

Design of a Perceptual-based Object Group Selection Technique

Hoda Dehmeshki*

Wolfgang Stuerzlinger†

York University

ABSTRACT

Selecting groups of objects is a common task in graphical user interfaces. Current selection techniques such as lasso and rectangle selection become time-consuming and error-prone in dense configurations or when the area covered by targets is large or hard to reach. This paper presents a new pen-based interaction technique that allows users to efficiently select perceptual groups formed by the Gestalt principle of good continuity.

1 INTRODUCTION

Selecting groups of objects is a common task in graphical user interfaces and is required for many standard operations. Current selection techniques such as lasso and rectangle selection become time-consuming and error-prone in dense configurations or when the area covered by targets is large or hard to reach. Perceptual-based selection techniques considerably reduce selection time when targets form perceptual groups, as predicted by Gestalt principles of proximity and good continuity. However, they use heuristic and not-validated grouping algorithms. Also, they do not allow editing of a selection or selecting of groups with random configurations. Dehmeshki and Stuerzlinger developed a perceptual-based object group selection technique for mouse-based user interfaces [1]. In their system double-clicking on an object that is part of multiple (curvi-)linear groups selects all the groups. To deselect an undesired group, the user alt-clicks on its first non-desired object. Three key elements distinguish that system from the present work. First, clicks can specify only the location of a group but not the direction in which a group of objects extends. This makes selection less efficient when objects belong to multiple groups. Second, it provides no support for selecting non-perceptual groups. Finally, their system relies heavily on multiple-clicks, which is not appropriate for pen-based systems. This problem is shared by other techniques that use multi-clicking to cycle through different perceptual interpretations [3].

2 PERSEL

This paper introduces PerSel, a new pen-based object group selection technique, which addresses the mentioned problems. PerSel consists of two components: The first component detects good continuation groups based on a neighborhood graph. The second provides a set of pen-based interaction techniques that use the detected groups to facilitate path-based selection.

2.1 Detecting Good Continuation Groups

The system first constructs a neighborhood graph. When the user performs a straight flick gesture starting from inside an object, the system examines all edges in the neighborhood graph that are connected to this object and picks the one which has the closest direction and distance to the gesture. The object and the edge are called the anchor object and the anchor edge, respectively.

PerSel is based on an implementation of Feldman’s model [2] for linear groups which models paths as groupings of four objects combined with a sliding window paradigm. Given an anchor object 1

and anchor edge e , the algorithm finds all paths of length four starting from 1 and along edge e , see also figure 1. We call these paths primary paths. Then, for each primary path, the method computes a linearity coefficient (LC) that indicates how strongly the four nodes are perceived as a line:

$$LC = \exp\left(-\frac{(a_1^2 + a_2^2 - 2ra_1a_2)}{2s^2(1-r^2)}\right),$$

where a_1 and a_2 are the angles between lines connecting the center of objects, and r and s are experimentally determined constants [2].

Primary paths with a LC smaller than a threshold are discarded as they are unlikely to be perceptually salient. If at least one primary path remains, the algorithm continues as follows: for each path it searches its potential continuations by identifying all neighbors of the last object in the path. Then, the four-object window is shifted to include each neighbor and the LC is computed in turn. If the new LC is smaller than a threshold, that completion is ignored, otherwise, the extension of the path with this neighbor is added to a stack. If all of the neighbors of the last node are unacceptable, the original path is kept; otherwise, the original path is discarded since at least one good extension has been found. The algorithm continues until all paths inside the stack have been processed.

Figure 1 illustrates this. In Fig. 1-a, the anchor node ‘1’ and anchor edge ‘ e ’ are represented by thick borders. All four-vertex windows with small LC are visualized with ovals, while the ones with large LC are shown in rectangles. There are three primary paths, see Fig. 1-b, but only the one inside the rectangle is considered for further extension. After two more iterations, see Fig. 1-c and d, only the straight path is returned as being perceptually plausible, see Fig. 1-e.

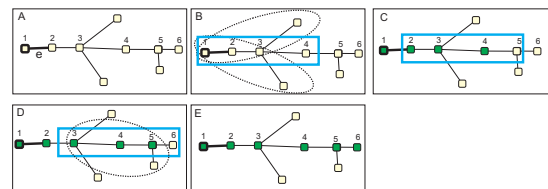


Figure 1: Illustration of Good continuity grouping algorithm. (a) shows the anchor object 1 and edge e . (b) the rectangle and ovals visualize primary paths (c) the path inside the rectangle is extended. (d) shows new potential primary paths. the one inside the rectangle is extended. (e) objects 1-6 are grouped.

For line gestures that cross an object, the gesture is first decomposed into two half-gestures, using the closest point of the gesture relative to the center of the object. Then, the linear groups corresponding to each half is found using the above methods, and then the two groups are merged.

Curvilinearity group detection: Similar to the linear case, when the user performs an arc gesture starting from inside an object, the system examines the edges in the neighborhood graph that are connected to this object. The edge closest to the gesture is picked, as defined by the sum of distances between gesture points and the edge. The rest of algorithm works similar to the linear case,

*e-mail: hoda@cs.yorku.ca

†e-mail: wolfgang@cs.yorku.ca

except that: (1) we use the curvilinearity coefficient that adapts the formula for LC by using the deviation from the average angle instead of the angles themselves, and (2) in the initial phase we only consider primary paths that turn in the same direction as the gesture.

2.2 Gestural Interaction

In this section we explain the gestural interaction techniques available in PerSel. As common in pen-based systems, tapping on a single object selects it. Also, PerSel cancels all selections whenever the user taps the pen on the background. This affords a simple and fast way to cancel erroneous selections.

Path Selection: Performing a straight gesture across an object selects the Good Continuation group aligned with the gesture direction. Similarly, an arc gesture across an object selects the curvilinear group that has a similar direction as the gesture. In both cases, the selected group is visualized by links connecting the successive objects, see Fig. 2.

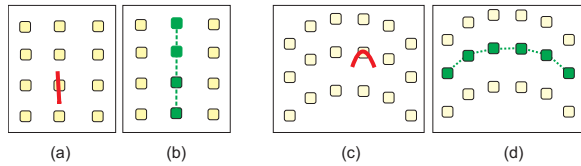


Figure 2: Good continuity group selection. Performing a line (a) or an arc (c) gesture, selects the corresponding linear (b) or curvilinear (d) group and visualizes it by links.

Partial Path Selection: There are two alternatives for partial selection of paths. The first way is to select the complete path and then cut undesired part(s) by drawing a flick gesture across one (or two) visualized links, see Fig. 3. The second alternative is to initiate the selection by a flick gesture from *inside* an object, see Fig. 4.

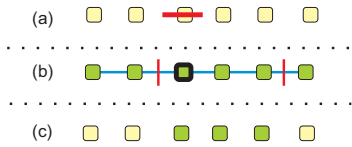


Figure 3: Partial group selection. (a) A flick gesture across an object selects the whole group. (b) Two flick gestures deselect all objects beyond these “cut” gestures.

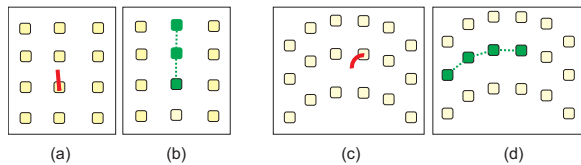


Figure 4: Partial group selection. In (a) respectively (c) a line and an arc gesture start from inside an object. Only objects that are on the same side of gesture are selected, as in (b) and (d).

Resolving Non-Perceptual Groups If a gesture corresponds to multiple potential curvilinear groups, all of them are selected. The user can then disambiguate the section by deselecting the non-desired groups. This is similar to the partial selection technique, in that the paradigm of “cutting” links is used to separate the non-desired objects from the targets, see Fig. 5.

Selecting Paths With Multiple Segments: More complex paths often consist of connected Good Continuity groups (segments). To

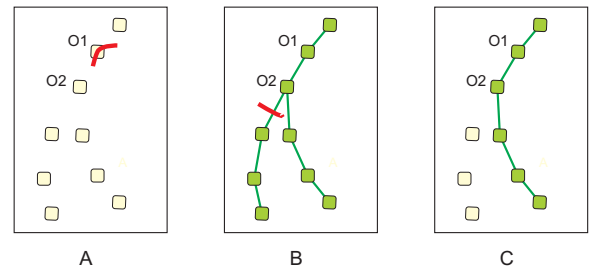


Figure 5: Resolving ambiguity. (a) An arc gesture on object O1 selects both curvilinear groups. (b) A “cutting” gesture disambiguates which objects to select. (c) Only desired objects remain selected.

enable selection of such paths, we introduce a new path editing feature. Assume that a path is already selected. If the user draws a gesture across an *already selected node* (called an anchor), a supplementary anchor is created. Then the selected path is modified by (1) automatically deselecting objects on the path beyond the new anchor, and (2) adding the new (curvi-)linear group corresponding to the new gesture and anchor to the selection, see Fig. 6.

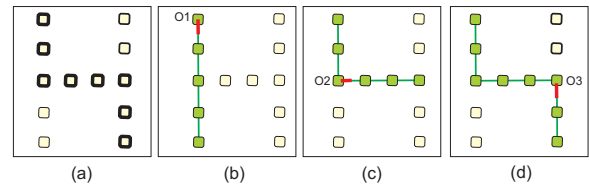


Figure 6: (a) Target objects have thick borders. (b) A line gesture over O1 selects the corresponding group. (c) A gesture from O2 guides the selection. (d) A gesture from O3 adds the remainder of the desired objects.

3 CONCLUSION AND FUTURE WORK

This paper presented PerSel, a new gesture-based selection technique that is based on the Gestalt principle of Good Continuation. Performing a flick gesture crossing an object selects the (curvi-)linear group(s) that the object belongs to and is aligned with the gesture direction. PerSel also provides interaction techniques that allow users to perform partial group selection and selecting groups with arbitrary configurations. As future work we will include Gestalt principle of similarity and extend PerSel to deal with objects with different visual features such as shape and size.

REFERENCES

- [1] H. Dehmeshki and W. Stuerzlinger. Intelligent mouse-based object group selection. In *Smart Graphics*, 2008.
- [2] J. Feldman. Curvilinearity, covariance, and regularity in perceptual groups. *Vision Research*, 37(20):2835–2848, 97.
- [3] E. Saund and T. Moran. A perceptually-supported sketch editor. In *Proceedings of the ACM Symposium on User Interface Software and Technology—UIST’94*, pages 175–184, New York, 1994. ACM.