An Agent-Based Approach to Point-Feature Label Placement

Hyeyoung Kim and Ningchuan Xiao

ABSTRACT: The goal of solving a point feature label placement problem is to find desirable positions for the labels of all point features on a map. Though the cartographic literature has a general agreement on the optimal position of each individual point label, finding optimal positions for all labels remains a difficult problem mainly because of its computational intensity. In this paper, we developed an agent-based approach to addressing the label placement problem. In our method, a label is assigned one of the eight positions. Each position has a weight ranging from 1 (the worst position directly underneath the point symbol) to 8 (the optimal upper-right position). Each label is programmed as an autonomous agent that tries to be placed at the optimal position. The overall goal -- placing the maximum number of labels at optimal positions while minimizing overlap between labels -- is achieved through a set of rules that govern the behavior of each individual label throughout an iterative simulation process. Specifically, each agent (i.e., a label) is assigned to the optimal position initially and agents start to move to a less preferred position to avoid overlap. An agent will receive a credit if it moves to a position with a lower weight. Among the overlapping agents, those with the lowest credit will move first. If multiple overlapping agents have the same credit level, the program will randomly pick one to move to a less desirable position. The process terminates when no movement can be negotiated or total weight of all agents does not decrease. We demonstrate the effectiveness of our method through computational experiments using data sets with different number and spatial distributions of point symbols. Our experiments suggest that the method is robust and efficient in finding high quality (i.e., optimal or near-optimal) label placement schemes.

KEYWORDS: point-feature label placement, agent-based approach, cartography

Introduction

Point feature label placement on a map is a common issue among cartographers. In general, the quality of label placement depends on the displayed information it conveys and the aesthetics of the map it presents. In order to enhance the quality of label placement, many cartographers have studied and proposed new approaches (Agarwal *et al.*, 1998; Beus and Crockett, 1994; Christensen *et al.*, 1993, 1995; Doerschler and Freeman, 1992; Formann and Wagner, 1991; Freeman, 2005; Huffman and Cromley, 2002; Imhof, 1975; Jones, 1989; Kakoulis and Tollis, 1997, 1998, 2003; Knuth and Raghunathan, 1992; Poon *et al.*, 1998; Wagner, 1994; Wagner and Wolff, 1995; Zhu and Poon, 1999, 2001). A rule based approach is popular (Ahn and Freeman, 1984; Cook and Jones, 1990; Freeman and Ahn, 1987; Imhof, 1975; Jones, 1989; Doerschler and Freeman, 1992; Wood, 2000). In this approach, a set of rules can be used to guide the cartographer to place the maximum number of labels in preferred positions and to avoid overlap between labels or between label and feature symbols in order to arrange the labels aesthetically on the map.

With efforts to place labels on a map automatically, algorithms for label placement have been developed since the early 1970s (Christensen *et al.*, 1995; Cook and Jones, 1990; Doerschler and Freeman, 1992; Ebinger and Goulette, 1990; Ebner et al., 2003; Edmondson et al., 1996;

Freeman and Ahn, 1984, 1987; Hirsch, 1982; Jones, 1989; Kameda and Imai, 2003; Mower, 1993; Stadler et al., 2006; Van Dijk et al., 2004; Van Kreveld et al., 1999; Verner et al., 1997; Wagner et al., 2001; Yamamoto and Lorena, 2005; Yoeli, 1972; Zoraster, 1986, 1991, 1997). Automated label placement, however, is one of the more complex and difficult processes since not only the position of a label but also the relationship between labels should be considered simultaneously. Though the cartographic literature has a general agreement on the optimal position of each individual point label, finding optimal positions for all labels still remains a difficult problem, mainly because of its computational intensity.

In this paper, we propose a new method to solve the point feature label placement problem using agent-based approach. In our method, a label is treated as an agent who can negotiate with other agents (labels) to find a desirable position throughout an iterative simulation process. The robustness of our model is illustrated by use of three different data sets consisting of simple data, randomized data and real-world data.

Model Design

Criteria for point feature label placement

Since automated label placement was first proposed by Yoeli (1972), many cartographers have used his rules in their research (Ahn and Freeman, 1984; Doerschler and Freeman, 1992; Ebinger and Goulette, 1989, 1990; Freeman and Ahn, 1984; Hirsh, 1982; Huffman and Cromley, 2002; Jones and Cook, 1989; Mower, 1986; Wagner and Wolff, 1997, 1998). Yoeli's basic rules for the quality of labeling include

- A label on the map cannot overlap with other labels or other feature symbols,
- Each label should be associated with exactly one feature symbol, and
- Each label should be placed in the more preferred position.

In an overcrowded map region, it is possible to encounter the unavoidable case that a label intersects with another label or feature symbol. In order to avoid this case, Langran and Poiker (1986) and Huffman and Cromley (2002) removed the overlapping label and placed the remaining labels without any overlaps. Since the overall goal of our model is to place the maximum number of labels at optimal positions while minimizing overlap between labels, we placed all the labels on the map, including the overlapping labels.
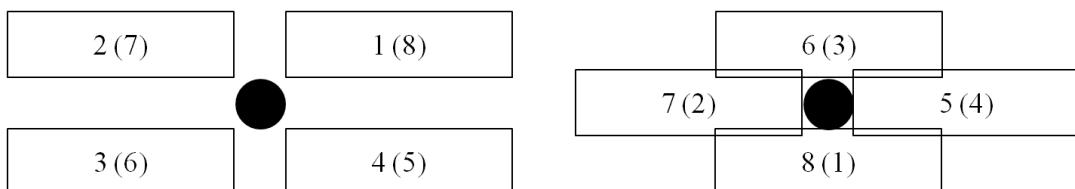

Figure 1: Typical set of potential label positions.

The potential label positions are decided by the rules which capture aesthetic quality of label positions. The most desirable position of label in a typical rule is on the right and above the point. Figure 1 shows a typical set of potential label positions for a point feature. In our method, a label is assigned to one of the eight positions by following the typical rule. Each position has a weight ranging from 1 to 8. The number one position, which is the optimal upper-right position, has been given the highest weight and the number eight position, the worst position directly underneath the point symbol, has been given the lowest weight. In Figure 1, the numbers in parentheses refer to the individual position weights.

An agent-based approach

In this study, we applied an agent-based approach to placing labels for point features. The agent-based model is a useful method to analyze complex systems. In the agent-based model, each agent learns from dynamic interactions among other agents or its surrounding environment and acts to pursue its goals. We assumed that each label is programmed as an autonomous agent that tries to position at the optimal position.

In our model, we introduced a credit to compensate for movement of an agent. That is, an agent will receive a credit if it moves to a lower weighted position. Among the overlapping agents, the agent with the lowest credit will move first. If multiple overlapping agents have the same credit level, the program will randomly pick one to move to a less desirable position. At the same time, the agent checks their position whether it is located inside the map region or whether the agent is overlapping with other feature symbols. If the agent has been placed in the desirable position and there is no overlapping with other labels or other feature symbols, then the position of the agent becomes fixed. If not, the agent continually moves such that the overlapping area between agents or an agent and a feature symbol is minimized. Moreover, each agent is being checked for the possibility of being moved to the previous positions with higher weights, because the previous position may remain empty after movement of other agents. As a result, each agent can be placed again in their best position. After that, the program adds up the weights of all agents and calculates the number of overlapping labels with other labels and feature symbols. It also determines the overlapping area for the entire region of the map. The process terminates when no movement can be negotiated or the total weight of all agents does not decrease with overlapping area being minimized. The algorithm used in our approach is as follows.

Algorithm for point feature label placement

Input: A set of the point features *S* to be labeled.
Output: A label placement with minimum overlapping area.

1. Place the agents (labels) *A* of each point feature in *S* in the most desirable position.
2. Find overlapped agents A_i and A_j.
3. *If* Priority of A_i == Priority of A_j *then*:
    3.1. *If* Credit of A_i == Credit of A_j *then*:
        3.1.1. Pick A_i or A_j randomly and move it to a less desirable position and give a credit.
        3.1.2. *If* it is overlapped with a feature in *S* or is placed outside the map region *then*:
            3.1.2.1. Move it to a less desirable position and give credit.

3.2. *Else*

    3.2.1. Move the agent with lowest credit to a less desirable position and give a credit.

    3.2.2. *If* it is overlapped with *S* or is placed outside the map region *then*:

        3.2.2.1. Move it to a less desirable position and give credit.

4. *Else*

    4.1. Move the agent with lower priority to a less desirable position and give credit.

    4.2. *If* it is overlapped with *S* or is placed outside the map region *then*:

        4.2.1. Move it to a less desirable position and give credit.

5. *If* all agents are moved *then*:

    5.1. *If* there is no overlapping with the more desirable position of the agent and other agents *then*:

        5.1.1. Move the agent to the more desirable position and take a credit from the agent.

6. Sum up the weights of all agents and calculate the overlapping area.

7. Repeat *Steps* 2, 3, 4, 5 and 6 until total weight does not decrease and the overlapping area is minimized.

Experiments and Results

In order to verify our model, we applied three different data sets consisting of simple data, randomized data and real-world data. For this experiment, we implemented an automated label placement simulator on top of the MASON[1] in Java. We set up the size of a map 600×600 pixels.

Simple data set

Firstly, we used a simple data set with 50 point symbols arranged in a vertical line with one pixel interval. Each symbol has the radius of 3 pixels. The names were extracted from a gazetteer database. In this experiment, a 11 point typeface is used for each label and we draw the bounding box of each label without margin. Figure 2 shows the simulation result with the simple data set. In the beginning of the simulation (Figure 2-a), all labels are placed in their most desirable position(upper-right of the point), however they are overlapping with each other. Figure 2-b illustrates that after simulation all labels are placed in the optimal position with no overlaps.

Randomized data set

In this experiment, we tested our method with a randomized data set. We first generated the data set where places feature points are located at random positions on the map. Then the priority was assigned to all points to identify the importance of the label. Based on the priority, the point features were divided into two classes: symbols with radius of 5 pixels and labels with the font size of 13 (first priority) and symbols with radius of 3 pixels and font size of 11 (second priority). Figure 3 illustrates the simulation result based on this data set.

---

[1] MASON is a single-process discrete-event simulation core and visualization toolkit (http://cs.gmu.edu/~eclab/projects/mason/).

According to the rules we set in our model, the goal is to place each label inside the map region at the preferred location without any overlaps. Although our method tries to place the labels of higher priority in a more desirable location (*Step* 4 in the algorithm), it also allowed to place the label in the next desirable location if there was a feature symbol in a more desirable location (see the blue circle in Figure 3-b).

(a)   (b)
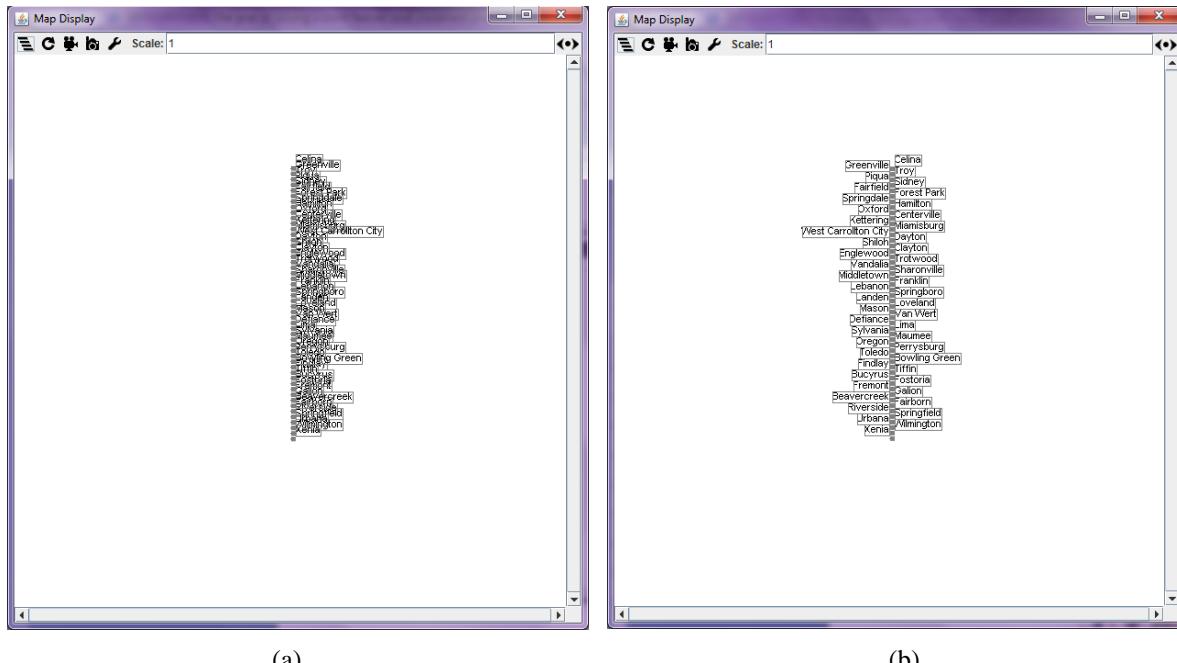
Figure 2: Simple data set with labels placed in:

(a) the most desirable position and (b) optimal position.

(a)   (b)
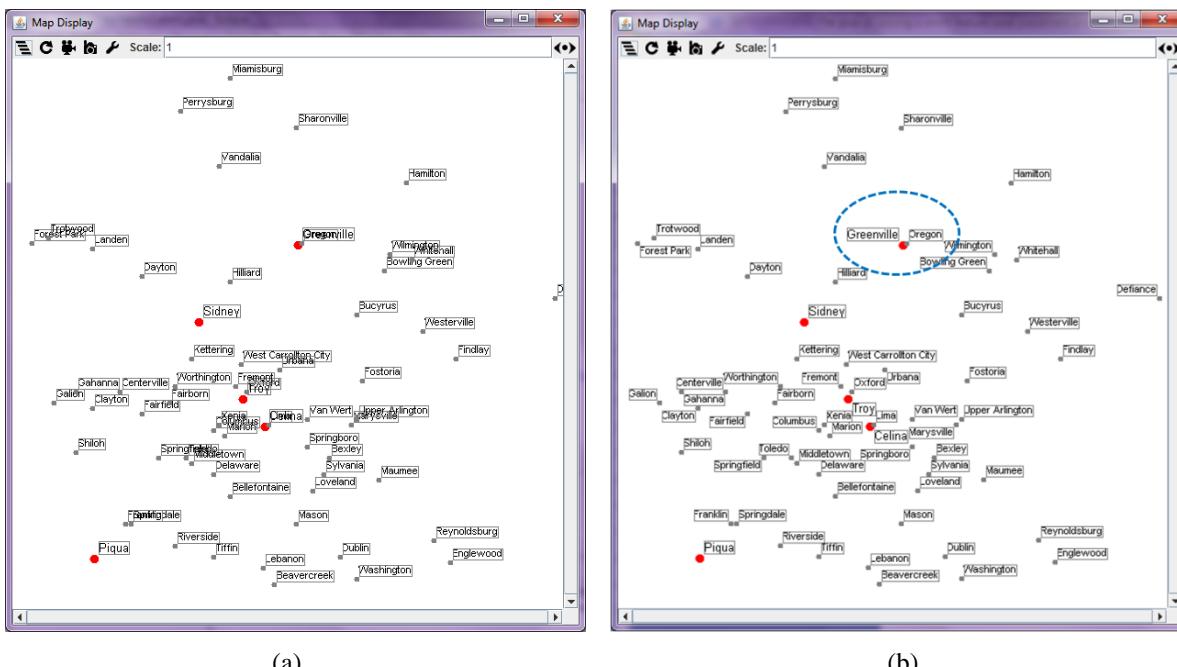
Figure 3: Randomized data set with labels placed in:

(a) the most desirable position and (b) optimal position.
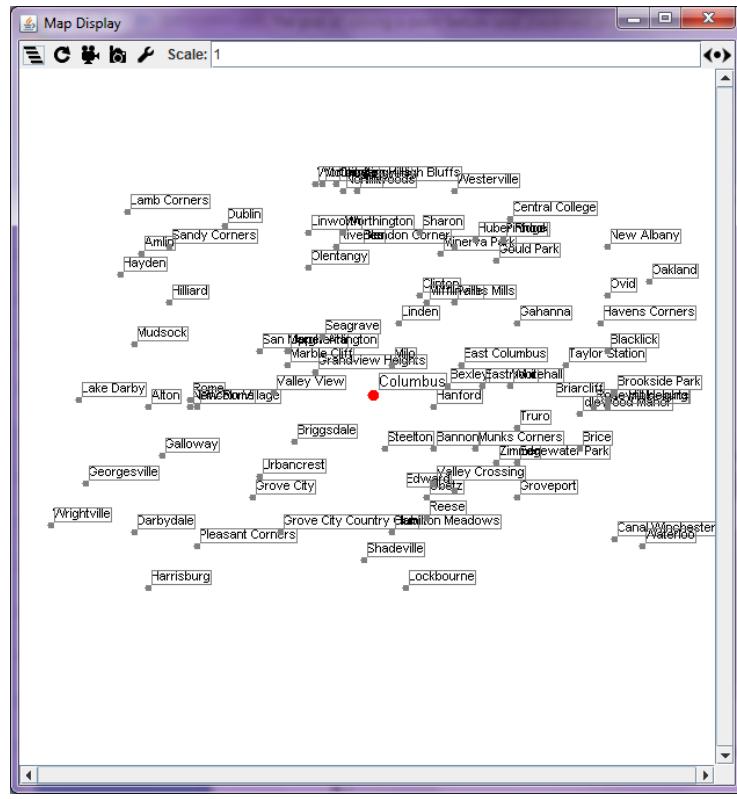
(a)
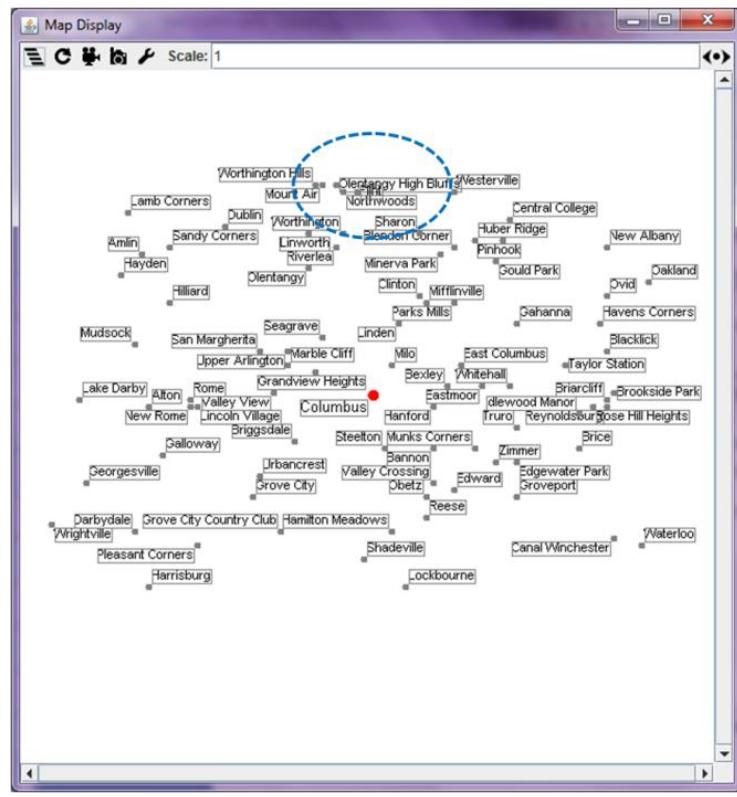
(b)

Figure 4: Real-world data set with labels placed in: (a) the most desirable position and (b) optimal position.

Real-world data set

Lastly, we ran our model using the real-world data set extracted from the census data. We restricted the study area to Franklin County in Ohio and picked the point data which had an attribute such as population so that we could divide the data into two classes. We assigned the points with population over 500,000 as more important and the other points with population under 500,000 as less important. Size of symbols and fonts in the two classes was set same way as the experiment with randomized data set, respectively. The simulation result is illustrated in Figure 4. In this map area, there is an overcrowded region and also there is not enough space for all labels to be placed without overlaps. Although there were some overlapped labels after the end of the simulation, our approach found the optimal case when the overlapping area was minimized (see the blue circle in Figure 4-b).

Concluding remarks

In this paper, we have suggested an effective method with the cartographic rules to address the point feature label placement issue. By applying the agent-based approach, we have shown the process of how to place labels in a preferred position. We have also demonstrated the effectiveness of our method using different kinds of data set. Through the simulation, our method displayed the labels at a desirable position with minimizing the overlapping. We believe that the method developed in this paper can be extended to line and area feature label placement.

References

Agarwal, P., Van Kreveld, M. and Suri, S. (1998) Label placement by maximum independent set in rectangles. Computational Geometry: *Theory and Applications*. 11, pp. 209–218.

Ahn, J. K, and Freeman, H. (1984) A program for automatic name placement, *Cartographica*. 21, 2/3, pp. 101-09.

Beus, D., and Crockett, D. (1994) Automated production of 1:24,000 scale quadrangle maps. *In the proceedings of 1994 ASPRS/ACSM Annual Convention and Exposition*, 1, pp. 94–99.

Christensen, J, Marks, J. and Shieber, S. (1993) Algorithms for cartographic label placement. *In the proceedings of 1993 ASPRS/ACSM Annual Convention and Exposition*. 1, pp. 75–89.

Christenson, J., Marks, J. and Shieber, S. (1995) An empirical study of algorithms for point-feature label placement. *ACM Transactions on Graphics*. 14, pp. 203–222.

Cook, A. and Jones, C. (1990) A Prolog rule-based system for cartographic name placement. *Computer Graphics Forum 9*. pp. 109-126.

Doerschler, J. and Freeman, H. (1992) A rule-based system for dense map name placement. *Communications of the ACM*, 35, 1, pp. 68-79.

Ebinger, L. and Goulette, A. (1989) Automated names placement in a non-interactiveenvironment. *In Proceedings of Auto-Carto 9*, pp. 205-14.

Ebinger, L. and Goulette, A. (1990) Noninteractive automated names placement for the 1990 Decennial Census. *Cartography and Geographic Information Systems*. 17, 1, pp. 69-78.

Ebner, D., Klau, G. and Weiskirscher, R. (2003) Force-based label number maximization. *Vienna, Austria: Vienna University of Technology*.

Edmondson, S., Christensen, J., Marks, J. and Shieber, S. (1996) A general cartographic labeling algorithm. *Cartographica*. 33, 4, pp. 13-23.

Freeman, H. and Ahn, J. (1984) AUTONAP - An expert system for automatic map name placement. *In proceedings of the International Symposium on Spatial Data Handling*. 2, pp. 44- 69.

Freeman, H. and Ahn, J. (1987) On the problem of placing names in a geographic map. *International journal of pattern recognition and artificial intelligence. I*. 1, pp. 12-140.

Freeman, H. (2005) Automated cartographic text placement. *Pattern Recognition Letters*. 26, pp. 287-297.

Formann, M. and Wagner, F. (1991) A packing problem with applications to lettering of maps. *In the proceedings of the 7th Annual ACM Symposium on Computational Geometry*. pp. 281–288.

Hirsch, S. (1982) An algorithm for automatic name placement around point data. *American Cartographer*, 9, 1, pp. 5-17.

Huffman, F. and Cromley, R. (2002) An automated multi-criteria cartographic aid for point annotation. *The Cartographic Journal*. 39, 1, pp. 51-64.

Imhof, E. (1975) Positioning names on maps. *The American Cartographer*. 2, pp. 128–144.

Jones, C. (1989) Cartographic name placement with Prolog. IEEE Computer Graphics and Applications. 9,5, pp:36-47.

Jones, C. and Cook, A. (1989) Rule-based name placement with Prolog. *In Proceedings of Auto-Carto 9*, pp. 231-39.

Kakoulis, K. and Tollis, I. (1997) An algorithm for labeling edges of hierarchical drawings. *In the proceedings of Graph Drawing `97*. pp. 169–180.

Kakoulis, K. and Tollis, I. (1998) A unified approach to labeling graphical features. *In proceedings of the 14th Annual ACM Symposium on Computational Geometry*. pp. 347–356.

Kakoulis, K. and Tollis, I. (2003) A unified approach to automatic label placement. *International Journal of Computational Geometry and Applications*. 13, 1, pp. 23-59.

Kameda, T. and Imai, K. (2003) Map label placement for points and curves. *IEICE Transaction Fundamentals*. E86, A4, pp.835-840.

Knuth, D. and Raghunathan, A. (1992) The problem of compatible representatives. SIAM *Journal on Discrete Mathematics archive*. 5, 3, pp. 422–427.

Langran, G. and Poiker, T. (1986) Integration of name selection and name placement. *In Proceedings of Auto-Carto 8*, pp. 50–64.

Mower, J. (1986) Name placement of point features through constraint propagation. *In proceedings of the 2nd International Symposium on Spatial Data Handling*, pp. 65-73.

Mower, J. (1993) Automated feature and name placement on parallel computers. *Cartography and Geographic Information Science*. 20,2, pp.69-82.

Poon, C., Zhu, B. and Chin, F. (1998) A polynomial time solution for labeling a rectilinear map. *Information Processing Letters*. 65, 4, pp.201-207.

Stadler, G., Steiner, T. and Beiglbock, J. (2006) A practical map labeling algorithm utilizing morphological image processing and forcedirected methods. *Cartography and Geographic Information Science*. 33, 3, pp.207-215.

Van Dijk, S., Thierens, D. and De Berg, M. (2004) On the design and analysis of competent selecto-recombinative GAs. *Evolutionary Computation*. 12, 2, pp.243-267.

Van Kreveld, M., Strijk, T. and Wolff, A. (1999) Point labeling with sliding labels. *Computational Geometry*. 13, pp.21-47.

Verner, O., Wainwright, R. and Schoenefeld, D. (1997) Placing text labels on maps and diagrams using genetic algorithms with masking. *INFORMS Journal on Computing*. 9, 7, pp.266-275.

Wagner, F. (1994) Approximate map labeling is in Ω (n log n). *Information Processing Letters*. 52, 3, pp 161–165.

Wagner, F. and Wolff, A. (1995) Map labeling heuristics: Provably good and practically useful. *In proceedings of the 11th Annual ACM Symposium on Computational Geometry*. pp. 109–118.

Wagner, F. and Wolff, A. (1997) A practical map labeling algorithm. *Computational Geometry: Theory and Applications*, 7, pp. 387-404.

Wagner, F. and Wolff, A. (1998) A combinatorial framework for map labeling, *In proceedings of the Symposium on Graph Drawing '98*. 1547, pp. 316-31.

Wagner, F., Wolff, A., Kapoor, V. and Strijk, T. (2001) Three rules suffice for good label placement. *Algorithmica*. 30, 2, pp. 334-349.

Wood, C. (2000) Descriptive and illustrated guide for type placement in small scale maps. *The Cartographic Journal*. 37, 1, pp. 5-18.

Yamamoto, M. and Lorena, A. (2005) A constructive genetic approach to point-feature cartographic label placement. *Metaheuristics: Progress as Real Problem Solvers*. New York, NY: Springer.

Yoeli, P. (1972) The logic of automated map lettering. *The Cartographic Journal*. 9, 1, pp. 99-108.

Zhu, B. and Poon, C. (1999) Efficient approximation algorithms for multi-label map labeling, In the proceedings of the 10th International Symposium on Algorithms and Computation. pp. 143 - 152.

Zhu, B. and Poon, C. (2001) Efficient approximation algorithms for two-label point labeling. *International Journal of Computational Geometry and Applications*. 11, 4, pp. 455–464.

Zoraster, S (1986) Integer programming applied to the map label placement problem. *Cartographica*. 23, 3, pp.16-27.

Zoraster, S (1991) Expert systems and the map label placement problem. *Cartographica*. 28, 1, pp.1-9.

Zoraster, S (1997) Practical results using simulated annealing for point feature label placement. *Cartography and Geographic Information Science*. 24, 4, pp.228-238.

Hyeyoung Kim, Ph.D student, Department of Geography, The Ohio State University, Columbus, OH 4321 0. Email <kim.3594@osu.edu>

Ningchuan Xiao, Associate Professor, Department of Geography, The Ohio State University, Columbus, O H 43210. Email <xiao.37@osu.edu>