Intelligent Mouse-Based Object Group Selection

Hoda Dehmeshki* and Wolfgang Stuerzlinger**

Department of Computer Science and Engineering
York University, Toronto, Canada

Abstract. Modern graphical user interfaces support direct manipulation of objects and object groups. Current object group selection techniques such as lasso and rectangle selection can be time-consuming and error-prone. This paper presents a new approach to group selection that exploits the way human perception naturally groups objects, also known as Gestalt grouping. Based on known results from perception research, we present a novel method to group objects via models of the Gestalt principles of proximity and (curvi-)linearity. Then, we introduce several new mouse-based selection techniques that exploit these Gestalt groups. The results of a user study show that our new technique outperforms lasso and rectangle selection for object groups with an implicit structure, such as (curvi-)linear arrangements or clusters.

1 Introduction

Object group selection is an integral part of most graphical user interfaces. Most systems implement rectangle and/or lasso selection as well as shift-clicking. For the first two techniques the user drags a rectangle or a loop around the target items. Auto-complete lasso speeds up selection by connecting the end points of the loop automatically [9]. Shift-clicking involves clicking on each object in turn with the shift-key held down.

Although quite simple and powerful, rectangle and lasso selection are time consuming when the mouse-movement distance is large, e.g. when selecting large groups of objects or on large displays. Rectangle selection requires only traversal of the diagonal of the region and hence is often faster than lasso. However, it works only well for horizontally and vertically aligned arrangements. Shift-clicking is relatively time consuming for groups of targets with more three objects, but is the only alternative that can deal with randomly scattered targets.

In this paper, we present a new approach for group selection that addresses the shortcomings of current selection techniques for spatially contiguous target groups. Our approach is based on the way human perception naturally groups objects, also known as Gestalt grouping [6]. Gestalt grouping is normally explained via a set of principles. Proximity and good continuation are among the most important principles in the context of graphical user interfaces. Proximity states that *"being all other factors equal, the closer two elements are to each*

* e-mail: hoda@cse.yorku.ca
** www.cse.yorku.ca/~wolfgang

other, the more likely they are to be perceived as belonging to the same form". Good continuation states that *"co-linear or nearly co-linear visual items tend to be grouped".*

2 Related Work

A substantial body of research in human perception has focused on measuring the strength of proximity and good continuity in dot patterns. Kubovy *et al.* [7] and Oeffelen *et al.* [13] model proximity as decreasing exponential functions of relative distances between dots. Feldman [3, 4] introduced a model that describes the strength of collinearity among three or four dots as a function of inter-dot angles (angles between lines connecting successive dots).

Contour grouping has been studied extensively in the field of computer vision. Most relevant to the present discussion is work, which investigates perceptual grouping in the digital ink domain [1, 10, 12]. All these approaches are based on heuristic grouping functions, with the exception of [12]. Unfortunately, no usability evaluation of these approaches has been presented.

Spatial parsers employed in hypertext systems automatically recognize implicit structures (typically representing semantic relationships) among objects, e.g. [5, 8]. These systems deal only with horizontal and vertical lists as well as clusters. They cannot deal with diagonal and curvilinear configurations. Moreover, they are typically based on heuristic functions that need to be tuned on a *per-user* basis. The only interaction technique offered is multi-clicking for hierarchical group selection. In ambiguous cases where there is more than one interpretation, most of the approaches visualize only the "best" interpretation. Neither interaction with secondary groups nor selection of multiple configurations is possible.

In the image editing domain, Saund *et al.* [11] applied a lattice-based grouping structure in which an object can be part of multiple groups. This system can detect curvilinear structures as well. Similar to spatial parsers, the only interaction technique is multiple-clicking for hierarchical group selection.

In our own previous work, we presented a system that recognizes implicit structures [2]. In the current paper, we improve on this work in multiple aspects. We utilize a perceptual-based scale-invariant proximity model and generalize the good continuity model to (curvi-)linear groups. Furthermore, we introduce several novel interaction techniques and present the results of a user study.

3 Motivation and Contributions

The approach we present in this paper is able to detect linear and curvilinear arrangements in arbitrary orientations as well as any form of clusters. Unlike previous work, our grouping methods are directly based on established models from perception science. This significantly increases the accuracy of Gestalt group detection and obviates the need to tune the system for individual users. Moreover, for ambiguous cases with more than one perceptual grouping, our system shows all interpretations at once. Based on different visual cues, the user can then not only distinguish the different configurations, but can also select the desired one(s). This enables selection of multiple groups at a larger scale without extra steps, for example. Finally, our approach extends existing interaction techniques in several ways. Most prominently, it enables partial group selection and allows the user to deal with ambiguous cases.

4 Grouping Objects by Gestalt Principles

Our system initially constructs a nearest neighbor graph. Then, it searches this graph to detect two types of perceptual groups: proximity and good continuity, where the last handles co-linearity as well as curvi-linearity.

4.1 Proximity Groups

Our proximity model is based on CODE, a scale-invariant dot-grouping algorithm [13]. In this algorithm, the grouping strength of each element (dot) exerted onto the others is modeled by a normal distribution function. This function is centered at each element and its standard deviation is half of the distance between the object and its closest neighbor. The strength of a proximity group is then defined as the summation over all individual functions. When the strength of a region surpasses a threshold, all elements in that region are grouped. Varying the threshold detects different groups at different scales. Figure 1 illustrates this on a group of four dots, labeled A, B, C, and D. Dashed and solid lines represent strength functions and the overall grouping strength, respectively. Applying threshold 1 puts all objects in the same group. Threshold 2 puts A, B, C in the same group and D in a separate group, etc.

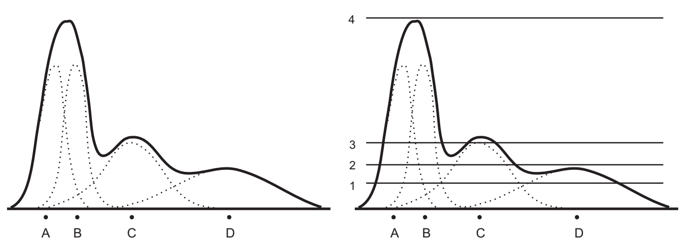

Fig. 1. Proximity grouping using CODE. Left: Objects are labeled A, B, C, and D. The dashed and solid curves represent the spread functions and the gradient strength, respectively. Right: Visualization of different thresholds.

4.2 Good Continuity Groups: Collinearity and Curvilinearity

For each set of four neighboring objects, our algorithm computes a linear coefficient (LC) indicating how strongly these four objects are perceived as a straight line. It is defined by:

$$\mathrm{LC} = \exp\left(-\frac{(a1+a2)^2}{2s^2(1+r)}\right) \times f(l_1, l_2, l_3).$$

where $a1$ and $a2$ are the angles between lines connecting the center of objects (see Fig. 2), r and s are constants, and f is a decaying exponential function of inter-object distances. This is based on Feldman's model for linear groupings of four consecutive dots [4]. We utilize a simpler model for groups of 3 objects [3].

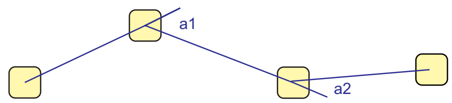

Fig. 2. Illustration of parameters used for good contiuity grouping.

We extended these models to deal with arc groupings as follows: In the above equation, we substitute every inter-line angle α_i by $(\alpha_i - \alpha_{Avg})$ where α_{Avg} is the average of all line angles α_i's. Hence, a uniform curvilinear path gets a high grouping coefficient similar to the case of a straight path.

In an extra step, the initial collinear and curvilinear sets are repetitively merged to form longer groups. In each merging step, smaller groups are discarded only if the grouping coefficient is relatively weaker than a threshold.

5 User Interaction with Gestalt Group

Here we introduce several novel interaction techniques that allow single, multiple, or partial Gestalt group selection for spatially contiguous groups. Furthermore, we present a technique to resolve ambiguity.

Proximity Group Selection: The fundamental approach is similar to the multi-click approach used in text editors and spatial parsers. Clicking on an object in a cluster, i.e. a proximity group, selects the object itself. Double-clicking selects the cluster. Each successive click extends the selection with the closest cluster. For example, in Fig. 3 the first click on an object in cluster $C1$ selects the object. Double-clicking selects $C1$, the next click adds $C2$ to selection, etc.

Good Continuity Group Selection: Similar to cluster selection, clicking on an object in a good continuity group selects only the single object. Moreover, the good continuity group is also visualized as colored links between successive objects. Double clicking then selects the whole group. If the clicked object is part of multiple groups, all the groups are selected, see Fig.4.

Partial Good Continuity Group Selection: To select a subgroup, the user first selects the whole group by double-clicking on an object, called anchor. Then the user deselects all undesired objects by clicking on the first non-desired one while holding down the alt-key. All objects on the path from the anchor "behind" this point will then be deselected, see Fig. 5.

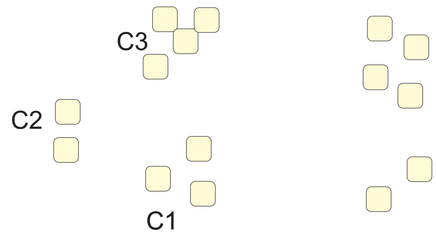

Fig. 3. Proximity group selection. Clicking on an object in cluster $C1$ selects the object. Double clicking selects $C1$, the next click adds cluster $C2$ to selection, etc.

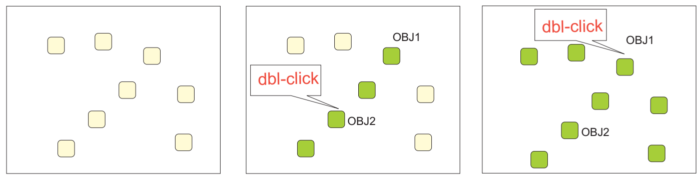

Fig. 4. Selecting good continuity group(s). Middle) Given the layout shown on the left, double clicking on $OBJ2$ selects the line. Right) Double clicking on $OBJ1$ selects the line and the arc.

5.1 Resolving Ambiguity

Ambiguity occurs when there is more than one visual interpretation of a scene. This is almost inevitable as soon as more than a few objects are involved. Our new approach permits the user to resolve ambiguities as follows:

– **Ambiguity in Clusters:** Grouping by proximity may not result in a unique grouping, as proximity can operate on multiple levels ranging from local to global. For example, in Fig. 3, three different configurations can be seen: five small groups, two large groups, or one whole group. Our grouping algorithm can detect all these configurations by changing the proximity threshold. Our novel interaction technique enables the user to change this threshold as follows: subsequent clicks on the same (anchor) object while holding the shift-key down changes the threshold to the next lower level, which enables group selection at larger scales. In contrast, subsequent clicks while holding the

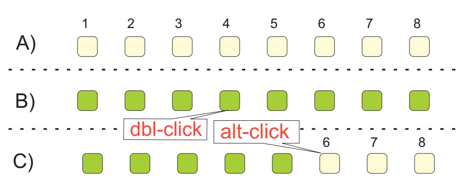

Fig. 5. Partial selection: A) a linear structure, B) double-clicking on object 4 selects the whole group, C) alt-clicking on object 6 then deselects objects 6, 7, and 8.

alt-key down changes the threshold to the next higher level, which selects groups at a smaller scale, see Fig. 6.

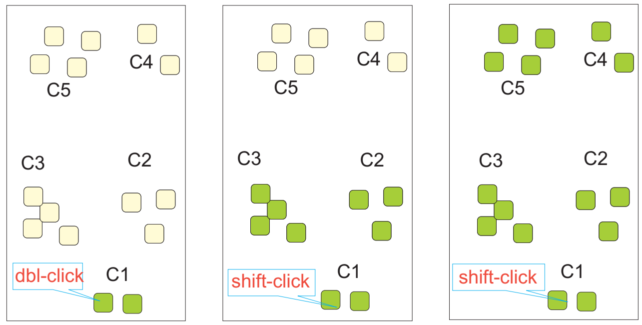

Fig. 6. Resolving proximity ambiguity: Left) Double-clicking on an object in $C1$ selects $C1$, i.e. the first level in the hierarchy. Middle) Then, a shift-click adds $C2$ and $C3$ to the selection (the next level). Right) Another shift-click selects all the clusters (the top level). At any level, an alt-click moves a level down in the hierarchy, i.e. one step left.

– **Ambiguity in Curvilinear Groups:** As mentioned before, double-clicking on an object shared by multiple groups selects all the groups. If only one of them is desired, the rest can be deselected by alt-clicking on non-desired node(s), as with partial selection. Figure. 7 illustrates such a scenario. The user double clicks on $OBJ1$ to select the diagonal group. However, horizontal and vertical groups are also selected as they share $OBJ1$. Clicking on $OBJ2$ and $OBJ3$ while holding the alt-key down deselects them.

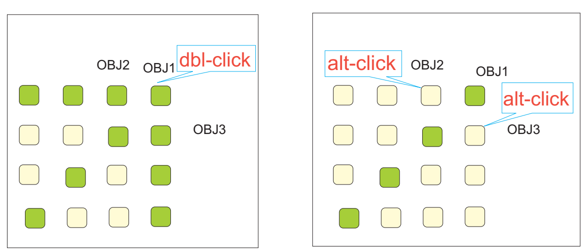

Fig. 7. Resolving curvilinear ambiguity: Left) double clicking on $OBJ1$ selects three groups (highlighted objects are selected). Right) clicking on $OBJ2$ and $OBJ3$ while holding the alt-key down deselect both groups.

6 Experiments

We conducted a within subject study to assess the efficiency of our technique in comparison to rectangle selection and auto-complete lasso. For rectangle and lasso selection shift-clicking a single object adds it to the group. Ctrl-clicking toggles selection, similar to most current GUI applications.

To ensure a fair comparison, we designed the layouts so that the bounding box of all targets contained no distracters. The issue here is that close-by distracters affect each selection technique in a different way and hence distracters would act as a confounding factor (a fact confirmed by pilot studies). For example, for non-axis aligned configurations, distracters make rectangle selection much more difficult. For lasso selection, distracters make the "tunnel" the user has to traverse smaller, which usually results in a reduction in movement speed. Similarly, for perceptual based techniques, very dense configurations can contain many different perceptual groups, which force the user to choose among those groups. Hence, we designed the layouts so that for all three techniques no subsequent modification to the selected group was required. We designed 36 layouts for groups of square targets with different structures, classified as follows:

- 3 Arrangements: linear, arc, cluster
- 3 Sizes: small, medium, large
- 4 Orientations: horiz, vert, sq45, sq135

In linear and arc arrangements targets were placed along straight lines and arcs, respectively. In cluster arrangements, targets were randomly spread out over an area. Size was defined by the diameter of the bounding box of the target group (small ≈ 250, medium ≈ 450, and large ≈ 750 pixels). The horizontal and vertical orientations had a bounding box with a significant difference in height vs. length, at least 4 to 1 ratio; the diagonal arrangements had (almost) square bounding boxes with objects arranged at roughly 45 or 135 degrees.

6.1 Tasks and Stimuli

In each task, participants were asked to select targets within the above-mentioned layouts using auto-complete lasso, rectangle, or the new Gestalt-based technique. Targets and distracters were displayed in green or black, respectively. When an object was selected, its border became stippled and its color was desaturated. Moreover, correctly selected target objects changed their color to yellow for better discrimination. If only the correct targets were selected, a brief sound was played and the software advanced to the next task, with a different layout. Selection time was measured from the first mouse click after the layout was displayed to the time when only the correct targets were selected. The number of cancelations, i.e. when the user clicked on an empty area to de-select everything, or drew a new rectangle/lasso, was also recorded.

6.2 Experimental Design

We used a repeated measure within subject design. The independent variables were: Selection Technique (Gestalt-based, auto-complete lasso, or rectangle), Group Arrangement (linear, arc, or cluster), Size (small, medium, or large), and Orientation (horiz, vert, sq45, or sq135). Dependant variables were selection time and error rate. We counterbalanced the order of selection techniques with a 3x3 Latin square to compensate for potential learning effects. Participants were trained between 10 to 20 minutes on all three techniques by asking them to perform various selections on 12 practice layouts. The main experiment used 36 layouts (|shape|*|size|*|orientation| $= 3 * 3 * 4$), which were shown 3 times for each technique. This whole sequence was repeated 3 times in different orders. Hence, each participant performed a total of $36 * 3 * 3 = 324$ selections during the experiment. In summary, the experiment design was as follows:

- 12 training layouts
- 36 trial layouts, categorized by target structure:
  - shape (line, arc, cluster)
  - size (small, medium,large)
  - orientation (horz, vert, sq45, sq135)
- 3 selection techniques (lasso, rectangle, Gestalt)
- 3 repetitions
- 11 participants

A grand total of 3564 selections was performed in the experiment. At the end, each participant was asked to fill out a questionnaire to evaluate ease of use and learn-ability of our technique.

Apparatus: The experiments were conducted on a computer with a Pentium M 1.6 Ghz processor and 1 GB memory. Screen resolution was 1024x768 and an optical mouse was used. The software was written in Python.

Participants: Eleven students from a local university campus were recruited to participate in the experiment: six females and five males, between 25-35 years of age. None of them had used our technique before. Most of them were unfamiliar with auto-complete lasso.

Hypotheses: Based on the fact that our Gestalt-based technique requires much less mouse movement, we hypothesize that selection time will be shorter for this technique when selecting common salient groups. Moreover, we hypothesize that users will make fewer errors with the new technique as it conforms better to human perception.

7 Results

Selection Time: A repeated measure ANOVA revealed that technique had a strong main effect on selection time ($F_{2,20} = 40.31, p \ll 0.001$). The mean selection time for the Gestalt-based technique was 0.38 seconds, with the means for rectangle and lasso 0.84 and 1.2 seconds, respectively, see Fig. 8. A Tukey-Kramer test reveals that all three techniques are different. As illustrated in

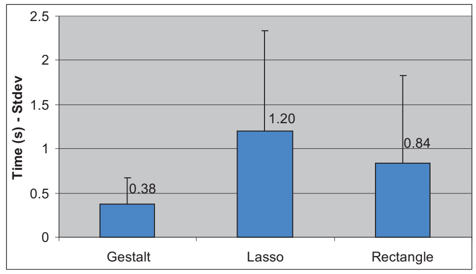

Fig. 8. Comparing selection time among techniques.

Fig. 9 and 10, orientation and shape had no significant effect on selection time, while size had a significant effect $F_{2,20} = 98.86, p \ll 0.001$. There is a strong interaction between technique and size of the layout, $F_{4,20} = 34.39, p \ll 0.001$. While size had no effect on Gestalt-based technique, lasso and rectangle selection time depended strongly on the size of the target group, see also Fig. 11.

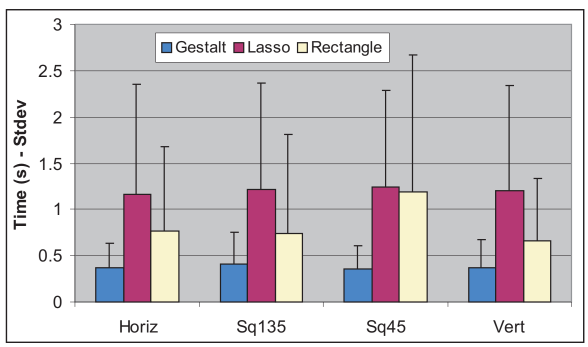

Fig. 9. Orientation does not have a significant effect on selection time.

Finally, we analyzed learning for each technique by plotting selection time vs. repetitions. The Gestalt-based technique showed no improvement over time, but participants got moderately faster with lasso and rectangle selection, see Fig. 13. When comparing performance per technique across participants, it is notable that most variation occurred within the rectangle technique, while the Gestalt technique had the least, see Fig. 12. As there was noticeable learning observable in the experiment, we analyzed the average number of cancelations only for the final, third, repetition. For this, there is not significant difference between the cancelation rates, see also Fig. 14.

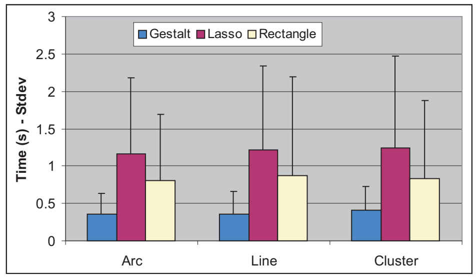

Fig. 10. Shape does not have a significant effect on selection time.

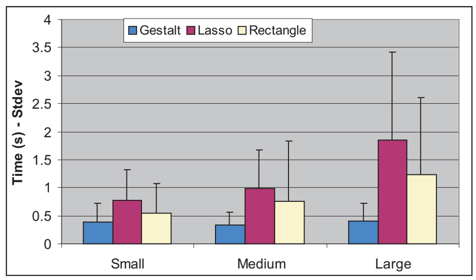

Fig. 11. Interaction between size and selection technique. Note that Gestalt is insensitive to size.

8 Discussion

Our technique is 2.3 respectively 3.2 times faster than rectangle or lasso. This is remarkable as the there were no nearby distracters and hence the trial layouts were equally well suited for rectangle and lasso. If distracters existed, we would expect that each technique would be affected differently (e.g. rectangle selection by non-axis aligned structures, lasso by narrow tunnels, Gestalt by ambiguous cases).

Orientation has a noticeable effect on rectangle selection. In particular, the time for the sq45 layout (layouts along $y = x$) is relatively longer than all other conditions. One explanation for this is that most people draw rectangles from the top-left corner to the bottom-right. For the sq45 layouts, identifying a good top-left corner point that covers only the targets is not trivial and errorprone.

The most likely explanation for the fact that our technique is faster is that much smaller mouse movements are needed: instead of traversing the circumference of a target group or the diagonal of the corresponding bounding box, the user just (double-)clicks on one of the group members. Rectangle is faster than lasso as the diagonal is shorter than the (partial) circumference, even with auto-complete lasso. Also, when using the mouse, most people find it easier to drag out a rectangle compared to drawing a curve that traverses a tunnel. Clearly, in the presence of nearby distracters, there exist layouts and target groups where

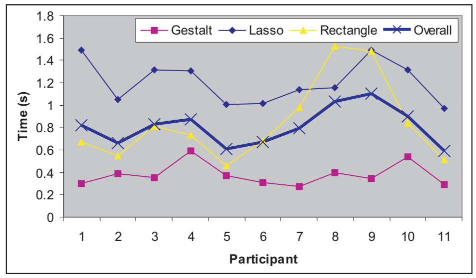

Fig. 12. Comparing overall and per technique performance of participants.

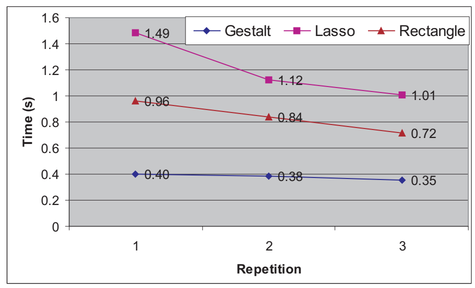

Fig. 13. Effect of repetition on selection time (learning effect).

rectangle or lasso selection may perform better than our technique. For example, lasso can outperform all other techniques for partial selection of a dense, random cluster. In general, we believe that rectangle and lasso are well suited for selection of groups that have significant two-dimensional spatial extent (i.e. area arrangements) or that are axis-aligned. On the other hand, our technique is highly well suited for groups with curvilinear layouts (i.e. one-dimensional arrangements), while still being competitive for cluster selection. Hence, we believe Gestalt selection nicely *complements* rectangle and lasso selection.

9 Conclusion and Future Work

We introduced a new perceptual-based object group selection technique for spatially contiguous groups. Based on established models from perception research, we presented a new approach to automatically detect salient perceptual groups with the Gestalt principles of proximity and good continuity. Then, we introduced several new, simple, and efficient mouse-based interaction techniques to select and deselect such Gestalt groups. The results of our user study show that our technique outperforms lasso and rectangle selection when selecting groups with implicit structures.

We have not yet formally evaluated our technique in more complex scenarios, such as partial group selection and the resolution of ambiguous cases. Informal

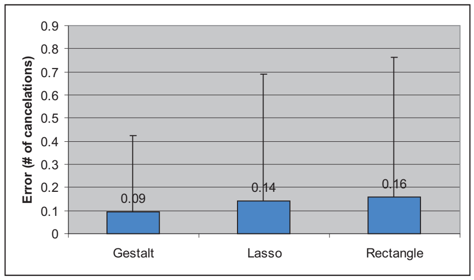

Fig. 14. Average cancelation rate for different techniques.

evaluations indicate that the technique works well for these cases, but we plan to do a complete user study in future work. Moreover, we will investigate extensions of our technique for highly dense configurations. Finally, we will investigate the complex interplay of spatial and similarity cues and extend our system to deal with objects with different visual features (such as shape, color, and size).

References

1. P. Chiu and L. Wilcox. A dynamic grouping technique for ink and audio notes. In *UIST*, pages 195–202, 1998.
2. H. Dehmeshki and W. Stuerzlinger. Using perceptual grouping for object group selection. *CHI Extended Abstracts*, pages 700–705, 2006.
3. J. Feldman. Perceptual models of small dot clusters. *DIMACS*, 19, 1993.
4. J. Feldman. Curvelinearity, covariance, and regularity in perceptual groups. *Vision Research*, 37(63), 1997.
5. T. Igarashi, S. Matsuoka, and T. Masui. Adaptive recognition of implicit structures in human-organized layouts. *Visual Languages*, pages 258–266, 1995.
6. K. Koffka. *Principles of Gestalt Psychology*. Routledge and Kegan Paul, 1935.
7. M. Kubovy and A. Holcombe. On the lawfulness of grouping by proximity. *Cognitive Psychology*, 35:71–98, 1998.
8. C. C. Marshall, F. M. Shipman, and J. H. Coombs. Viki: spatial hypertext supporting emergent structure. In *Hypermedia Technology*, pages 13–23, 1994.
9. S. Mizobuchi and M. Yasumura. Tapping vs. circling selections on pen-based devices: evidence for different performance-shaping factors. In *SIGCHI*, pages 607–614, 2004.
10. T. P. Moran, P. Chiu, W. van Melle, and G. Kurtenbach. Implicit structure for pen-based systems within a freeform interaction paradigm. In *CHI*, 1995.
11. E. Saund, D. Fleet, D. Larner, and J. Mahoney. Perceptually-supported image editing of text and graphics. In *UIST*, pages 183–192, 2003.
12. E. Saund and T. P. Moran. A perceptually-supported sketch editor. In *UIST*, pages 175–184, 1994.
13. M. P. van Oeffelen and P. G. Vos. An algorithm for pattern description on the level of relative proximity. *Pattern Recognition*, 16(3):341–348, 1983.