Extending ActionSketch for New Interaction Styles: Gestural Interfaces and Interactive Environments

Gil Barros

FAU-USP: Faculty of Architecture and Urbanism, University of São Paulo
Rua do Lago, 876 - 05508.080, São Paulo, SP, Brazil
`gil.barros@formato.com.br`

Abstract. Sketching is considered a core activity in design and is a well-established practice in many areas of design. Interaction design is a new field of design that poses challenges for sketching, due to particular characteristics of the field such as being dynamic and "immaterial". To address these challenges a technique was proposed to improve the process of sketching for interaction design. It was originally proposed for WIMP interfaces and has been extended for multi-touch interfaces. In this article we verify how to apply the technique to other interaction styles, namely gestural interfaces and interaction environments. We then analyze these adaptations and extract some general rules for this kind of modification. We conclude the article with some future directions.

Keywords: Sketching, technique, gestural interface, interactive environment, ActionSketch.

1   Introduction

Sketching is considered a core activity in design [7, 11, 14] and since the seminal work of Schön [20], who describes the act of sketching as having a "conversation" with the drawing, it has been increasingly clear the importance of sketching and its dialogic nature during early stages of the design process [5, 10, 22].

Since the term "sketch" can be interpreted in several ways, in this article when we refer to sketches we use Lawson's [14] definition of "proposition drawings", those drawings that the designer creates to propose a possible outcome to a problem.

In a process where the designer almost simultaneously creates and reinterprets the representation, sketching is "not merely an act of representation of a preformulated image ... it is, more often than not, a search for such image" [11]. While the designer tries to define what a particular shape should be, sketching helps with "a back and forth swaying movement which helps translate particulars of form into generic qualities, and generic rules into specific appearances" [11].

Sketching is a well-established practice in many areas of design, such as product design and architecture [14, 17], but is still in its infancy in interaction design (IxD) [5], a field that can be defined as "the design of the subjective and qualitative aspects of everything that is both digital and interactive" [15].

Interaction design also poses particular challenges for sketching because it deals with objects that are dynamic [4, 5, 9] and which have "immaterial" qualities [4, 16], and so traditional sketching practices are sometimes insufficient [5, 8].

For example, one way to represent interactions is in textual form, on annotations that describe what happens [12]. The problem here is that either these annotations don't describe visually how the interaction is done, which leaves too much room for interpretation, or they tend to get very long and cumbersome to read, which hinders the creative process in sketching [22].

Another alternative when communicating with a colleague is to complement the drawings and to explain the interaction orally or gesturally [21]. The problem in this situation is that the interaction is not registered in any physical support and it relies only on the memory of the designers who participated in the conversation.

Considering these challenges for sketches in interaction design ActionSketch was proposed as a technique to improve the process of sketching in interaction design [2]. The technique was originally proposed for WIMP interfaces and was latter extended to encompass multi-touch interfaces [1].

In this article we analyze how the technique can be further extended to other styles of interaction and try to identify general rules for doing so.

First we present the technique and how it was extended for multi-touch. Then we give examples of how we applied it to two other styles of interaction: gesture recognition interfaces and interactive environments. We then analyze these adaptations and extract some general rules for this kind of modification. We conclude the article with some future directions.

2   ActionSketch for WIMP and Multi-touch

Based on the descriptions found in other articles we now present how the technique was originally planned [2, 3] and how it was extended for multi-touch interactions [1].

2.1   ActionSketch: Frames, Colors, Symbols and Rules

To represent the interaction happening over time the technique proposes to use frames, arranged as a storyboard. Each frame is also divided into three steps: *initial state, user actions and system actions*. This is done to better represent the interaction, in other words, the dialogue between the user and the system.

In order to indicate these stages in the same frame the technique divides the frames in layers, and indicates them using color, with the following association:

- Black: initial state;
- Green: user actions;
- Orange: system actions.

Symbols are defined to represent screen elements, the user actions and responses from the system. Each of the three stages has a specific set of symbols.

For the initial state the technique proposes to draw a wireframe by hand and adds four symbols presented in Figure 1 to improve consistency.
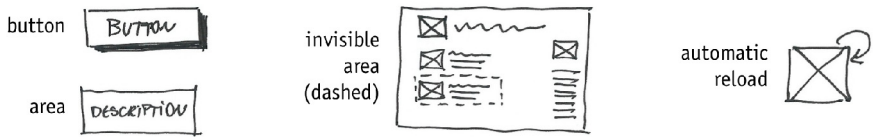

Fig. 1. Four symbols proposed for the initial state [3]

To represent the user actions 12 symbols are proposed, which cover basic actions that the user can do with a pointing device (typically a mouse) and a keyboard, presented in Figure 2.
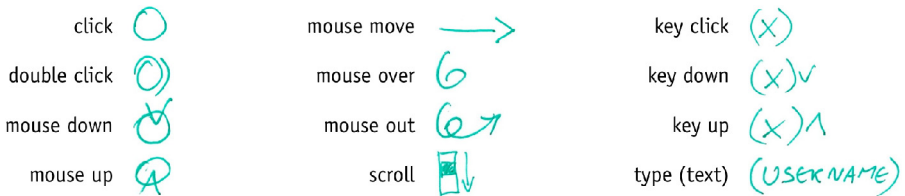

Fig. 2. 12 symbols for the user actions [3]

And to represent actions from the system eight symbols are proposed, presented in Figure 3.
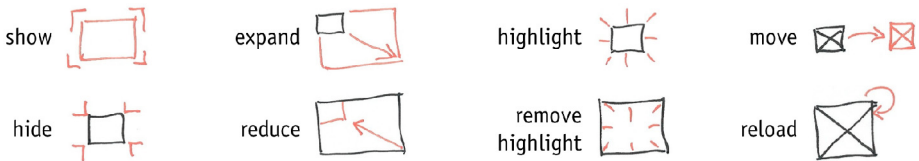

Fig. 3. Eight symbols for the system actions [3]

Rules of use are the fourth and last part of the technique. These three rules are meant to help applying the technique, and can be described as:

1. Draw only what changes between frames.
2. On each storyboard stay focused on just one idea
3. The goal is to express an idea, employ the technique when useful and modify it as needed.

To demonstrate how the technique is applied in practice an example is provided, seen on Figure 4.
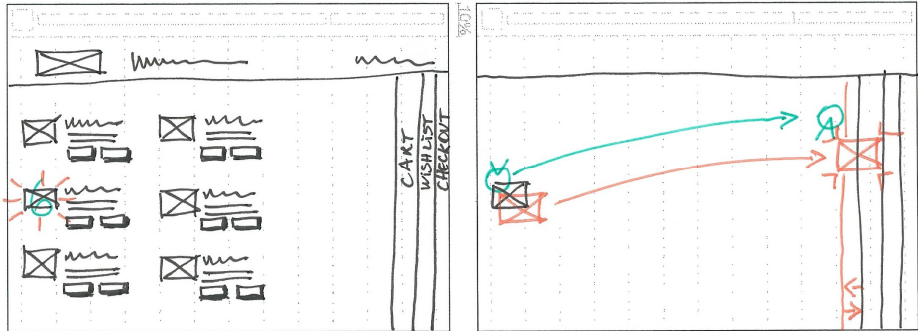
Fig. 4. Using the technique to represent how to add an item through a drag and drop mechanism [3]

- Frame one: the user puts the mouse over the icon and the system highlights it.
- Frame two: the user grabs the icon, drags it to the tab of the panel and releases it. The system carries a shadow of the icon with the cursor and when the user releases it the shadow disappears and the tab moves to indicate that the item was added.

2.2   Extending the Technique for Multi-touch Interaction

The only part of the technique that needed to be extended for multi-touch interactions was the symbols for user actions, shown on Figure 5.
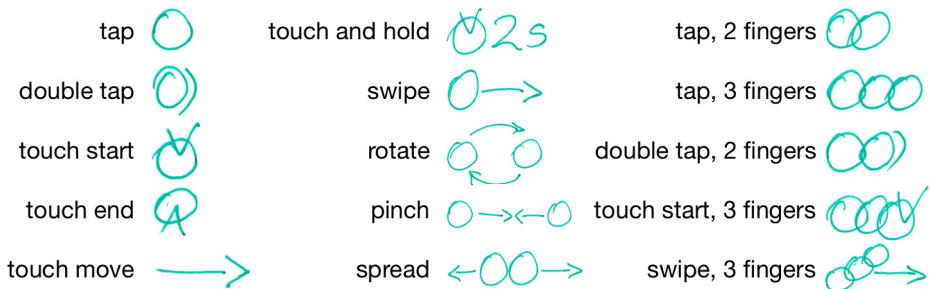

Fig. 5. Symbols for user actions on the multi-touch version of the technique [1]

The five symbols on the leftmost column already existed in the WIMP version, but their names were changed to match the terminology used in multi-touch. This means that the same symbols can be used for both types of interfaces and equivalence between the two styles of interaction was set.

The five symbols on the middle column are new actions, which exist in multi-touch but not on WIMP. And the five symbols on the rightmost column are multi-touch versions of the basic symbols, they represent similar actions but done with more that one touch.

From these examples we see that the symbols were extended so as to form a consistent grammar of meaning. If a click is the same as a tap, they should have the

same symbol. Also, if a circle represents a tap, a tap with two fingers should be represented by two circles. To achieve this consistency between symbols there are a few basic shapes (circle, arrow, etc.) and these shapes are then combined to create new meanings, extending the original ones.

It is also interesting to note that the only changes needed for this extension were in the symbols for user actions. We believe that the main reason for this is because the basic difference between WIMP and multi-touch interfaces is exactly on the input mechanisms: mouse and keyboard (for WIMP) and touches on the screen (for multi-touch). In terms of the output, both interfaces rely on interactions that happen on a screen and so the representation of that "side" of the interface (the initial state and system actions) is the same in both situations.

Considering these factors, we now present two of the tests we made when trying to extend the technique no new interaction styles.

3 Testing the Technique with New Interaction Styles

To test how well ActionSketch could be further extended we proposed to test it with two other styles of interaction: gesture recognition interfaces and interactive environments.

We chose these two interaction styles because they have important differences when compared to the WIMP paradigm. In the case of gesture recognition part of the interaction happens "outside" the screen and in the case of interactive environments there might not even exist a "screen" and the whole interaction can happen in space.

3.1 Gesture Recognition Interfaces

Since there is still some variation on the definitions of gesture recognition interfaces we find it useful to define it a bit better. Gestural interfaces can be roughly divided into touchscreen and free-form [19].

In this article, when we refer to gesture recognition interfaces we mean free-form gestural interfaces, meaning an interface that doesn't require the user to touch or handle any device in particular, and where the body is the "input device".

This kind of interface has become increasingly common, especially after the release of the Kinect input device for the Xbox gaming consoles. Several gestural interfaces exist, both based on Kinect[1] or on other devices[2], but for this exercise we chose the implementation of Gesture Control found on Samsung Smart TVs[3].

---

[1] One example is DepthJS, which is an Open Source JavaScript framework that allows users to control web pages from Kinect (`http://depthjs.media.mit.edu`).

[2] Two examples of other input devices are the Myo Armband (`http://thalmic.com/en/myo/`) and the Leap Motion Controller (`http://leapmotion.com/product/`).

[3] This implementation is part of a new group of interface styles available on new models from this manufacturer, called "Smart Interaction". It includes voice recognition, gestures control, face recognition and a touchpad (`http://www.samsung.com/us/2013-smart-tv/#smart-tv-5`).

We found this to be a good example for the exercise because the actions are simple and well known (change channels, change volume, turn device on/off, navigate through menus, etc.), which makes it easier to explain the interactions and to test the sketching technique. Also, the interface is well documented both by the manufacturer as well as by third party reviews and is an interface that we have access to, which allowed direct tests.

On this interface we sketched several of the possible interactions, and here we demonstrate how to change channels. On Figure 6 we see how the technique can be used to represent the equivalent of a click in this kind of interface.
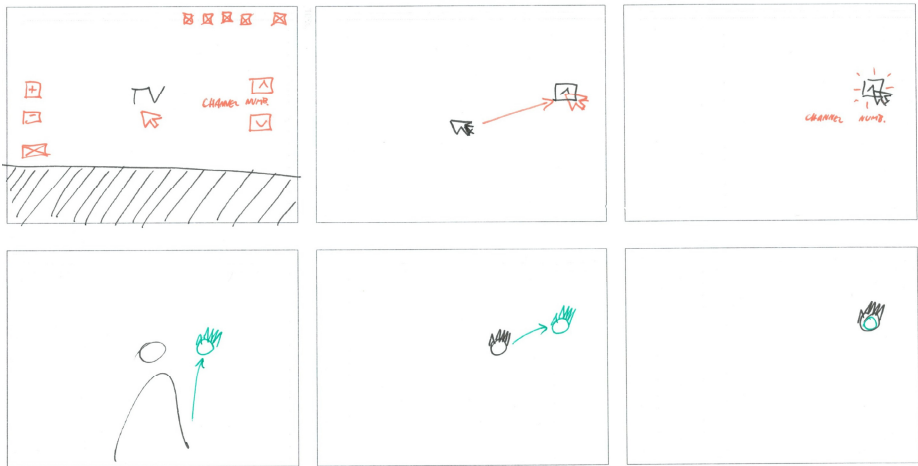

Fig. 6. Using the technique to represent how to change channels using a click

- Frame one, bottom: the user raises his hand and holds it open in front of the device;
- Frame one, top: the system shows the cursor and buttons on the screen.
- Frame two, bottom: the user moves his hand to the right;
- Frame two, top: the system moves the cursor and places it on top of the "channel up" button.
- Frame three, bottom: the user closes and opens his hand;
- Frame three, top: the system highlights the button and changes the channel.

Even with this simple example we can see three important differences on the technique: the use of two frames instead of one, the presence of new symbols and how the template provided had to be adapted. We describe these differences in further detail.

When representing this interface we found the need to represent not only what was on the screen, but also what was the gesture from the user that generated a response. To do so we decided to use pairs of frames, one to represent what was happening in space (the user movements) and another to depict the interface on the screen.

At first we used the top frames for user movements and the bottom frames for the screen, because that usually corresponds to how the interaction happens over time: first the user does something in space and then the system responds on the screen.

However, with time it became more common to think about the drawings as if we were seeing the user "from behind" and so it seemed more logical to draw the screen on top and what was happening in space bellow.

The top frame was "further away" (on the Z axis) and the space bellow it was "closer" to the designer, similarly to what happens in an isometric projection of an object.

The second difference was that we needed to create symbols for objects in space. An inverted parabolic shape with a circle on top (frame one, bottom) was used to represent the user (seen from behind) and a circle with five triangles was proposed as a representation of the users hand. Also, the movement of the user closing and opening his hand, which is equivalent to a click, was the combination of the position of the users hand with a circle, which is the symbol for a click.

Finally we realized that the templates provided with the WIMP version of the technique had a few shortcomings. First, the screen proportions were not adequate for a TV set, which is a lot wider (or shorter) than a normal desktop monitor. Also, since we were using pairs of frames there should be an indicator of that on the template. Finally we found that the size of the rectangles used to represent the screen (top) and the user space (bottom) doesn't have to be the same. This indicates that while the storyboard structure was useful, the template should be modified.

3.2   Interactive Environments

Interactive environments is also a field with several definitions, so in this article the term refers to spaces that allow direct interaction, implemented through digital technology [6].

In particular we are interested in spaces where the dissemination of sensors and actuators allows it to respond to people's usual behavior, that is, people can interfere in the space simply because they are present, talk to each other or move inside it.

For this exercise we chose a space described by Carneiro et. al [6], which is the office space for a digital agency. The space has a pattern of 25 triangles on one of its walls that are visible only when lit, which is used as the "output" of the system.

As "input" the space has six ultrasonic sensors, a microphone and a magnetic switch at the door. On Figure 7 we have the pattern of triangles (left) and a schematic blueprint of the space with the position of the sensors (right). The triangles are placed on the wall to the right of the entrance door.
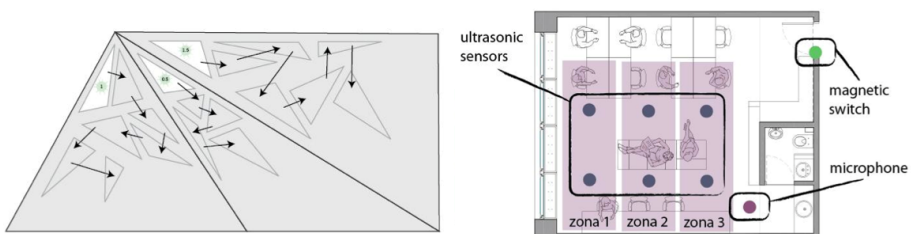

Fig. 7. The output (left) and input (right) used the interactive environment [6]

For this space we demonstrate here how the system would respond to a person walking inside it. On figure 8 we see how the technique can be used to represent this situation in this interactive environment.
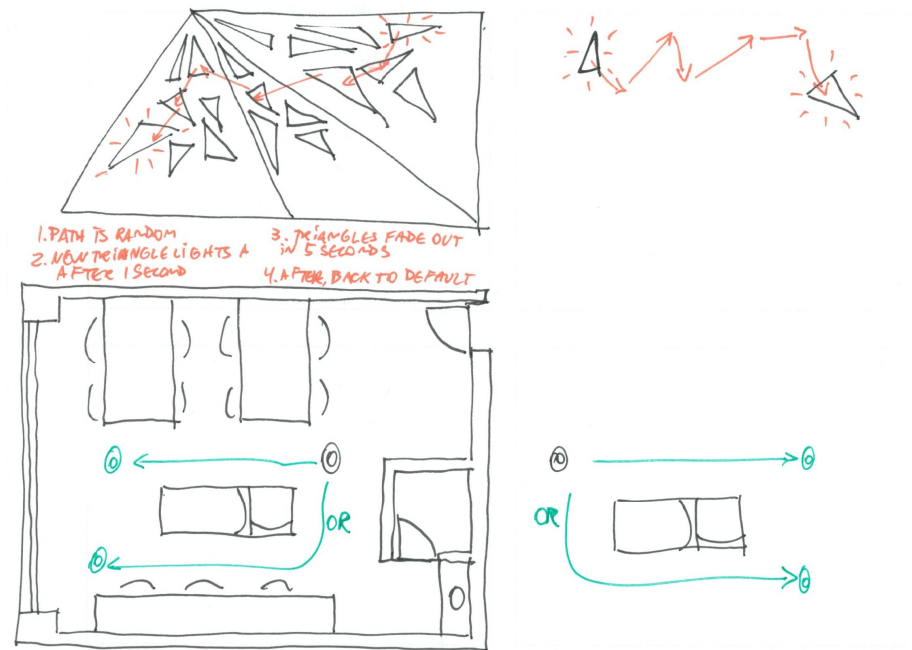

Fig. 8. Using the technique to represent how the environment responds to a person walking inside it

- Bottom left: a person walks towards the windows (from right to left);
- Top left: the triangles flash on a random sequence, from right to left.
- Bottom right: a person walks away from the windows (left to right);
- Top right: the triangles flash on a random sequence, from left to right.

Again, on a very simple example we felt the need for some important modifications on the technique. The three modifications made for gestural interfaces were also important here: the use of two frames instead of one, new symbols and the need for adaptations on the template provided. Besides this, we also found a greater need for using side notes to describe the interaction. Now we describe these differences in further detail.

Even more strongly than on the gestural interface, we felt the need to use separate drawings to represent what was happening in the space (the input) and the response from the system (the output). For the space we used a sketch of the floor plan and for the panel with the triangles it was used an elevation (a side-view).

We can see that these are modes of representation typically used in architecture. This seems quite logical, since we were representing architectural spaces, and indicates a way in which ActionSketch can be combined with other representation techniques.

Again we found it more interesting to put the output above the input on the drawings, but for a different reason. In this case we found it more interesting to represent the panel of triangles next to the wall where it is located, as if we were looking from "inside" the space. This is a common way to combine an elevation with a floor plan in architecture.

Also in this example we created new symbols for objects in space. In this example we can see the symbol for a person (a circle inside an ellipse), as seen from above. To represent common objects (walls, doors, windows, tables, etc.) we used representations already applied in architectural sketches.

In terms of the templates, we found that not only they weren't useful, but also they were hindering the process, and used blank paper instead. Since the space was quite simple it wasn't difficult to draw it quite accurately, but for other situations tracing paper might be more adequate.

Finally, we found that the need to use more side notes than in other interaction styles. As seen between the two drawings on the left side, there are some notes explaining the order and the timing for the triangles to light up and fade out. We found it very hard to describe these details on the drawings.

4   Specific and Generic Aspects of the Technique

Comparing the WIMP and multi-touch versions of the technique with the exercises created for the gestural interface and the interactive environment, it was possible to find some interesting results in terms of: the creation of new symbols; the use of templates; the structure of frames and colors and the use of side notes. We now present these in greater detail.

In terms of the symbols used, it was clear that this is the aspect of the technique that changed the most between interaction styles. For each new interface the symbols had to be reviewed.

In some situations there are equivalencies of actions between the interaction styles. For example, a click on the WIMP style was equivalent to a tap on multi-touch or the movement of closing and opening the hand for gestural interface. Since these actions are equivalent, they should use the same symbol, the circle.

On other situations there are new elements, which calls for new symbols to represent them. For example the presence of multiple touch points in multi-touch, the physical body in gestural interfaces and the movement of a person through space in interactive environments. In this case new symbols were created following two scenarios.

In the first scenario they recombine the basic shapes that were used for the elementary symbols (circle, arrow, etc.) in order to extend them. For example, in the case of new actions such as rotate and "pinch" used in multi-touch.

In the second scenario, the symbols are based on other representation codes, which are used together with the symbols of ActionSketch. For example, in the case of the interactive environment, the objects and the person seen from above used symbols "borrowed" from architectural sketches.

Considering that the technique relies on symbols to communicate efficiently, we consider it essential that in both situations, either for establishing equivalencies or when creating new symbols, there should be great care to preserve the semantic quality of the symbols and to avoid confusion with symbols brought from other representation techniques.

We also found that the templates used were very strongly related to the interfaces being represented. This means that the templates created for one kind of interface might not be useful when used for sketching another interface. Even worse, sometimes they might hinder the process.

Even further, in some cases the templates might not be an interesting alternative. For example, for interactive environments the use of tracing paper might be a better alternative, since each floor plan (and each configuration of a floor plan) would be a different "template" for the interaction to take place.

From these results we believe that the symbols and the templates are strongly tied to the interface being represented, and that each new kind of interface might require it's own set of symbols and templates.

In contrast, we found that the use of frames, and the use of colors to represent the three steps in each frame, was applied to all styles of interface with little or no modification.

When we separated the representation of one frame in two drawings, it was mostly a question of how to represent actions that occur at the same time, but in different locations (physical space and the screen). The concept of representing one instant of the interaction in one frame was still valid. Also, it was also applicable that each instant can be divided in three steps: initial state, user actions and system actions.

Contrary to the symbols and templates, we consider that this temporal structure, frames of time divided in three steps, as a generic aspect of the technique, which seems to be applicable for any kind of interface being sketched.

Finally, the use of side notes is related to how much of the interaction is already implicit as a common practice. Even when sketching the gestural interface, the interface was still using common interface elements, like icons and buttons, which needed little explanation.

In contrast, when doing sketches of the interactive environment, there were no predefined rules to dictate how the triangles would flash (for how long, how they would turn of, etc.). Since we considered these details important in this case, we had to describe them with side notes.

From this result we believe that the use of side notes tends to increase when the interaction is different from the default behavior, either because the designer wants a different one, or because there is simply no default behavior for that case.

5   Conclusions

In this article we presented some general rules that show how ActionSketch can be applied to different styles of interaction. We explained the technique and demonstrated how it was used to sketch interactions for WIMP, multi-touch, gestural and environmental interfaces.

From these variations we extracted general rules, which can be summarized as:

- Symbols are tightly connected to the interface being represented, and it is expected that for each new interface a new set of symbols should be prepared.
- Templates are also tightly connected to the interface. As with the symbols, each new interface might require new templates.
- In contrast, the temporal structure proposed by the technique seems adequate for all the interfaces tested. It appears that the use of frames to describe consecutive moments of the interaction, and the subdivision of these moments in three basic steps (initial state, user actions and system actions), might be applicable to many styles of interfaces.
- The use of side notes seems to be more evident when the interaction being described is different from the default behavior.

Based on these rules we feel encouraged to further extend the technique and to test it with other styles of interaction, such as audio or haptic interfaces, as it is the case with the Smart Audio City Guide project [18].

We also feel that with interactive environments new forms of interaction will become more important. One example are the implicit interactions, those that occur "without the explicit behest or awareness of the user" [13]. We believe that these different forms of interaction will also require new adaptations on the technique.

References

1. Barros, G., Velloso, L., Carli, L.: A Technique to Improve Freehand Sketches of Multi-touch Interactions. In: Anacleto, J.C., Clua, E.W.G., da Silva, F.S.C., Fels, S., Yang, H.S. (eds.) ICEC 2013. LNCS, vol. 8215, pp. 151–155. Springer, Heidelberg (2013)
2. Barros, G., Carneiro, G.: A technique to improve sketches of rich interactions. In: Proceedings of the 12th Brazilian Symposium on Human Factors in Computing Systems, pp. 22–31. Brazilian Computer Society, Manaus (2013)
3. Barros, G., Velloso, L.: How the communication between designers was affected by ActionSketch, a technique to improve sketches in interaction design. InfoDesign - Rev. Bras. Des. Informação Braz. J. Inf. Des. 10(01), 1–17 (2013)
4. Budgen, D.: Design models' from software design methods. Des. Stud. 16(3), 293–325 (1995)
5. Buxton, B.: Sketching User Experiences: Getting the Design Right and the Right Design. Morgan Kaufmann, San Francisco (2007)
6. Carneiro, G., et al.: Collaborative design of behaviors for interactive environments. V!RUS 6, 1 (2011)
7. Cross, N.: The nature and nurture of design ability. Des. Stud. 11(3), 127–140 (1990)
8. Dubberly, H.: On Modeling: Models of models. Interactions 16(3), 54–60 (2009)
9. Eikenes, J., Morrison, A.: Navimation: Exploring time, space & motion in the design of screen-based interfaces. Int. J. Des. 4(1), 1–16 (2010)

10. Goldschmidt, G.: Design representation: Private process, public image. In: Goldschmidt, G., Porter, W.L. (eds.) Design Representation, pp. 203–217. Springer, London (2004)
11. Goldschmidt, G.: The dialectics of sketching. Creat. Res. J. 4(2), 123–143 (1991)
12. Greenberg, S., et al.: Sketching user experiences: the workbook. Morgan Kaufmann, San Francisco (2011)
13. Ju, W., Leifer, L.: The Design of Implicit Interactions: Making Interactive Systems Less Obnoxious. Des. Issues 24(3), 72–84 (2008)
14. Lawson, B.: What designers know. Architectural Press, Oxford (2004)
15. Moggridge, B.: Designing Interactions. MIT Press, Cambridge (2007)
16. Ozenc, F.K., et al.: How to support designers in getting hold of the immaterial material of software. In: Proceedings of the SIGCHI Conference, pp. 2513–2522. ACM, New York (2010)
17. Pipes, A.: Drawing for designers. Laurence King Publishers (2007)
18. Rozestraten, A.S., et al.: Smart Audio City Guide: um sistema colaborativo para apoio ao deslocamento urbano de pessoas com deficiência visual. In: Proceedings of the SBSC Conference, pp. 175–178. Brazilian Computer Society, Porto Alegre (2013)
19. Saffer, D.: Designing gestural interfaces. O'Reilly, Beijing (2009)
20. Schön, D.: The Reflective Practitioner: How Professionals Think in Action. Temple Smith, London (1983)
21. Tholander, J., et al.: Where all the interaction is: sketching in interaction design as an embodied practice. In: Proceedings of the DIS Conference, pp. 445–454. ACM, New York (2008)
22. Verstijnen, I., et al.: Sketching and creative discovery. Des. Stud. 19(4), 519–546 (1998)