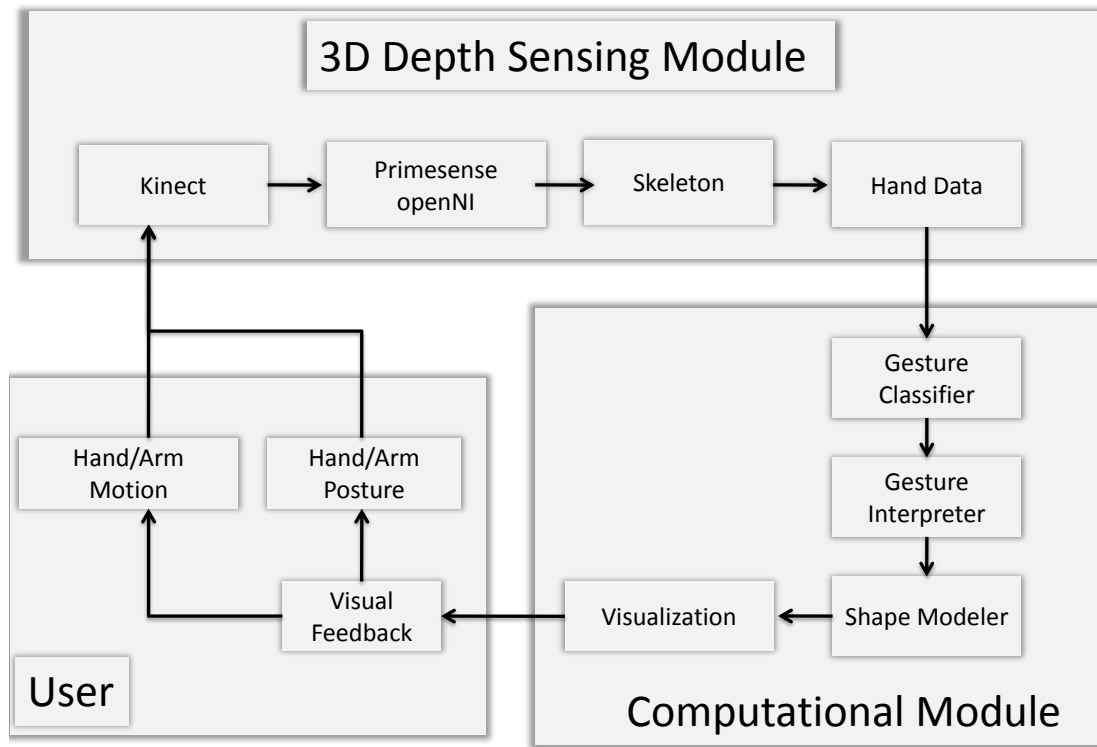


Figure 3.14. Pipeline showing the flow of information from the user to the proposed system.

practical purposes, it is difficult to get both the hands simultaneously close to the same section. To overcome this issue, we allow a range of sections (about 10% of the total number of sections) for the proximity of the left and right hands. We select the section in

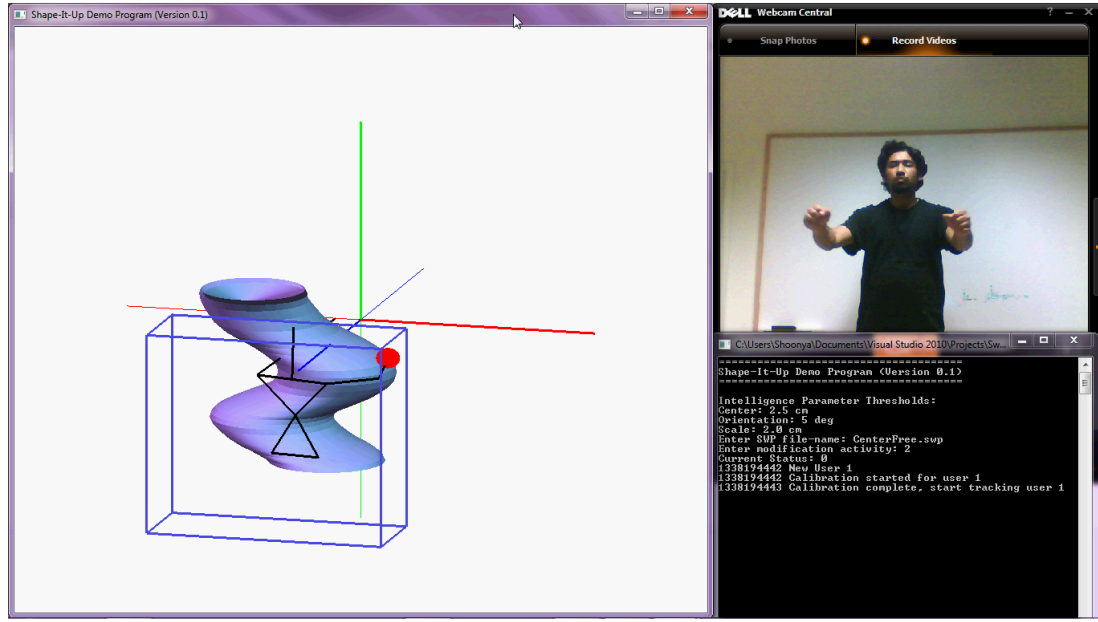


Figure 3.15. User interface of the prototype.

the middle of this range as the active section. Figure 3.13 shows a sequence of recursively scaled IGC's through sectional scaling using the deformation transfer function $f_3(d)$.

3.7 IGC Manipulation

The manipulation of an IGC is a global transformation activity which was described in section 3.3.5. In the context of hand based modeling, we use the *point* posture in conjunction with bimanual proximity (i.e. both hands near the surface) to define the context of manipulation. Here, the scaling, translation and rotation are given by Equation (3.14), with different values of the intelligence parameter for the scaling. In the context of manipulation, i and $i - 1$ are instances of time. In this case, we intend to scale the object only with significant change in the inter-hand distance. Thus, we set the ε_3 to $2.5cm$ in the present work.

3.8 Implementation

To demonstrate natural shape exploration with the SGC-I paradigm using IGC representation, we developed a prototype to support the creation of GC's. In the current implementation we assume the shape of the cross-section to be given apriori. In this case, we decided to take a circular cross-section with resolution of 100. The development of the tool was done using C++ and OpenGL was used for the rendering. We use an off-the-shelf Kinect camera in conjunction with the skeletal tracking capability provided by the openNI library to track the hand locations of the user. For the recognition of user intent through gesture classification, we implemented a hand-posture classification method for gesture recognition. In our situation, each hand data was an 8080 depth image as seen by the Kinect. Since the hand poses are perceived in different viewpoints and there are large variances, we choose to use random forest [96] to classify a given observation. Random forest is one of the most accurate learning algorithms available; especially, it is fast, only several binary decisions are needed in recognition. The pipeline for the implementation is shown in Figure 3.14.

The interface developed in this tool is simple; the user only sees the global frame of reference and a working volume wherein the user can create the IGC's. Taking into account the spatial extent within which a user would typically feel comfortable working, we define the virtual slab at a distance of 75% of the total arm-length of the designer along the view direction of the Kinect camera. Figure 3.15 shows a typical interactive session using the interface developed.

3.9 Results

Figure 3.16 shows a wide variety of shapes using the IGC representation within the SGC-I approach. It can be appreciated that the description of shapes in the examples shown involves, to a large extent, the interpretation of what the designer wants to create. Since the actual dimensional details of a shape can always be specified as a post-process by standard



Figure 3.16. Shapes modeled using Shape-It-Up.

parametric CAD techniques, the 3D shape creation process demonstrated in this chapter is more intuitive and natural to use.

3.10 Conclusions

In this chapter, we proposed the idea of shape-gesture-context interplay (SGCI) towards natural 3D shape modeling, wherein the interpretation of gestures in the spatial context of a 3D shape directly deduces the designer’s intent and the subsequent modeling operations. Our primary focus is to propose SGCI as a generic framework which can be used to design not one but many novel gesture-driven interfaces for applications in shape modeling. We implemented “Shape-It-Up”, a prototype tool for gesture-driven expression of 3D shapes using IGC as the shape representation. We demonstrated the potential of our modeling framework through high-speed creation of shapes without the actual need for training which are otherwise difficult to model.

Shape-It-Up offers only a glimpse at the variety of rich spatial interactions enabled by depth camera for 3D shape exploration. In the current implementation, the tool allows the designer to create 3D shapes. Modeling on surfaces with holes and multiple cross-sections will be an important area which we will consider. Other future work includes deformations of the model with enforced symmetric constraints, constrained bending with natural limits and twists in cross-sections. The general logical architecture behind SGCI can be used for implementing a wide variety of contexts with different combinations of gestures and shapes. A complete user study based on factors like fatigue, time of modeling, ease of use etc. is, in its own right, a research problem involving significant effort. In the forthcoming chapters, we will revisit this framework in the context of virtual pottery and show an evaluation of the symbolic approach.

4. GEOMETRIC APPROACH FOR MID-AIR DEFORMATION THROUGH PROXIMAL ATTRACTION

We learned from the previous chapter that in the symbolic approach for shape ideation, converting user input into a meaningful shape modeling task involves (a) acquisition, segmentation, and processing of hand data, (b) extracting a virtual representation of the hand, (c) mapping the gestures to a shape modeling task, and (d) generating an appropriate response of the shape as intended by the user's input. The key element here is that it is interface designer who must decide what gestures to use and how to map them to a given set of geometric operations. In fact, in our exposition of SGC-I, we followed a similar approach.

Dourish [17, p.170] states: *“Traditional interactive system design ascribes two sets of responsibilities to the designer. The first is responsibility for the artifact for its form and function, and for how they are related. The second is responsibility for its use for the sets of activities in which people will engage with that artifact.”* Thus, in the context of embodied interactions, a symbolic approach does not inherently allow users to manage coupling. One might argue that users' preference could be accommodated through gesture elicitation where they may define what the set of gestures should be and how they must be mapped to a given set of geometric operations. However, hand and finger movements in real-world shaping processes (such as pottery or clay sculpting) are complex, iterative, and gradual. Such processes are essentially governed by the geometry of contact between hand and clay. Thus, eliciting a finite set of symbols and associating them to dynamic tasks such as deformation is not as simple as defining gestures for operations such as selecting, picking, or placing objects. In this chapter, we will investigate a geometry-driven approach for shape deformation that does not depend on a prescribed set of gestures. Instead, it implicitly utilizes the geometry of contact between the user's hand and a virtual object towards mid-air shape deformation.

To cater to our goal of investigating mid-air deformation, we choose the example of virtual pottery. Our choice of pottery offers a well-defined and intuitive relationship between the use of hands and the shaping of pots to a user. The simplicity of the geometric representation and deformation in pottery lends itself to quantitative measurement of the user's response to our system. As we will see in the subsequent chapter, this concrete and practical implementation enables us to study how hand and finger motions can allow users to deform a shape *as they see fit*).

4.1 Related Work

Three-dimensional user input has been extensively studied, evaluated, and reviewed [97–100] in the context of 3D selection, manipulation [101], control, and navigation, in virtual environments. Mid-air interactions have found significant use in gaming [102] and art [103–105]. Within the context of 3D conceptual shape design, we find two broad classes of mechanisms that enable mid-air user input.

4.1.1 Instrumented Controllers

The first class comprises of instrumented controllers such as gloves [31, 106, 107], hand-held trackers [32, 108], and haptics devices [33, 109, 110]. Special devices and setups [111–116] have also been demonstrated for 3D interactions. These hardware systems offer great control, feedback, and unambiguity to the user while interacting in mid-air. In these approaches, the user provides explicit commands via the hand-worn or hand-held controller to indicate design intent such as starting, stopping, or selecting a modeling operation when desired. However, such systems are not accessible to the common user outside a lab environment. Further, wearing or holding can be intrusive to the user during a focused modeling task.

4.1.2 Gesture-Based Input

The second class of mechanisms for mid-air user input are the so-called *bare-hand* gesture-based interactions [30, 35, 36, 95, 117, 118]. With the recent commercialization of depth cameras, gesture-based interactions have become accessible to the common user. Creative applications for free-form shape modeling [119] in mid-air have gained significant popularity. The user input in these applications is represented as a combination of a hand posture (such as pointing with a finger) and the motion of a representative point (such as the palm or fingertip) on the hand.

Both classes of mid-air interactions (instrumented controllers and bare-hand gestures) have one common characteristic, namely, there is a clear distinction between interaction and geometric modeling. In this work, we take a different approach; we pose mid-air interaction itself as a geometric problem. Our focus is the geometric investigation of spatial interactions in the specific context of shape deformation. Particularly, we address the problem of determining how the shape and motion of a user’s hand and fingers geometrically relates to the user’s intent of deforming a shape.

4.2 Problem

Consider a mid-air interaction scenario of selecting and displacing a mesh vertex for deforming the mesh. Unlike a controller-based approach, there is no explicit physical mechanism for triggering events. Here, gestures serve two fundamental purposes. First, they help define a beginning (e.g. reaching and clutching some region of interest) and end (e.g. de-clutching the region after required deformation) of an interaction [120, 121]. Secondly, they help define the exact operation from a set of operations defined in the context of an application. For example, the type of deformation could be selected by using different gestures (e.g. fist to pull, point to push, open palm to flatten).

There are two issues with this approach. First, gesture-based interactions rely heavily on (a) the robustness of gesture recognition to hand occlusions and (b) stability of recognized gestures over time. Even slight instability in the accuracy of gesture recognition

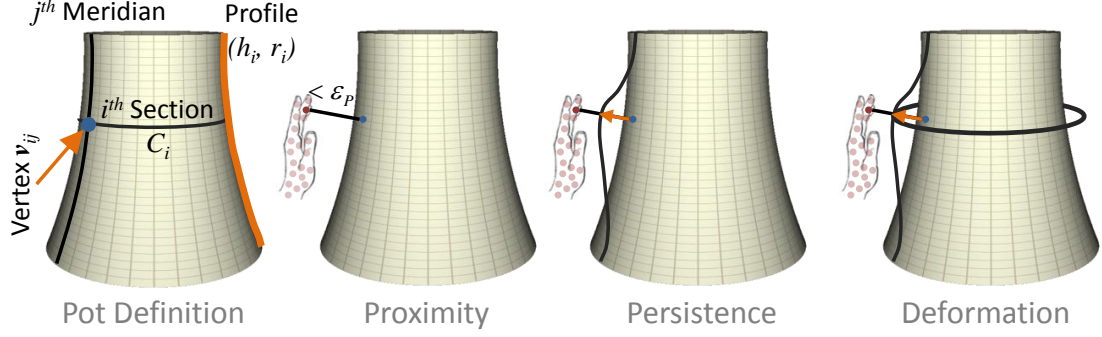


Figure 4.1. Definition of a pot and the deformation using proximal persistence.

can disrupt the modeling process leading to user frustration. Secondly, predefined gestures impose rigid constraints on users. Having to learn too many gestures is memory-intensive in a creative activity, but too few gestures can result in ambiguity while expressing design intent.

Works by Sheng et al. [112], Kry et al. [113], and Pihuit et al. [114] have leveraged finer finger level movements and grasping for 3D shape deformation by designing ingenious hardware systems. In their systems, users can actually grasp and deform virtual objects. However, for marker-less camera-based systems, the true expressive potential of finger movements remains underutilized despite advances in hand pose and skeleton estimation [50, 122]. We aim to determine user’s intent from fine finger-level movements while retaining the non-intrusiveness and accessibility of depth cameras. In doing so, we demonstrate that it is unnecessary to *prescribe* shape modeling interactions with a predefined set of rules using classified hand gestures or full-scale hand skeletons.

4.2.1 Overview

Our basic idea is to progressively conform the shape of a pot (represented as a 3D mesh) to that of the user’s hand. This idea, dubbed *proximal persistence*, is a generalization of the notion of *dwel-time* used in 3D object selection [123]. Our algorithm is a combination of *exponential smoothing* and selective *Laplacian smoothing* (Figure 4.1). Here, each point in

the hand’s PCL *attracts* a local region on the pot, hence deforming the pot’s section. The combination of many such local deformations, due to each point on the hand, amounts to a progressive convergence of the pot-profile to the shape of the user’s hands. We found two approaches that appear similar to our method in terms of the representation of the hand and the idea of persistence-based deformation.

The first approach is *Data miming* [36] wherein the hand is modeled as a set of active voxels in a volumetric domain. Thus, this approach uses hands without explicit determination of gestures for recognizing user intent, as in our case. Our method, however, is inherently different from this approach in two ways. Firstly, *Data miming* focuses on the specification existing objects through the user’s hand motions rather than creating new shapes through shape deformation. Secondly, it differentiates the relevant voxels from the irrelevant ones by imposing a threshold on the number of times the user’s hand visit each voxel. Our method does not require any threshold or statistic to distinguish intentional actions from unintentional ones. Instead, we use the rate of attraction as our parameter to determine when and how much to deform a surface (Section 4.3.3). We also provide an explicit relationship between our parameters and the responsiveness of shape deformation (Section 4.4). We determined appropriate parameter values of our algorithm through our pilot experiments. Such a study was not the focus of the *Data miming* approach.

The *magnet tool* demonstrated by Schkolne et al. [31] captures the idea of persistence. It is a custom made hardware addendum that allows a user to change a region of surface locally by *waving the hand close to the surface while holding the tool*. This tool, as mentioned by the authors, enables an overdrawing metaphor for surface deformation. In our case, the input is dynamic and unstructured data (the PCL of the user’s hand) and there is no explicit user expression (such as waving) that triggers the deformation of a shape. Further, our method is capable of both local and global changes to the shape depending on how the user makes contact with the surface of the shape.

4.3 Proximal Attraction

Proximal attraction can be described as a spatio-temporal model according to which a 'modifiable object' in space deforms when in close proximity to a 'modifying object'. Deformation occurs over a length of time, until the shape of the modifiable object matches the profile of the modifying object. In our case, the modifiable object is the surface of a virtual pot, the modifying operation is deformation, and the modifying object is a point-sampled surface of a hand or a hand-held tool. Thus, the idea is to define the modeling process of a 3D shape as a time-series of its deformed states converging, so as to conform to the shape of the hand or the tool (Figure 4.1).

To provide a realistic and enjoyable experience of pottery, there are four considerations which drive our modeling technique. These are: (a) smooth appearance, (b) behavioral realism, (c) intuitive interactions, and (d) the possibility of using real-world and virtual objects as tools for pot creation. We will first define the required components of the technique and describe a general strategy for the deformation of pots. Our strategy is based on Laplacian smoothing which has been extensively used in literature (see [124] for a detailed review).

4.3.1 Pot Definition

We describe a pot $\tilde{P}(h, r(h))$, as a simple homogeneous generalized cylinder (SHGC) containing a set of circular sections at heights $\tilde{h} \in [h_1, h_2]$ ($h_1, h_2 \in \mathbb{R}$) whose radii are defined by a smooth function $\tilde{r} : \mathbb{R} \rightarrow \mathbb{R}$ applied to the closed and connected interval $[h_1, h_2] \in \mathbb{R}$. Thus, the function $\tilde{r}(\tilde{h})$ can also be interpreted as the *profile curve* of the pot. In a discrete setting, a pot is given by:

$$P_{n,m} := \{(h_i, r_i) \mid r_i = r(h_i), h_1 < \dots < h_n \in \mathbb{R}\} \quad (4.1)$$

The surface $S(P_{n,m})$ of the pot is defined as a vertical sequence of n circular sections C_i of radii r_i at heights h_i . In a discrete setting, a circular section is approximated by a closed regular polygon with m sides (Equation (4.2)) thus allowing for a simple quad-mesh representation.

$$C_i = [\mathbf{v}_{i,1}, \dots, \mathbf{v}_{i,m}]^T \quad (4.2)$$

$$\mathbf{v}_{i,j} = \left[-r_i \sin\left(\frac{2j\pi}{n}\right), h_i, -r_i \cos\left(\frac{2j\pi}{n}\right) \right]^T$$

A *manipulator* is defined as a point denoted by $\mu(t, \mathbf{p})$, where t is the time stamp and $\mathbf{p} \in \mathbb{R}^3$ is the position of the manipulator. The manipulating object (user's hand or a hand-held tool) can thus be represented as a point cloud (PCL) given by $M(t) = \{\mu(t, \mathbf{p})\}$ where each point is a manipulator. A *handle*, $\mathbf{q}(t) = \mathbf{v}_{i,j} \in V$ ($1 \leq i \leq n$ and $1 \leq j \leq m$), is defined as the vertex on the pot which is closest to the manipulator in euclidean space. We define a *convergence threshold* ε_C as the distance between a handle and its manipulator at which the handle is considered to be coincident with the manipulator. Finally, we define a *proximity threshold* as the maximum distance, ε_P , at which a manipulator should be from a handle in order for a deformation to occur. Making use of the structured topology of the pot mesh, we compute the handle for a given hand by first computing the index of nearest section and subsequently the point closest to the hand on the nearest section. Thus, the handle \mathbf{q} at an instance t , is given by:

$$\mathbf{q}(t) = \mathbf{v}_{i,j} \quad (4.3)$$

$$i = \underset{i}{\operatorname{argmin}}(h_i - p_y(t))$$

$$j = \underset{j}{\operatorname{argmin}}(\|\mathbf{v}_{i,j} - \mathbf{p}(t)\|)$$

Here, $\|\cdot\|$ is the Euclidian distance defined in \mathbb{R}^3 . Further, i gives the index of the cross-section closest to the manipulator. It is clear that the relationship between a manipulator and its corresponding handle is that of *proximity*, the first component of *proximal persistence*.

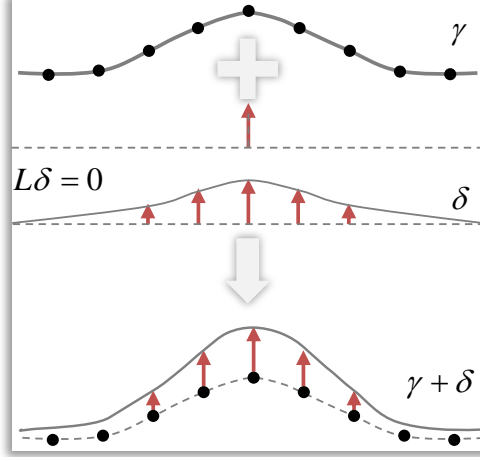


Figure 4.2. Deformation strategy of a curve γ (row 1) involves initializing a seed displacement (row 2), application of the Laplace operator to obtain a smooth deformation curve δ (row 3), and finally the computation of deformed curve $\gamma + \delta$.

4.3.2 Deformation Method

In real world pottery, deformations due to the potter's hands are either local or global based on the extent of the profile region in contact with the potter's hands or the tools held in the hands. The smoothness of deformation is an essential constraint as well. Consider the hand as a set of points in 3D space. Each point in the PCL deformed a small local region on the pot using the proximal attraction approach. On the whole this amounts to a gradual and progressive convergence of the pot-profile to the shape of the user's hand. We achieve this convergence by using the idea of selective Laplacian smoothing. This method performs local or global deformation of the profile based on the distribution of the manipulators on the profile. For manipulators within a small region of the profile, the deformation is local. If the manipulators cover a broader span of the profile, the deformation is global.

Given the profile function $r(h)$, the problem of deforming the surface of a pot is essentially transformed to that of deforming the profile curve. For a curve $\tilde{\gamma}(u) \in \mathbb{R}^3$ ($u \in [0, 1]$), deformation on $\tilde{\gamma}$ can be defined as a curve $\tilde{\delta}(u)$ such that $\tilde{\delta} : \tilde{\gamma} \rightarrow (\tilde{\gamma} + \tilde{\delta})(u)$. Our goal is to determine a deformation $\tilde{\delta}(u)$ which satisfies Laplace equation ($\nabla^2 \tilde{\delta} = 0$).

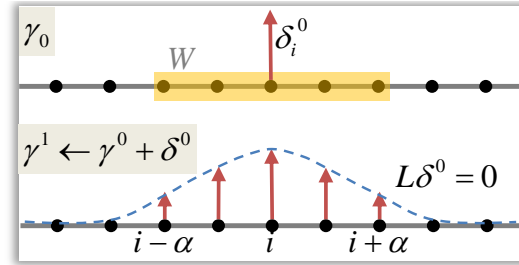


Figure 4.3. The selective application of Laplace operator is based on the window operator α which defines the neighborhood around some given vertex i in a profile.

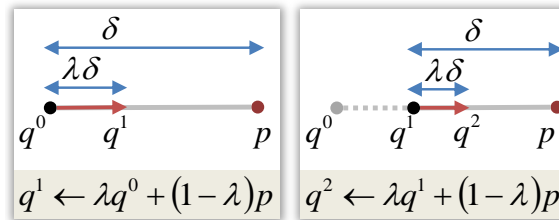


Figure 4.4. Recursive attraction of a handle q towards a manipulator p is shown. The parameter $\lambda \in [0, 1]$ defines the rate of convergence and is akin to the *smoothing constant* in single exponential smoothing.

With seed displacements defined on selected vertices of the profile, we compute a smooth deformation and apply it to the profile (see Figure 4.2). Let $\gamma := \{(h_i, r_i)\}$ be the discrete profile curve as given in (Equation (4.1)) and $K \subset [1, n]$ define a set of selected vertices in γ . Given a set of seed displacements on the vertices in K , our goal is to determine the displacement of all other vertices so as to preserve the smoothness of γ . To achieve this, we first initialize a seed displacement curve δ (such that $\delta_i = 0 \forall i \notin K$) and iteratively apply discrete Laplacian smoothing, $\mathcal{L}(\delta) = 0$, (Equation (4.4)) with Neumann boundary conditions. The deformed profile is then obtained by setting $r_i \leftarrow r_i + \delta_i$, $\forall i$. The derivatives at the boundary are set to zero, i.e, the radii at the boundary sections of the pot are the same as the radii at their neighboring sections. We use inverse edge weights to determine the non-diagonal elements of \mathcal{L} (Equation (4.5)).

$$\mathcal{L}_{i,j} = \begin{cases} -1, & |i - j| = 0 \\ 1, & |i - j| = 0, \quad i \in \{1, n\} \\ w_{i,j}, & |i - j| = 1, \quad i \notin \{1, n\} \\ 0, & \text{otherwise} \end{cases} \quad (4.4)$$

$$w_{i,j} = \frac{l_j}{\sum_{k \in K} l_k} \quad (4.5)$$

$$K = \{k : |i - k| = 1\}$$

In order to deform the profile in a smooth manner, we begin by defining W to be a set of all *contiguous* subsets in $[1, n]$. Note that a single section $i \in [1, n]$ and the whole set $[1, n]$ trivially belong to W . Subsequently, we define a mapping $\alpha : W \rightarrow W$ as given in Equation (4.6). Given a section i , it is clear that $\alpha(i)$ determines the window of sections above and below the section i , which are subject to Laplacian smoothing (Figure 4.3).

$$\alpha : \{i \dots j\} \mapsto \{\max(1, i - \alpha) \dots \min(n, j + \alpha)\} \quad (4.6)$$

$$\alpha \in \mathbb{N}$$

$$i, j \in [1, n]$$

Thus, for a given seed displacement $\delta(t)$, the deformation of the profile can be controlled using parameters, namely the window operator α and a parameter $\beta \in \mathbb{N}$ specifying the number of iterations for Laplacian smoothing. Note that a single point manipulator will instantiate a smooth but local deformation as should be expected in a typical real-world scenario - *a thin tool creates a thin impression*. However, we aim to allow for both global and local deformations. In our case, note that a manipulating object is a point-sampled surface. Thus, global deformations naturally occur due to the combination of window operators for spatially proximal manipulators.

4.3.3 Attraction

We define attraction as a continuous temporal response of the pot profile to the proximity of a given manipulator. The main step towards the definition of attraction is one involving initializing the seed displacement. For a discrete pot profile curve γ and a handle i , the idea is to *attract* a handle towards its corresponding manipulator by a fraction of distance between them (Figure 4.4). Our idea is inspired by exponential smoothing [125]. Given a parameter $\lambda \in [0, 1]$, we converge the handle towards the manipulator over time in a smooth manner (Figure 4.5). Note that this parameter, when constant, is analogous to the *smoothing constant* in single exponential smoothing. Similarly, a temporally varying parameter $\lambda = \lambda(t) \in [0, 1]$ is analogous to adaptive exponential smoothing. We call λ the *attraction parameter*.

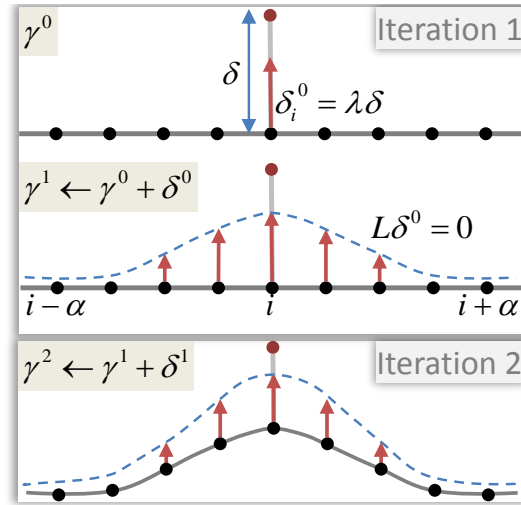


Figure 4.5. Algorithm for the application of proximal attraction to a discrete profile curve of the pot for one manipulator.

4.3.4 Pushing vs. Pulling:

Generally, a deformation to any given shape can either be inward (push) or outward (pull). The exact characterization of the deformation (shape, size, and location) is subject to the type of contact that is maintained on the deforming object. Along similar lines, our modeling technique has three general responses to a user input, namely push, pull, and no response. The goal is to recognize which of these represents the response to the actual intent of the user without asking the user to remember any prescribed rule (for instance, a button press or a static hand gesture). Note that the recognition of intent for “no response” is important, since it represents the robustness to accidental user input in cases when the user wishes to take rest or explore other features in a given modeling interface.

A push is characterized by an inward displacement i.e. when $\hat{\delta}_i^H < 0$. This is the simplest case wherein a user would typically approach the pot and subsequently recede away once the desired deformation has occurred. A pull is characterized by an outward displacement i.e. when $\hat{\delta}_i^H > 0$. This is a non-trivial intent to recognize since a user would invariably approach the surface first and then recede to pull. The overall motion of the hand is similar to that of a push. However, unlike a push, a pulling action typically

involves the articulation of hands so as to grasp the surface that is being deformed. In order to account for grasping actions in pulling tasks, we defined the attraction parameter as a smooth function given by $\lambda^P e^{-A(\hat{\delta}_i^N)^2}$ (Figure 4.7. Here, A defines the rate of decrease of the function with respect to $\hat{\delta}_i^H$ and $\hat{\delta}_i^N$ is the normalized horizontal distance of a manipulator from a given section of the pot, given by:

$$\hat{\delta}_i^N = \frac{\hat{\delta}_i^H}{\hat{\delta}_{max}^H - \hat{\delta}_{min}^H} \quad (4.7)$$

$\forall i$ such that: $\hat{\delta}_i^H > 0$

Our initial experiments showed that the normalization of $\hat{\delta}_i^H$ allowed us to standardize the design of bandwidth A . Also, λ^P is the maximum rate of pulling (for $\hat{\delta}_i^H = 0$).

4.3.5 Accidental Deformation:

Periods of rest, reflection, and accidental hand movements can sometimes lead to unintended deformations. We propose a temporal approach for avoiding accidental or unintended deformation of the pot by allowing for the pot to deform only when contact with the pot is maintained for a sufficient amount of time. We employ adaptive exponential smoothing to achieve this wherein we vary the attraction parameter λ according to a monotonically increasing function of time with a bounded range, i.e. a fixed duration of time T . Note that the same approach is also applicable to λ_P . In this work, we implemented a linear function to determine λ as follows:

$$\lambda(t) = \max\left(\frac{t - T_0}{T} \tilde{\lambda}, \tilde{\lambda}\right) \quad (4.8)$$

Here, $\tilde{\lambda}$ and $\tilde{\lambda}_P$ represent pre-defined maximum values of the attraction parameters for push and pull respectively; T represents a pre-defined duration of time taken for λ to vary from 0 to $\tilde{\lambda}$ from the starting time T_0 . The idea is to reset λ for every initial contact made

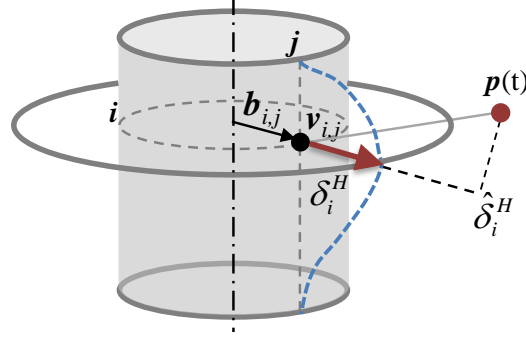


Figure 4.6. An illustration of proximal attraction for hand PCL-based deformation is shown.

with the pot and linearly increase it to a maximum prescribed value within a stipulated duration of time.

Given a handle $v_{i,j} \in C_i$ and its corresponding manipulator $\mu(t, \mathbf{p})$, the application of attraction translates to defining the increase or decrease in the radius of C_i such that $v_{i,j}$ converges to the $\mu(t, \mathbf{p})$ (Figure 4.6). The rate at which this convergence takes place is decided by the parameter λ as represented by the following equation:

$$v_{i,j} \leftarrow \lambda v_{i,j} + (1 - \lambda)(\hat{\delta}_i^H \mathbf{b}_{i,j}) \quad (4.9)$$

$$\hat{\delta}_i^H = |\langle \mathbf{b}_{i,j}, \mathbf{p} - \mathbf{v}_{i,j} \rangle|$$

$$\mathbf{b}_{i,j} = \left[-\sin\left(\frac{2j\pi}{n}\right), 0, -\cos\left(\frac{2i\pi}{n}\right) \right]^T$$

Using Equation (4.9), the seed displacement and the corresponding section transformation are given by Equation (4.10). Note that $\hat{\delta}_i^H$ is a signed displacement and can take both positive and negative values for outward (pull) and inward (push) displacements respectively.

$$C_i \leftarrow \left(\frac{r_i + \delta_i^H}{r_i} \right) C_i \quad (4.10)$$

$$\delta_i^H = \lambda \hat{\delta}_i^H$$

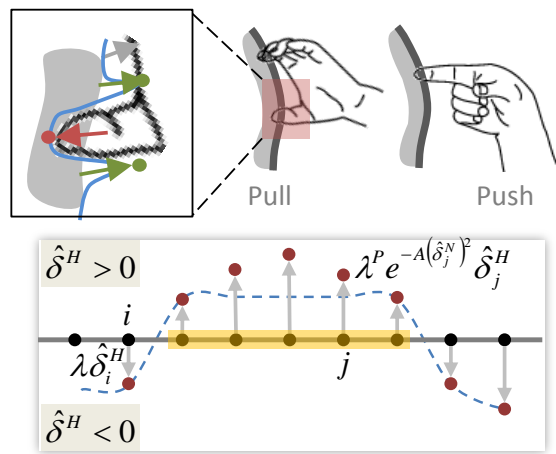


Figure 4.7. The intent for push and pull can be disambiguated by the identifying the shape of a manipulating object even when the overall motion is similar (top row). We apply outward attraction on the profile in terms of a smooth function decreasing with respect to the distance between that manipulators and the handle (bottom row).

4.4 Guidelines for Controllability

The controllability of our deformation method is affected by two factors: (a) the disparity between what a user intends for the shape to be and what the shape actually becomes after the deformation, and (b) the responsiveness of the deformation. The goal is to minimize the disparity and *optimize* the responsiveness. Here we investigate parameters which affect the deformation behavior and responsiveness of the interaction as described below.

We can define proximal attraction by the set of parameters as $\mathcal{P}(\varepsilon_C, \varepsilon_P, \tilde{\lambda}, \tilde{\lambda}^P, A, \lambda(t), T)$ wherein the first two parameters characterize proximity and the last three parameters characterize persistence. Combined with α and β described in section 4.3.2, we define deformation parameters denoted as $\mathcal{D}(\alpha, \beta)$.

4.4.1 Sampling and Resolution

From our deformation strategy, it follows that our approach is affected by the density of points in the manipulating object and the resolution of the pot mesh (defined by m and n). Note that the maximum point density in the manipulating object is dependent on the resolution of the depth camera. Thus, we define the pot resolution such that the hand or tool PCL's are densely sampled in comparison to the pot. The rationale is that the sparsity of the hand or tool PCL will result in undesired effects when the user intends a global deformation.

4.4.2 Deformation Parameters ($\mathcal{D}(\alpha, \beta)$)

The window operator defined by α primarily decides how local the deformation will be for a *single point manipulating object*. Larger values of α result in global deformation. This parameter is also a factor in producing global deformations for a manipulating object with several points. The parameter β affects responsiveness. Recall that we apply Neumann boundary conditions for Laplacian smoothing of the deformation curve which is an *open 1-manifold*. Thus, applying the Laplace operator selectively near an internal section

(i.e. a section not at the profile boundary) for a reasonably large amount of time (i.e. increasing β sufficiently) would make the deformation converge to zero making the profile non-responsive to any deformation.

4.4.3 Attraction Parameters ($\mathcal{P}(\varepsilon_C, \varepsilon_P, \tilde{\lambda}, \tilde{\lambda}^P, A, \lambda(t), T)$)

These parameters specifically affect the responsiveness of deformation. Assuming $\lambda(t)$ to be a constant function, the attraction parameters $\tilde{\lambda}$ and $\tilde{\lambda}^P$ are directly proportional to responsiveness. The parameter A defines the bandwidth of the exponential function for pulling. Lower values of A will distribute the rate of pull along the profile while higher values will lead to localized pulling at the points of contact on the pot. Low values of the proximity threshold ε_P require precision from the user while deforming the pot. Higher values require the user to recede from the pot at higher speed once a desired deformation has been achieved. The value for ε_P is decided based on the geometric properties of the modeling scene, the size of the pot and size of the hand. Finally, the parameter T signifies the amount of time required for the pot deformation to vary from being completely non-responsive to optimally responsive. Higher values of T would require the user to maintain contact for a longer time, thus making the deformation under-responsive.

4.5 Implementation

We explore our approach with two depth sensors to obtain such an input, as described below.

4.5.1 System Description

Our system setup consists of a ThinkPad T530 laptop computer with Dual Core CPU 2.5GHz and 8GB RAM, running 64 bit Windows 7 Professional with a NVIDIA NVS 5400M graphics card, a depth sensor (such as a SoftKinetic DS325 sensor or a Leap Motion Controller). The placement of the depth sensor varies according to needs (Figure 4.8). Our

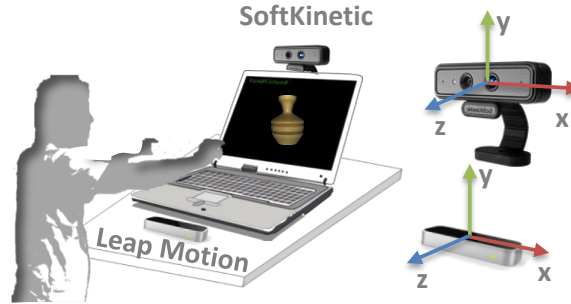


Figure 4.8. The overall setup consists of a computer and a depth camera. We implemented our approach using two such cameras, SoftKinetic DS325 and the Leap Motion controller.

applications were developed in C++ and OpenGL with the OpenGL Shading Language (GLSL). We implemented two instances of our pottery application using the two different depth sensors, namely Leap Motion controller and SoftKinetic DS325 sensor.

SoftKinetic DS325 is a close range (0.1m-1.5m) time-of-flight depth sensor that provides a live video stream of the color and depth image of the scene. Every pixel on a given depth image can be converted to a 3D point using the camera parameters. We use the DepthSense™SDK to obtain the depth-map of the scene. Our first step was to segment the hand from the scene based on a depth threshold. We then convert the segmented depth map to the PCL of the hand using the camera parameters. This is akin to using a pre-defined a volumetric workspace as the active region in front of the computer screen.

Since the main data provided by the sensor is a depth map, several image processing algorithms can be applied to extract skeletal and boundary points as we will demonstrate in future sections (see Figure 4.9). Leap Motion provides a skeletal representation of the hand comprised of the palm and fingers. We used this device to implement virtual tools, with each tool defined by a prescribed set of manipulators. We used the palm position and orientation, provided by the Leap SDK, for tool manipulation.

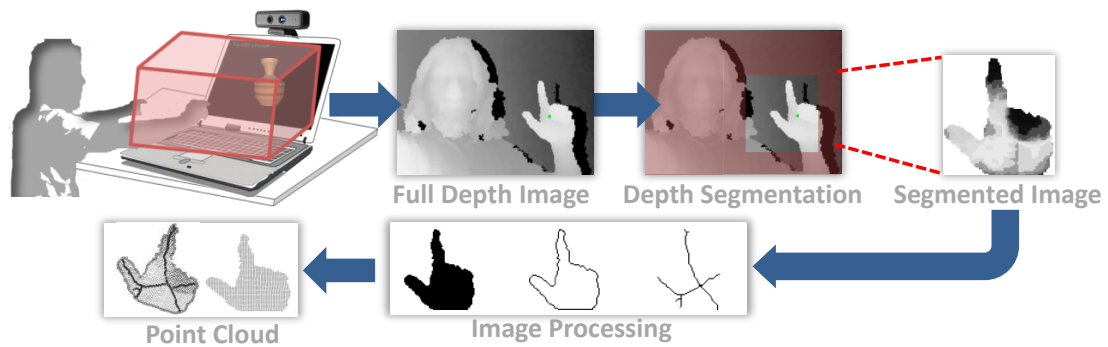


Figure 4.9. The depth sensing module for SoftKinetic involves the segmentation of hand's depth image from the full depth image using the DepthSense SDK. This is followed by optional image processing steps for skeletal and boundary images, and subsequently the segmented depth image is converted to a PCL using camera parameters.

4.5.2 Interface

Our interface comprises of an integrated 3D scene with a rotating pottery wheel, a set of shaping tools overlaid on natural outdoor background. In our pottery application, we combine three spatial interactions, namely, (a) 3D environment navigation through interaction space partitioning and 3D camera transitions, (b) robust tool selection using cylindrical zoning, and (c) gesture-free shaping interactions using proximal persistence. The interfaces for the two depth sensors, are similar in appearance. However, we designed additional interactions for tool selection for our Leap motion version.

4.5.2.1 Leap Motion Controller:

We partitioned the 3D scene into three distinct interaction spaces for unambiguous interactions (Figure 4.10). Each of these partitions, when active, map to the interaction space of the Leap Motion device. This allows for precise hand-motions in real space and constrains the user's focus to the active area. The user can freely transition between the spatial partitions by moving towards the right or left extremities of the Leap motion device. The *potter's wheel* partition is the main workspace wherein a user can shape the pot by modifying the lump of clay into a pot. The right and the left partitions represent the shaping and decoration tools respectively.

4.5.2.2 SoftKinetic:

The interface for SoftKinetic is simpler in that we did not need to use any tool selection interactions. Thus, this interface consists of the whole 3D scene without any spatial partitions. The user sees only the potter's wheel and a PCL representation of their hands, or the tools held in their hands.

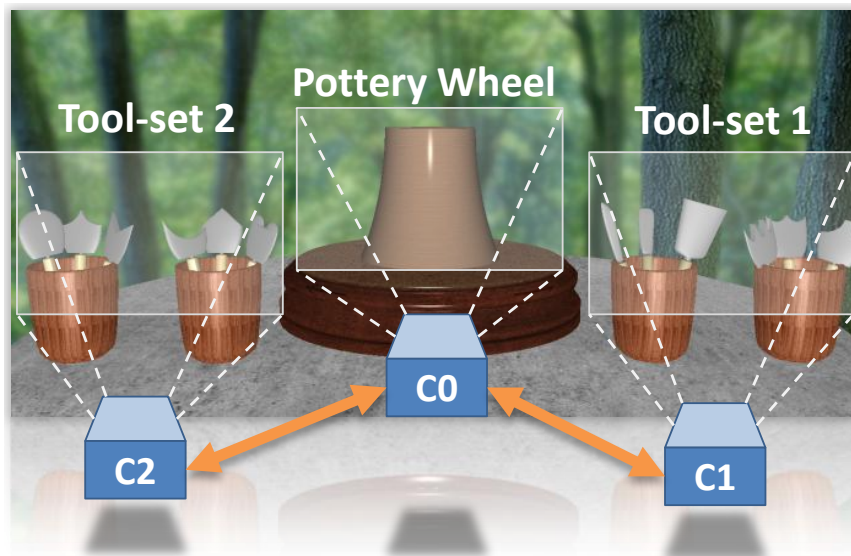


Figure 4.10. Our interface consists of active partitions of the total interaction space where each partition is associated with a virtual camera. Cameras C0, C1, and C2 correspond to the pottery wheel, tool-set 1, and tool-set 2 respectively. User's motion towards the left or right extremity results in camera transition and a subsequent re-mapping of the users physical interaction space to the active partition. The selection of tools is based on a dwell-time strategy.

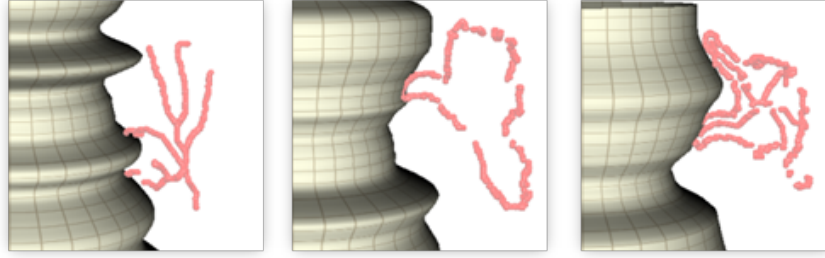


Figure 4.11. Attraction parameters identified with sufficiently-responsive deformation were applied for skeletal (left), boundary (middle), and hybrid (right) PCL's of the hand. While the boundary and hybrid PCL's are generally more controllable, the skeletal PCL shows better controllability for local-pulling tasks.

4.6 Results

In this section, we will discuss and demonstrate three use cases for our approach, namely (a) hand, (b) physical tools, and (c) virtual tools. For the first two cases, we further explore multiple modes of sampling such as (i) full PCL, (ii) boundary PCL, and (iii) skeletal PCL. Based on the designed interaction space, we fixed the pot to be 0.6 units in length with radius in the range $[0.1875, 0.3125]$. Further, we fixed the resolution of the pot by assigning $m = 314$ and $n = 100$, i.e. the difference in height between each section is set to 6×10^{-4} units. After spatial mapping and PCL scaling, the average distance between two neighboring points in the PCL were observed to be 2×10^{-3} , 4×10^{-4} , and 1.5×10^{-3} units along x , y , and z directions respectively. The deformation thresholds were set as $\varepsilon_C = 10^{-8}$ $\varepsilon_P = 0.1$. Our deformation parameters $\mathcal{D}(\alpha, \beta)$ were set to $\mathcal{D}(5, 50)$. This is based on experiments we conducted for the chosen pot resolution.

4.6.1 Hand and Physical Tools

The proximal attraction approach involves several parameters (section 4.4) which must be appropriately determined in order to design controllable interactions with our interface. Thus, our first step was to study the effect of attraction parameters ($\tilde{\lambda}$, $\tilde{\lambda}^P$, A , and T) on

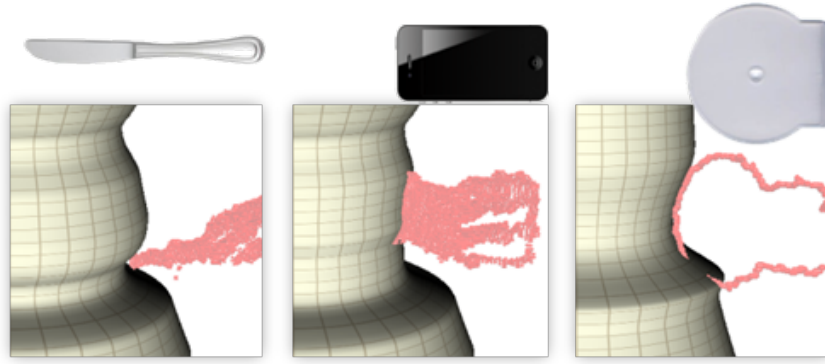


Figure 4.12. Three use cases are shown for using day-to-day physical objects (left - kitchen knife, middle - mobile phone, and right - CD-case) as shaping tools.

the intuitiveness and responsiveness of the pot deformation process. We note that effect of these parameters are not independent. Thus, an exhaustive study of all combinations is prohibitively difficult. We conducted an informal pilot study where participants used our system with six set of parameter combinations. Here, the ranges of parameters were:

- $0.1 \leq \tilde{\lambda} \leq 0.4$
- $0.1 \leq \tilde{\lambda}^P \leq 0.4$
- $A \in \{0.5, 1.0, 1.5, 2.0, 2.5\}$
- $T \in \{0.5, 1.0, 1.5, 2.0, 2.5\}$

The general trend we observed is that $\tilde{\lambda}_P > \tilde{\lambda}$ and $A < 2.0$ resulted in uncontrollable pulls. We also found that despite sufficient responsiveness, the deformations were unstable for $T < 1.5$ seconds, in that they alternated between inward and outward directions. We believe this to be a consequence of the Laplacian smoothing parameter β . Our final parameters provided by user's feedback were $\tilde{\lambda} = 0.3$, $\tilde{\lambda}^P = 0.1$, $A = 2.0$, and $T \in 1.5$. These values worked well for boundary and skeletal PCL of the hand (Figure 4.11). Use of physical tools proved to be more stable in comparison to the hand (Figure 4.12). This, we believe, can be attributed to the highly articulated nature of hand-based manipulations in comparison to the relatively rigid PCL obtained from the tools.

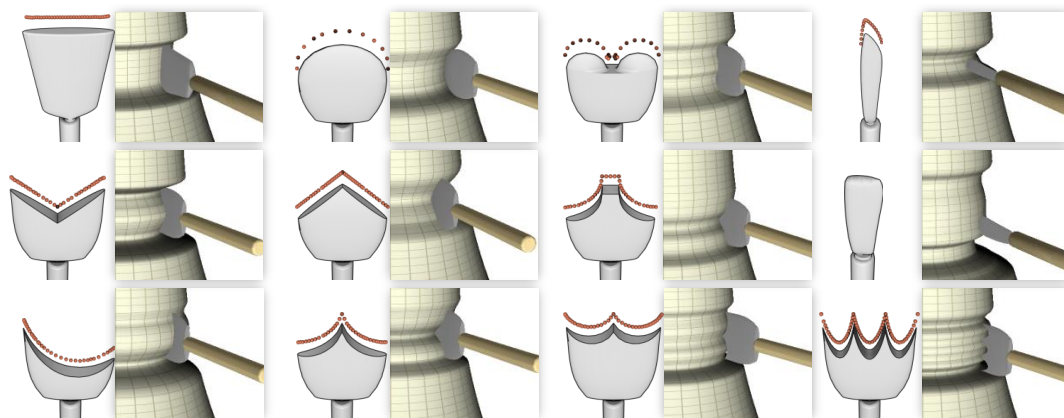


Figure 4.13. A variety of virtual tools are shown with their respective deformations on the pots.

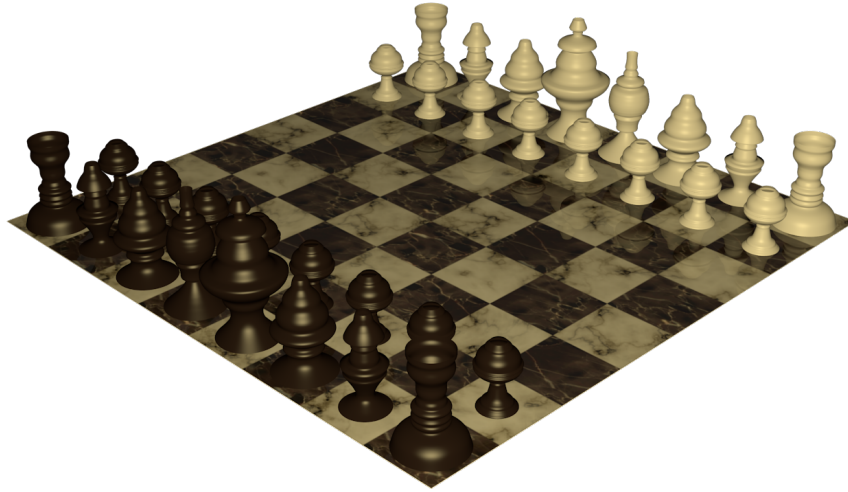


Figure 4.14. The results of the pot composition are shown Pots were created by participants using our full hand PCL-based interface.

4.6.2 Virtual Tools

Based on parameters determined from our experiments, we implemented virtual tools using the Leap Motion (Figure 4.13). In this case, we constrained the deformations to be purely inward. This was a design decision considering the typical use of tools in real world pottery. We implemented a smoothing tool wherein we did not associate any PCL with the tool. This tool simply applies the Laplace operator on the profile of the pot itself. The use of virtual tools lends itself to rapid conceptualization of several shapes. To demonstrate this, we created a virtual chess set (Figure 4.14) where each piece was created in less than one minute, taking a total time of approximately six minutes to create all the pieces.

4.6.3 Efficiency

The main cause of computational bottleneck is the need to determine handles corresponding to each manipulator. This can be a major cause of disruption and dissatisfaction for users. For arbitrary meshes, the typical solutions in such a case would be to use spatial hashing or the GPU. In our case, however, the SHGC representation of the pot eliminates

Table 4.1. The times (millisec.) taken for the steps of our algorithm are shown for different combinations of pot resolution and PCL sampling resolution.

Pot Resolution	# Points in PCL	# Handles	Normal Computation	Pot-Pcl Map	Deformation Computation	Sectional Scaling	Total Time
314×100	3524	51	4.0	12.5	0.06	0.36	17
471×150	3142	69	9.7	17.3	0.11	1.62	29
314×100	816	49	6.6	5.2	0.10	0.71	13
471×150	805	66	6.6	3.1	0.07	1.30	11
628×200	821	71	26.9	10.3	0.21	4.37	42
628×200	352	73	26.5	5.1	0.20	4.33	36



Figure 4.15. Pots that were created by young participants using our Leap-based interface.

the need to use any such special method. For a pot mesh $S(P_{n,m})$, the worst case complexity to determine handles for k manipulators is $O(k(n+m))$. This is attributed to the fact that the computation of a handle $\mathbf{v}_{i,j}$ is sequentialized in first finding the closest section (i.e. i) based on only the heights of the manipulators and sections, and then finding the closest vertex per section (i.e. j). We experimented with different pot and hand PCL resolutions and found that the total time taken for a pot with 314×100 vertices deformed by a hand with 3524 manipulators was about 17 *ms*. Increasing the pot resolution to 470×150 vertices resulted in a total time of 29 *ms* for 3142 manipulators on the hand (see Table 4.6.2). Given that the average frame-rate for SoftKinetic is 60 *Hz*, our computational efficiency is well within the required range for an interactive design application.

4.7 User Evaluation with Virtual Tools

In this study, we wanted to evaluate how well our method allows young users to rapidly conceptualize shapes through direct spatial interactions. In order to do so, we wanted to provide the users with a simpler and finite set of deformation tools for pottery. Thus, we used the virtual tools implemented using the Leap Motion controller (section 4.6.2).

This study was conducted as an informal workshop wherein we invited twelve young participants studying in fourth, fifth, and sixth grades. We introduced our application and demonstrated the interactions. Following this, all participants were asked to create as many

pots as possible in a duration of ten minutes as in the pot composition task. Figure 4.15 shows the pot concepts brainstormed by each participant during the pot composition task.

4.8 Discussion

4.8.1 Three-Dimensionality

There is no limitation in the method which restricts a user from manipulating the pot from any direction, as demonstrated in our results. However, we observed that the use of 2D displays is a factor, due to which users tended to use side configurations. We believe that 3D visual feedback would encourage users to access the front and back faces. Further, hand skeleton tracking provided in the Leap API is not sufficiently robust to use in front configurations due to occlusions. Our PCL-based method works for hand configurations where gesture recognition and skeleton tracking will fail.

4.8.2 Limitations

We see four limitations with our current approach. First, severe occlusion resulting from camera position and hand orientation is a problem particularly for skeleton-based gesture recognition. We partly addressed this challenge using our PCL-based approach which can make use of partial data even when the full hand skeleton is intractable. However, occlusion is an inherent problem in any camera-based method. Investigation of optimal camera position and the use of multiple cameras at strategic locations is important. Secondly, we provided a method for temporally adaptive persistence. However, this does not take the structure of the hand or the geometry of the physical tool into consideration. For instance, the optimal attraction parameters for the palm may be different from those of the fingers because of the nature of hand articulation. Hence, spatially adaptive estimation of attraction parameters needs further attention. Thirdly, in our current implementation, the definition of window operators is in terms of 2D profile topology, rather than actual distances in real space. Thus, our implementation is dependent on PCL sampling, relative to the mesh

resolution of the pot. Independence from the sampling resolution may be achieved with an adaptive approach wherein new sections could be added according to manipulators, or old ones removed based on geometric properties of the pot profile such as curvature. Finally, we observed that our method gives plausible outcomes for different representations of manipulating objects such as full, boundary, and skeletal PCL. The performance and controllability in each of these modalities may differ according to the context and the user's intent. A method to determine the optimal modality for each modeling task requires further investigation.

4.9 Conclusions

In this chapter, we demonstrated that it is possible to tailor a geometric modeling method to suit the needs of controllable spatial interactions that use hands and finger motions for 3D shape modeling. We make two main contributions. First we demonstrate, with a practical implementation, that it is possible to achieve controllability in bare-hand mid-air shape deformation using unstructured PCL data of the user's hand. We describe a method that does not compute any finite set of gestures or hand skeleton. Instead, our method uses the actual shape of the user's hands for deforming the shape of a pot in 3D space. This directly allows a user to shape pots by using physical objects as tools.

Intent disambiguation, user experience, and computational efficiency are affected by the representation of shapes, the modeling technique, and hand representation. Our approach allows for the accommodation of different representations of the hand, ranging from discrete gesture-classes to kinematic hand models. A comparative investigation of different hand representations as well as traditional interactions would provide a comprehensive understanding of controllability of the shaping process. The augmentation of geometric information, such as normals to the PCL, may provide valuable insights on the articulation of hands and allow for determining parameter combinations adaptively based on the motion of the hands. Works by Rosman et al. [126, 127] provide frameworks for motion segmentation of point clouds and demonstrate specific examples related to hand segmentation were

demonstrated. Augmentation of such frameworks with our technique could lead to several novel interaction mechanisms.

At its core, proximal attraction is a general notion which can be instantiated in a variety of ways by combining different hand representations with different shape representations and modeling metaphors. It will be interesting to investigate methods which could automatically deduce appropriate hand representations for different modeling metaphors. With upcoming mid-air geometric design applications, achieving simultaneous efficiency, robustness, and controllability is a challenging problem towards enhancing the user's creative process and outcome. We believe that proximal attraction takes a fundamental step towards problem.

5. HAND GRASP & MOTION FOR MID-AIR SHAPE DEFORMATION

So far, we have detailed two approaches, namely SGC-I and proximal attraction, for transforming human action into design intent. First, we described the SGC-I framework wherein the intent was modeled as a combination of arm motion and a finite set of symbolic hand gestures. The aim was to contextualize this intent with respect to the geometric content that the user intended to create, modify or manipulate. Subsequently, we generalized the notion of shape deformation through proximal persistence wherein we demonstrated the use of point-sampled representation of hands to interact with 3D shapes. Here, the main idea was to embed the users' understanding of a real-world shaping process within the virtual interactions for shape modeling. In particular, we considered pottery as a creative context to instantiate the proximal persistence approach.

We note that the two approaches provide fundamentally different modes of interactions with 3D geometry. On one hand SGC-I can be viewed as a state-based approach wherein an hand-based modeling application could be designed by a finite set of pre-defined associations between geometric objects and processed user inputs such as gestures. On the other hand, proximal attraction captures the notion of spatio-temporal continuity of hand-to-shape interactions by making use of raw user-inputs such as point clouds.

In this chapter, we will extend the idea of proximal attraction (presented in Chapter 4) by investigating how design intent can be deduced for shape deformation by determining the user's hand grasp and motion on a given shape. Our broader goals are to (a) identify aspects of real-world interactions that can be emulated in free-form 3D shape deformation, (b) understand the expression of design intent in shape deformation in terms of the user's hand grasp and motion, and (c) design an interaction that integrates the geometric information in user's actions with shaping operations in virtual space. To this end, we design, implement, and evaluate a method for geometric characterization of the contact made by the hand's point-cloud (PCL) on the surface model of a shape. Our approach uses the kernel-

density estimate (KDE) of the hand’s PCL to extract the grasp and motion for deforming the shape of a pot in 3D space.

For the sake of completeness, we will present the complete evolution of our algorithm in three stages of iterative design (section 5.2.3) that have been described in detail in Chapters 3 and 4. At the end of each stage, we will describe a user evaluation that informs the algorithm development of the subsequent stage. Our evaluations help reveal two core aspects of mid-air interactions for shape deformation, namely, intent & controllability. We characterize user behavior in pottery design in terms of (a) common hand & finger movement patterns for creating common geometric features, (b) user perception of intent, and (c) engagement, utility, and ease of learning provided by our approach.

5.1 Related Works

5.1.1 Mid-Air Gestures

Gestures can be designed effectively for pointing, selection [128, 129], and navigation, since they define an unambiguous mapping between actions and response. Such tasks are implemented using deictic gestures [130] and can usually be segmented into discrete phases, with each phase triggering an *event* or a *command* [120]. Pointing in the direction of a virtual object creates the association between the user and the object. A recent study [123] shows dwell-time to be an effective method of pointing and selecting objects without hint to the users. In manipulative tasks such as ours, a direct spatial mapping is required between the user’s input and the virtual object [130, 131]. Particularly in our case, such an association would be in terms of the proximity of the user’s virtual hand to the shape being deformed.

5.1.2 Mid-Air Virtual Pottery

Work specifically related to virtual pottery using direct hand manipulations has in itself made a niche within the hand-based modeling paradigm. One of the earliest works we

found is by Korida et al. [106] wherein the authors describe a 3D stereoscopic display based interface which allows users to create pottery with their hands by wearing two-handed instrument gloves. Han et al. [95] proposed an augmented reality (AR) based pottery system for realistic pottery experience. Lee et al. [110] proposed a haptic interface for creating realistic pottery experiences using the *PHANTOM Omni* device. Cho et al. [115] presented *Turn*, a system which augmented a tangible interfaces like a wooden table with 3D hand based virtual sculpting to create digital pottery using the Kinect. They used volumetric geometric representation based on the well known virtual sculpting [132] using the marching cubes method. Han and Han [117], demonstrated an interesting surface-based approach with particular focus on audiovisual interfaces for creating 3D sound sculptures.

5.1.3 Hand Grasp

Prehension is a common phenomenon in real-world interactions. Jeannerod [133] notes two functional requirements of finger grip during the action of grasping, (a) adaptation of the grip to the size, shape, and use of the object to be grasped and (b) the coordination between the relative timing of the finger movements with hand transportation (i.e. whole hand movements). Intended actions strongly influence motion planning of hand and finger movements [134]. This suggests that the intent for deformation can be recognized before the user makes contact with the surface being deformed. Grasp classification [135] and patterns of usage and frequency [136] have been integral to robotics research. Literature in virtual reality [137, 138] has studied and implemented grasping in the context of object manipulation (pick-and-place). Kry et al. [113] implemented a novel hardware system to emulate grasping for desktop VR applications such as digital sculpting. It is worth noting that the primary methodology for investigating grasp taxonomies is mostly derived from the geometry of the hand in relation to a physical object that is held or manipulated by the hand. What we aim to do is to understand what is the minimal and sufficient characterization of the user's hand and finger movements, that could be used for mid-air deformation. Our

goal is not to explicitly detect the hand grasp, but to design a deformation approach where the grasp is automatically and implicitly taken into consideration.

5.2 Overview

5.2.1 Intent & Controllability

The general term *intent* is literally defined as “*the thing that you plan to do or achieve : an aim or purpose*”. In our case, intent (*what one wants to achieve*) can be described in terms of the context of shape deformation (*what operations one can perform on the shape*). Based on Leyton’s perceptual theory of shapes [139], Delamé et al. [140] proposed a process grammar for deformation by introducing structuring and posturing operators. Here, structuring operators involve adding/removing material to the shape, while posturing operators allow for modifications such as bending or twisting some portion of the shape. Since our context is that of deformation, we define the intent in terms of two basic operations: pulling and pushing. These are analogous to structuring operators.

We see controllability as the quality of intent recognition and disambiguation *as perceived by the user*. Specifically, in our context, controllability is defined as a function of two factors: (a) the disparity between what a user intends for the shape to be and what the shape actually becomes after the deformation and (b) the responsiveness of the deformation. The goal is to minimize the disparity and optimize the responsiveness.

5.2.2 Rationale for Pottery

We have two goals in this paper. First, we seek a concrete geometric method that takes a general representation of the user’s hand (PCL) and allows the user to deform 3D geometry. Second, we want to investigate this geometric method in light of intent and controllability. Thus, our focus here is not to build a comprehensive and feature-rich 3D modeling system. Instead, we intend to investigate spatial interactions for 3D shape deformation with an unprocessed representation of the hand.

Our broader motivation in this work is to cater to the creative needs of individuals that are inclined towards 3D modeling and design and but do not have the expertise require for working with design tools. With this in view, we use pottery as our application context for two reasons. First, it offers a well-defined and intuitive relationship between the use of hands and the shaping of pots to a user. This allows us to concretely construct a geometric relationship between the shape of the hand PCL and the corresponding user intent. Secondly, the simplicity of the geometric representation and deformation lends itself to quantitative measurement of the user’s response to our system.

5.2.3 Evaluation Approach

Given the context of pottery, our approach involved the following three stages:

Stage 1: Using hand as one-point manipulator, we implemented *proximal-attraction*, an interaction technique for clutching and de-clutching without hand gestures. Our technique (section 5.3) generalizes the notion of *dwelt-time* in the context of mid-air shape deformation. We conducted a preliminary study to evaluate the feasibility and effectiveness of this technique.

Stage 2: We extended the *proximal-attraction*¹ method to the whole shape of the hand (section 5.4). Here, the hand was represented as a collection of multiple points (i.e PCL) obtained via a depth sensor. Each point in the PCL deformed a small local region on the pot using the proximal-attraction approach. On the whole this amounted to a gradual and progressive convergence of the pot-profile to the shape of the user’s hands. Through experimentation, we found that users had significant difficulty in creating convex (pulling) and flat (fairing) features on the pot. This method was also found to be agnostic to the user’s grasp and hand movements.

Stage 3: Based on our experiments, we implemented our final technique for pot deformation using hand PCL (section 5.5). We used kernel-density estimation to characterize the

¹The technical details of this approach were provided in Chapter 4

contact between the hand and the pot. This allowed us to classify the users' intent to push, pull or fair the surface of the pot depending on the hand grasp, finger movements, and motion of the hand on the pot's surface. We conducted a final user evaluation to investigate the efficacy of this approach.

5.3 Hand as a Point: Clutching by Proximal Attraction

In the first stage, we developed a method wherein the hand is represented as a single point manipulator, as is the case with many gesture-based methods. The main goal was to allow users to deform the surface the of pot without using hand gestures for clutching and de-clutching the pot.

5.3.1 Technique

Let h be the location of the hand in 3D space and p be the point on the pot that is closest to h . The main idea of proximal-attraction is to deform the pot gradually by attracting p towards h in the *horizontal* plane. The condition of proximity is that the distance between h and p should be less than a pre-defined threshold (say ε). We implement the approach in the following steps:

1. Given h and A , compute p
2. if($\|h - p\| < \varepsilon$)
 - (a) Set δ to horizontal distance between h and p
 - (b) Set attraction at p to $\alpha\delta$
 - (c) Compute smooth deformed profile using Laplacian smoothing ($\nabla^2\delta = 0$ for all points in A)
3. Rescale pot sections

Here, $\alpha \in [0, 1]$ is the rate of attraction where $\alpha = 0$ implies no attraction and $\alpha = 1$ implies maximum attraction. Our idea is inspired by exponential smoothing [125]. The

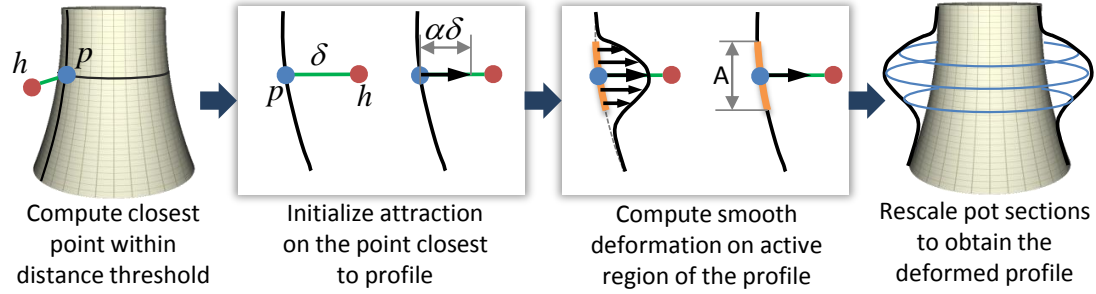


Figure 5.1. Algorithm for one-point pot deformation is illustrated for proximal-attraction. The pot is gradually deformed by attracting the profile towards the hand (represented by a point). Subsequently, each section is re-scaled to obtain the deformed pot surface.

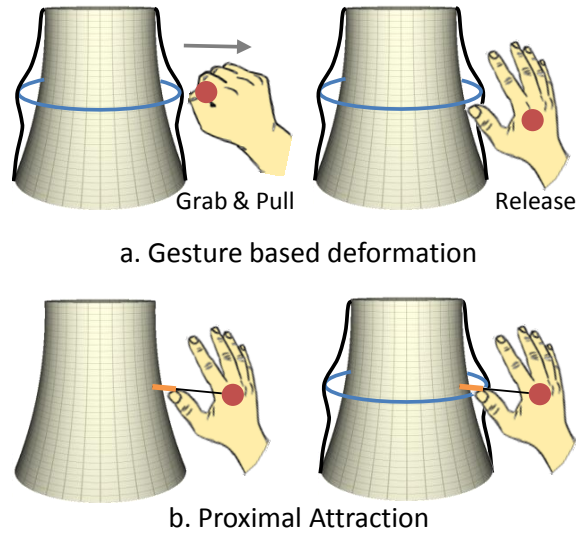


Figure 5.2. Two strategies are shown for clutching and deforming a pot using hand as a single point. In the first approach (a) grab and release gestures. The second (b) is the proximal-attraction approach.

main step was to determine the right balance between the rate of attraction and the distance threshold. The responsiveness of deformation is directly proportional to both, attraction rates and distance threshold. From our pilot studies, we found $\alpha = 0.3$ and $\varepsilon = 0.05$ to be the optimal values. Here, the distances are in the normalized device coordinates. In our current implementation, we pre-defined the active region A to be 50% of the total profile length.

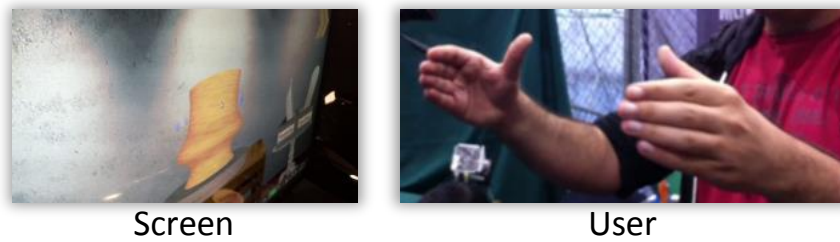


Figure 5.3. An example of common behavior is shown wherein users shaped their hands to express their intent for deformation.

5.3.2 Preliminary Evaluation

Our main goal was to examine the feasibility and effectiveness of the proximal-attraction approach for pot shaping in terms of user performance and behavior. We also wanted to determine the differences between our method and a typical gesture-based approach. Additionally, we wanted to understand the reception of a creative application such as pottery for a wide variety of participants - particularly those without prior knowledge of CAD tools. For this, we conducted a two-day field study ² in an exhibition setting.

5.3.2.1 Apparatus

Our hardware setup consisted of a ThinkPad T530 laptop, a 60" display, and the Microsoft Kinect camera. The Kinect camera was placed on a tripod below the display facing a user standing at a distance of around 1.5 – 2.0 meters from the display. Our pottery prototype was developed in C++ and OpenGL.

5.3.2.2 Implementation

We implemented two versions of our pottery application, one using mid-air gestures and the other based on the proximal-attraction approach. We first obtained the position of the hand using the skeletal tracking algorithm provided by the *openNI* API. Owing to the nature of the venue, the study was not conducted in a controlled environment leading to disturbances in skeletal tracking, posture recognition, and ambient noise. Thus, appropriate measures were taken to isolate the user from the audience.

The gesture-based prototype uses two simple hand postures, *grab* and *release*, which correspond to closed and open palms respectively (Figure 5.2(a)). We used the random forest algorithm for posture recognition as detailed in [141]. The grab and release postures allowed the user to *clutch* and *de-clutch* a certain region of interest on the pot. The user could create concave and convex profiles of the pot by *grab-and-push* and *grab-and-pull*

²MakerFaire, Bay Area (2013)

Table 5.1. Behavioral observations in our preliminary evaluation.

Age	Value	Behavior
5-10	Fun, Play	Excitement, Random hand movements
11-15	Entertainment, Education	Controlled movements, Explored tool features
16-30	Entertainment, Art, Education	Controlled movements, Investigated pot behavior
30-60	Entertainment, Meditative	Controlled movements, Expected real-world like response

actions at the desired location of the pot surface in 3D space. In the second prototype, we implemented our proximal-attraction technique (Figure 5.2(b)).

5.3.2.3 Participants & Procedure

Participants within a wide age range (5-60 years) were invited to use our pottery prototype wherein, the task for each participant was to create a pot as per the participant's liking. Although we did not carry out a formal demographic survey, we found that the participants were from a variety of backgrounds including non-technical users, engineers, designers, artists, and professional potters. Our evaluation was mainly informal and observational wherein we recorded videos of sessions subject to the participant's permission and the time taken to complete the creation of a pot. Due to the nature of our venue, we constrained the maximum time for each participant to about 8-10 minutes.

A total of 360 participants responded to our invitation and used our prototype to create pots. In the first session (day 1), 180 participants used the prototype implemented using the grab and release gestures. In the second session (day 2), 180 participants used the proximal-attraction technique for pot deformation. There were participants that were either completely unable to create any meaningful shape of the pot or did not find the resulting shape as the intended one. These attempts we removed from our database leaving us with the recorded times for 113 participants per session (i.e. 226 participants in total).

5.3.3 Results

We categorized the perceived utility and user behavior during the use of the pottery applications on the basis of age. Young participants (5-10 years) were mostly interested in simply playing around with the application and usually applied arbitrary hand movements during the deformation of the pot's profile. Participants in the age range of 11-15 years provided more controlled movements of the hands during pot shaping with slower and more careful hand movements and accurate hand gestures. They also adopted a more exploratory approach towards the applications in that they were primarily interested in the various software features rather than the realism in the pot's deformation.

However, in case of participants above the age of 15, we observed that they instinctively shaped their hands according to geometry of the pot on the screen. Specifically, users within 16 and 30 years of age were mainly interested in investigating how the gesture and motion of the hand was related to the deformation of the pot. They would frequently expect the pot to deform according to how they shaped and moved their hands on the pot's surface. This strongly suggested that the internal learning of physical interactions, combined with some prior expectation of the pot's response, increased with the participants' age. In case of the gesture-based approach, this was also a cause for intermittent gesture misclassification, resulting in user frustration. Despite their simplicity, the *grab* and *release* gestures were tedious to use while using virtual tools. This was mainly the case with participants who were completely new to interfaces developed for RGBD cameras.

On the other hand, users found the proximal-attraction approach easier to learn and use. The participants could immediately start deforming the pot, and at the same time they could shape their hands as they saw fit. A common mental model that the users seemed to create was that of a surface which "*sticks*" to their hands upon coming close. Thus, the users were invariably slower while approaching the pot (so as to reach the right location) and retreated faster when they wanted to release contact with the pot. For some users, fast retreat also caused accidental deformation leading to frustration.

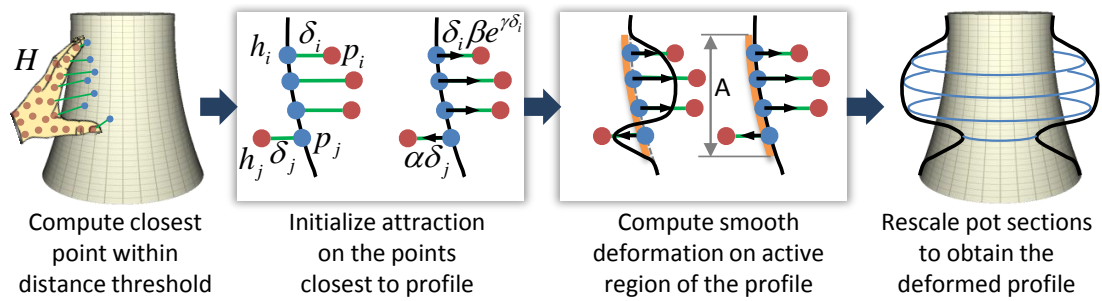


Figure 5.4. Algorithm for pot deformation is illustrated for proximal-attraction. The profile is deformed based on the proximity of the points on a given hand PCL. Subsequently, each section is re-scaled to obtain the deformed pot surface.

5.3.4 Takeaways

The two main insights we gained were: (a) the intent for deformation directly translates to how users shape their hand and (b) the rate of attraction for pulling and pushing must be determined separately so as to make them consistent. We found that full-body interactions caused significant fatigue and difficulty in controlling deformation. Thus, our subsequent stages, we implemented interactions at close range wherein a user could perform pottery sitting in front of a desktop or a laptop computer.

5.4 Hand as a PCL: Shaping by Proximal Attraction

Our main objective in this stage was to adapt the proximal-attraction method that could use the shape of the whole hand to deform the pot. Thus, we used a representation of the hand as a collection of multiple points (i.e PCL) obtained via a depth sensor³.

5.4.1 Technique

Consider the hand H as a set of points $\{h_i\}$ in 3D space. Each point in the PCL deforms a small local region on the pot using the proximal-attraction approach. On the whole this

³See Chapter 4 for complete technical details

amounts to a gradual and progressive convergence of the pot-profile to the shape of the user's hands (Figure 5.4).

5.4.1.1 Pushing vs. Pulling

A push is characterized by an inward displacement ($\delta < 0$). This is the simplest case wherein a user would typically approach the pot and subsequently recede away once the desired deformation has occurred. A pull is characterized by an outward displacement ($\delta > 0$). This is a non-trivial intent to recognize since a user would invariably approach the surface first and then recede to pull. The overall motion of the hand is similar to that of a push. In order to distinguish pulling and pushing, we used two different rates of attraction. For pulling, we defined the attraction rate as a smooth function of the distance between the hand point and pot. The function is given by $\beta e^{\gamma \delta_i}$. For pushing, we defined the rate of attraction as α . This essentially allows the user to first approach the pot without deforming it during the process of approach. The algorithm is as follows:

1. For each section i

Compute unique h_i such that $\|h_i - p_i\| < \varepsilon$ is minimum.

Set δ_i to horizontal distance between h_i and p_i

2. Set δ_r to $\delta_{max} - \delta_{min}$

3. Set γ to $\frac{0.1}{\delta_r}$

4. For each i on profile

if($\delta_i < 0$): Set attraction at p_i to $\alpha \delta_i$

else: Set attraction at p_i to $\beta e^{\gamma \delta_i} \delta_i$

5. Compute Active region A

6. Smooth deformation ($\nabla^2 \delta = 0$ for all points in A)

7. Compute deformed profile
8. Rescale pot sections

5.4.1.2 Initialization Time

In order to avoid accidental or unintended deformation of the pot, we implemented an that allows for the pot to deform only when contact with the pot is maintained for a sufficient amount of time. We achieved this in two steps. First, we reset α and β to 0 at every new contact that the hand made with the pot. Subsequently, we linearly increase them to their maximum values within a stipulated amount of time T . We call this the initialization time. Intuitively, T is the time taken by the pot to gradually initiate the response to the user's hand after a contact is made.

5.4.2 Experiment

The main motivation behind our user evaluation was to understand was the relationship between the **design outcome** (“what users want in the end”) and **design process** (“how they want to get there”). In order to understand how users perceive and perform mid-air shape deformations, we conducted a lab experiment to evaluate the proximal-attraction approach. Our primary goals in this evaluation were: **(a)** to observe common and uncommon user patterns during the shaping process in terms of hand grasp and **(b)** to get user's feedback on the effectiveness of controllability. The results of this experiment led us to develop the final approach for characterizing grasp and motion for the deformation of pots. To cater to these goals, we performed an observational and user perception analysis. In order to support our observations and findings, we first wanted to characterize how well and how fast users could create basic shape features using our proposed algorithm. To assess the quality of shapes created by the users in relation to the time taken for creating the shapes, we utilized an approach based on curvature cross-correlation. Below, we describe our evaluation in detail.

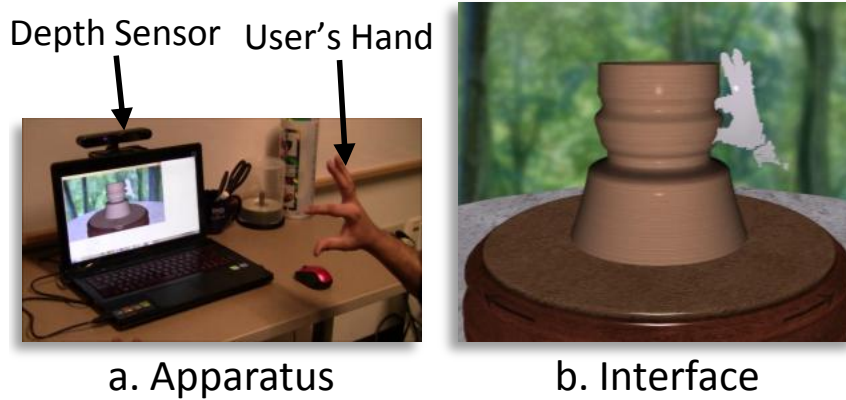


Figure 5.5. The apparatus (a) consists of the user, a computer and a depth camera. The user sees a PCL of their hand deforming a rotating pot (b).

5.4.2.1 Apparatus

Our setup consisted of a Lenovo ideaPad Y500 laptop computer with an intel i7 processor and 8GB RAM, running 64-bit Windows 8 operating system with a NVIDIA GeForce GT 750M graphics card, and the SoftKinetic DS325 depth sensor (Figure 5.5(a)). SoftKinetic DS325 is a close range (0.1m-1.5m) time-of-flight depth sensor that provides a live video stream of the color and depth image of the scene. Every pixel on a given depth image can be converted to a 3D point using the camera parameters.

5.4.2.2 Implementation & Interface

After segmenting the hand from the scene, we use the SoftKinetic iisu API for tracking the hand PCL. However, the tracking method provided in this API does not work with hand-held objects - a feature that we required in order to allow users to utilize physical objects for deformation. Thus, we used a pre-defined a volumetric workspace as the active region in front of the computer screen. Our interface comprises of a 3D scene with a rotating pottery wheel on natural outdoor background (Figure 5.5(b)). The user sees the potter's wheel and the PCL of their hands, or the tools held in their hands. We designed this interface based on

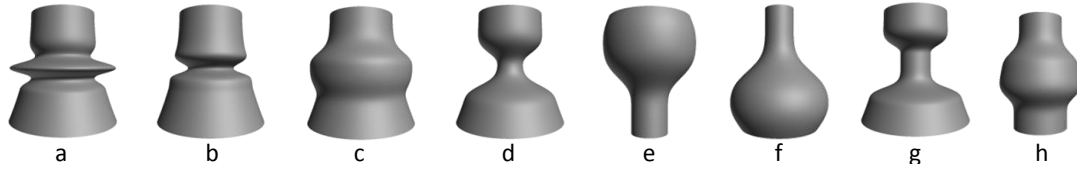


Figure 5.6. Eight pre-defined pots were shown to participants in the quiz. These are: (a, b) thin convex and thin concave, (c,d) fat convex and concave, (e, f) round and flat, and (g, h) flat at center and ends.

the guidelines provided by Stuerzlinger and Wingrave [142]. Finally, we provided keyboard shortcuts to allow the participants to undo and redo a particular deformation at any time. Additionally, we also made provisions for the participants to reset the current shape to the blank pot.

5.4.2.3 Participants

The participants of this evaluation comprised of 15 (13 male, 2 female) science and engineering graduate students within the age range of 20 – 27 years. Out of the 15 participants, 5 participants self-reported familiarity with mid-air gestures and full body interactions through games (Kinect, Wii). Due to engineering background, most participants (12 of 15) reported familiarity with 3D modeling and computer-aided design. Incidentally, we also had 3 participants who had prior experience with physical ceramics and pottery.

5.4.2.4 Procedure

The total time taken during the experiment varied between 45 and 90 minutes. We began the study with a demographic surface where we recorded participants' background regarding their familiarity with depth cameras, full-body games, and pottery. Subsequently, we provided a verbal description of the setup, the purpose of the study, and the features of the pottery application. This was followed by a practical demonstration of the pottery appli-

cation by the test administrator. The participants were then asked to perform the following tasks:

- P** *Practice*: To get an overall familiarity with the interaction of their hands with the pot surface, each participant was allowed to practice with our interface for a maximum time of three minutes. The participants were allowed to ask questions and were provided guidance when required.
- T1** *Quiz*: A pre-defined target shape was displayed on the screen and the participant was asked to shape a “blank” pot so as to roughly match the most noticeable feature of the target shape. We showed a total of eight target shapes in a randomized sequence (Figure 5.6). The participants were allowed to undo, redo, and reset the pot at any given time and for as many times as they required.
- Q1** *Questionnaire 1*: Each participant answered a series of questions regarding the association of the deformation to the shape of the hand, responsiveness of the deformation, and consistency of pushing and pulling.
- T2** *Composition*: The participants were asked to think of (and verbally describe) a set of *intended* pot shapes and subsequently create those shapes using their hands. Although the maximum duration of time for each shape was fixed to five minutes, we allowed the participants to complete their last composition that was started before the end of the specified duration.
- Q2** *Questionnaire 2*: Finally, each participant answered a series of questions regarding enjoyability, ease of use and learning. The participants also commented on what they liked and disliked about the application, interface and interaction.

5.4.3 Metric for Quality of User Response

The main aspect that we sought from the *Quiz* was the quality of the final outcome across participants for a given quiz problem. We also wanted to understand what features

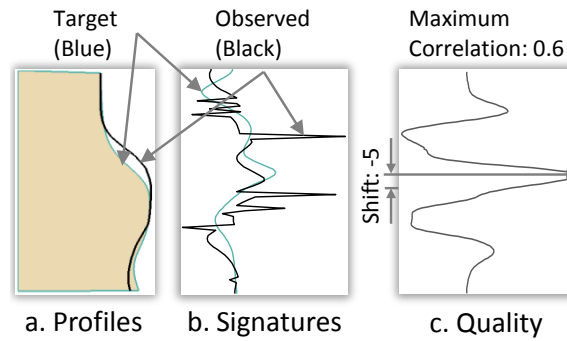


Figure 5.7. Curvature cross-correlation is used as measure of quality of user created pots. Given the observed and target profiles (a), we compute curvature signatures (b), and subsequently compute maximum correlation between the signatures (c). Notice the sensitivity of the curvature signature (b) for a seemingly high visual similarity. In the example shown, the profile of the observed pot are shifted upwards by 5 sections with respect to the target pot.

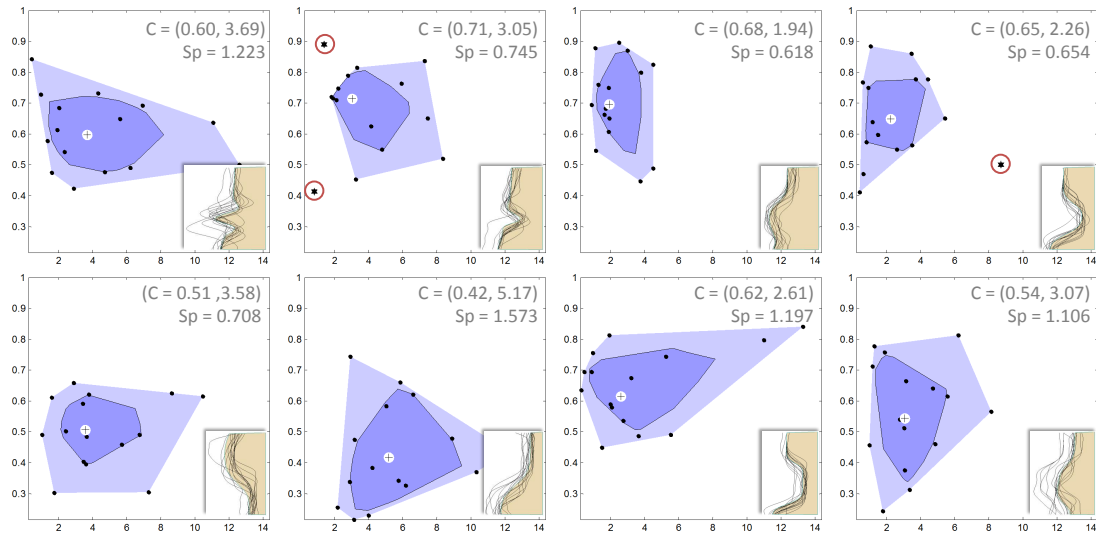


Figure 5.8. User performance is shown for the each quiz problem as a bag-plot. The x-axis is time in the range $[0, 14]$ minutes and the y-axis is the curvature cross-correlation in the range $[0, 1]$. The dark and light blue regions show the bag and fence regions, respectively. The white circle is the Tukey depth median and the points marked with red circles are the outliers. The insets show the actual pot profiles (black lines) created by the users in comparison to the target shapes (beige region) of the Quiz. The coordinates of the tukey median (C) and the spread (Sp) are provided for each target shape.

were difficult for the users to create, i.e. a comparison of the final outcomes across features. Thus, we sought to compare experimentally observed (user created) pot profiles with respect to the target shapes (quiz problems). The requirements for our metric were: (a) invariance to the shift between the features of the profiles along vertical and horizontal directions and (b) sensitivity to capture small local dissimilarities across users. Thus, we used curvature cross-correlation as a measure to compare the quality of user created profiles. We first compute the curvature signature of an observed profile wherein each point on the signature is the curvature of a point in the profile (Figure 5.7(a)). Subsequently, we compute the normalized cross-correlation [143] between the curvature signatures of the observation with that of the ground truth (Figure 5.7(b)). The quality is then defined as the maximum value of the correlation (Figure 5.7(c)). Since the signatures are normalized before cross-correlation, the value of the measure of quality is in the range $[0, 1]$. Here, higher values represent better quality (1 corresponding to perfect match and 0 no match).

5.4.4 Results

5.4.4.1 User Performance (T1)

Each user perceived and approached a given target shape; there was no observable correlation between the time taken by each user and the quality (curvature cross-correlation) of the final pot created by the user for any of the target shapes. Hence, we chose to represent the user performance as a bag-plot [144] (Figure 5.8), where the time taken and the response quality are considered as independent variables. Rousseeuw et al. [144] state: *Like the univariate boxplot, the bagplot also visualizes several characteristics of the data: its location (the depth median), spread (the size of the bag), correlation (the orientation of the bag), skewness (the shape of the bag and the loop), and tails (the points near the boundary of the loop and the outliers).* In this section, we will identify notable aspects of user response by looking at the location, spread, and skewness of the bag-plots for each of the quiz questions. In the subsequent sections, our goal will be to report our visual obser-

vations of user behavior and usage patterns across the different quiz questions based on the bag-plot observations.

Users performed best for thin-concave and fat-convex targets with Tukey median values of (3.80, 0.71) and (1.90, 0.69) respectively. In particular, the fat-convex case shows a nearly vertical orientation indicating a strong correlation between the time taken and the quality of response. The average time taken by users was highest for round-flat features with significant variation in the quality. As expected, the thin-convex target feature was difficult to shape. This is also shown by the large spread (1.223) and the Tukey median of (0.60, 3.69). Contrary to our expectation, the top-flat-bottom-round feature (Figure 5.6(f)) was most difficult for the users to create, as indicated by the maximum spread (1.573) and lowest Tukey median of (0.42, 5.17). Similar difficulty was observed in the central flat feature (Figure 5.6(h)). The spread was consistently higher for all flat features (bottom row of Figure 5.8) indicating that the key problem the users faced was due to the lack of an explicit method for *smoothing* the surface of the pot. We found that the quality of responses in relation to the completion time are closely linked to how the users perceived correspondingly approached the shaping process. Below, we provide a detailed description of how the measurements of the response quality and time are related to the behavior and strategies of the users for approaching, grasping, and deforming the pots.

5.4.4.2 Hand Usage (T1)

Users generally preferred small finger level movements for thin features. For fat and flat features, we observed that the users first formed a grasp according to the amount of deformation required and then moved the whole hand to achieve the feature as expected [134]. Pushing generally required smaller finger movements in comparison to whole hand movements. Most users spent time smoothing and refining the surface of the pot after the general shape had been obtained. The motion of the hand was performed vertically along the surface of the pot (Figure 5.9). This led to frustration, for two reasons. First, the accidental unintended deformations caused due to the contact of the hands with regions of the pot other

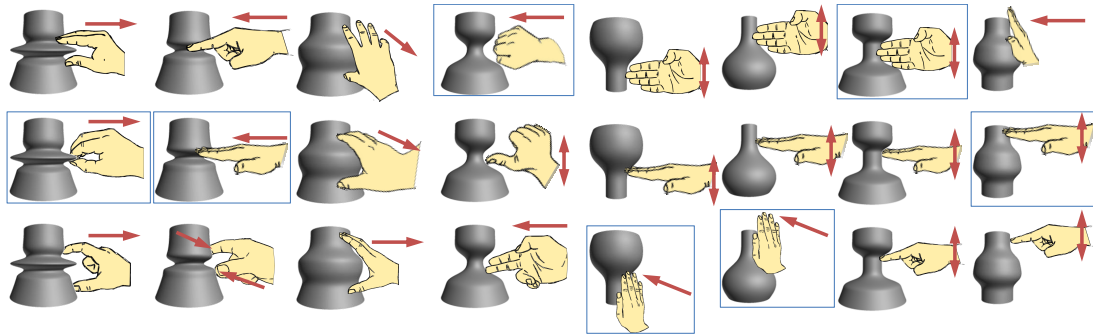


Figure 5.9. Common user patterns are shown in terms of grasp and motion performed by users for each target shape (in decreasing order of occurrence along columns). The hand images represent the grasp and the arrows (red) show the motion of the hand. The most successful strategies are indicated by blue boxes for each target shape.

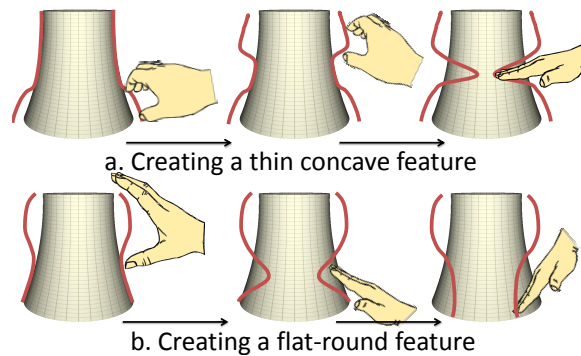


Figure 5.10. Two examples are shown of common deformation strategies are shown through which users created (a) thin concave and (b) flat-round features.

than the parts which the users were attempting to refine. Secondly, although our algorithm allowed for smooth deformations, there was no explicit way for the users to smooth or straighten a region of the pot. Another interesting observation was that most users avoided using the key-board commands for undo, redo and reset. Instead, they preferred using their hands for reversing an accidental deformation. In some cases, users had to be reminded of the undo, redo and reset functionalities.

5.4.4.3 Hand Usage (T2)

On average, users created 5 pots (max: 9, min: 3) within 5.75 minutes (std. dev.: 1.00 min). Most users felt that the control of the pot was significantly better during the composition phase. One user stated: *“I thought it was easier to learn the software when I was trying to make my own pot not a model one”*. This was expected because of the learning and practice that the users had during the quiz. However, according to the users’ comments, the cause of difficulty in the quiz turned out to be split attention between the target shape and the user’s pot. There were two common observations that we made. First, almost all users tried making a pot that they perceived to be the most difficult ones to make. Surprisingly, these were the round-flat combinations (Figures 5.6(e),(f)), rather than the thin convex one (Figures 5.6(a)). Most of the users tried to make pots with large and straight stems, such as in a wine glass or chalice wherein most of their effort went in smoothing and straightening long vertical regions of the pot. This, in conjunction with their difficulty in the *Quiz*, strongly indicated the need for an explicit method for recognizing the intent for smoothing or fairing the surface of the pot.

5.4.4.4 Reaching, Grasping, & Deformation Strategies

Each user had a different perception of the process necessary to achieve the profile of a given target shape. Most users attempted the quiz problems in multiple trials, wherein they would refine their strategy to deform the profile in every trial. However, we observed that these strategies of reaching, grasping, and deforming the profile converged to patterns common across users (Figure 5.9). Typically, users would first estimate the size and shape of the grasp according to the geometric feature of the profile and then move the whole hand in the intended grasp to deform the profile [134]. The most common usage pattern observed across users was the recursive smoothing and refining of the pot after deforming the profile reasonably close to the target shape. This was typically done by moving the hand vertically along the surface of the pot (Figure 5.9). This was the cause of frustration for two reasons. First, the accidental contact of the hand with the pot’s surface resulted

in unintended deformations. Second, the proximal attractions did not allow for an explicit way to smooth or straighten a region of the pot. Despite being reminded of the undo, redo, and reset functionalities, most users preferred using their hands for reversing an accidental deformation.

For the thin-convex profile, most users first created a convex feature in the center followed by pushing the top and bottom portions inward. For concave features, users first pulled the top and the bottom portions of the pot and subsequently pushed the central region of the pot (Figures 5.10(a)). This was an interesting common pattern since we had assumed that users will create concave features in a single inward action. This was also the case with flat-round features (Figures 5.10(b)) wherein many users first pulled out the round feature followed by straightening the flat regions of the pot. The *pointing* posture of the hand was commonly observed during the creation of thin concave features. However, in subsequent trials, most users resorted to using an open palm. This was because the *pointing* pose limited the depth to which the users could push the surface inwards, owing to the interference of the fingers other than the index finger. The cupping of the hands in conjunction with vertical movement of the hands was a common approach for round features.

The use of two hands was particularly prevalent for round-flat combinations. Due to arm fatigue, some users also changed from their dominant hand to the non-dominant hand. This was a cause for frustration due to the limited volume of the workspace and unintended deformations caused by the asynchronous motions of two hands. Most users commonly approached the pot from the sides. The reason, as stated by a user, was: ” *“my own hand blocks the view of the pot”*. Difficulty in depth perception caused many users to inadvertently reach behind the pot’s surface. This caused further unintended deformations when the user did not expect one, or the lack of response when it was expected.

5.4.4.5 Intent & Controllability (Q1)

In general, users agreed that the shape of the profile behaved in correspondence to shape of the hands (Figure 5.11(a)). However, only 50% of the users agreed that the re-

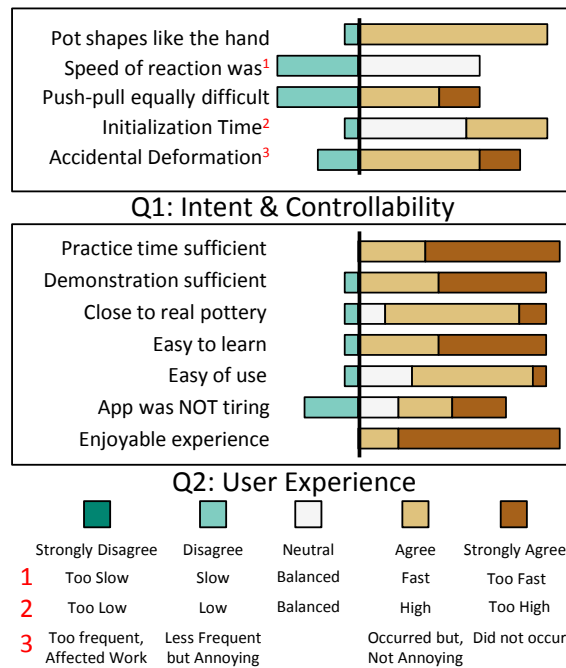


Figure 5.11. User response to are shown for proximal-attraction. The main issue in terms of controllability (a) was the slow response and difficulty of pushing in comparison to pulling.

sponse speed of the deformation was balanced. There was a common agreement on the initialization time and robustness to accidental deformation. There were two common and expected difficulties that the users faced. These were: (a) pulling specific regions of the pot and (b) creating straight and flat features on the top portion of the pot. As a user stated: *“Pushing seems easier than pulling. Part of the reason I suspect is the visual feedback. It is easier to determine if my hand starts to touch the pot, while it’s not as easy to determine if my hand is still attached with the pottery or leaving it.”*. This indicated that perceiving the depth difference between the hand and the pot was difficult for the users.

5.4.4.6 User Experience (Q2)

Despite the frustrations with pulling and smoothing, most users enjoyed the non-symbolic aspect of the interaction. According to a user: *“I enjoyed the lack of constraints in the design process; free-forming”*. Another user commented: *“This is the first time I have seen something like this. The app is very sensitive so that way it gives the user a lot of freedom. i really loved that part”*. The main aspects that the users liked were (a) realism of pottery, (b) ease of learning, and (c) the freedom of choosing how to deform the pot (Figure 5.11(b)).

5.4.5 Takeaways

There were two main issues with the proximal-attraction approach. First, pulling was clearly more difficult since the rate of attraction was *designed* to be lower than that of pushing. Secondly, the users clearly distinguished between several operations of fairing, straightening, carving, pulling and pushing. However, the proximal-attraction approach, was not designed to explicitly identify or classify the type of operation the user intended to perform. Our main goal in our third and final stage was to resolve these two issues. Our first step was to identify the main characteristics of users’ preferences towards grasping to pull and motion patterns for smoothing the pot. Subsequently, the aim was to design a geometric approach that could recognize these identified characteristics and broadly classify the intended actions from the hand PCL.

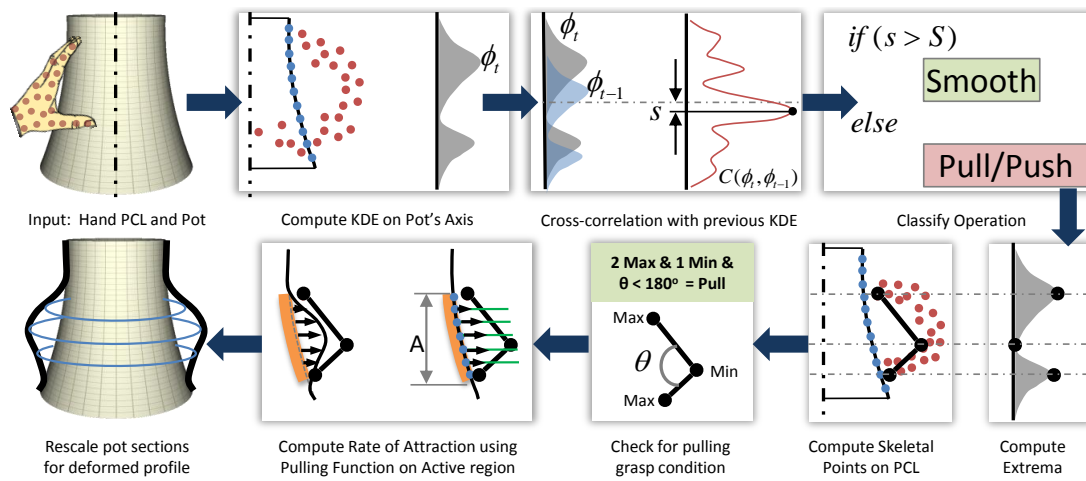


Figure 5.12. Algorithm for grasp+motion technique is illustrated. The main steps involve computation of axial KDE for hand PCL, detection of intent for smoothing, differentiation between pulling and pushing, and deformation of the pot. In this example, we show the details of the pulling deformation (row 2).

5.5 Hand as a PCL: Grasp + Motion

Our observations strongly indicated that users distinguished their intent in three broad categories: pulling, pushing, and smoothing. In our final stage, we implemented a grasp and motion based approach to identify these three classes of intent.

5.5.1 Technique

The basic idea of the *grasp+motion* approach is to *summarize* the grasp of the hand *in relation to* the surface of the pot and subsequently classify the user's action (Figure 5.12). We achieve this by using kernel-density estimation of the point cloud on the axis of the pot. In our context, this kernel-density estimate (KDE) is essentially a smoothed histogram of the distribution of the hand's PCL on the pot's. We use the exponential function to determine the KDE. For a given section i , the KDE is given by:

$$\phi_{i,j} = \sum_{j=1}^{j=|H|} e^{a\|\delta_{i,j}^2\|} \quad (5.1)$$

There were three main observations (Figure 5.9) that helped us use the KDE to classify the user's intent. First, users moved their hands in a fixed pose along the surface of the pot to express their intent for smoothing. This corresponds to detecting the vertical shift of the KDE. We used normalized cross-correlation [143] between the two consecutive KDE signals to determine the shift. Secondly, for pulling the pot, we observed that users used specific grasps. In this case, we note that the KDE has two maxima and one minima (Figure 5.13(a)). Here, each maxima corresponds to the fingers making contact with the pot and the minima corresponds to the center of the grasp. This essentially allows us to track a basic skeletal representation of the hand. We then define the attraction rate using a based on the angle of grasp (ϕ) (Figure 5.13(b)). Finally, all actions that do not correspond to either smoothing or pulling, are assigned as pushing. For pushing, we use the proximal-attraction approach for deformation. The steps of the algorithm are:

1. Compute the KDE ϕ_t at time t

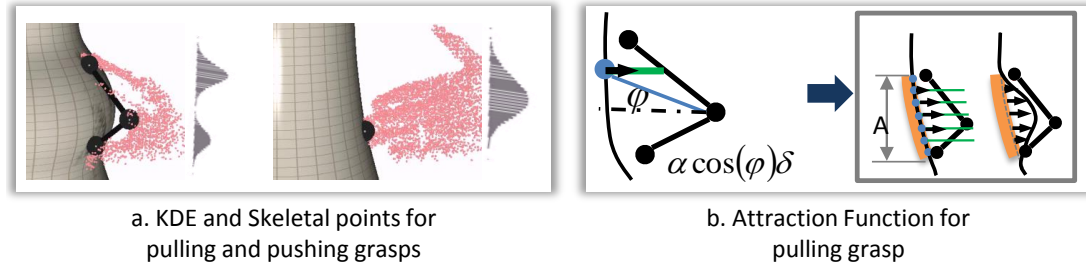


Figure 5.13. a. KDE functions are shown for a pulling (left) and pushing (right) intents, b. Computation of attraction rate using the angle of grasp is illustrated.

2. Compute normalized cross-correlation $C(\phi_t, \phi_{t-1})$
3. Compute Active region A
4. Set s to the shift of correlation
5. if($s < S$): Smooth pot profile in A
6. else:
 - Compute extrema
 - Detect skeleton
 - Compute θ
 - if($\#maxima = 2$ & $\theta < 2\pi$): Apply pulling in A
 - else: Apply proximal-attraction in A
7. Smooth deformation ($\nabla^2 \delta = 0$ for all points in A)
8. Compute deformed profile
9. Rescale pot sections

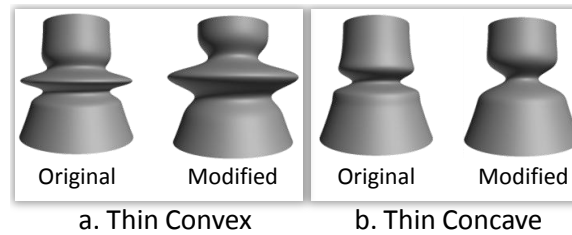


Figure 5.14. The thin convex and concave features were modified according to the capability provided by the *grasp+motion* technique.

5.5.2 Experiment

We used identical apparatus and interface to evaluate our final stage. Additionally, we made two important modifications to the interface. First, we added a shadow of the hand on the surface of the pot. The goal was to enable users to estimate their proximity to the surface. Secondly, we clamped the hand PCL so as not to allow points on the hand to reach behind the surface of the pot.

5.5.2.1 Participants

We recruited 15 (11 male, 4 female) participants within the age range of 19 – 30 years. None of these participants had prior knowledge of mid-air interactions or had participated in any of our previous studies with pottery interface. All participants were from science and engineering background wherein 10 participants had familiarity with mid-air gestures and full body interactions, and 11 participants reported familiarity with 3D modeling and computer-aided design. 5 participants reported that they had practical familiarity with real ceramics via informal workshop sessions but did not pursue pottery as a regular activity or professional practice.

5.5.2.2 Procedure

Our overall experimental procedure was identical to the one that we used for evaluating the proximal-attraction approach (Section 5.4.2, *Procedure*). However, we made three modifications to the evaluation procedure as listed below:

1. One of the main goals of our work was to enable users to invoke their tacit knowledge of deforming physical objects. To this end, we designed the grasp+motion approach such that it is geometrically-driven and can potentially be used even for user inputs that used other physical objects as tools in addition to the use of hands. In order to verify the generality of our approach with respect to user input, we added another composition task (**T3**) wherein participants were given a duration of five minutes to create pots using a set of physical artifacts as tools. Our “tools” comprised of day-to-day objects (e.g. white-board marker, pair of scissors, ruler) and also some special objects such a Shapescapes^{TM4}.
2. In order to understand user experience with physical objects tools, we also added questions to the questionnaire **Q2** regarding the utility, ease of use, and preference of tools over hands.
3. We modified the target shapes for the thin convex and concave features (Figure 5.14). The rationale behind this modification was that the graph+motion technique is sensitive to the size of the hands, finger thickness. Thus, the detection of single-point pulling intent is not possible, as in the case of proximal-attraction.

For each participant and task (**T1**, **T2**, and **T3**), we recorded the completion time and the profiles of the pots shaped by the users. Even though we designed **T1** towards statistical analysis, we observed that each user perceived the target shapes differently and consequently the measured data did not provide sufficient insights regarding the strengths and weaknesses of our approach. With this in view, we present a visual comparison of the

⁴www.shapescapes.com

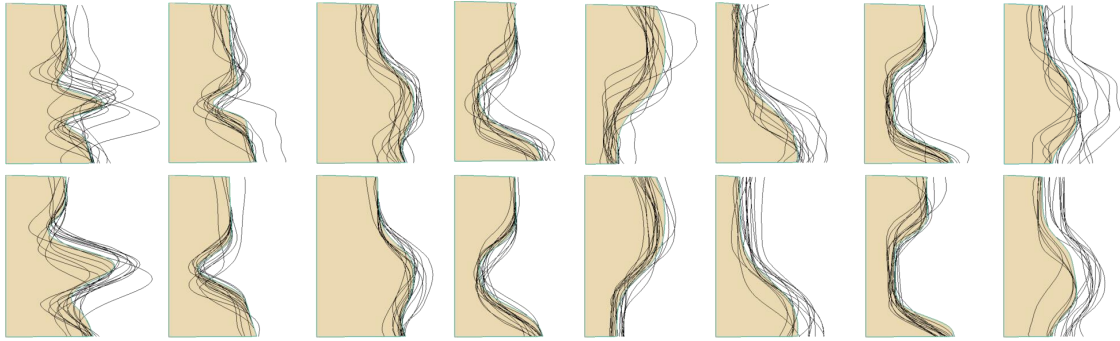


Figure 5.15. User created pot profiles (black curves) are shown relative to the target shapes (light brown cross sections). The top and bottom rows shows the results for proximal-attraction and grasp+motion approaches respectively. Visual inspection evidently shows improvements in the creation of flat, round and smooth features. More significant improvements were observed in the creation of fat convex features in comparison to proximal-attraction.

numerical data recorded during the evaluation of proximal-attraction and grasp+motion techniques.

5.5.3 Results

5.5.3.1 User Performance (T1)

Visual similarity with respect to the target shapes evidently increased in comparison to the proximal-attraction approach (Figure 5.15). This was primarily due to the explicit smoothing. Overall, the completion time (Figure 5.16(a)) was reduced as expected. Surprisingly, the maximum completion time across all users and all target shapes was recorded for the thin-concave feature (14.4 minutes) followed by the thin-convex feature (13.2 minutes). The mean completion time was highest for the thin-convex feature (3.4 minutes) followed by the central-flat feature (3.3 minutes). The main aspect that we sought from **T1** was the quality of the final outcome across participants for a given quiz problem. We used curvature cross-correlation (*CCC*) as a measure of the quality of user created profiles (see [145] for details). As expected, the smoothness of the results was notably superior

in comparison to the proximal-attraction (Figure 5.16(b)). We also recorded the number of trials per user per target shape (Figure 5.16(c)). The global maximum number of trials were 7 and 5 for proximal-attraction and grasp+motion techniques respectively. In case of grasp+motion, most users required only one trial for fat-convex, central-flat, and top-bottom-flat features. On the other hand, thin-concave and thin-convex features required more iterations.

Each user perceived and approached a given target shape in different ways. Consequently, there was no evident correlation between the time taken by each user and the quality (*CCC*) of the final pot created by the user for any of the target shape. To account for this, we represent the user performance as a bivariate dataset given by the ordered pair of the response quality and completion time. We visualize performance as a bag-plot [144] (Figure 5.17). Here, the spread of the data (i.e. variations in user responses) is given by the area of the *bag*. Users clearly performed best for thin-concave targets with Tukey median value of (0.94, 1.46). Performance was most consistent for the fat-concave feature (Figure 5.17(d)). Users also performed consistently for round-and-flat features (Figures 5.17(e) and (f)). Variations were significant for central flat feature (Figure 5.17(g)). Further, the pot-profile quality was very low for the central-flat and top-bottom-flat features (Figures 5.17(g) and (h)). This was mainly because users typically spent considerable time pulling and smoothing the top and bottom regions after performing an initial push. Consequently, the median completion times were also higher for the round-flat and central-flat features (Figure 5.17(f) and (g) respectively).

5.5.3.2 Hand Usage (T1)

The general user behavior in terms of reaching the pot was similar to the proximal-attraction approach. Both the algorithm and its description was different in this case. The users were explicitly made aware of pushing, pulling and smoothing as three distinct operations. This obviously led to variation in user behavior as compared to proximal-attraction.

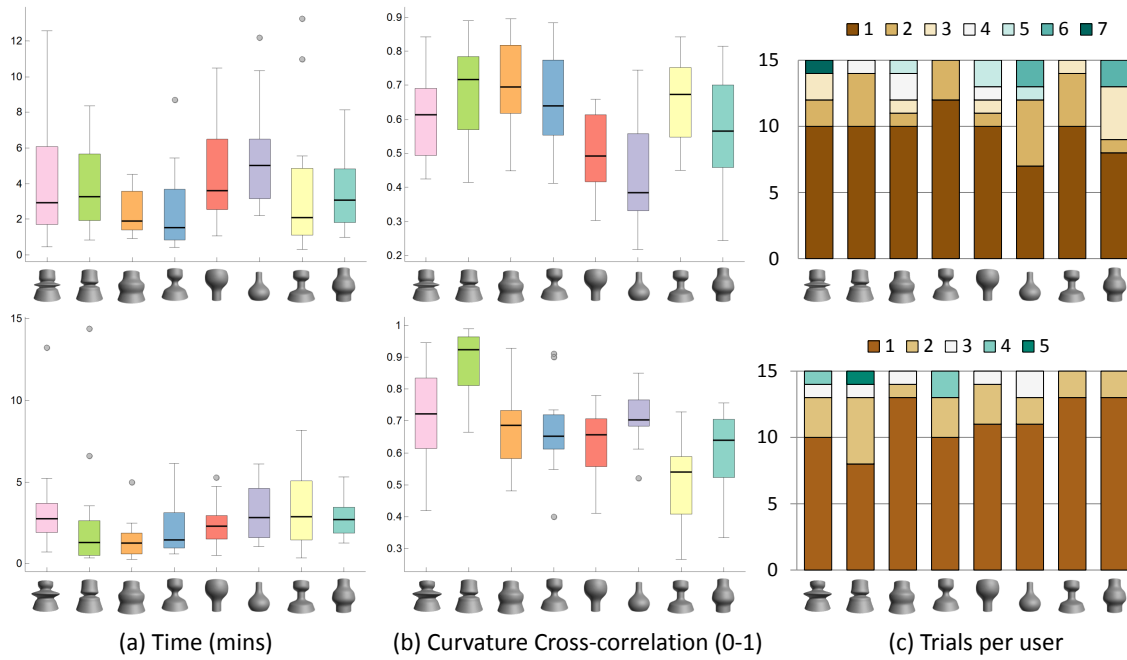


Figure 5.16. A comparison between proximal-attraction (top row) and grasp+motion (bottom row) is shown in terms of (a) the time taken by users to shape a target profile, (b) the quality of users' responses in terms of curvature cross-correlation of profiles, and (c) the distribution of users with respect to the number of trials per target profile.

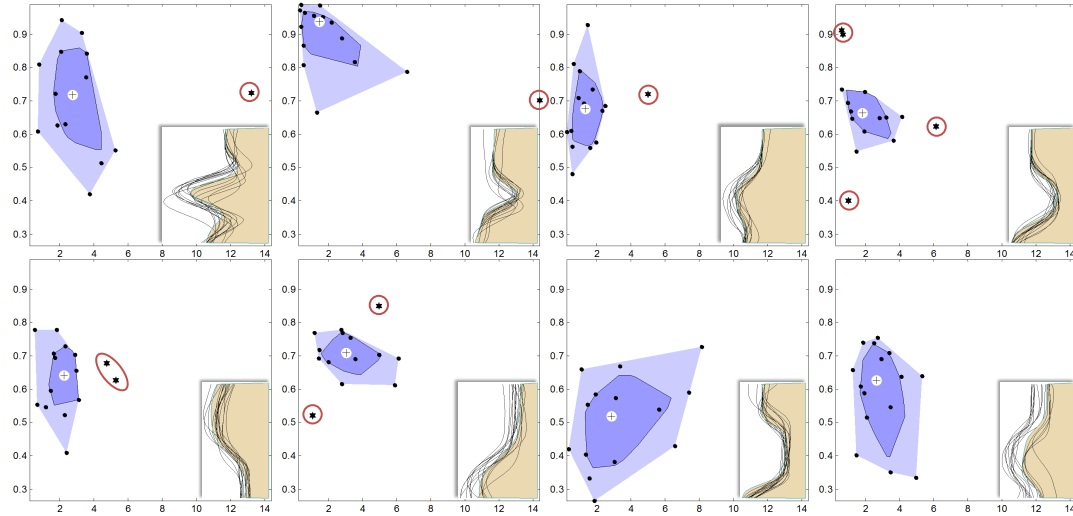


Figure 5.17. User performance is shown for the each quiz problem as a bag-plot. The x-axis is time in the range $[0, 14]$ minutes and the y-axis is the curvature cross-correlation in the range $[0, 1]$. The dark and light blue regions show the bag and fence regions, respectively. The white circle is the Tukey depth median and the points marked with red circles are the outliers. The insets show the actual pot profiles (black lines) created by the users in comparison to the target shapes (beige region) of the Quiz. The coordinates of the depth median (C) and the spread (Sp) are provided for each target shape.

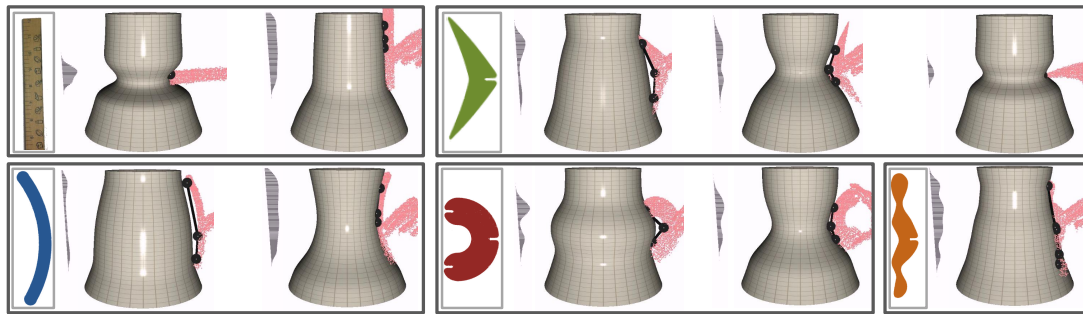


Figure 5.18. The characterization of tool geometry is visualized for five different physical objects. The objects were chosen to represent concave, convex, flat, and round contacts for deformation.

5.5.3.3 Hand Usage (T2)

On average, users created 5 pots (max: 12, min: 2) within 5.80 minutes (std: 0.66 min). We made two interesting observations in **T2**. First, we found that users were able to repeat the process of getting from an initial shape to the same final shape across multiple trials. Similarly the users could also deform a current shape back to some previous shape, akin to the *undo* operation, but with the hands. In fact, most participants preferred using their hands to *undo* a pot deformation instead of the keyboard-shortcut. One user stated: “*I thought it was easier to learn the software when I was trying to make my own pot not a model one*”. This was expected because of the learning and practice that the users had during the quiz (**T1**). However, during **T1**, users mentioned that their attention was divided due to the need to intermittently look at the target shape during the shaping process. Thus, they generally perceived **T1** to be more demanding than **T2**.

We made two observations that were not evident in the earlier stages. First, we found that the ability to repeat the process of getting from an initial shape to the same final shape. Similarly, the ability to get to some previous state from the current state was increased substantially. We observed that most of the participants were successfully able to use their hands to *undo* a pot deformation instead of the keyboard-shortcut.

5.5.3.4 Geometric Characterization of Tools

The choice of everyday objects and ShapeScapes™ was mainly helpful in providing a reasonable variety of geometric profiles for pot deformation. However, in order to better understand how users would use these objects, we wanted to pre-determine how the intent of pulling and pushing translates to the use of physical objects. Thus, we conducted a set of experiments (Figure 5.18) to verify if the users could in fact extend their understanding of the grasp+motion approach and apply it to the use of physical tools. Our experiments showed that the geometry of the tool can indeed be interpreted in terms of the nature of the KDE of the tool’s PCL and the grasping angle of the skeleton computed from the KDE.

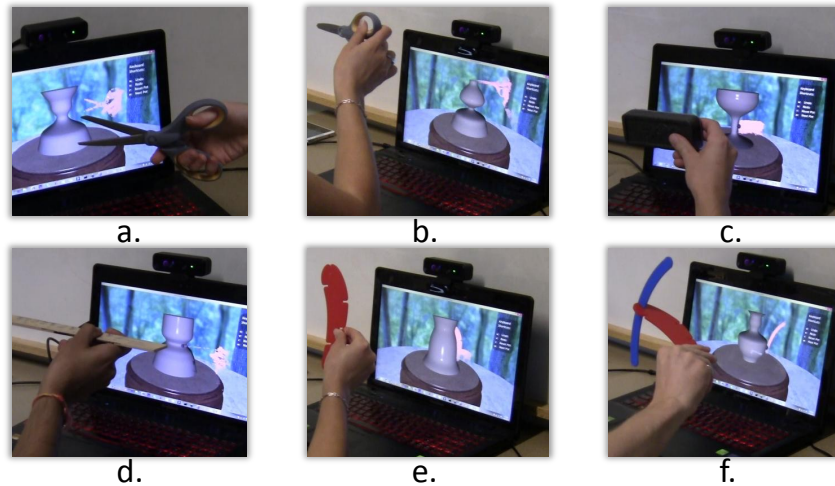


Figure 5.19. Examples of tool usage are shown.

Below, we summarize how this observation came into play during the usage of tools by our participants.

5.5.3.5 Tool Usage (T3)

Users showed immediate enthusiasm during the use of tools. Almost all users first inspected the objects provided to them and planned how to use them for shaping the pots. Users created 4 pots on average (max: 8, min: 2) within 6.0 minutes (std: 0.8 min). In contrast to the use of hands, we observed exploratory behavior in users while using tools. Rather than creating pots, most users were more interested in finding out the effect of each of the objects provided to them. This explained the decrease in the average number of pots in the composition task. One of the difficulties with the use of hands was the inability to create thin concavities. With the use of tools (Figure 5.19(a),(d)), users could achieve this easily. The most interesting behavior that was observed was the tendency to create convex deformations, which the users achieved by combining two different objects, so as to simulate a grasping hand. This was evident from the users' fascination with scissors (Figure 5.19(b)).

Another important observation was the direct association the users made between the shape of the tool and the purpose it could be used for. The motion of the hand was affected by this association. For instance, while using a white-board eraser (Figure 5.19(c)), the most common motion was that of smoothing the pot. Similarly, for objects with grasp-like geometries, users invariably tried convex deformations by pulling (Figure 5.19(e)). One user fashioned a new tool by combining different Shapescapes™ parts. This provided the convenience of holding the tool at the “handle” and deforming the pot using fine hand movements (Figure 5.19(f)).

5.5.3.6 Intent & Controllability (Q1)

We see evident improvements in the perception of intent recognition quality, initialization time, and robustness to accidental deformations (Figure 5.20). However, despite the decrease in completion time (task **T1**) there was no significant improvement in the user’s perception of inconsistency between pulling and pushing. In this case, reason for this perception was primarily related to the visual and tactile feedback rather than the algorithm for pulling itself. This was evident from the user’s comments such as: *“I think the reason pushing and pulling were different were because the pulling you had to 2 contacts with the pot and pushing you only needed one. I had a hard time understanding the depth of the pot making it hard to get two contacts on the pot”*. One user also suggested: *“I think it would be better if I get some feeling when I touch the pottery. It [would] make me feel more real and easier to control my hand. Then it would be better to have some sounds when I touch the pottery”*.

5.5.3.7 User Experience (Q2)

The experience was mostly positive, similar to the proximal-attraction approach (Figure 5.20(b)). In particular, users liked the use of tools and the smoothing operation the most. One user commented: *“The freeform design with tools was the most fun, as I could spend most of my time focusing on the design aspect as opposed to focusing on minimizing*

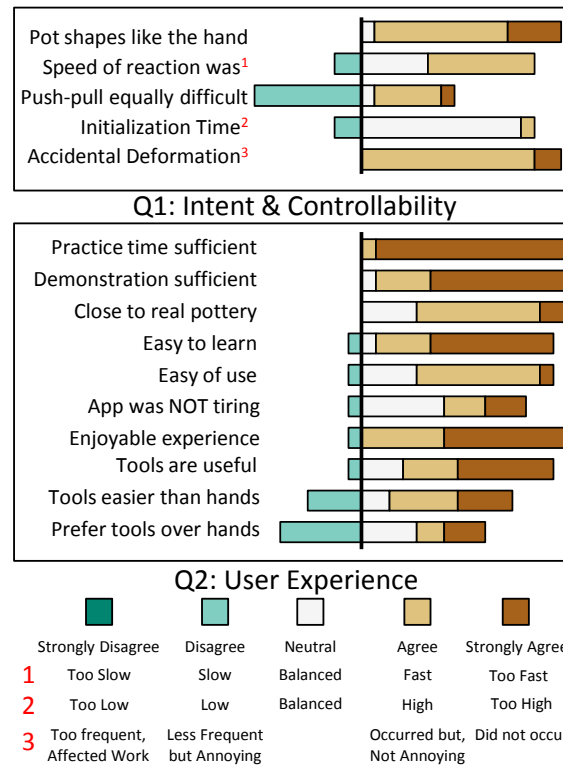


Figure 5.20. User response to are shown for grasp+motion. While the robustness to accidental deformations was perceived to be negligible (a), many users still perceived pulling to be difficult. Users agreed regarding the usefulness of tools but were not in general agreement about preferring them over hands.

errors.”. According to another user: *“The pottery changing according to my hand shape is so real. While smoothing, I could shape it as well, I like to do it this way a little bit.”*.

5.5.4 Limitations

Our method is currently implemented for pottery, which is essentially a one dimensional deformation. Further, we observed that the use of 2D displays is a factor due to which users tend to use side configurations. We believe that 3D visual feedback will encourage users to access the front and back faces. One user noted: *“This application with haptic feedback could train people for pottery before they actually perform it”*. This strongly indicates that the lack of tactile feedback is a critical component that is missing from our current system.

Severe occlusion resulting from camera position and hand orientation is an issue particularly for skeletal based gesture recognition. We partly addressed this challenge using our PCL-based approach which can make use of partial data even when the full hand skeleton is intractable. However, occlusion is an inherent problem in any camera-based method. Investigation of optimal camera position and use of multiple cameras at strategic locations is important. Secondly, we provided a method for temporally adaptive persistence.

In our current implementation, the definition of active regions is in terms of 2D profile topology rather than actual distances in real space. Thus, our implementation is dependent on PCL sampling relative to the mesh resolution of the pot. Independence from the sampling resolution may be addressed with an adaptive approach wherein new sections could be added according to manipulators or old ones removed based on geometric properties of the pot profile such as curvature.

In terms of our evaluation approach, our participants were primarily from science and engineering background. Even though some users had prior experience with creative tasks such as pottery and computer-aided design, studying our approach with art students would provide additional insights on user experience and utility of our approach.

5.6 Discussion

5.6.1 Spectrum of Expressiveness

One aspect that is both advantageous and disadvantageous in our approach is that different users can achieve the same target shape using different strategies for grasping, reaching, and deforming a shape. While this provides flexibility and intuitiveness to the user, it also results in increasing the time taken by the user to reach to a desired shape. The evaluation of proximal-attraction evidently indicated that there needs to be a balance between completely free-form interaction and symbolic approaches. This is what we attempted through the grasp+motion approach. The main advantage that our process provided was the discovery of relevant grasp information that is useful to design continuous operations such as shape deformation. Our grasp based approach can serve as a starting point for designing grasp-based interactions using cleaner data such as hand-skeleton [122].

5.6.2 Definition of Intent:

We began with a simple classification of intent through the analogy of structuring operators inspired by Delamé’s [140] work. However, users’ description of actions and expectation strongly indicates towards a richer and more complex mental model for deformation processes. To this effect, we had to include a third class of operation, namely “smoothing” which evidently improved the performance of the user. Though this aspect is not new in 3D modeling in general, this aspect of refinement is certainly worth investigating from a perceptual point of view.

5.6.3 Generalization

Although we demonstrated intent classification for rotationally symmetric shapes, the general approach of computing KDE to characterize grasp and motion can be extended to the deformation of arbitrary shapes. Here, we propose such an extension in two steps. First, we will consider asymmetric deformation in the context of pottery itself. For this, we begin

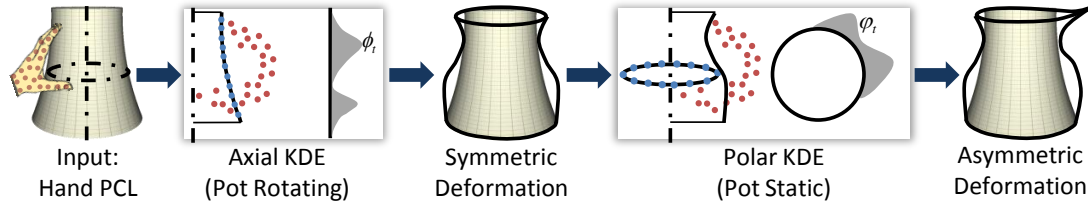


Figure 5.21. Asymmetric deformation can be applied to a pot in two steps. When the pot is rotating, we apply the *axial* KDE (top row) of the hand PCL for deforming the profile of the pot. Subsequently, users can stop rotating the pot and deform the pot locally using the *polar* KDE (bottom row).

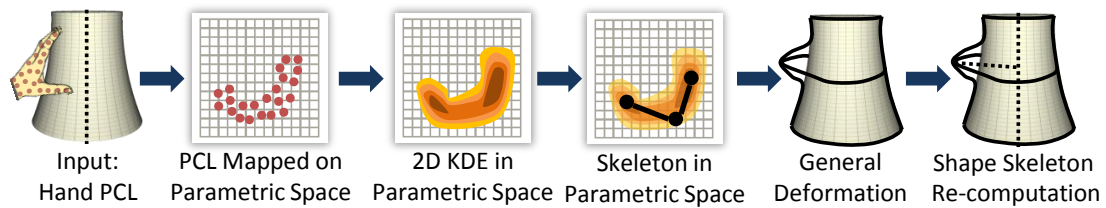


Figure 5.22. The computation of two-dimensional KDE in the parametric space of a cylindrical surface leads to the computation of grasp and motion for an arbitrary orientation of the hand PCL with respect to the surface. This allows for arbitrary deformation of the surface. Recomputing and segmenting the deformed surface using the method of Bærentzen et al. [146] provides a generalized deformation approach using our KDE based approach.

by noting that our approach summarizes the hand grasp and motion by computing a one-dimensional *axial* KDE of the hand PCL on the pot’s surface. In the same way, we can also compute the one-dimensional *polar* KDE of the PCL (Figure 5.21). Thus, by combining two one-dimensional KDE computations (axial and polar), we can enable users to create asymmetric features on the pots.

To see how these ideas can be used to conceptualize an arbitrary deformation of a shape, we make two observations. First, the pot is a cylindrical shape with a simple parametric representation and the axis of the cylinder is essentially its skeleton. Thus, given the hand’s PCL in an arbitrary orientation with respect to the cylinder’s surface, its two-dimensional KDE can be computed in the parametric space as a simple means to determine the grasp and motion of the hand (Figure 5.22). The consequent deformation of the cylinder would inevitably result in the need for re-computing the skeletal structure of the surface. This is where we invoke our second observation that an arbitrary 3D surface model can be converted to a set of connected cylinders using the recent work by Bærentzen et al. [146] that demonstrates the conversion of arbitrary triangle meshes into polar-annular meshes (PAM). The PAM representation effectively segments 3D shapes into generalized cylinders. Thus, the combination of two-dimensional KDE with the PAM representation can be used for deforming arbitrary meshes.

5.6.4 Precise & Selective Reachability

One user aptly commented: “*Sometimes it is hard to use the palm because it may deform the surface too much. The context of barely touching does not seem too well implemented. However, if you do this very carefully you can do the barely touching but may make your arm tired a little.*”. This is the problem of *precise and selective reachability* wherein one is required to reach and manipulate a local region of an object without affecting neighboring regions. There is extensive volume of work that investigates *distal* selection, manipulation, and navigation [98, 99, 147] of objects. We believe that precision and selectivity are problems worth investigating for close-range, i.e. *proximal* 3D manipulations in mid-air.

5.7 Conclusions

Extending the grasp+motion approach for arbitrary meshes would involve several computational challenges since distance computations and KDE computation would be on 2-manifolds. Secondly, it will also be important to study how user perception and performance is affected by adding 3D visual feedback and also tactile feedback. Finally, with our approach, it is not possible to perform deformation using existing hand skeleton tracking approaches. We intend to investigate this in comparison to the PCL based hand representation. One key advantage of using tracked skeletons is that there is a direct correspondence between the fingers and palm which can give useful movement information for better intent detection. This would help segmenting users intentional and unintentional movements [148]. One of the main observations in our preliminary exploration was that users from different backgrounds and age group had different ways of using the pottery tool. In future works, it would be interesting to investigate how experience, performance, and creative outcomes vary with respect different user groups such as artists, engineering designers, and young participants.

In this chapter, we presented a spatial interaction technique that uses hand grasp and motion for intent expression in virtual pottery. This approach enables a shift from existing gesture-based procedural events towards non-procedural and temporally continuous processes in the context of shape deformation. In other words, our work enables users to achieve what they intend in the way they see fit. To the best of our knowledge, no existing hand-based spatial modeling scheme offers such diverse contexts of user input, for instance the use of everyday real objects as tools for virtual shaping, with controllable outcomes.

6. SPATIAL DESIGN IDEATION USING A SMARTPHONE AS A HAND-HELD REFERENCE PLANE

The demonstration of tools in the geometric approach added anew dimension to spatial interactions for ideation: tangibility. In this chapter, we will investigate the role of tangibility in further detail through mobile spatial user interfaces (M-SUI's). M-SUI's have received significant research focus towards 3D manipulation, navigation and design on large-screen public displays, immersive environments, and mixed-reality setups. In this chapter, we explore *spatial design ideation* through the association of physical human movement to the design outcome. Our broader goal is to explore the role of embodied interactions in enabling spatial ideation during early phase design by employing M-SUI's. As a concrete step towards this goal, we present *MobiSweep*, a prototype application for creation of 3D compositions comprised of swept surfaces through constrained spatial interactions with a smartphone.

As the name suggests, *MobiSweep* makes use of sweep surfaces as the underlying shape representation. Sweep surface representations are fundamental in computer-aided geometric design (CAGD) and provide a simple and powerful means for defining 3D shapes of arbitrary complexity. Further, sweep surfaces inherently lend themselves to an intuitive association to the action of *sweeping* a 2D shape in 3D space. In this context, we inspire our work with two observations from CAGD and M-SUI. First, traditional construction of sweeps relies heavily on the procedural specification of datum planes as spatial references and 2D curves profiles and trajectories. Within conventional CAGD systems, this is a tedious process, involves a number of operations, and requires parametric configuration of each plane [35]. Secondly, even though sketch-based interactions are common to both geometric modeling and 2D mobile applications, their utilization in existing M-SUI's [149] has been severely limited towards mid-air shape creation. We posit that by combining the spatial freedom in mid-air interactions with multi-touch capabilities of smartphones, novel

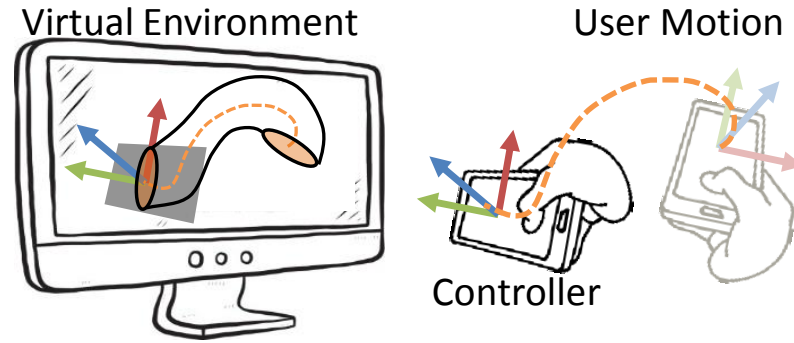


Figure 6.1. The idea behind *MobiSweep* is to utilize the spatial relation between the action of sweeping with the creation of the swept surface.

and creative work-flows can be constructed to enable expressive design exploration. To this end, we introduce an interaction metaphor that uses a smartphone as a hand-held reference plane. Our interaction is an extension of the free plane casting method proposed by Katzakis et al. [150]. By adapting our metaphor for sweep surface creation, we demonstrate how users can directly create, modify, and compose 3D design concepts through tilt and touch interactions on the mobile phone.

6.1 Related Works

6.1.1 Mobile Devices for 3D Manipulation

Mobile devices offer a unique combination of computational power, wireless data communication, 3D sensing capabilities, ergonomic manipulability, and multi-touch input mechanisms. Although mobile devices have been previously explored as spatial controllers for several virtual applications, the primary disadvantage in case of a mobile device is the lack of an explicit position tracking method. While there is literature that uses smartphone accelerometers to detect *motion-gestures* [151, 152], a significant amount of work has used touch-tilt combinations for 3D object selection and manipulation [153, 154]. The multi-touch capability of mobile devices provides additional affordances for both direct and indirect manipulations of the virtual objects [150, 155, 156]. Combinations of touch and tilt

interactions have also been utilized for precise object selection [157–159], scene navigation [160], and immersive interactions [161–163]. A study by Tsandilas et al. [164] identified the best tilt-touch combinations in terms of user performance, motor coordination, and user preferences. Similar interactions have also been utilized within 3D exploratory applications such as volumetric data annotation [165] and scientific visualization [166]. In contrast to these works, our focus is on the exploration of novel interactions for 3D shape composition, i.e. activities that involve shape creation tasks (e.g. cross section sweeping) and modification tasks (bending, twisting) in addition to manipulation.

6.1.2 Design with Mobile Devices

Interactive design and shape modeling has also been studied with mobile interfaces. Xin et al. [167] demonstrated the use of a tablet as an augmented reality (AR) canvas for 3D sketching, akin to creating wire-sculptures. Scheible and Ojala [168] proposed *MobiSpray*, a system for intuitive graffiti on large physical environments using mobile phones as *spray-cans*. Lakatos et al. [169] and Leigh et al. [170] proposed the use of mobile devices as *spatially-aware* hand-held controllers in conjunction with hand-worn gloves for 3D shape modeling and animation. Their work was more focused on demonstrating general interactions for modeling scenarios rather than exploring a concrete design work-flow for shape composition. Mine et al. [171] described and discussed an immersive M-SUI system and demonstrated an immersive adaptation of the SketchUp application. Though their work provides an excellent set of guidelines for mobile-based modeling, their focus was towards an immersive system augmented with additional hardware for positional tracking. Our work differs from these works in two ways: (a) our focus is on supporting creative 3D composition through interaction and work-flow design (in contrast to [167, 168]) and (b) our system does not use any additional hardware or vision based method for explicit position tracking (such as in [169–171]).

Xin et al. [167] demonstrated the use of a tablet as an augmented reality (AR) canvas for 3D sketching, akin to creating wire-sculptures. Similarly, Lue and Schulze [149]

demonstrated the *3D Whiteboard* system using smartphone AR technique with fiducial markers. Lakatos et al. [169] demonstrated the use of tablets as *spatially-aware* hand-held controllers in conjunction with hand-worn gloves for 3D shape modeling and animation. However, their work was more focused on demonstrating general interactions for modeling scenarios rather than exploring a concrete design work-flow for shape composition. Mine et al. [171] described and discussed an immersive adaptation of the SketchUp application using a tracked smartphone in a CAVE setting. Our work differs from these works in two ways: (a) our intention is to support quick creative compositions with actual 3D surfaces in contrast to [21, 149, 167] and (b) our system does not use any additional hardware or vision based method for explicit position tracking (such as in [169, 171]).

6.2 MobiSweep

6.2.1 System Setup

The *MobiSweep* interface comprises of a hand-held controller (smartphone), and the virtual environment (i.e. a modeling application running on a personal computer) (Figure 6.2). The virtual environment consists of a *reference plane* with a local frame of reference mapped to the phone's coordinate system.

6.2.2 Design Rationale

The design goal behind *MobiSweep* is to strike a balance between modeling constraints, interaction techniques, and system workflow to enable direct spatial ideation. There are mainly two fundamental aspects that we considered while designing *MobiSweep*: (a) 3D manipulation and (b) sweep surface generation. For 3D manipulation, the critical aspect under consideration is to minimize fatigue for precise manipulations and minimize the interaction time for coarse manipulations. Instead of imposing full mid-air movements, we employ touch gestures to allow controlled and precise 3D manipulation of virtual objects. In order to minimize learning time, we take advantage of the fact that most users are al-

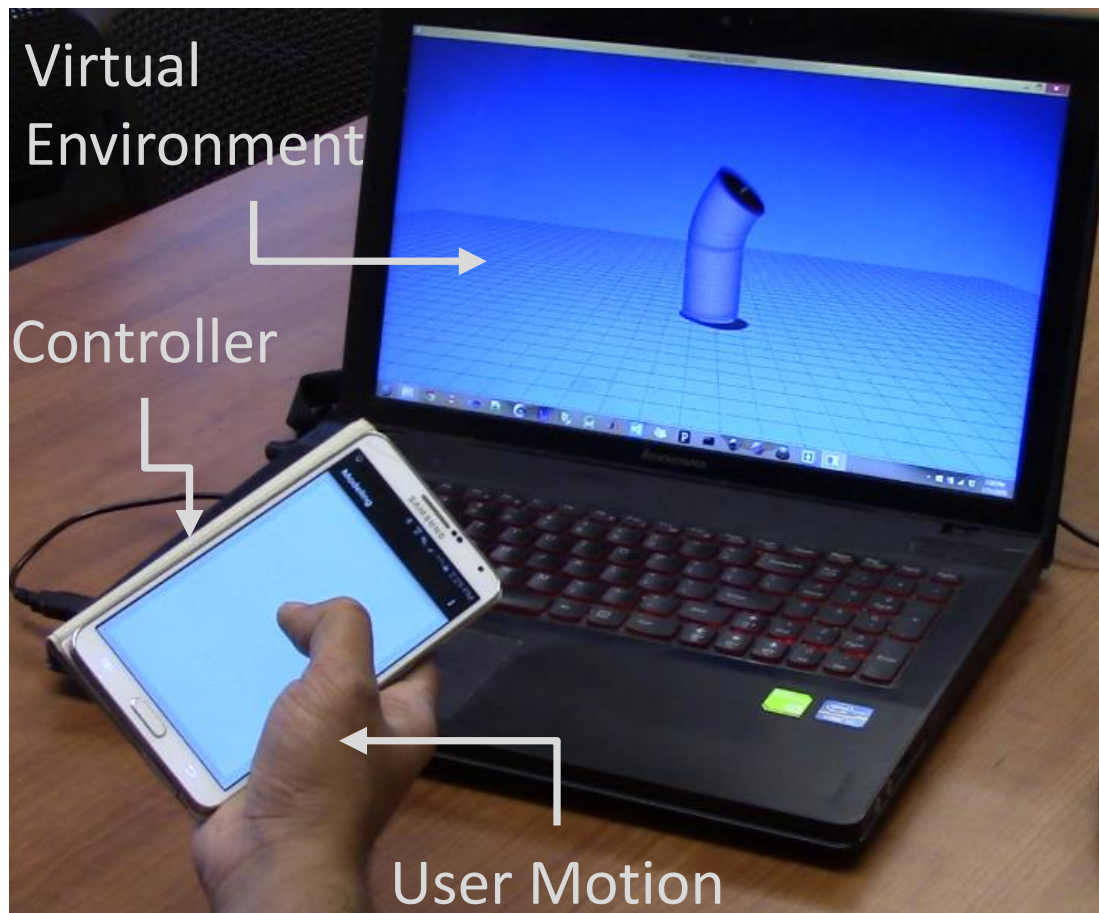


Figure 6.2. Setup for *MobiSweep* comprises of a visual display of the virtual environment and a smartphone that acts as a reference plane in the virtual environment.

ready familiar with multi-touch gestures for manipulating objects. Thus, we define a single *context-aware* interaction metaphor that: (a) uses known multi-touch gestures and (b) is shared between several modeling tasks.

Drawing from the key insight of Jacob et al. [172], we find that the separation of degrees-of-freedom (DoF) can be effective if the interactions for the task (sweeping a section) are synergistic with the input mode provided by the device (the smartphone). Based on this, we inspire our approach from the free plane casting method proposed by Katzakis et al. [150] by combining direct orientation control with indirect gesture based position control. We introduce an interaction metaphor - *phone as a reference plane* - that emulates the action of sweeping a sketched cross-section that is held in the user's hand (Figure 5.5). In doing so, we do away with the procedural specification of planes as spatial references for drawing 2D curves to define profiles and trajectories, as is predominantly done in conventional CAD systems. The key advantage of our metaphor is that in addition to creation, it naturally lends to spatial actions such as on-the-fly bending, gesture-based cross-sectional scaling, and in-situ modification of the cross-sectional shape by sketching.

6.2.3 Gesture Definition

In order to define the interaction work-flow for *MobiSweep*, we begin with the definition of our interaction metaphor - *phone as a reference plane*. Given a hand-held phone, we can define a reference plane in the virtual 3D space with a local coordinate frame. Subsequently, the objective is to allow the user to specify the location and orientation of the reference plane. We define the following gestures to achieve this objective:

Rotate: Here, the orientation (and hence the local coordinate frame) of the phone is directly mapped to that of the reference plane. Thus, simply rotating the phone results in the rotation of the reference plane (Figure 6.3: column 1, row 2).

Pan: Using the two finger sliding gesture, users can translate the reference plane on the x-y plane of the local coordinate system (Figure 6.3: column 2, row 2). This is similar to *in-plane* panning in the *free plane casting* interaction [150].


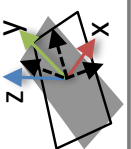

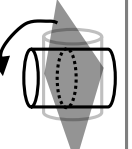



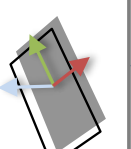
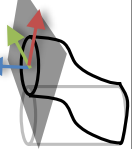
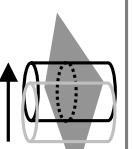
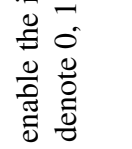



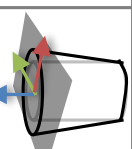
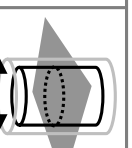
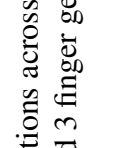

<u>Gestures</u>		0F Rotation (Rotate)	2F Slide (Pan)	2F Pinch/Spread (Scale)	1F Press/Hold (Offset)	3F Pinch/Spread (Offset)	1F Move (Sketch)
<u>S1: Configure</u> Reference plane empty							
<u>S2: Author</u> Reference plane attached to the top- most sweep section							
<u>S3: Manipulate</u> Reference plane attached to the sweep surface							

Figure 6.3. There are six gestures (row 1) that enable the interactions across three modeling states: Configure (row 2), Author (row 3), and Manipulate (row 4). 0F, 1F, 2F, and 3F denote 0, 1, 2, and 3 finger gestures respectively.

Scale: Users can also perform in-plane scaling by using a two finger pinch gesture. However, scaling is a context dependent operation that is allowed only when the reference plane either contains a sweep section (Figure 6.3: column 3, row 3) or is attached to a 3D object during a manipulation task (Figure 6.3: column 3, row 4).

Offset: The use of one-finger press/hold (Figure 6.3: column 4, row 2) gesture allows for automatic translation of the reference plane along its normal with a predefined constant speed. Users can also offset the reference plane by applying a three-finger pinch/spread gesture (Figure 6.3: column 4, row 2). In this case, the magnitude of offset defined according to the area of pinching or spreading¹. The one finger gesture provides a quick but imprecise method for offsetting. On the other hand, the three finger gesture requires more effort but allows for a more precise and bi-directional control of the reference plane.

Sketch: Users can sketch a curve on the reference plane using the traditional one finger movement. Similar to scaling, we allow sketching only when the user wants to modify the cross-section of a sweep surface.

6.2.4 Modeling States

The gestures defined for manipulating the reference plane form the basis of *MobiSweep*'s work-flow. For any given state in the work-flow, the input gestures (Figure 6.3: row 1) remain the same but the reference plane takes a different meaning according to the context of the states (Figure 6.3: rows 2-4) as defined below:

Configure (S1): In this state, the reference plane is detached from all existing shapes (if any). This empty plane can be manipulated to a desired location and orientation in 3D space using the gestures described above (Figure 6.3: row 1). Such as manipulation may occur either during the creation of the first shape of a composition or during in-situ composition where a user is directly creating one shape on an existing shape. Alternately, users can also move the reference plane in order to select an existing shape in the virtual environment.

¹See supplementary material for details.

Author (S2): In this state, the reference plane is attached to the top-most section of a sweep surface. Users can (a) create a swept surface by offsetting (Figure 6.3: columns 4-5, row 3), (b) bend and twist a sweep surface by rotating the phone (Figure 6.3: column 1, row 3), (c) pan and scale a section using two-finger gestures (Figure 6.3: columns 2-3, row 3), (c) modify a section's shape by sketching on the phone (Figure 6.3: column 6, row 3).

Manipulate (S3): This state involves rigid transformation of a swept surface for composing through assembly. Here, the reference plane serves as a container for the swept surface through which users can translate, rotate, or scale the surface. Additionally, users can also copy an existing shape and reuse a transformed version of the copy within the composition.

6.2.5 Modeling Work-Flow

In the *MobiSweep* work-flow, the configure state (**S1**) is the base state from where users can transition to either the authoring state (**S2**) or the manipulation state (**S3**). The transition between these states are enabled using a combination of menu and gestures. The controller interface for *MobiSweep* is a single-screen Android application that allows for two distinct modes of interactions: (a) *multi-touch input* for reference plane manipulation, sketching, and state transition and (b) *menu navigation* for state transitions and general software tasks. Below, we describe the three canonical examples for creation, modification, and manipulation of swept shapes.

6.2.5.1 Shape Creation

The creation of a swept surface involves the transition from the **configure (S1)** to the **author (S2)** state (Figure 6.4(a)). For this, the user selects the “**Add Shape**” button on the menu, thus expressing the intent to begin the creation of a sweep surface. Once the user has expressed the intention to add a shape, the visual representation of the reference plane changes to a default circular section. The user can now sweep the section by using the *one finger press-hold* (or *three-finger pinch-spread*) gestures. This corresponds to the *offsetting*

operation occurring along the reference plane normal. By continuously re-orienting the phone during the sweeping process, users can create curved sweeps. Users can also modify the swept surface as described in the following section. Once the user has created a desired shape, the swept surface can be detached from the reference plane using the *double-tap* gesture effectively bringing the user back to the **configure** state.

6.2.5.2 Shape Modification

Once the user has created a swept surface, the authoring state allows users to modify it *as long as the user has not detached the reference plane from the surface*. The reference plane is attached to the top-most section of the sweep surface (Figure 6.3: column 3). Hence, all interactions performed by the user affect the top most section only and correspondingly changes the remaining sections of the sweep surface (Figure 6.4(b)). For instance, simply re-orienting the smartphone results in the rotation of the top-most section effectively allowing the user to bend and twist the swept surface. Similarly, using the two-finger gestures allows for panning and scaling the top-most section of the swept surface.

The modification of the shape of the top-most section involves three steps. The user first selects the “**Sketch Section**” button on the menu to activate the sketching mode. Once in sketching mode, the user simply sketches a desired curve on the smartphone. In our current implementation the user is required to sketch the section in a single stroke. Every time the user finishes drawing a sketch, the sweep surface is immediately modified according to the new sketched section. Thus, the user can simply keep over-drawing the sketch in order to explore different varieties of shapes. Once satisfied with the modified section, the user finalizes the modification using the “*Confirm Section*” button on the menu. Similar to shape creation, the swept surface can be detached from the reference plane by using a *double tap* gesture.

6.2.5.3 Shape Manipulation

Manipulation of an existing shape involves two steps (Figure 6.4(c)): *hover* (**S1**) and *selection* (**S3**). Translating the center of the reference plane inside a swept surface is defined as *hovering* on the surface. The user can select an object by first hovering on the object followed by a *double tap* gesture on the phone. Similarly, using the *double tap* on a selected object reverts the state to *hover* again. Thus, *double tap* acts as a toggle between the attachment and detachment of a shape from the reference plane. The use of *double-tap* enables users to perform selection without looking at the controller. *Selection* signifies the attachment of a 3D object with the reference plane, i.e. all rigid transformations applied on the reference plane are transferred to the selected object. In addition to manipulation, the *hover* state can also be used to perform operations such as copying, deleting, and coloring a shape by using the menu.

6.3 Implementation

6.3.1 Hardware & Software

Our hardware comprises of a ThinkPad T530 laptop computer with Dual Core CPU 2.5GHz and 8GB RAM, running 64 bit Windows 7 Professional with a NVIDIA NVS 5400M graphics card, and the Samsung Galaxy Note 3 as the hand-held controller. We implemented a one-way Bluetooth serial port communication to stream input data from the controller (phone) to the *MobiSweep* application (running on the PC). The input data packet consisted of device orientation, touch coordinates, menu events and multi-touch gestures. Our controller interface was implemented using the Android SDK and the application was developed in C++ with OpenGL Shading Language for rendering ².

²See supplementary material for details on menu and calibration implementations

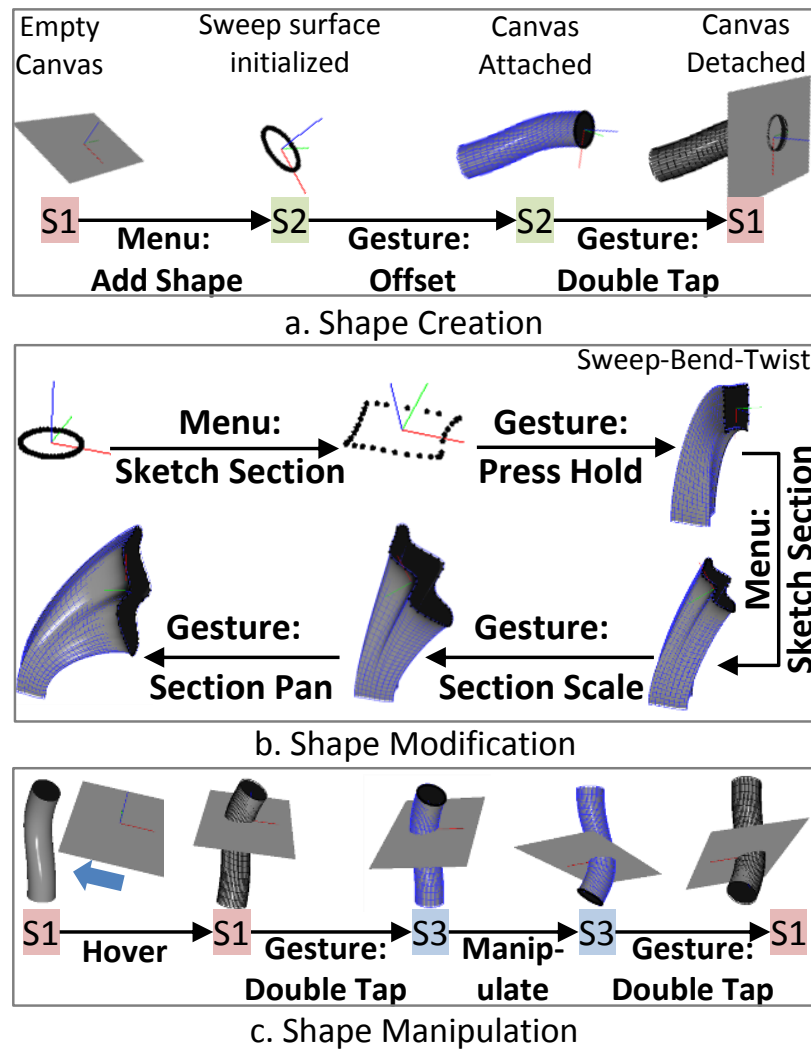


Figure 6.4. For shape creation (a), the user selects the **Add Shape** menu item and creates a sweep surface using one or three finger offsetting gesture. For manipulating a shape (b), the user first hovers on a desired sweep surface and selects the shape using the *double-tap* gesture. In the shape modification example (c) the user modifies the initial section by sketching, creates a sweep surface, and modifies the final section by sketching, scaling, and panning.

6.3.2 Algorithms

6.3.2.1 Sweep Surface Generation

The sweep surface is represented as a stack of cross-sections. Once the users starts the offsetting interaction, the sweep surface is incrementally generated in three steps: (a) adding a new section and (b) translating the top-section along the reference plane normal at until a stipulated time has elapsed, and (c) repeating addition and translation as long as the user is offsetting the reference plane. This process of incremental generation provides the visual continuity of sweeping to the users and the translation time defines the distance between consecutive sections.

In this work, we implemented a variant of the control-section based sweeping technique [173] wherein every sweep surface consists of two control sections at the two ends of the sweep surface. Each control section comprises of equal number of points and the information about its local coordinate frame (i.e. the frame of the reference plane). Hence, there is a one-to-one point correspondence between the control sections. For a given pair of control sections, we interpolate each meridian of the sweep surface by using the cubic hermite basis functions (Figure 6.5(a)). The interpolation requires four boundary conditions, namely, the position and tangents at the end points. These are conveniently provided by the vertices and the normal of the section's local coordinate frame respectively. Our approach removes the need for explicit computation of the individual section transformations and avoids frame rotation minimization and section blending. This simplifies the operations (bending, twisting, panning, scaling and section modification) in the authoring state.

6.3.2.2 Section Modification

Currently, we allow single stroke sketching in our implementation and the number of points in each section of the sweep surface is constant and pre-defined. For a sketch input, we first determine if the sketch is an open or a closed curve based on a simple distance threshold between the two end-points of the sketch input. For a closed curve, we imple-

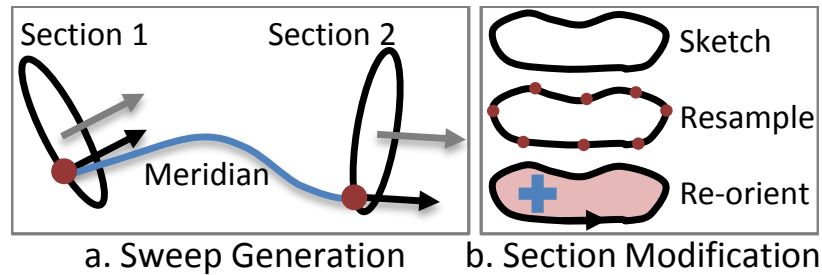


Figure 6.5. Algorithms for (a) sweep generation and (b) section sketching.

mented a three stage processing of the sketch input (Figure 6.5(b)). First, we perform an equidistant curve re-sampling [174] to match the number of points on the sketch to the initial control section of the sweep surface. Subsequently, we determine if the orientation of the curve is the same as that of the initial control section. This involves the comparison between the signs of the areas enclosed by the sketched curve and the initial section. If the initial and sketched sections have opposite orientations, we correct the sketch orientation by reversing the order of vertices in the re-sampled sketch input. Finally, we minimize the twist between the sketch input and the initial section [175].

6.4 User Evaluation

The goals for our study, were to (a) understand how users perceive the interaction workflow embodied by *MobiSweep*, and (b) explore and characterize user ideation and creation enabled our system.

6.4.1 Participants

We recruited a total of 14 (11 male, 3 female) participants in the range of 19 – 40 years. Our user population consisted of 9 mechanical engineering students (with 1 user with expertise in CAD and design practices) and 5 students from other fields including engineering, liberal arts, and sciences. All participants were dominantly right handed and owned an Android-based or an Apple smartphone.

6.4.2 Procedure

The length of the study varied between 60 to 75 minutes. In the beginning of the study, each participant was given a verbal description of the setup, the purpose of the study and functionality of the *MobiSweep* application. The participants were then asked to perform the following tasks:

- P** *Guided Practice:* Each participant was taken through a guided composition process wherein the participant used *MobiSweep* to create an abstract tree concept. The goal was to introduce the participants with features and constraints of the system in an organized manner. During this phase, the participants were encouraged to think-aloud, ask questions and were provided guidance when required.
- T** *Idea Exploration:* This task focused on understanding the usefulness of *MobiSweep* as a tool for design brainstorming. Participants was given 1 among 3 pre-determined product contexts (tea-kettles, jars, lamps) and were asked to create variants of the given context in a fixed time duration of 15 minutes. Although the duration of time was fixed, we allowed the users to complete their last composition that was started before the end of the specified duration. Once the participant was satisfied with a composition, they would clear the virtual environment and start with a new composition.
- Q** *Questionnaire:* After completing the tasks, participants were asked to complete an online questionnaire regarding the workflow, intuitiveness of the gestures, and task preference. the participants were asked to complete an online questionnaire for evaluating: (a) effectiveness of interactions and gestures and (b) the usefulness of *MobiSweep* towards ideation and creation activities in early design. For assessing the usefulness of *MobiSweep* for design ideation, we used the creativity support index [176]. We also asked them questions regarding the potential future use of the application.

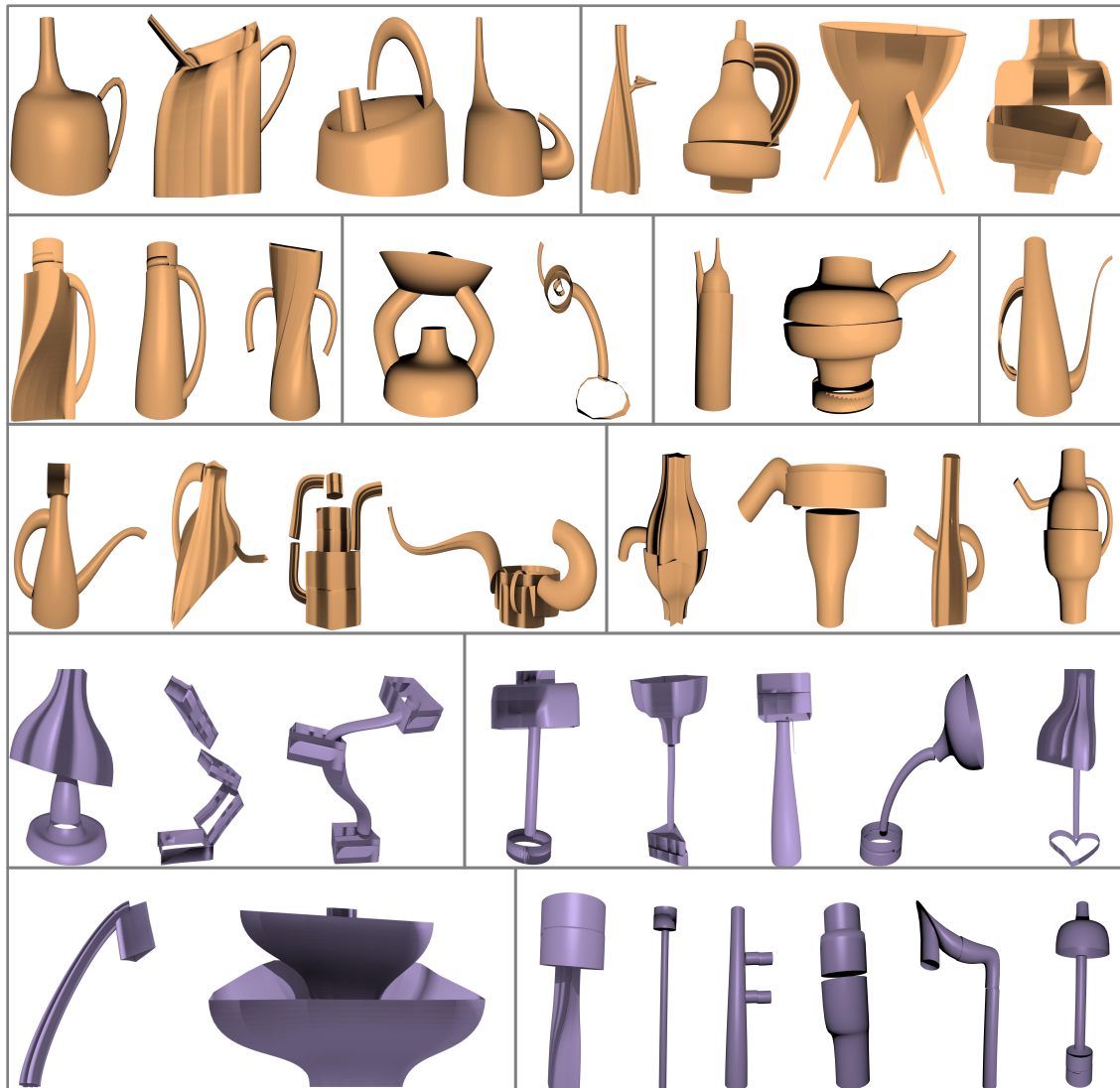


Figure 6.6. Design concepts generated by the users are shown (kettles and jars are shown in the top three rows and lamps in the bottom two rows). Each box represents concepts generated by one user.

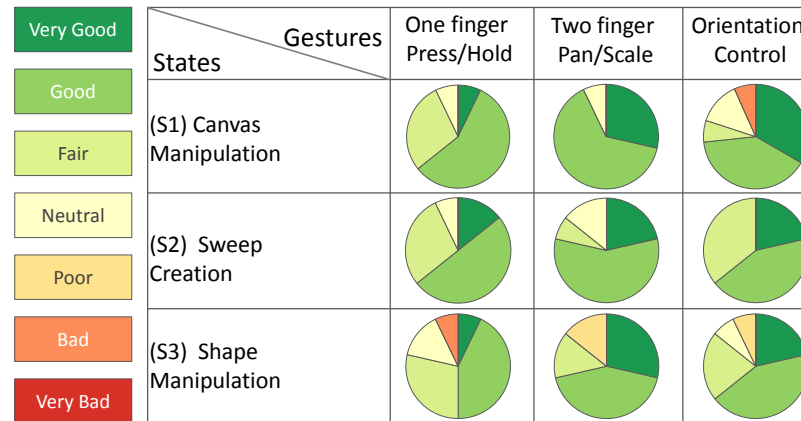


Figure 6.7. User feedback for interaction ratings in the context of the work-flow states.

6.4.3 Results

We found that almost all users were able to rapidly generate ideas in the product contexts provided to them (Figure 6.6). With an average practice time of 19 minutes (min: 11, max: 30), users generated between 3 to 4 (min: 1, max: 6) concepts within an average ideation time of 15.7 minutes (min: 6, max: 21). Typically, each concept comprised of at least 2 and at most 4 parts (sweep surfaces). As expected, the number of concepts reduced for compositions with more geometric detail at the part level. In the context of these results, we will discuss our observations and users' feedback regarding interactions, creative support, and perceived utility of *MobiSweep*.

6.4.3.1 Interactions

A significant majority of the ratings were positive across interaction types and work-flow states (Figure 6.7). The two main problems users faced were (a) manipulation of a shape/part (**S3**) using the offsetting operation with one finger press and (b) controlling the reference plane orientation (**S1**). Interestingly, many users actually moved their hands along the trajectory of a sweep surface during shape authoring despite having the knowledge regarding the lack of explicit position tracking. One user commented: "*I felt 3D*

objects [were] alive while I was sweeping and manipulating them.” This corroborates the proprioceptive nature of these activities, making the case for spatio-kinesthetic awareness for mapping spatial motion of smartphone-based controllers for 3D shape creation.

For the reference plane offset operation, we asked users to compare the one finger press, with the three finger scale on ease of use, physical comfort, intuitiveness, and controllability. All but three users indicated that the three finger press was better in terms of controllability. However, we found no significant preference towards ease-of-use, physical comfort, or intuitiveness. Users commented that the three finger pinch was more controlled, however it took some practice to understand how to apply the gesture correctly. They also perceived the one finger press as simple and natural, but only controllable in one direction (upwards). This is a useful insight that could be used to improve the offset operation by introducing auto-rotation features based on the ergonomics of wrist movements in one-handed manipulations.

Users found the sketching mode to be an intuitive and direct method for specifying cross-sections. A user commented: *“Section sketching granted me quite a lot of flexibility in producing the desired shapes. I also found that section sketching allows me to select even the end sketch giving even more flexibility”* Additionally, the default circular section was also considered useful by users. One user pointed out that *“Having the circle as a default was very helpful, as more often than not, I wanted a circular cross-section. When I didn’t need a circle, I felt it was simpler to just sketch the shape. Having other options (polygon selection, for instance), may have been annoying.”*.

6.4.3.2 Creative Support

A large majority of users responded favorably in terms of the exploration capability, expressiveness, engagement, and enjoyment provided by *MobiSweep* (Figure 6.8). In particular, the user feedback strongly validated our primary goal - quick design ideation in 3D space. As a user pointed out: *“Quickly sampling ideas in 3D shortens the discussion on any subject that requires a solution and closes the gap between individuals who can’t ex-*

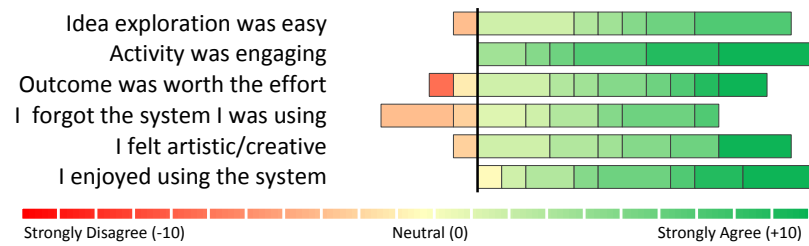


Figure 6.8. User feedback for creativity support in *MobiSweep*.

plain what they see in their heads and individuals who can't visualize what is explained to them. Normally such discussions would end with - I'll have to show you later." Regarding quick ideation, a user stated: *"This tool can be very useful for people who are afraid to make mistakes and can also help people to formulate spatial perceptions."*

6.4.3.3 Utility

Users confirmed *MobiSweep's* utility in real design problems in individual and team settings. In particular, users with mechanical engineering and design experience found such a tool particularly useful in the context of their design projects. One user commented: *"I can see myself using this tool for a quick mock-up of ideas, something to do right after the sketching stage. Assuming that a future version of the system will allow me to navigate my creation in 3D (instead of offering a single-port view as it does currently), I would be able to use this to mock up an idea in 3D to discuss issues like space, access, scale etc. with my team."* Most participants with prior design experience perceived our system as a useful mode of coarse design followed by fine refinement using a professional CAD tool. One user with expertise in CAD and professional engineering design experience stated: *"I can see a multi-user scenario of this system, where you can perform 3D modeling versions of the C-sketch or Gallery methods of ideation. It would make for a fun activity, with each user using their own device to move between ideas and interact with shapes."*

6.4.3.4 Limitations

One user who was focused on precise manipulations, mentioned: “[it is] hard to keep a steady orientation when manipulating”. We believe this can be rectified using simple measures such as smoothing the smartphone orientation data and snapping the reference plane orientation along primary axes. Another user mentioned that: “depth is so hard to perceive on screen.”. Improving visual feedback and allowing view manipulation would allow for better assembly of shapes. The use of cubic-hermites in our implementation constrains the control of the spine of the sweep surface. Our early experiments showed that this was a necessary constraint to achieve controllability while maintaining reasonable design flexibility. Extending our interactions for piecewise will help improve the expressiveness of ideation at the part level. Our indirect multi-touch control for 3D translations provided low-fatigue interaction and was effective in terms of controllability. Although users commented that 3D position tracking will improve their efficiency in translation, their primary reason was the repetitive nature of the two-finger panning while moving long distances rather than unintuitive interaction design. One user commented on the offsetting gesture: “I would still prefer on occasion to use the single tap for coarse movement, and the three-finger touch for fine movement.” This strongly indicates that the allowing users to customize interaction parameters such as the offsetting speed and panning sensitivity will significantly improve user performance in 3D translation allowing for both coarse and fine translations.

6.5 Discussion

The primary motivation behind *MobiSweep* was to adapt existing parametric geometry representations in a conceptual design work-flow using mobile spatial interactions. In this respect, the creative outcomes, observations and feedback from our user evaluations make a strong case in favor of the underlying canvas metaphor presented in our work-flow. Fundamentally, there are two aspects of the metaphor that played a central role: the offsetting operation and the sketching modality. Even though it is theoretically possible to span the whole 3D space using in-plane panning in conjunction with the orientation (free-plane

casting [150]), the offsetting interaction turned out to be a necessary addition, particularly in the sweep surface modeling context. Second, despite the constrained screen space on the smartphone, enabling users to provide sketch inputs for 2D curve creation proved to be equally essential for allowing them to create shapes with reasonably complex geometric features.

Extending the arguments above, the aspect most critical in our work was the combination of two fundamental interactions pertinent to geometric design: sketching and spatial configuration. Sketch-based 3D modeling forms an exclusive area for early phase design due to its accessibility and natural interface [177]. However, the two-dimensionality of the interactions involved in sketching interfaces necessitates additional interactions to achieve a complete 3D modeling work-flow. We believe that the combination of canvas interactions with sketch-based modeling is a simple but powerful idea that could lead to several new design work-flows reveal new challenges in mobile spatial interaction. To this end, *MobiSweep* demonstrated how both 2D and 3D interactions can be coupled *in their intended forms* to refine the design work-flow in a concrete modeling system.

6.6 Conclusions

In this chapter, we explored an embodied approach for spatial design ideation through a sweep-based shape composition work-flow using a smartphone. At its core, *MobiSweep* allowed for two important geometric modeling interactions: rigid transformations and curve creation (both 2D and 3D). Our goal in the immediate future is to perform a quantitative evaluation of the reference plane metaphor for these three operations. *MobiSweep* revealed an untapped design space that emerged from the combination of M-SUI and CAD towards novel work-flows for creative shape conceptualization in early phase design.

7. CONCLUSION

We will conclude this thesis by first summarizing the technical contributions, reiterating the implications of this work on spatial design ideation, and finally discussing the avenues for future research on spatial design ideation.

7.1 Summary

In this work, we explored three approaches for transforming physical human action into the intent for designing of 3D shapes. We began with the SGC-I framework wherein the intent was modeled as a combination of arm motion and a finite set of symbolic hand gestures. The aim was to contextualize design intent with respect to the geometric content that the user intended to create, modify or manipulate. The symbolic transformations embodied by SGC-I are representative of a *state-based approach* through a finite set of pre-defined associations between geometric objects and user input.

Subsequently, we introduced the geometric approach for enabling intent expression in shape deformation for the design of constrained 3D artifacts – pots. In doing so, we introduced two new interaction techniques, proximal attraction and grasp+motion, that captured the spatio-temporal continuity of use input by making use of raw user-inputs (PCL). There were three key outcomes of this approach. First, we practically demonstrated how users’ understanding of a real-world shaping process can be embedded within the virtual interactions for shape modeling. Second, we presented the formal evaluation of the two core components of interactions for ideation: intent & controllability. Finally, the geometric approach helped us introduce the role of tangibility in the ideation process through the use of physical objects as hand-held tools for virtual shape deformation.

In the final stage of this work, we investigated the tangible approach towards spatial design ideation enabled by the use of smartphones as hand-held controllers. In particular,

we investigated a new interaction metaphor for transforming the smartphone into a tangible mediator of design intent. Here, the idea was to capture the physical action of creating swept surfaces towards their creation. We demonstrated the effectiveness of this approach towards spatial design ideation through the *MobiSweep* application. From the general perspective of interaction techniques, the tangible approach was as a middle-ground between the symbolic and geometric approaches through the combination of prescriptive gestures and spatial manipulation.

7.2 Design Implications

While the embodied interaction framework served as the source of broader guiding principles, there are some specific aspects of spatial interactions that we have discovered through the symbolic, geometric, and tangible approaches presented in this work. Below, we describe the broader insights in terms of the implications for designing spatial interactions for the ideation of shapes.

- Context-aware Interaction Approaches:** There are several aspects to a software tool that must be considered from the point of view of usability, utility, and context. Especially when the tool is targeted towards facilitating creative idea generation, no single approach (symbolic, geometric, or tangible) can be sufficient to cater to all the requirements that the tool is intended to satisfy. The most important implication in the design of interactive consideration while designing a virtual interface for ideation is to infuse the awareness of the *currently active context* during its usage. We have learned that the geometric approach is suitable for gradual and reflective processes such as shape deformation and creation. On the other hand, symbolic interactions are suitable for contexts such as selecting objects and identifying modeling operations.
- Value of Geometric Constraints:** An important problem in computer-supported ideation is to determine a minimal set of modeling features that channel the designer's thinking process towards the variety of ideas while retaining expressiveness

of their creations. Our focus being the study of spatial interactions, one of the most important outcomes of this thesis is the demonstration of the fact that a careful selection of geometric constraints is critical to the investigation of interaction techniques towards idea generation. For instance, the choice of the pottery metaphor helped us focus on the geometric characterization of intent and controllability for shape deformation. Similarly, the assembly based context in *MobiSweep* allowed for the exploration of the ideation process. The key insight gained in this respect is that properly defined geometric constraints can be instrumental for studying the role of interactions as well as engaging users through carefully defined modeling contexts.

- Levels of Spatiality:** For a spatial task there can be several input mechanisms with different degree-of-freedom (DoF). To this end, we explored two input mechanisms: (a) purely mid-air (6 DoF) input in symbolic and geometric approaches and (b) combination of multi-touch and 3 DoF orientation control in the tangible approach. The main differences between these mechanisms is in their utilization of manual effort (i.e. the extent to which they use and divide bodily effort for a spatial task). On one hand, even though interactions shown in the symbolic and geometric approaches were generally liked by users, these interactions also lacked in control and often lead to arm fatigue. On the other hand, the tangible approach demonstrated a balance between spatiality and control by constraining the interaction to a 3 DoF control. The additional use of touch interactions increased the expressiveness in the ideation process. This clearly points to the importance of manual constraints in designing spatial interactions for virtual tasks.
- Creativity Evaluation:** There are several studies in design literature that aim at quantifying the ideation process [178, 179]. In all these case, ideas are visually communicated through sketches and are judged by a set of independent reviewers. However, the quality of sketches severely affects the judgment of early phase design ideas. Recently, Kudrowitz et al. [180] stated that “*correct perspective and realistic proportions are factors of a sketch that can influence the perceived creativity of an idea*”.

Enabling the direct communication of ideas as 3D artifacts, as demonstrated in this thesis, enables a more realistic and reliable enforcement of the known metrics for ideation evaluation.

7.3 Future Directions

One of the core goals of this thesis was to demonstrate, through concrete software prototypes, that spatial interactions can be useful as well as usable for direct creation of 3D virtual artifacts towards quick idea generation. Below, we identify some directions that can help guide future research on spatial interactions for design ideation.

- Parametric Design:** The contexts presented in this work (e.g. pottery) were primarily defined towards the design of free-form shapes. However, there is still a large body of work that is required towards the design of parametric shapes using spatial interactions. In terms of interaction techniques, a deeper investigation of precise spatial input is required in the context of the design of 3D shapes. For instance, methods for selecting both geometric (e.g. vertex, edge, face) and semantic (e.g. protrusion, fillet) features on CAD models need be designed and investigated for engineering designs. Due to parametric interdependencies within 3D engineering models, modification of one model feature inherently necessitates adjustment of others. This is analogous to “brushing and linking” commonly studied in information visualization, where a changes in one form of data representation gets automatically reflected in others. Drawing from existing HCI guidelines, mobile interaction metaphors can be explored for representing, visualizing, and controlling these interdependent parametric relations in 3D models.
- Functional Design and Analysis:** While this work focused on the creation of static geometric designs and 3D assemblies, the functional aspect of early design ideas is a critical component of the ideation process. Thus, investigating the role of spatial interactions for defining functional elements to geometric shapes is an important direction for future research. For example, it would be interesting to investigate how

physical human action could be transformed into the kinematic relationships between multiple parts of a 3D design. Another example can be the design and evaluation of spatial methods for annotating attributes such as materials and functions on to shapes created during an ideation process. Similarly, the physicality of spatial interactions also lends itself to several novel scenarios for design analysis. Exploring function based design and analysis would be interesting topics of investigation in future research.

- **Comparison of Spatial Input:** From the interaction perspective, the generality of the symbolic and geometric approaches allows for the accommodation of different representations of three-dimensional user input. For instance, we have seen two fundamentally different representations of the hand: a point and a point-cloud. There is also a third possibility for representing the hand as a skeletal model. On the other hand, we have also seen a variety of geometric modeling contexts that are possible just with a single sweep surface representation. Thus, an in-depth evaluation and comparison of hand representations is an important direction for future research across the various geometric modeling contexts such as object manipulation, shape deformation and the creation of new shapes.
- **Collaborative Ideation:** One of the unexplored functionality provided by our phone-based tangible approach is that it enables new possibilities for collaborative ideation. There are several interesting aspects of early phase design collaboration that need to be investigated in the context of our tangible approach. The spatial design capabilities provided by systems such as *MobiSweep* can now be used to gain deeper insights on (a) how users segment design tasks during ideation involving multiple geometric parts and those involving a single shape, and (b) how collocated and distant collaborations occur during spatial design ideation using single large-screen displays and distributed personal devices respectively.

7.4 Closing Statement

Fundamentally, any instance of human-computer interaction is a cycle comprised of (a) input by the user through a hardware system, (b) computation in a software system, and (c) output to the user by a hardware system again. In the last five years, i.e. since the beginning of this work, several advancements in mid-air input and output technologies have either taken place or are in the making. After the Kinect, we have witnessed several other camera technologies such as PrimeSense Carmine, SoftKinetic, and Leap Motion, leading to the integration of depth sensors in tablets (Google Tango). We have also witnessed, in parallel, the commercialization of virtual and augmented reality hardware such as Oculus Rift and Meta SpaceGlass leading to advanced visual technologies such as HoloLens. Amidst these developments, this thesis dealt with the computational core in the specific context of ideation. Given the context, the overarching contribution of this work was to shift towards a new paradigm – **what you do is what you get**". Through three concrete embodiments this paradigm, we learned that creative tasks such as design demand an uninhibited flow between what a designer is *thinking* and what the designer is *doing* to communicate the thought. We believe that this work took a step towards enabling this flow through embodied interactions for spatial design ideation.

LIST OF REFERENCES

LIST OF REFERENCES

- [1] Engelbart, D. C. X-y position indicator for a display system (us patent# 3,541,541), 1970.
- [2] Hsu, W. and Liu, B. Conceptual design: issues and challenges. *Computer-Aided Design*, 32(14):849 – 850, 2000.
- [3] Goel, V. *Sketches of thought*. MIT Press, 1995.
- [4] Olofsson, E. and Sjöln, K. *Design sketching*. KEEOS Design Books, 2007.
- [5] Schembri, M., Farrugia, P., Wodehouse, A. J., Grierson, H., and Kovacevic, A. Influence of sketch types on distributed design team work. *CoDesign*, 11(2):99–118, 2015.
- [6] Horváth, I. On some crucial issues of computer support of conceptual design. *Product engineering ecodeign technologies and green energy*, page 123142, 2004.
- [7] Dorta, T. S. Design flow and ideation. *International Journal of Architectural Computing*, 6(3):299–316, 2008.
- [8] Geyer, F., Budzinski, J., and Reiterer, H. Ideavis: A hybrid workspace and interactive visualization for paper-based collaborative sketching sessions. In *ACM Nordic Conference on Human-Computer Interaction: Making Sense Through Design (NordiCHI '12)*, pages 331–340, 2012.
- [9] Zhao, Z., Badam, S. K., Chandrasegaran, S., Park, D. G., Elmqvist, N. L., Kisselburgh, L., and Ramani, K. skwiki: A multimedia sketching system for collaborative creativity. In *ACM Conference on Human Factors in Computing Systems (CHI '14)*, pages 1235–1244, 2014.
- [10] Jonson, B. Design ideation: the conceptual sketch in the digital age. *Design Studies*, 26(6):613 – 624, 2005.
- [11] Varga, E. Using hand motions in conceptual shape design: Theories, methods and tools. In Talaba, D. and Amditis, A., editors, *Product Engineering*, pages 367–382. Springer Netherlands, 2008.
- [12] Bae, S.-H., Balakrishnan, R., and Singh, K. Everybodylovessketch: 3d sketching for a broader audience. In *ACM Symposium on User Interface Software and Technology (UIST '09)*, pages 59–68, 2009.
- [13] Sproull, R. F. Parts of the frontier are hard to move. 24(9), 1990.
- [14] Scali, S., Shillito, A. M., Wright, M., Place, L., Clerk, J., and Building, M. Thinking in space: concept physical models and the call for new digital tools. *Craft in the 21st Century Theorising Change and Practice Edinburgh*, 2002.

- [15] Stappers, P. and Hennessey, J. Toward electronic napkins and beer mats: Computer support for visual ideation skills. In Paton, R. and Neilson, I., editors, *Visual Representations and Interpretations*, pages 220–225. Springer London, 1999.
- [16] Cheshire, D., Evans, M., Dean, C., and Al., E. *Haptic modeling – an alternative industrial design methodology?*, volume 2001, page 124128. Citeseer.
- [17] *Where the Action is: The Foundations of Embodied Interaction*. MIT Press, Cambridge, MA, USA, 2001.
- [18] Fogtman, M. H., Fritsch, J., and Kortbek, K. J. Kinesthetic interaction: Revealing the bodily potential in interaction design. In *Proceedings of the ACM Australasian Conference on Computer-Human Interaction: Designing for Habitus and Habitat (OZCHI '08)*, pages 89–96, 2008.
- [19] Jetter, H.-C., Leifert, S., Gerken, J., Schubert, S., and Reiterer, H. Does (multi-)touch aid users' spatial memory and navigation in 'panning' and in 'zooming' & panning' uis? In *Proceedings of the ACM International Working Conference on Advanced Visual Interfaces (AVI '12)*, pages 83–90, 2012.
- [20] Klemmer, S. R., Hartmann, B., and Takayama, L. How bodies matter: Five themes for interaction design. In *ACM Conference on Designing Interactive Systems (DIS '06)*, pages 140–149, 2006.
- [21] Willis, K. D., Lin, J., Mitani, J., and Igarashi, T. Spatial sketch: Bridging between movement & fabrication. In *In Proceedings of the Fourth International ACM Conference on Tangible, Embedded, and Embodied Interaction (TEI '10)*, pages 5–12, 2010.
- [22] Piya, C. and Ramani, K. Proto-tai: Quick design prototyping using tangible assisted interfaces. In *Proceedings of The ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages V01AT02A097–V01AT02A097, 2014.
- [23] Jacob, R. J. K., Girouard, A., Hirshfield, L. M., Horn, M. S., Shaer, O., Solovey, E. T., and Zigelbaum, J. Reality-based interaction: a framework for post-WIMP interfaces. In *Proceedings of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, CHI '08, pages 201–210, New York, NY, USA, 2008. ACM.
- [24] Wilson, M. Six views of embodied cognition. *Psychonomic Bulletin and Review*, 9(4):625–636, 2002.
- [25] Clark, A. Embodiment and the philosophy of mind. *Royal Institute of Philosophy Supplement*, 43:35–51, 1998.
- [26] Chang, T.-I., Lee, J.-H., Kim, M.-S., and Hong, S. J. Direct manipulation of generalized cylinders based on b-spline motion. *The Visual Computer*, 14(5/6):228–239, 1998.
- [27] Horváth, I. and Vergeest, J. S. M. Theoretical fundamentals of natural representation of shapes generated with gestural devices. In *Proceedings of TMCE 98*, pages 393–409, 1998.

- [28] Pavlovic, V. I., Sharma, R., and Huang, T. S. Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:677–695, 1997.
- [29] Seth, A., Vance, J. M., and Oliver, J. H. Virtual reality for assembly methods prototyping: a review. *Virtual Reality*, 15(1):5–20, 2011.
- [30] Segen, J. and Kumar, S. Gesture vr: vision-based 3d hand interace for spatial interaction. In *Proceedings of the sixth ACM international conference on Multimedia, MULTIMEDIA '98*, pages 455–464, New York, NY, USA, 1998. ACM.
- [31] Schkolne, S., Pruett, M., and Schröder, P. Surface drawing: creating organic 3d shapes with the hand and tangible tools. In *Proceedings of the SIGCHI conference on Human factors in computing systems, CHI '01*, pages 261–268, New York, NY, USA, 2001. ACM.
- [32] Llamas, I., Kim, B., Gargus, J., Rossignac, J., and Shaw, C. D. Twister: a space-warp operator for the two-handed editing of 3d shapes. *ACM Trans. Graph.*, 22(3):663–668, July 2003.
- [33] Keefe, D. F., Zeleznik, R. C., and Laidlaw, D. H. Drawing on air: Input techniques for controlled 3d line illustration. *IEEE Trans. Vis. Comput. Graph.*, 13(5):1067–1081, 2007.
- [34] Fuge, M., Yumer, M. E., Orbay, G., and Kara, L. B. Conceptual design and modification of freeform surfaces using dual shape representations in augmented reality environments. *Computer-Aided Design*, June 2011.
- [35] Wang, R., Paris, S., and Popović, J. 6d hands: markerless hand-tracking for computer aided design. In *Proceedings of the 24th annual ACM symposium on User interface software and technology, UIST '11*, pages 549–558, New York, NY, USA, 2011. ACM.
- [36] Holz, C. and Wilson, A. Data miming: inferring spatial object descriptions from human gesture. In *Proceedings of the 2011 annual conference on Human factors in computing systems, CHI '11*, pages 811–820, New York, NY, USA, 2011. ACM.
- [37] Erol, A., Bebis, G., Nicolescu, M., Boyle, R. D., and Twombly, X. A review on vision-based full dof hand motion estimation. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops - Volume 03*, pages 75–, Washington, DC, USA, 2005. IEEE Computer Society.
- [38] Argyros, A. A. and Lourakis, M. I. A. Binocular hand tracking and reconstruction based on 2d shape matching. In *Proceedings of the 18th International Conference on Pattern Recognition - Volume 01, ICPR '06*, pages 207–210, Washington, DC, USA, 2006. IEEE Computer Society.
- [39] Vassilis, A. and Stan, S. Estimating 3d hand pose from a cluttered image. In *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR03)*, pages 432–439, 2003.
- [40] Manresa, C., Varona, J., Mas, R., and Perales, F. J. Hand Tracking and Gesture Recognition for Human-Computer Interaction. In *Electronic Letters on Computer Vision and Image Analysis*, volume 5, pages 96–104, 2005.

- [41] Panin, G., Klose, S., and Knoll, A. Real-time articulated hand detection and pose estimation. In *Proceedings of the 5th International Symposium on Advances in Visual Computing: Part II*, ISVC '09, pages 1131–1140, Berlin, Heidelberg, 2009. Springer-Verlag.
- [42] Rehg, J. M. and Kanade, T. Model-based tracking of self-occluding articulated objects. In *In ICCV*, pages 612–617, 1995.
- [43] Rezaei, A., Vafadoost, M., Rezaei, S., and Daliri, A. 3d pose estimation via elliptical fourier descriptors for deformable hand representations. In *Bioinformatics and Biomedical Engineering, 2008. ICBBE 2008. The 2nd International Conference on*, pages 1871–1875, may 2008.
- [44] Wu, Y., Lin, J. Y., and Huang, T. S. Capturing natural hand articulation. In *In ICCV*, volume 2, pages 426–432, 2001.
- [45] Tangkuampien, T. and Suter, D. Real-time human pose inference using kernel principal component pre-image approximations. In *In British Machine Vision Conference*, 2006.
- [46] Hamer, H., Schindler, K., Koller-Meier, E., and Gool, L. V. Tracking a hand manipulating an object. In *IEEE International Conference on Computer Vision*, 2009.
- [47] Kato, M., Chen, Y.-W., and Xu, G. Articulated hand motion tracking using ica-based motion analysis and particle filtering. *Journal of Multimedia*, 1(3):52–60, 2006.
- [48] Sudderth, E. B., Mandel, M. I., Freeman, W. T., and Willsky, A. S. Visual hand tracking using nonparametric belief propagation. In *Propagation, IEEE Workshop on Generative Model Based Vision*, page 189, 2004.
- [49] Suryanarayan, P., Subramanian, A., and Mandalapu, D. Dynamic hand pose recognition using depth data. In *Proceedings of the 2010 20th International Conference on Pattern Recognition, ICPR '10*, pages 3105–3108, Washington, DC, USA, 2010. IEEE Computer Society.
- [50] Iason Oikonomidis, N. K. and Argyros, A. Efficient model-based 3d tracking of hand articulations using kinect. In *Proceedings of the British Machine Vision Conference*, pages 101.1–101.11. BMVA Press, 2011. <http://dx.doi.org/10.5244/C.25.101>.
- [51] Choi, J. *Developing a 3-Dimensional Kinematic Model of the Hand for Ergonomic Analyses of Hand Posture, Hand Space Envelope, and Tendon Excursion*. PhD thesis, The University of Michigan, 2008.
- [52] Cobos, S., Ferre, M., Sanchez Uran, M. A., Ortego, J., and Pena, C. Efficient human hand kinematics for manipulation tasks. *2008 IEEEERSJ International Conference on Intelligent Robots and Systems*, pages 2246–2251, 2008.
- [53] Lin, J., Wu, Y., and Huang, T. S. Modeling the constraints of human hand motion. In *Proceedings of the Workshop on Human Motion (HUMO'00)*, HUMO '00, pages 121–, Washington, DC, USA, 2000. IEEE Computer Society.
- [54] van Nierop, O. A., van der Helm, A., Overbeeke, K. J., and Djajadiningrat, T. J. A natural human hand model. *The Visual Computer*, 24(1):31–44, January 2008.

- [55] Ingram, J. N., Körding, K. P., Howard, I. S., and Wolpert, D. M. The statistics of natural hand movements. *Experimental brain research*, 188(2):223–36, 2008.
- [56] Pitarch, E. P., Yang, J., and Abdel-Malek, K. Virtual human hand: Grasping and simulation. In *HCI (11)*, pages 140–149, 2009.
- [57] Marler, T., Johnson, R., Goussous, F., Murphy, C., Beck, S., and Abdel-Malek, K. Human grasp prediction and analysis. In Pham, H., editor, *Safety and Risk Modeling and Its Applications*, Springer Series in Reliability Engineering, pages 397–424. Springer London, 2011.
- [58] Feix, T., Schmiedmayer, H.-B., Romero, J., and Kragić, D. A comprehensive grasp taxonomy. In *in Robotics, Science and Systems Conference: Workshop on Understanding the Human Hand for Advancing Robotic Manipulation*, 2009.
- [59] Rusak, Z., Antonya, C., van der Vegte, W., Horváth, I., and Varga, E. A new approach to interactive grasping simulation of product concepts in a virtual reality environment. *ASME Conference Proceedings*, 2007(48035):213–221, 2007.
- [60] Mäntylä, M. *An introduction to solid modeling*. Computer Science Press, Inc., New York, NY, USA, 1987.
- [61] Rogers, D. F. and Adams, J. A. *Mathematical Elements for Computer Graphics*. McGraw-Hill Higher Education, 2nd edition, 1989.
- [62] Zhang, D. and Lu, G. Review of shape representation and description techniques. *Pattern Recognition*, 37(1):1–19, 2004.
- [63] Lipman, Y., Sorkine, O., Cohen-Or, D., Levin, D., Rössl, C., and Seidel, H.-P. Differential coordinates for interactive mesh editing. In *Shape Modeling International*, pages 181–190. Society Press, 2004.
- [64] Welch, W. and Witkin, A. Variational surface modeling. *SIGGRAPH Comput. Graph.*, 26:157–166, July 1992.
- [65] Taubin, G. A signal processing approach to fair surface design. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, SIGGRAPH '95, pages 351–358, New York, NY, USA, 1995. ACM.
- [66] Joshi, P., Meyer, M., DeRose, T., Green, B., and Sanocki, T. Harmonic coordinates for character articulation. *ACM Trans. Graph.*, 26, July 2007.
- [67] Zhou, K., Huang, J., Snyder, J., Liu, X., Bao, H., Guo, B., and Shum, H.-Y. Large mesh deformation using the volumetric graph laplacian. *ACM Trans. Graph.*, 24:496–503, July 2005.
- [68] Alexa, M., Cohen-Or, D., and Levin, D. As-rigid-as-possible shape interpolation. pages 157–164, 2000.
- [69] Forstmann, S., Ohya, J., Krohn-Grimberghe, A., and McDougall, R. Deformation styles for spline-based skeletal animation. pages 141–150, 2007.
- [70] Jacobson, A., Baran, I., Popović, J., and Sorkine, O. Bounded biharmonic weights for real-time deformation. *ACM Transactions on Graphics (proceedings of ACM SIGGRAPH)*, 30(4):78:1–78:8, 2011.

- [71] Igarashi, T., Moscovich, T., and Hughes, J. F. As-rigid-as-possible shape manipulation. *ACM Trans. Graph.*, 24:1134–1141, July 2005.
- [72] Botsch, M. and Kobbelt, L. An intuitive framework for real-time freeform modeling. *ACM Trans. Graph.*, 23:630–634, August 2004.
- [73] Singh, K. and Fiume, E. Wires: a geometric deformation technique. pages 405–414, 1998.
- [74] Sederberg, T. W. and Parry, S. R. Free-form deformation of solid geometric models. *SIGGRAPH Comput. Graph.*, 20:151–160, August 1986.
- [75] Hirota, G., Maheshwari, R., and Lin, M. Fast volume-preserving free-form deformation using multi-level optimization. *Computer-Aided Design*, 32(8-9):499 – 512, 2000.
- [76] Barr, A. H. Global and local deformations of solid primitives. *SIGGRAPH Comput. Graph.*, 18:21–30, January 1984.
- [77] Hsu, W. M., Hughes, J. F., and Kaufman, H. Direct manipulation of free-form deformations. In *SIGGRAPH '92: Proceedings of the 19th annual conference on Computer graphics and interactive techniques*, pages 177–184, New York, NY, USA, 1992. ACM Press.
- [78] Angelidi, A., Canif, M., Wyvill, G., and King, S. Swirling-sweepers: constant-volume modeling. In *Computer Graphics and Applications, 2004. PG 2004. Proceedings. 12th Pacific Conference on*, pages 10 – 15, oct. 2004.
- [79] Choi, B. K. and Lee, C. S. Sweep surfaces modelling via coordinate transformation and blending. *Computer-Aided Design*, 22(2):87–96, 1990.
- [80] Lee, J.-H. Modeling generalized cylinders using direction map representation. *Computer-Aided Design*, 37(8):837 – 846, 2005. *CAD '04 Special Issue: Modelling and Geometry Representations for CAD*.
- [81] Kim, M.-S., Park, E.-J., and Lee, H.-Y. Modeling and animation of generalized cylinders with variable radius offset space curves. *The Journal of Visualization and Computer Animation*, 5:189–207, 1994.
- [82] Hyun, D.-E., Yoon, S.-H., Chang, J.-W., Seong, J.-K., Kim, M.-S., and Jüttler, B. Sweep-based human deformation. *The Visual Computer*, 21(8-10):542–550, 2005.
- [83] Yoon, S.-H. and Kim, M.-S. Sweep-based freeform deformations. *Comput. Graph. Forum*, 25(3):487–496, 2006.
- [84] Igarashi, Y. and Igarashi, T. Designing plush toys with a computer. *Commun. ACM*, 52:81–88, December 2009.
- [85] Nealen, A., Igarashi, T., Sorkine, O., and Alexa, M. Fibermesh: designing freeform surfaces with 3d curves. *ACM Trans. Graph.*, 26, July 2007.
- [86] Talton, J. O., Gibson, D., Yang, L., Hanrahan, P., and Koltun, V. Exploratory modeling with collaborative design spaces. In *Proceedings of the 2nd Annual ACM SIGGRAPH Conference and Exhibition in Asia*, New York, NY, USA, 2009. ACM Press.

- [87] Chaudhuri, S. and Koltun, V. Data-driven suggestions for creativity support in 3d modeling. *ACM Trans. Graph.*, 29(6):183, 2010.
- [88] Chaudhuri, S., Kalogerakis, E., Guibas, L., and Koltun, V. Probabilistic reasoning for assembly-based 3D modeling. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 30.
- [89] Merrell, P. Example-based model synthesis. In *In I3D 07: Proceedings of the 2007 symposium on Interactive 3D graphics and games*, pages 105–112. ACM Press, 2007.
- [90] Merrell, P. and Manocha, D. Continuous model synthesis. *ACM Trans. Graph.*, 27(5):158, 2008.
- [91] Merrell, P. and Manocha, D. Constraint-based model synthesis. In *2009 SIAM/ACM Joint Conference on Geometric and Physical Modeling, SPM '09*, pages 101–111, New York, NY, USA, 2009. ACM.
- [92] Merrell, P. and Manocha, D. Model Synthesis: A General Procedural Modeling Algorithm. *IEEE Transactions on Visualization and Computer Graphics*, 17(6):715–728, June 2011.
- [93] Lee, G., Fradet, L., Ketcham, C., and Dounskaia, N. Efficient control of arm movements in advanced age. *Experimental Brain Research*, 177:78–94, 2007. 10.1007/s00221-006-0648-7.
- [94] Han, L. and Rudolph, L. Inverse kinematics for a serial chain with joints under distance constraints. In *Robotics: Science and Systems*, 2006.
- [95] Han, G., Hwang, J., Choi, S., and Kim, G. J. Ar pottery: Experiencing pottery making in the augmented space. In Shumaker, R., editor, *Virtual Reality, Second International Conference, ICVR 2007, Held as part of HCI International 2007, Beijing, China, July 22-27, 2007, Proceedings*, volume 4563 of *Lecture Notes in Computer Science*, pages 642–650. Springer, 2007.
- [96] Liaw, A. and Wiener, M. Classification and regression by randomforest. *R news*, 2(3):18–22, 2002.
- [97] Hinckley, K., Pausch, R., Goble, J. C., and Kassell, N. F. A survey of design issues in spatial input. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, pages 213–222, 1994.
- [98] Hand, C. A survey of 3d interaction techniques. *Computer Graphics Forum*, 16(5):269–281, 1997.
- [99] Hinckley, K., Pausch, R., Proffitt, D., and Kassell, N. F. Two-handed virtual manipulation. *ACM Transactions on Computer-Human Interaction*, 5(3):260–302, 1998.
- [100] Jankowski, J. and Hachet, M. A survey of interaction techniques for interactive 3d environments. In *Eurographics State of the Art Reports*, 2013.
- [101] Özacar, K., Takashima, K., and Kitamura, Y. Direct 3d object manipulation on a collaborative stereoscopic display. In *Proceedings of the Symposium on Spatial User Interaction*, pages 69–72, 2013.

- [102] Apostolellis, P., Laha, B., and Bowman, D. A gaming interface using body gestures for collaborative navigation. In *IEEE Symposium on 3D User Interfaces*, pages 185–186, March 2012.
- [103] Keefe, D. F., Feliz, D. A., Moscovich, T., Laidlaw, D. H., and LaViola, J. J., Jr. Cavepainting: a fully immersive 3d artistic medium and interactive experience. In *Proceedings of the 2001 symposium on Interactive 3D graphics*, I3D '01, pages 85–93, New York, NY, USA, 2001. ACM.
- [104] Keefe, D. F. From gesture to form: The evolution of expressive freehand spatial interfaces. *Leonardo*, 44(5):460–461, 2011.
- [105] LaViola, J. J. and Keefe, D. F. 3d spatial interaction: Applications for art, design, and science. In *ACM SIGGRAPH 2011 Courses*, pages 1:1–1:75, 2011.
- [106] Korida, K., Nishino, H., and Utsumiya, K. An interactive 3d interface for a virtual ceramic art work environment. In *Virtual Systems and MultiMedia, 1997. VSMM '97. Proceedings., International Conference on*, pages 227–234, 1997.
- [107] Fuge, M., Yumer, M. E., Orbay, G., and Kara, L. B. Conceptual design and modification of freeform surfaces using dual shape representations in augmented reality environments. *Computer-Aided Design*, 44(10):1020 – 1032, 2012.
- [108] Agrawala, M., Beers, A. C., and Levoy, M. 3d painting on scanned surfaces. In *ACM Symposium on Interactive 3D Graphics (I3D)*, pages 145–ff., 1995.
- [109] McDonnell, K. T., Qin, H., and Wlodarczyk, R. A. Virtual clay: A real-time sculpting system with haptic toolkits. In *Proceedings of the 2001 Symposium on Interactive 3D Graphics*, I3D '01, pages 179–190, New York, NY, USA, 2001. ACM.
- [110] Lee, J., Han, G., and Choi, S. Haptic pottery modeling using circular sector element method. In *Proceedings of the 6th international conference on Haptics: Perception, Devices and Scenarios*, EuroHaptics '08, pages 668–674, Berlin, Heidelberg, 2008. Springer-Verlag.
- [111] Galyean, T. A. and Hughes, J. F. Sculpting: An interactive volumetric modeling technique. *Proceedings of the ACM Conference on Computer Graphics and Interactive Techniques*, 25(4):267–274, 1991.
- [112] Sheng, J., Balakrishnan, R., and Singh, K. An interface for virtual 3d sculpting via physical proxy. In *ACM International Conference on Computer Graphics and Interactive Techniques in Australasia and Southeast Asia*, pages 213–220, 2006.
- [113] Kry, P. G., Pihuit, A., Bernhardt, A., and Cani, M.-P. Handnavigator: Hands-on interaction for desktop virtual reality. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, pages 53–60, 2008.
- [114] Pihuit, A., Kry, P., and Cani, M.-P. Hands on virtual clay. In *IEEE Conference on Shape Modeling and Applications*, pages 267–268, 2008.
- [115] Cho, S., Heo, Y., and Bang, H. Turn: a virtual pottery by real spinning wheel. In *ACM SIGGRAPH 2012 Emerging Technologies*, SIGGRAPH '12, pages 25:1–25:1, New York, NY, USA, 2012. ACM.

- [116] Hilliges, O., Kim, D., Izadi, S., Weiss, M., and Wilson, A. Holodesk: Direct 3d interactions with a situated see-through display. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pages 2421–2430, 2012.
- [117] Han, Y.-C. and Han, B.-J. Virtual pottery: a virtual 3d audiovisual interface using natural hand motions. *Multimedia Tools and Applications*, pages 1–17, 2013.
- [118] Song, J., Cho, S., Baek, S.-Y., Lee, K., and Bang, H. Gafinc: Gaze and finger control interface for 3d model manipulation in {CAD} application. *Computer-Aided Design*, 46(0):239 – 245, 2014.
- [119] Leap motion sculpting. <https://airspace.leapmotion.com/apps/sculpting/windows>, 2013.
- [120] Baudel, T. and Beaudouin-Lafon, M. Charade: Remote control of objects using free-hand gestures. *Commun. ACM*, 36(7):28–35, July 1993.
- [121] Rateau, H., Grisoni, L., and De Araujo, B. Sub-space gestures. Elements of design for mid-air interaction with distant displays. Research Report RR-8342, INRIA, August 2013.
- [122] Leap motion skeletal tracking. <https://developer.leapmotion.com/features>, 2015.
- [123] Walter, R., Bailly, G., Valkanova, N., and Müller, J. Cuenesics: Using mid-air gestures to select items on interactive public displays. In *Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices & Services, MobileHCI '14*, pages 299–308, New York, NY, USA, 2014. ACM.
- [124] Botsch, M., Kobbelt, L., Pauly, M., Alliez, P., and uno Levy, B. *Polygon Mesh Processing*. AK Peters, 2010.
- [125] Holt, C. C. Forecasting seasonals and trends by exponentially weighted moving averages. *International Journal of Forecasting*, 20(1):5 – 10, 2004.
- [126] Rosman, G., Bronstein, A. M., Bronstein, M. M., and Kimmel, R. Articulated motion segmentation of point clouds by group-valued regularization. In *Proceedings of the 5th Eurographics Conference on 3D Object Retrieval, EG 3DOR'12*, pages 77–84, Aire-la-Ville, Switzerland, Switzerland, 2012. Eurographics Association.
- [127] Rosman, G., Bronstein, A. M., Bronstein, M. M., Tai, X.-C., and Kimmel, R. Group-valued regularization for analysis of articulated motion. In Fusiello, A., Murino, V., and Cucchiara, R., editors, *Computer Vision ECCV 2012. Workshops and Demonstrations*, volume 7583 of *Lecture Notes in Computer Science*, pages 52–62. Springer Berlin Heidelberg, 2012.
- [128] Ren, G. and O'Neill, E. 3d selection with freehand gesture. *Computers & Graphics*, 37(3):101 – 120, 2013.
- [129] Walter, R., Bailly, G., and Müller, J. Strikeapose: Revealing mid-air gestures on public displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '13*, pages 841–850, New York, NY, USA, 2013. ACM.
- [130] Karam, M. and Schraefel, M. C. A taxonomy of gestures in human computer interactions. Technical report, University of Southampton, 2005.

- [131] Quek, F., McNeill, D., Bryll, R., Duncan, S., Ma, X.-F., Kirbas, C., McCullough, K. E., and Ansari, R. Multimodal human discourse: Gesture and speech. *ACM Trans. Comput.-Hum. Interact.*, 9(3):171–193, September 2002.
- [132] Kameyama, K.-I. Virtual clay modeling system. In *Proceedings of the ACM symposium on Virtual reality software and technology*, VRST '97, pages 197–200, New York, NY, USA, 1997. ACM.
- [133] Jeannerod, M. The formation of finger grip during prehension. a cortically mediated visuomotor pattern. *Behavioural Brain Research*, 19(2):99 – 116, 1986.
- [134] Armbruster, C. and Spijkers, W. Movement planning in prehension: do intended actions influence the initial reach and grasp movement? *Motor Control*, 10(4):311, 2006.
- [135] Cutkosky, M. On grasp choice, grasp models, and the design of hands for manufacturing tasks. *Robotics and Automation, IEEE Transactions on*, 5(3):269–279, Jun 1989.
- [136] Zheng, J., De La Rosa, S., and Dollar, A. An investigation of grasp type and frequency in daily household and machine shop tasks. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 4169–4175, May 2011.
- [137] Mapes, D. P. and Moshell, J. M. A two handed interface for object manipulation in virtual environments. *Presence*, 4(4):403–416, 1995.
- [138] Boulic, R., Rezzonico, S., and Thalmann, D. Multi-finger manipulation of virtual objects. In *In Proc. of the ACM Symposium on Virtual Reality Software and Technology (VRST '96)*, pages 67–74, 1996.
- [139] Leyton, M. A process-grammar for shape. *Artif. Intell.*, 34(2):213–247, March 1988.
- [140] Delamé, T., Léon, J. C., Cani, M. P., and Blanch, R. Gesture-based design of 2d contours: An alternative to sketching? In *Proceedings of the Eighth Eurographics Symposium on Sketch-Based Interfaces and Modeling*, SBIM '11, pages 63–70, New York, NY, USA, 2011. ACM.
- [141] Vinayak, Murugappan, S., Liu, H., and Ramani, K. Shape-it-up: Hand gesture based creative expression of 3d shapes using intelligent generalized cylinders. *Computer-Aided Design*, 45(2):277 – 287, 2013. Solid and Physical Modeling 2012.
- [142] Stuerzlinger, W. and Wingrave, C. The value of constraints for 3d user interfaces. In Brunnnett, G., Coquillart, S., and Welch, G., editors, *Virtual Realities*, pages 203–223. Springer Vienna, 2011.
- [143] Yoo, J.-C. and Han, T. Fast normalized cross-correlation. *Circuits, Systems and Signal Processing*, 28(6):819–843, 2009.
- [144] Rousseeuw, P. J., Ruts, I., and Tukey, J. W. The bagplot: A bivariate boxplot. *The American Statistician*, 53(4):382–387, 1999.
- [145] Vinayak and Ramani, K. A gesture-free geometric approach for mid-air expression of design intent in 3d virtual pottery. *Computer-Aided Design*, 69:11 – 24, 2015.

- [146] Bærentzen, J. A., Abdrashitov, R., and Singh, K. Interactive shape modeling using a skeleton-mesh co-representation. *ACM Trans. Graph.*, 33(4):132:1–132:10, July 2014.
- [147] Bowman, D. and Hodges, L. An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. In *Proceedings of the ACM Symposium on Interactive 3D Graphics*, pages 53–58, 1997.
- [148] Choumane, A., Casiez, G., and Grisoni, L. Buttonless clicking: Intuitive select and pick-release through gesture analysis. In *Virtual Reality Conference (VR), 2010 IEEE*, pages 67–70, March 2010.
- [149] Lue, J. and Schulze, J. P. 3d whiteboard: collaborative sketching with 3d-tracked smart phones. volume 9012, pages 901204–901204–12, 2014.
- [150] Katzakis, N., Kiyokawa, K., and Takemura, H. Plane-casting: 3d cursor control with a smartphone. In *ACM Asia Pacific Conference on Computer Human Interaction*, pages 199–200, 2013.
- [151] Baglioni, M., Lecolinet, E., and Guiard, Y. Jerktilts: Using accelerometers for eight-choice selection on mobile devices. In *ACM Conference on Multimodal Interfaces*, pages 121–128, 2011.
- [152] Ruiz, J., Li, Y., and Lank, E. User-defined motion gestures for mobile interaction. In *ACM Conference on Human Factors in Computing Systems*, pages 197–206, 2011.
- [153] Bertolo, D., Vivian, R., and Dinet, J. A set of interactions to rotate solids in 3d geometry context. In *ACM Extended Abstracts on Human Factors in Computing Systems*, pages 625–630, 2013.
- [154] Teather, R. J. and MacKenzie, I. S. Position vs. velocity control for tilt-based interaction. In *Graphics Interface Conference*, pages 51–58, 2014.
- [155] Knoedel, S. and Hachet, M. Multi-touch rst in 2d and 3d spaces: Studying the impact of directness on user performance. In *IEEE Symposium on 3D User Interfaces*, pages 75–78, March 2011.
- [156] Seifert, J., Bayer, A., and Rukzio, E. Pointerphone: Using mobile phones for direct pointing interactions with remote displays. In *Human-Computer Interaction*, volume 8119, pages 18–35. Springer Berlin Heidelberg, 2013.
- [157] Liang, H.-N., Williams, C., Semegen, M., Stuerzlinger, W., and Irani, P. User-defined surface+motion gestures for 3d manipulation of objects at a distance through a mobile device. In *ACM Asia Pacific Conference on Computer Human Interaction*, pages 299–308, 2012.
- [158] Nancel, M., Chapuis, O., Pietriga, E., Yang, X.-D., Irani, P. P., and Beaudouin-Lafon, M. High-precision pointing on large wall displays using small handheld devices. In *ACM Conference on Human Factors in Computing Systems*, pages 831–840, 2013.
- [159] Debarba, H. G., Grandi, J., Maciel, A., Nedel, L., and Boulic, R. Disambiguation canvas: A precise selection technique for virtual environments. In *Human-Computer Interaction*, volume 8119, pages 388–405. Springer Berlin Heidelberg, 2013.

- [160] Boring, S., Jurmu, M., and Butz, A. Scroll, tilt or move it: Using mobile phones to continuously control pointers on large public displays. In *ACM Australian Computer-Human Interaction Special Interest Group*, pages 161–168, 2009.
- [161] Benzina, A., Toennis, M., Klinker, G., and Ashry, M. Phone-based motion control in vr: Analysis of degrees of freedom. In *ACM Extended Abstracts on Human Factors in Computing Systems*, pages 1519–1524, 2011.
- [162] Medeiros, D., Teixeira, L., Carvalho, F., Santos, I., and Raposo, A. A tablet-based 3d interaction tool for virtual engineering environments. In *ACM SIGGRAPH Conference on Virtual-Reality Continuum and Its Applications in Industry*, pages 211–218, 2013.
- [163] Steed, A. and Julier, S. Design and implementation of an immersive virtual reality system based on a smartphone platform. In *IEEE Symposium on 3D User Interfaces*, pages 43–46, March 2013.
- [164] Tsandilas, T., Appert, C., Bezerianos, A., and Bonnet, D. Coordination of tilt and touch in one- and two-handed use. In *ACM Conference on Human Factors in Computing Systems*, pages 2001–2004, 2014.
- [165] Song, P., Goh, W. B., Fu, C.-W., Meng, Q., and Heng, P.-A. Wysiwyf: Exploring and annotating volume data with a tangible handheld device. In *ACM Conference on Human Factors in Computing Systems*, pages 1333–1342, 2011.
- [166] Bergé, L.-P., Serrano, M., Perelman, G., and Dubois, E. Exploring smartphone-based interaction with overview+detail interfaces on 3d public displays. In *ACM Conference on Human-computer Interaction with Mobile Devices and Services*, pages 125–134, 2014.
- [167] Xin, M., Sharlin, E., and Sousa, M. C. Napkin sketch: Handheld mixed reality 3d sketching. In *ACM Symposium on Virtual Reality Software and Technology*, pages 223–226, 2008.
- [168] Scheible, J. and Ojala, T. Mobispray: Mobile phone as virtual spray can for painting big anytime anywhere on anything. In *ACM SIGGRAPH 2009 Art Gallery*, pages 5:1–5:10, 2009.
- [169] Lakatos, D., Blackshaw, M., Olwal, A., Barryte, Z., Perlin, K., and Ishii, H. T(ether): Spatially-aware handhelds, gestures and proprioception for multi-user 3d modeling and animation. In *ACM Symposium on Spatial User Interaction*, pages 90–93, 2014.
- [170] Leigh, S.-W., Schoessler, P., Heibeck, F., Maes, P., and Ishii, H. Thaw: Tangible interaction with see-through augmentation for smartphones on computer screens. In *ACM Conference on Tangible, Embedded, and Embodied Interaction*, pages 89–96, 2015.
- [171] Mine, M., Yoganandan, A., and Coffey, D. Making vr work: Building a real-world immersive modeling application in the virtual world. In *ACM Symposium on Spatial User Interaction*, pages 80–89, 2014.
- [172] Jacob, R. J. K., Sibert, L. E., McFarlane, D. C., and Mullen, M. P., Jr. Integrality and separability of input devices. *ACM Trans. Comput.-Hum. Interact.*, 1(1):3–26, March 1994.

- [173] Coquillart, S. A control-point-based sweeping technique. *IEEE Computer Graphics and Applications*, 7(11):36–45, 1987.
- [174] Kristensson, P.-O. and Zhai, S. Shark2: A large vocabulary shorthand writing system for pen-based computers. In *ACM Symposium on User Interface Software and Technology*, pages 43–52, 2004.
- [175] Bloomenthal, J. Graphics gems. chapter Calculation of Reference Frames Along a Space Curve, pages 567–571. Academic Press Professional, Inc., 1990.
- [176] Cherry, E. and Latulipe, C. Quantifying the creativity support of digital tools through the creativity support index. *ACM Trans. Comput.-Hum. Interact.*, 21(4):21:1–21:25, 2014.
- [177] Olsen, L., Samavati, F. F., Sousa, M. C., and Jorge, J. A. Sketch-based modeling: A survey. *Comput. Graph.*, 33(1):85–103, 2009.
- [178] Shah, J. J., Smith, S. M., and Vargas-Hernandez, N. Metrics for measuring ideation effectiveness. *Design Studies*, 24(2):111 – 134, 2003.
- [179] Nelson, B. A., Wilson, J. O., Rosen, D., and Yen, J. Refined metrics for measuring ideation effectiveness. *Design Studies*, 30(6):737 – 743, 2009.
- [180] Kudrowitz, B., Te, P., and Wallace, D. The influence of sketch quality on perception of product-idea creativity. *Artif. Intell. Eng. Des. Anal. Manuf.*, 26(3):267–279, August 2012.

VITA

VITA

Vinayak was born on September 21, 1984 in Kanpur, India. He graduated with a Bachelors in Engineering in 2006 from Punjab Engineering College, Chandigarh, India, with Mechanical Engineering as his major field of study. During his undergraduate education, he conducted research on the kinematics of under-actuated legged robots. From 2006 to 2007, he worked as a research assistant at the Indian Institute of Science (IISc), Bangalore, where he developed kinematics software for the *Digital Human Modeling* project at the Center for Product Design and Manufacturing (CPDM). He continued his research at CPDM (IISc) towards a Masters of Science (Engineering) degree in 2010 with a dissertation on computational visual ergonomics.

In January 2011, he joined C-Design Lab at Purdue University in West Lafayette, Indiana, USA. As a graduate research assistant at Purdue University, he investigated spatial interactions for idea generation in early design. His areas of interest include computational geometry, human-computer interactions, geometry processing, graph theory, and digital human modeling.