more things, and project more, resulting in the production of a greater number of design variants. "You have the paper, you have the foam, you have CAD ... but when you have all of them together, this helps you to focus on creativity." If one observes collaboration within a group of people, one can see that the roles are more clearly distributed. People draw simultaneously in a model, or divide up the work to each of the different media, model and sketch. In future evaluations we will use a design task to compare established analog ways of working with designing using the CDP, with a view toward revealing the weaknesses and benefits of the system in practical use.

REFERENCES

3Dconnexion SpaceMouse Pro. http://www.3dconnexion.com/products/spacemousepro.html.

Achten, H. H., B. de Vries, and J. Jessurun. (2000). DDDOLZ: A Virtual Reality Sketch Tool for Early Design. Proceedings of CAADRIA 2000, Singapore, May 2000: 451–460.

Arnheim, R. (1971). Visual Thinking. Berkeley, CA: University of California Press.

Broll, W., MSCM. (2003). The Augmented Round Table: A New Interface to Urban Planning and Architectural Design. In Human-Computer Interaction—Interact '03, Rauterberg et al. (Hg.), 1103–04.

Butterworth, J., A. Davidson, S. Hench, and T. M. Olano. (1992). 3DM: A Three Dimensional Model Using a Head-Mounted Display. In Symposium on Interactive 3D Graphics. Proceedings of I3D '92. New York City: 135–38.

Buxton, W. (2007). Sketching User Experience. Getting the Design Right and the Right Design. San Francisco: M. Kaufmann.

Cross, N. (2008). Engineering Design Methods. Strategies for Product Design. 4th ed. (2008), reprinted. Chichester: John Wiley & Sons.

Fish, J., and S. Scrivener. (1990). Amplifying the Mind's Eye. Sketching and Visual Cognition. In Leonardo Electronic Almanac 23 (1): 117–26.

Gero, J., J. Damski, and J. Han. (1995). Emergence in CAAD Systems. Proceedings of CAD Futures. Singapore: Centre for Advanced Studies in Architecture, National University of Singapore.

Igarashi, T., S. Matsuoka, and H. Tanaka. (1999). Teddy: A Sketching Interface for 3D Freeform Design. Proceedings of the Siggraph 1999 (26): 409–16. Los Angeles: ACM.

Johnson, T. E. (1963). Sketchpad III: A Computer Program for Drawing in Three Dimensions. Proceedings AFIPS '63: 347–53.

Kurmann, D. (1995). Sculptor—A Tool for Intuitive Architectural Design. In CAAD Futures 1995. Proceedings of CAAD Futures: 323–30.

MAXON. Why This Package? http://www.maxon.net/en/products/bodypaint-3d/.

Lawson, B., and S. M. Loke. (1997). Computers. Words and Pictures. Design Studies 18: 171–84.

Purcell, A. T., and J. S. Gero. (1998). Drawings and the Design Process. Design Studies 19 (4): 389–430.

Resnick, M., B. Myers, K. Nakakoji, B. Shneiderman, R. Pausch, T. Selker, and M. Eisenberg. (2005). Design Principles for Tools to Support Creative Thinking. In Science 20(2), ed. Ben Shneiderman, 25–35.

Robertson, B. F., and D. F. Radcliffe. (2009). Impact of CAD Tools on Creative Problem Solving in Engineering Design. Computer-Aided Design 41, 136–46. Newton, MA: Butterworth-Heinemann.

Schneider, S., and F. Petzold. (2009). A Virtual Design Platform: Bridging Barriers When Designing with Computers. Proceedings of eCAADe 2009, Istanbul 27: 205–11.

Shneiderman, B. (2000). Creating Creativity: User Interfaces for Supporting Innovation. In ACM Transactions on Computer-Human Interaction 7, 114–38. New York: ACM.

Schön, D. (1983). The Reflective Practioner: How Professionals Think in Action. London: Temple Smith.

Schubert, G., E. Artinger, F. Petzold, and G. Klinker. (2011a). Bridging the Gap: A (Collaborative) Design Platform for Early Design Stages. Proceedings of the eCAADe (29), 187–93. Brussels, Ljubljana: eCAADe, Faculty of Architecture.

Schubert, G., E. Artinger, F. Petzold, and G. Klinker. (2011b). Tangible Tools for Architectural Design: Seamless Integration into the Architectural Workflow. Proceedings of ACADIA 2011 (31): 252–59. Stoughton, WI: Printing House Inc.

Schweikardt, E., and M. D. Gross. (1998). Digital Clay: Deriving Digital Models from Freehand Sketches. Proceedings of ACADIA 1998 (18): 202–11.

Sutherland, I. E. (1963). Sketchpad: A Man-Machine Graphical Communication System. Proceedings AFIPS '63, 1963. New York City.

Terry, M., and E. D. Mynatt. (2002). Recognizing Creative Needs in User Interface Design. Proceedings of the Fourth Conference on Creativity Cognition, CC 02, 38–44.

DANCING ON THE DESKTOP:

GESTURE MODELING SYSTEM TO AUGMENT DESIGN COGNITION

Chih-Pin Hsiao
School of Architecture,
Georgia Institute of Technology

Nicholas M. Davis
School of Interactive Computing,
Georgia Institute of Technology

Ellen Yi-Luen Do
School of Industrial Design,
Georgia Institute of Technology

ABSTRACT

During the design process, architects have traditionally used a range of media and tools, including sketches and physical models, to create external representations that communicate design concepts. These representations are rapid, support quick testing and advancement of ideas, and allow new patterns and ideas to emerge. However, building these models in computer-aided architectural design (CAAD) programs is cumbersome and disembodied. In this paper, we take a cognitive perspective to explore how designers distribute part of their spatial reasoning onto the materials and tools with which they work. From this cognitive theory, we have created a unique gesture-based modeling system, Dancing on the Desktop. In this prototype, two interactive displays are projected on a desktop and the adjacent wall to show the plan and perspective views of an architectural model, respectively. Visual images and text are projected on the user's hands to provide different types of feedback for the gestural interactions. A depth camera detects gestural interactions between these two displays to create an immersive gestural interaction space for model manipulation. We argue that Dancing on the Desktop helps users develop an embodied understanding of the spatial and volumetric properties of virtual objects that the current CAAD systems cannot afford. The details of the low cost, yet effective, gesture recognition technique are also described in this paper.

1 INTRODUCTION

Spatial cognition plays a key role in the early stages of architectural design. In the design process, architects reason about spatial relationships in floor plan layouts and building forms. Architects apply spatial cognition to discover design problems and define constraints (e.g., quality and quantity of spaces and distances of circulations). The problem space, constraints, and motivations are what Schön (Schön 1992) refers to as the "design world." Designers build up and use the "design world" to conduct consistent and objective evaluations of their design decisions. Schön describes design creativity as a "reflective conversation with the materials of a design situation," whereby architects make creative discoveries by exploring the emergent consequences of their design decisions and manipulations. Since many creative discoveries involve emergent spatial relationships, the cognitive process in design is not linear. Though the process may appear to occur solely inside the head of the architect, it is often distributed between mental processes and the multimodal feedback provided by the "materials of a design situation" (Schön 1992). Unfortunately, current computer-aided architectural design (CAAD) tools do not support spatial and distributed cognition in this manner. For example, interacting with a mouse only provides 2D XY coordinate information in CAD programs, which does not intuitively support moving a virtual object in a 3D environment. Furthermore, the Windows Icon Menu Pointer (WIMP) style of interaction requires users to remember and understand the commands associated with the buttons and menus.

In our prototype, Dancing on the Desktop (DoD), we introduce an immersive and natural gestural interaction space. It provides multimodal feedback to help designers develop an embodied understanding of the spatial and volumetric relationships of architectural models. By combining feedback from multiple modalities, including vision and movement, DoD augments the spatial cognition of users to facilitate emergent creative discoveries in the design process. With these modifications, we aim to fuse the computational power of CAD modeling with the flexible and embodied nature of early design sketching and prototyping.

Figure 1 depicts the use scenario for DoD in which a user employs hand gestures to create computational models. First, in Figure 1A, the designer uses a finger to draw a 2D shape on the horizontal surface. Then, in Figure 1B, the designer selects the shape and extrudes it to create a volumetric model using a pulling gesture (Figure 1C) with the hand movements in Z direction. He or she may "grab" these virtual models in the air to move them to the proper locations as shown in Figure 1D. The system projects three types of feedback on the user's hand: (1) categorical, (2) spatial, and (3) task-dependent gestural. Categorical gesture feedback communicates to users what types of gestures are valid or invalid. For instance, in Figure 1E, when a user shows an invalid gesture, the system projects a question mark. Spatial gesture feedback provides information about where in space the user's hand is relative to the virtual objects in the model. Task-based gesture feedback provides additional information about the actions, such as icons and texts that are relevant to the task the user is currently engaged in. The horizontal and vertical interactive surfaces host the reference plan and associated perspective of the current drawing.

For our DoD system, we employ a depth camera (Microsoft Kinect) to capture hand gestures with 3D motions and two projectors to project images for visual feedback. As shown in Figure 2, the first projector projects images horizontally on the desktop while the other projects vertically on the vertical surface in front of the user. These two displays form a unified interactive environment where the users use gestures to manipulate the models. In this paper, we will review related gestural prototypes and examine their shortcomings. Then, we will introduce distributed cognition and describe how it helped our system address the shortcomings of typical gestural systems. Next, we will describe the implementation details and explain each type of gestural interaction in detail. Finally, we will discuss our preliminary tests and conclusions.

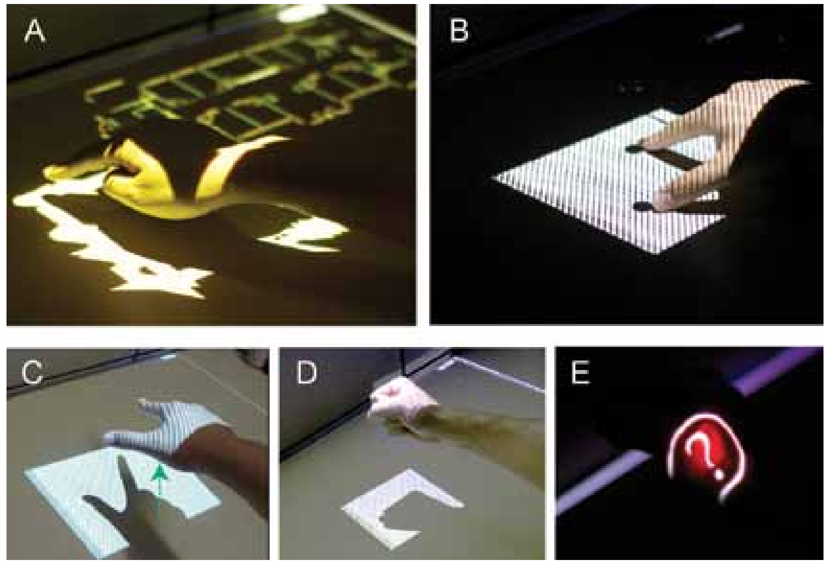

figure 1

figure 1

Using Dancing on the Desktop to build a model. The user would a) draw a shape on the horizontal surface; b) use two fingers to select the shape; c) pull the hand up to extrude the shape; and d) grab the created volumetric model and move it to a proper location; e) when a user shows an invalid gesture, the system projects a question mark.

figure 2

Dancing on the Desktop configuration. One projector is above the desktop for horizontal projection. The other projects vertical images in front of the user. The depth camera is hung on the desktop to detect the locations and gestures of both hands.

2 RELATED WORKS

In 1984, Nemeth made the first attempt to utilize hand input for a CAD system to aid designers' spatial cognition throughout the design process (Nemeth 1984). To increase user engagement with the modeling environment, Donath et al. adopted virtual reality (VR) with instrumented gloves to aid designers in creating models (Donath and Regenbrecht 1999). Their systems helped designers generate simple geometries and navigate them with a head-mounted display. Unfortunately, these early prototypes that adopted gestural interactions required heavily instrumented settings, customizations based on the particular usage scenarios. The resulting physical configurations were not comfortable for users over long periods of time.

Gesture Modeling is a project that reduced the size of the designer's workspace to that of a customized desk, which is a more ergonomic configuration (Gross and Kemp 2001). The system employs both gesture languages and VR to indicate where the user's hands are. The project employs a camera that performs image processing of the noninstrumented gloves to trigger the commands for 3D shape editing. The depth data is calculated by the size of the gloved hand in the scene.

In recent years, due to the wide availability of sensing technologies, many projects have adopted new interaction techniques to facilitate more natural ways to interact with the computers. However, few of them can be successfully applied in practical scenarios. Gustafson et al. argue that these failed prototypes employ too many categorical gestures, such as different hand gestures to trigger commands, rather than spatial ones in which users can trigger commands through utilizing the locations of interface objects (Gustafson et al. 2010). Many other gestural interaction researchers also believe we can take advantage of body motion and use visual feedback to reduce the learning curve for gestural interactions (Kratz and Rohs 2009).

Recently, Wang et al. created 6D Hands (Wang et al. 2011) for directly manipulating virtual objects in a mechanical CAD system with bare hands, a keyboard, and a mouse. However, they found that their prototype is confusing without any visual cues when interacting in the 3D space. LightSpace is

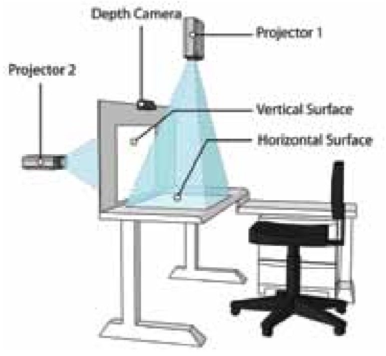

figure 2

another project that has demonstrated motion inputs with immersive computational images (Wilson and Benko 2010). This project illustrates the potentials of using hand gestures with interactive touch screens to enhance the user experiences in between the physical and the virtual worlds. They explored how projection images could help users to "feel" the virtual interface objects in the physical world.

3 COGNITIVE TASKS IN DESIGN

Our prototype, DoD, enables several types of cognitive offloading and can therefore be considered a cognitive technology. Similar to the prior projects, gestural interactions help users develop an embodied understanding of the spatial and volumetric properties of virtual objects. Unlike other projects, we introduce the multimodal feedback that projects images on the user's hands and two displays. The multimodal feedback can help the users remember the available and relevant commands as well as further enhance the spatial cognition of elements in the immersive gestural space. Each of the gestural feedback mechanisms (categorical, spatial, and task dependent) as well as the gestural interaction space itself enables the user to distribute unique cognitive tasks onto the technology. Before we introduce the multimodal feedback of DoD, we will discuss the idea of distributed cognition and explore its relevance to gestural interactions as well as the design tasks.

Distributed cognition is a cognitive science theory that argues that humans can offload cognitive processes onto artifacts and other individuals in the environment (Hutchins 1995). An everyday example of distributed cognition is writing a phone number on a piece of paper to help remember it (or asking a friend to help remember a part of it). Designers rely on this cognitive capacity when they sketch their designs on paper rather than trying to imagine the entire picture. Humans have limited attention resources and can only consciously attend to a small number of items in what is referred to as working memory (Baddeley 1992). Working memory temporarily retains information relevant to tasks that an individual is engaged in, such as symbolic or spatial representations that are used in cognitive processes (Thagard 2005). Psychologist George A. Miller famously discovered that humans could actively maintain seven plus or minus two chunks of information in working memory (Miller 1956). Although humans have a hard constraint on the capabilities of working memory, we have the unique ability to manipulate and restructure the environment in such a way as to distribute cognitive tasks onto it. Technologies that enable individuals to distribute cognitive processes are said to extend and augment human cognition and are referred to as cognitive technologies (Dror and Harnad 2008). Considering distributed cognition while designing technology and also analyzing how designed artifacts facilitate cognitive offloading is therefore a valuable goal when building technology to support cognitive tasks, such as design.

In addition to distributing cognitive tasks onto tools, Le Dantec also found that design teams utilize various embodied acts for hunting through design mistakes, exploring ideas, and solving problems (Le Dantec 2009). The team members observed by Le Dantec frequently adopted common gestures to indicate specific activities, scales, spatial properties, and the directions of angles. Viewing our system through this lens, DoD has the potential to facilitate social cognition in the design process by providing an immersive gestural interaction space that responds to gestures typically used to communicate features of designs. Rather than translating the discussion to a computer model after the meeting, DoD would dynamically adjust the model throughout the discussion.

4 MULTIMODAL FEEDBACK

4.1 Categorical Gestural Feedback

We designed four different types of multimodal feedback to offload cognitive tasks during design activities and enhance gestural interactions. The first and most common issues identified by Gustafson et al. (Gustafson et al. 2010) is that it is difficult to remember the plethora of available gestural commands in a gesture-based system. Instead of performing important design tasks, the user may waste cognitive resources trying to remember the proper gestural configuration that maps to the corresponding command. Our system eases this cognitive task in two ways. First, we project visual feedback on the user's hand to indicate the type of gesture they are engaged in and also inform the user when the system did not recognize their gestures, as in the case when the fingers are blocked from the sensor. Second, we constrain the gestural input the system recognizes for each interactive surface to those commands that are relevant to that particular surface. If the user performs irrelevant gestures, the system will give visual feedback such as the question mark projected on the user's hand (Figure 3A).

4.2 Spatial Gestural Feedback

As the user moves through areas of the gestural space that are occupied by three-dimensional virtual objects, the system provides visual feedback to indicate the presence of those virtual objects. For example, in Figure 3C, when the user's hand moves to the inside of the object, the system projects hatch marks onto the hand to signify that it has moved to the interior of a virtual object. Since an overhead projector is used in the system, the color and shape projected on the user's hand would naturally reflect the location of the building, but this does not provide any information about the height of the building. Our system helps the user understand the height of virtual objects by projecting hatch marks onto the user's hand whenever it is within the bounds of a virtual object. When the user's hand moves out of the virtual object in Z dimension, the

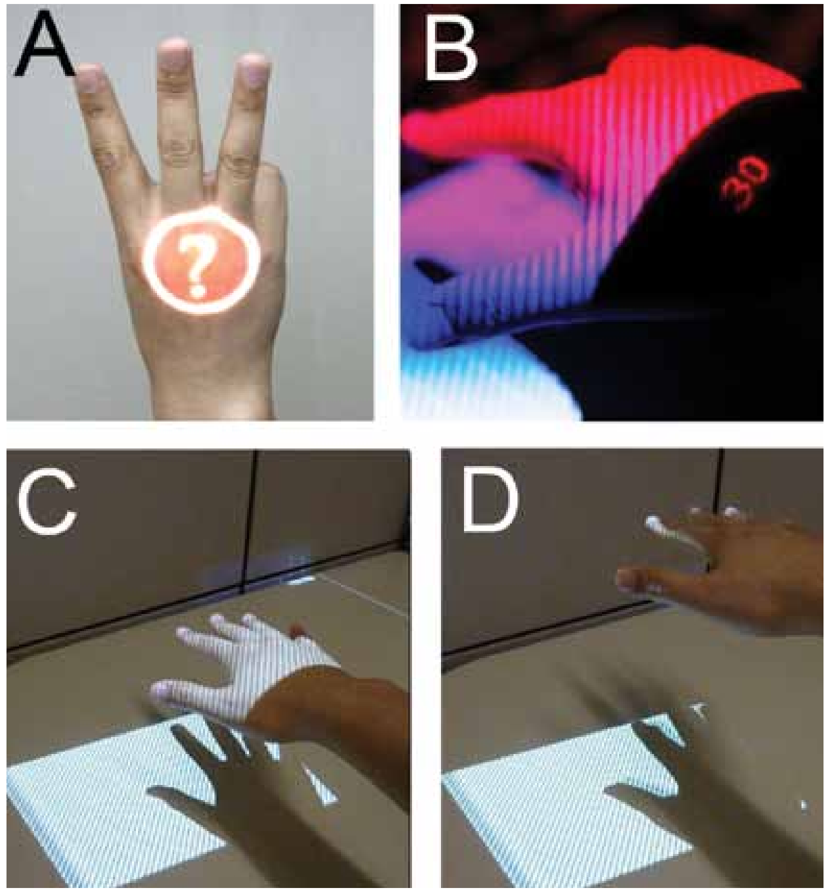

figure 3

figure 3

Different projections on the user's hand for multimodal feedback: a) categorical gestural feedback showing an invalid gesture; b) task-dependent gestural feedback projecting height of a building; c) spatial gestural feedback projecting hatch marks to inform the user that the hand is inside a virtual object; d) spatial gestural feedback masking light around the contour of the hand to show that it is above and outside of a virtual object.

figure 4

system creates a mask, and thus nothing appears on the user's hand (Figure 3D). It is then possible to raise one's hand to try to develop an embodied understanding of the height of the virtual object. Enabling this kind of embodied interaction helps the user develop a deeper understanding of the spatial and volumetric properties of the virtual objects. Varela et al. argue that a large degree of the human conceptual system is influenced through our bodily interaction with the world (Varela et al. 1999). Without this embodied interaction, the user would have to produce a mental simulation or visualization to make design decisions. However, DoD offloads and augments this task of embodied reasoning through its

4.3 Task-Dependent Gestural Feedback

Providing relevant task-based information accomplishes two things. First, it confirms the task that the user is engaged in to help avoid errors. Second, it provides relevant details that help the user make informed decisions. With these functionalities, this feature supports decision making and memory. For instance, Figure 3B shows the system projecting the height of the extruded artifact onto the hand of the user. The projected value represents the extrusion distance from the desktop surface. It informs the user how far he has extruded the current model. The extrusion value updates dynamically, based on the user's hand position, and provides real-time visual feedback to augment spatial reasoning tasks.

4.4 Dual-Display Visual Feedback

Figure 4 shows that DoD contains a dual-display configuration that projects both plan and three-dimensional views on two independent interactive surfaces. Both of these displays update dynamically and therefore provide visual feedback about how manipulations change the plan and three-dimensional properties of a model. Since the user doesn't have to mentally simulate how their decision affects the model in the third dimension, he can use those resources to explore and evaluate different designs, and therefore increase the creative potential of the designer. This is an example where DoD can be used as a creative technology in a process called distributed exploratory visualization (Davis et al. 2011). Distributed exploratory visualization describes a process whereby introducing cognitive technology into a creative task has the potential to transform cognitive processing from planning or simulating to exploration and evaluation. While interacting with DoD, users do not have to envision how their decisions are going to affect their creation. Rather, low-cost and playful embodied interactions enable the user to creatively explore and evaluate ideas in real time. The model of distributed exploratory visualization suggests that this low-cost creative exploration and evaluation can help facilitate creative discoveries.

5 IMPLEMENTATIONS

The system analyzes the image streams from a depth camera. We adopt OpenNI (OpenNI 2012) as the middleware, a natural interaction library to analyze the raw data. This software library helps the system detect hand locations and manage point cloud streams from the device. The raw depth data provides the distance from the camera to the objects in each pixel and produces grayscale images (Figure 5A). These grayscale images are used for recognizing the contours of hands and fingers to produce the meaningful gestures in our prototype system. First, the software recognizes the movements and gestures. Next, another module executes certain actions based on the recognized gesture, as well as projecting images on the user's hands for multimodal feedback.

5.1 Recognizing the Gesture

In the past, recognizing a gesture from a regular camera required heavy computer processing to filter out unnecessary pixels. Some projects also required instrumented or colored gloves. In our prototype, we use the depth frames from the depth camera to help filter out those unnecessary pixels to focus on analyzing where the fingers are. This enables bare hand interaction to aid usability.

figure 4

Dual-display visual feedback creates an immersive environment.

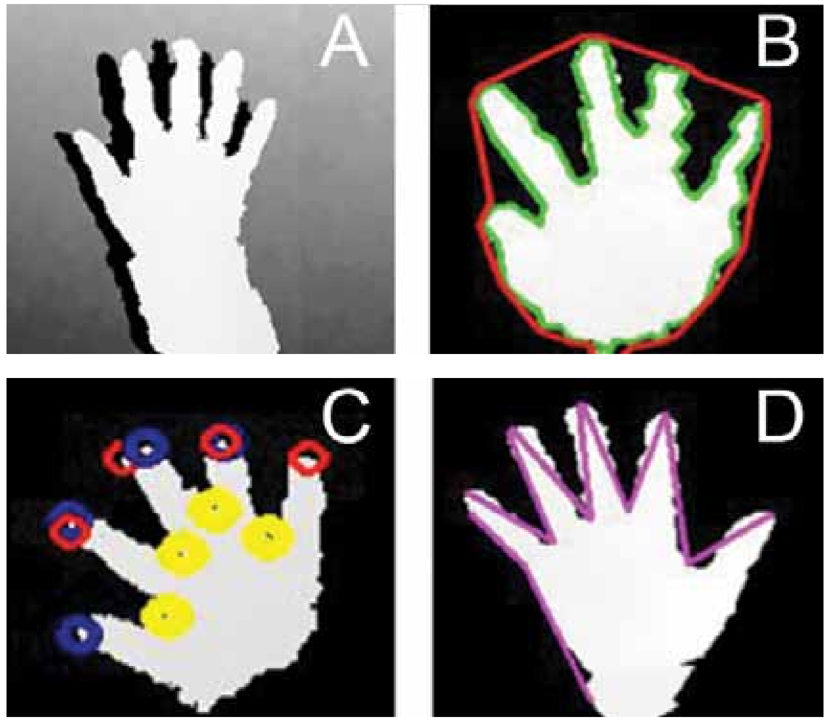

figure 5

OpenNI tries to detect the center of the hand locations once it receives the data from the depth camera. Computing the intersection points of the hand's contour and convex hull polygons yields the fingertip locations (the blue and red dots in Figure 5C). The fingertip locations are also used for querying the depth distance in the 3D scene. Thus, the vectors for each finger can be calculated by using the location of each fingertip and the finger depth point. These vectors are used to recognize the gestures. The location for the fingertips could also be the place at which the user is pointing. By using this method, we can easily recognize the gestures in more than 20 frames per second, which meets real-time interaction needs.

5.2 Gestures

We utilize the algorithm described above to detect four predefined gestures. The user's right-hand "pointing" gesture (Figure 5A) is recognized as a pointer in the system. The pointing gesture is mainly used to draw shapes on the horizontal surface. Instead of the mouse cursor in WIMP setting, the cursor is now the user's fingertips. The user's right hand "grabbing" gesture (Figure 6B) is used to select a virtual object in the immersive environment—the equivalent to hold-and-drag when using a mouse. When a user performs the gesture "releasing" (Figure 6C), the system knows that the user is placing the object onto the surface. When a user performs a "pulling" gesture (Figure 6D) on the polygon in the scenario of CAD, the system will execute the extrusion command on the polygon to create a 3D model. The combinations of these gestures form a sequence of actions that enable 3D manipulations in an immersive environment.

5.3 The Horizontal and Vertical Surfaces

The horizontal and vertical surfaces show the plan and perspective model view, respectively. They both support multitouch interaction; however, the horizontal surface specializes in drawing and

figure 5

a) The depth image; b) the contour (green) and convex hull (red) lines; c) the red/blue dots are the intersection points of the contour and convex hull lines, and the yellow dots are the depth points; d) the connected vectors of the yellow dots and the red/blue dots.

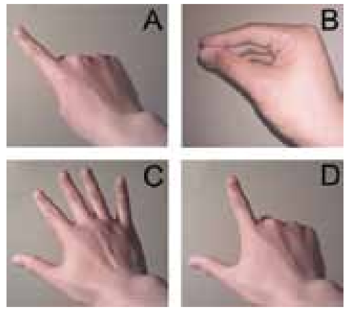

figure 6

figure 6

Basic gestures: a) pointing; b) grabbing; c) releasing; and d) pulling.

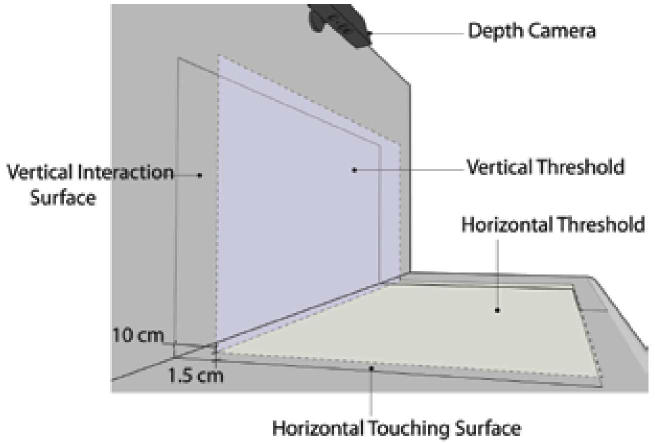

figure 7

figure 7

The horizontal and vertical interaction thresholds. The area between the thresholds is for gestural interactions.

manipulating 2D shapes. We adopt Wilson's method (Wilson 2010) to detect the user's touch points on the regular surfaces using only a depth camera. Figure 7 illustrates the threshold areas on top of each surface for the area where the user can perform gestures. Similar to 6D Hands, which supports user interactions with a mouse and keyboard besides hand movement in the air (Wang et al. 2011), this configuration also enables users to freely interact with objects right on a multitouch surface. Below we briefly describe interaction examples supported by the current configuration.

Extruding a polygon from horizontal surface:

If a user is trying to build a 3D model, he or she can draw a 2D profile path on the horizontal surface and easily pull it up for extrusion. This action begins on the horizontal plane and is continued into the gestural interaction space. When a user touches a polygon with two fingers, the system looks for the pulling gesture in the gestural interaction area. If the system detects the pulling gesture, it creates an extruded model with the height of the hand location from the surface of the desktop. If the user performs a release gesture, the system will stop extruding the shape in the model.

Grabbing virtual object in the air:

While trying to move a 3D object, such as a component of a model, the user can perform a "grabbing" gesture in the air, and then release the object to the desired location. The system highlights the virtual object when the location of the user's hands matches the location of the virtual object.

6 DISCUSSIONS AND EVALUATION

We installed DoD in a regular office environment, which demonstrates the flexibility of the system. The prototype is built based on an Intel Core i7 computer with a Microsoft Kinect depth camera. Our system runs at a sufficient rate (>20 HZ) for interactions. However, due to the inaccuracy for touch points (~5 mm), it is difficult to draw shapes in detail on the 2D horizontal surfaces. Gesture recognition was another challenge. Usually, the depth camera can recognize gestures correctly. However, if a finger is obstructed by any other object in the scene, the depth camera will not be able to recognize the finger. From our observations, this condition happens around 10–20 percent of the time during interactions.

We demonstrated the prototype and invited users to interact with it during an open house event for an audience of about 50 people. Through observations during the event, we obtained valuable suggestions and insights about how people interact with DoD. Most people found the multimodal feedback helpful. The projected images helped them locate the 3D models and manipulate them with a short learning curve. During the demo, we also found an interesting emergent interaction when grabbing virtual 3D objects. When one of the projectors is projecting from the ceiling to the desktop surface vertically, some users utilized the shadow of their hand on the desktop surface to interact with the virtual objects. This actually generates effective associations between the virtual objects and the physical locations with the projected images on the user's hand.

figure 8

figure 8

Pointing at a physical or virtual object to get information.

7 FUTURE WORKS

Our work so far presents an exploratory view of a 3D gesture-based workspace to augment design cognition. Clearly, in order to move the system toward a real-world implementation, we still need to add more modeling features and explore ways to create comprehensive geometries by using gestures. For instance, when a user tries to subtract one geometrical shape from another to make a hole, what gestures may he or she use to execute the command? We are currently implementing the straight-line extrusion for the user to make a volumetric model such as a building. In the future, we imagine that a user can draw a 3D curve line in the space freely. With a profile shape and the curve, the user could sweep a freeform shape, which some current CAAD systems cannot easily create.

We are also interested in investigating the interactions between virtual and physical objects such as projecting simulation images on physical models (Hsiao, Do, and Johnson 2011). The combination of gestures and physical models would form a flexible, powerful, and uniquely immersive experience to help designers understand and reason about spatial relationships. As shown in Figure 8, we have implemented a way of pointing to physical or virtual models, and then displaying the information associated with the object.

8 CONCLUSIONS

This paper explored the benefits of using projected images in gestural interactions. Our system, DoD, creates a gestural interaction space with two orthogonal multitouch surfaces and a depth camera. We analyzed how this immersive gestural interaction space can be used to help users develop an embodied understanding of the spatial properties of virtual objects.

Using the theory of distributed cognition, we demonstrated how certain feedback could aid design cognition. DoD provides two types of feedback. The first is gestural feedback (categorical, spatial, and task dependent) projected on the user's hands. This type of feedback helps users remember the proper gestures and understand where their hands are relative to the virtual objects in the gestural interaction space. The second source of feedback is the plan and perspective view that are simultaneously projected in real time. These displays reflect gestural manipulations of the model and help the user creatively explore and evaluate ideas in real time.

Although CAD systems are extremely useful in late stages of the design process, architects still rely heavily on hand-drawn sketches and low-fidelity prototypes in the early stages of design. Dancing on the Desktop is an attempt to fuse digital modeling with the flexible and embodied nature of early design sketching and prototyping.

ACKNOWLEDGMENTS

We thank the members of ACME Lab and colleagues at Health System Institute at Georgia Institute of Technology for giving us insightful feedback and suggestions.

REFERENCES

Baddeley, A. (1992). Working Memory. Science 255 (5044), 556–59.

Davis, N., B. Li, B. O. Neill, and M. Nitsche. (2011). Distributed Creative Cognition in Digital Filmmaking. Proceedings of the Creativity & Cognition, 207–16. Atlanta.

Donath, D., and H. Regenbrecht. (1999). Using Immersive Virtual Reality Systems for Spatial Design in Architecture, 1–10. Proceedings of AVOCAAD 99, Brussels.

Dror, I. E., and S. R. Harnad. (2008). Cognition Distributed: How Cognitive Technology Extends Our Minds, Vol. 16. John Benjamins Publishing Co.

Gross, M. D., and A. J. Kemp. (2001). Gesture Modeling. Proceedings of the Computer Aided Architectural Design Futures 2001, 33–46. Eindhoven, Netherlands.

Gustafson, S., D. Bierwirth, and P. Baudisch. (2010). Imaginary Interfaces: Spatial Interaction with Empty Hands and Without Visual Feedback. Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology, 3–12. New York, NY: ACM.

Hsiao, C.-P., E. Y.-L. Do, and B. R. Johnson. (2011). Tangible User Interface for Architectural Modeling and Analyses. Proceedings of the Tangible Embedded and Embodied Interaction, 157–62. Funchal, Portugal.

Hutchins, E. (1995). Cognition in the Wild. Cambridge, MA: MIT Press.

Kratz, S., and M. Rohs. (2009). HoverFlow: Expanding the Design Space of Around-Device Interaction. Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services, 1–8. Bonn, Germany: ACM.

Le Dantec, C. A. (2009). Situated Design: Toward an Understanding of Design through Social Creation and Cultural Cognition. Proceedings of the Seventh ACM Conference on Creativity and Cognition, 69–78. New York, NY: ACM.

Miller, G. A. (1956). The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. Psychological Review 63 (2): 81–97.

Nemeth, C. M. (1984). A Gesture Recognition System. Illinois Institute of Technology.

OpenNI. Accessed April 13, 2012. http://www.openni.org/.

Schön, D. A. (1992). Designing as Reflective Conversation with the Materials of a Design Situation. Knowledge-Based Systems 5 (1): 3–14.

Thagard, P. (2005). Mind: Introduction to Cognitive Science. Cambridge, MA: MIT Press.

Varela, F. J., E. Thompson, and E. Rosch. (1999). The Embodied Mind: Cognitive Science and Human Experience. Cambridge, MA: MIT Press.

Wang, R., S. Paris, and J. Popović. (2011). 6D hands: Markerless Hand-Tracking for Computer Aided Design. Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, 549–58. New York, NY: ACM.

Wilson, A. D. (2010). Using a Depth Camera as a Touch Sensor. ACM International Conference on Interactive Tabletops and Surfaces, 69–72. New York, NY: ACM.

Wilson, A. D., and H. Benko. (2010). Combining Multiple Depth Cameras and Projectors for Interactions On, Above and Between Surfaces. Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology, 273–82. New York, NY: ACM.

CONVENTIONS OF CONTROL:

A CATALOG OF GESTURES FOR REMOTELY INTERACTING WITH DYNAMIC ARCHITECTURAL SPACE

ABSTRACT

The intent of this project is to create a catalog of gestures for remotely controlling dynamic architectural space. This research takes an essential first step toward facilitating the field of architecture in playing a role in developing an agenda for control. The process of the project includes a sequence carried out in four stages: 1) research of gestural control; 2) creating an initial catalog of spatial architectural gestures; 3) real-world testing and evaluation; and 4) refining the spatial architectural gestures. In creating a vocabulary for controlling dynamic architectural environments, the research builds upon the current state-of-the-art of gestural control, which exists in integrated touch- and gesture-based languages of mobile and media interfaces. The next step was to outline architecturally specific dynamic situational activities as a means to explicitly understand the potential to build gestural control into systems that make up architectural space. A proposed vocabulary was then built upon the cross-referenced validity of existing intuitive gestural languages as applied to architectural situations. The proposed gestural vocabulary was then tested against user-generated gestures in the following areas: frequency of "invention," learnability, memorability, performability, efficiency, and opportunity for error. The means of testing was carried out through a test-cell environment with numerous kinetic architectural elements and a Microsoft Kinect sensor to track gestures of the test subjects. We conclude that the manipulation of physical building components and physical space itself is more suited to gestural physical manipulation by its users than to control via device, speech, cognition, or other. In the future it will be possible, if not commonplace, to embed architecture with interfaces to allow users to interact with their environments, and we believe that gestural language is the most powerful means of control because it enables real physical interactions.

Michael Fox
California State Polytechnic University, Pomona

Allyn Polancic
California State Polytechnic University, Pomona