Introducing Leader Lines into Scale-Aware Consistent Labeling

Hsiang-Yun Wu[1], Shigeo Takahashi[2],
Sheung-Hung Poon[3], and Masatoshi Arikawa[4]

[1] Keio University, Japan
[2] University of Aizu, Japan
[3] Universiti Teknologi Brunei, Brunei Darussalam
[4] The University of Tokyo, Japan

Abstract. Consistently placing annotation labels across map scales often poses a problem due to the restriction of the screen space. This problem becomes further exacerbated when we navigate by arbitrarily zooming in and out of digital maps on mobile devices. In this paper, we introduce leader lines to conventional techniques for scale-aware consistent labeling to accommodate more annotation labels on the map domain while retaining their plausible arrangement. The overall visibility of annotation labels is optimized using genetic algorithms while avoiding their unwanted popping effects and sudden leaps regardless of the change in the map scale. The feasibility of the proposed approach is demonstrated by experimental results including comparison with relevant techniques.

Keywords: Active range optimization, dynamic labeling, leader lines, genetic algorithms

1 Introduction

The development of mobile devices and digital map services allows us to read maps not only on papers but also on display devices of various sizes. This technology advancement enables us to zoom in and out of the map content for an overview and details of areas of interest and is especially important to dynamically explore map content across multiple scales within limited screen space. In addition, annotation labels help precisely locate specific geographic positions on maps and clearly discriminate them from the names of other places. Thus, laying out such labels consistently across the map scales will provide better readability of the map content on digital devices. However, consistency in label placement leads to technically challenging problems. Figure 1 shows specific types of such unwanted behaviors in which we can see popping effects and sudden leaps by the annotation labels as the scale changes. Avoiding visual distractions in label placement across multiple scales usually requires high computational costs because we have to investigate all possible combinations of labels and their spatial placement.
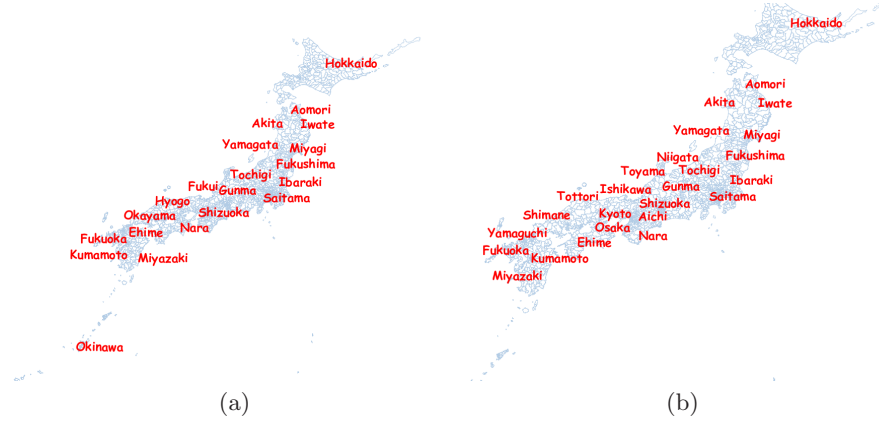
Fig. 1. Naïve inconsistent placement of labels as the scale changes from (a) to (b). All the name labels that appear in (a) are expected to stay in (b) while some labels disappear (e.g., "Shimane" and "Yamaguchi") and other labels (e.g., "Kumamoto" and "Miyazaki") unexpectedly jump between different positions at the different map scales.
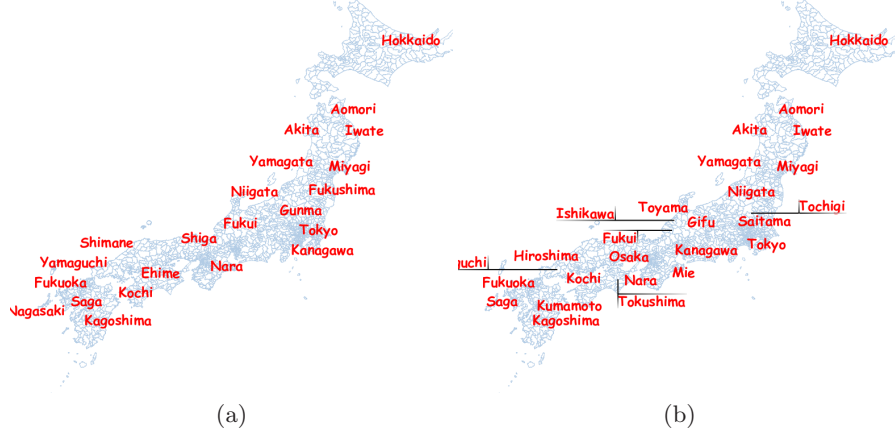

Fig. 2. Label placements (a) without leader lines (#{labels} = 22) and (b) optionally with leader lines (#{labels} = 26). We can place more labels by selectively employing leader lines to take advantage of available space within the map domain.

One promising solution to this problem is to retain consistency in the label positions on a map irrespective of the change in the map scale. This consistent map labeling problem was formulated as *active range optimization (ARO)* (Been et al., 2006, 2010). The active range refers to the range of the map scale in which the corresponding label sticks to the unique position on the map domain. Thus, the consistent labeling problem can be solved by maximizing the total sum of active ranges in such a way that as many annotation labels as possible remain and as long as possible within the space spanned by the map domain and scale. To achieve this, we extend the conventional ARO approaches by introducing *leader*

lines that connect between the point features and the annotation labels. This permits us to spare a little space between a point feature and its corresponding label, which eventually relaxes the degree of freedom in the label placement and thus is more likely to accommodate more labels within the map domain. For maximizing the total active range, we employ a genetic-based approach to seek better placement of the annotation labels across the multiple map scales. Specifically, we optimize the order of the labels to be placed and find the best order, where we try to place the labels in a greedy fashion according to the corresponding order. This allows us to finalize the best order of the labels that maximizes the visibility of consistent label placement across the entire range of the map scale.

Our technical contribution lies in the sophistication of previous techniques for consistently placing labels across multiple scales, by introducing leader lines that connect point features and labels to take advantage of the space around the point features. This labeling design is particularly useful for visualizing areas that have heterogeneous distributions of point features in such a way that they form a set of local clusters. This is because we can place annotation labels even when they are a little away from the corresponding point features with the help of leader lines and thus can increase the total number of labels to be placed in the map. Figure 2 gives an example of such consistent placement of labels together with leader lines. After providing several experimental results, we compare the conventional label placement without leader lines and the proposed label placement with leader lines to demonstrate the feasibility of our labeling approach.

The remainder of this paper is structured as follows. In Section 2, we briefly summarize most relevant literature to this study. In Section 3, we present an overview of our visualization framework, including the design of the leader drawing styles for consistent dynamic label placement. In Section 4, we then detail an algorithm for the extended ARO, which is effectively solved by the genetic-based heuristic and greedy label placement scheme. In Section 5, we provide several results generated by our system to demonstrate the feasibility of our approach. Finally, in Section 6, we conclude the paper and propose potential future extensions.

2   Related Work

In this section, we review the most relevant techniques for the label placement problem.

2.1   Static map labeling

Cartographers have investigated the conception of text discrimination, which is evaluated for noticing the difference between features in cartography and has been shown to be a key factor for map design (MacEachren, 1995; Chiang et al., 2014). To improve text discrimination, labeling principles have been studied

and summarized in several pioneering works (Yoeli, 1972; Imhof, 1975). Wolff (2009) conducted an extensive survey of techniques for automatic map labeling and thus provided instructions for computer science in cartography. For example, gradient-descent optimization (Hirsch, 1982) and the integer programming model (Zoraster, 1986) are classical static map labeling techniques, which follow an empirical investigation to demonstrate the feasibility of each algorithm (Christensen et al., 1995). Afterward, a genetic algorithm was introduced to accelerate the computation time (Raidl, 1998). Other map features, such as lines and areas, have also been considered to improve the readability of labels on maps (Edmondson et al., 1997; Luboschik et al., 2008), and domain-assisted label placement called boundary labeling (Bekos et al., 2007, 2015) was developed for this purpose. Appropriately drawing different leader line styles for label placement enhances map readability effectively (Lin, 2010; Meng et al., 2015).

2.2 Dynamic map labeling

In practice, mobile devices are used for viewing maps, and thus, effective navigation, such as annotating labels within a lens, has become a popular solution (Fekete & Plaisant, 1999). Fink et al. (2012) enhanced this layout by optimizing multiple design rules, such as minimizing the total leader length and distance between labels, and thus provided beautiful label placement in static and dynamic environments. A two-step labeling approach was introduced by Petzold et al. (2003), where the system computes conflicted labels in a preprocessing process and then applies the results to the user navigation. Gemsa et al. (2013) also conducted trajectory-oriented map labeling, where consistent label placement is computed that refers to moving history. These techniques were developed in a fixed scale and less consider the visual consistency of label placement during user interaction.

Schwartges et al. (2014) presented a slider model to move labels along the associated site points during scale changes, while a waiting function is employed to avoid flickering effects. As pioneering work on consistent label placement, Poon & Shin (2005) conducted another study on consistent label placement; however, in this approach, the orientation of a label is fixed. A significant extension of this approach was then proposed by Been et al. (2006, 2010), in which they relaxed the restrictions on the algorithm to place labels of various sizes. The method optimizes the active ranges of all labels by plotting rectangular pyramids in the three-dimensional (3D) space to compute the conflicts of labels at individual scales. Recently, this work was extended by Zhang et al. (2015), in which the active ranges are designed to be selectable, and the total weight of the associated ranges is maximized.

In this study, we relax the limitation of the approach proposed by Been et al. (2006, 2010), so that labels with leader lines can be employed to improve map readability. A genetic-based approach is introduced to solve the ARO problem (Raidl, 1998; Wu et al., 2011), which provides an effective means for the map labeling problem, especially for optimizing fitness functions without being trapped by local extrema.
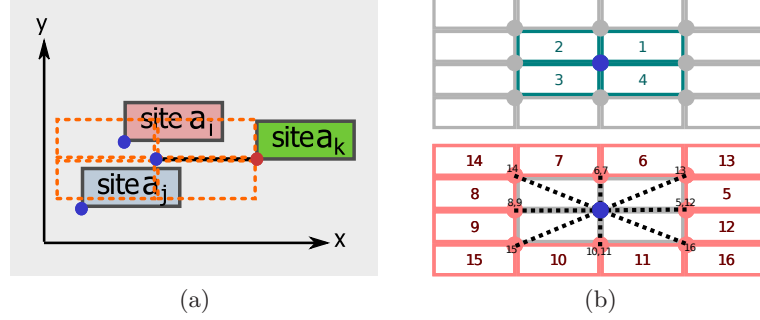
Fig. 3. Possible label positions to a given site point. (a) The conventional four-position model (in orange) usually suffers from the lack of available space around the point feature. (b) Our new 16-position model (bottom) consists of not only the conventional four positions (top) but also 12 additional positions. For each, we use the leader lines to connect the label and the point feature.

3 Design Principles for Label Placement

In this section, we provide an overview of the proposed approach with an explanation of the design principles for label placement. The primary technical challenge here is to develop scale-adaptive label placement while introducing leader lines in the ARO model. The key idea is to appropriately select a subset of visible labels at each scale to make full use of the available space around the corresponding point features. Leader lines effectively make it possible to increase such available space by connecting the labels and point features even when they are a little away from each other.

In our approach, several design rules for placing annotation labels are introduced for this purpose. First, for each point feature, we allow an additional number of possible label positions that are a bit apart from the point feature as shown in Figure 3(b). The conventional label placement does not have leader lines as shown in Figure 3(a), which are all in touch with the point feature and commonly employed in geographical information systems (GISs). However, in this case, we often cannot find available labeling space especially in densely annotated regions where the point features are locally concentrated as a cluster because the four possible positions are more likely to be occupied by other labels in the vicinity beforehand. In our approach, in addition to the conventional four positions (Figure 3(b) top), we introduce 12 surrounding label positions (Figure 3(b) bottom) and connect the label and the feature point with a leader line if it can be placed in one of these 12 positions, where the leader line is assumed to be *straight*. Thus, leader lines will help us take advantage of more available space around the respective point features and still maintain the visual correspondence between the labels and the point features.

Please also note that each possible label position is indexed by the ID ranging from 1 to 16 as shown in Figure 3(b), where we employ horizontal leader lines for the positions indexed by 5, 8, 9, and 12, vertical leader lines for the

positions indexed by 6, 7, 10, and 11, and diagonal leader lines for the positions indexed by 13, 14, 15, and 16. Employing horizontal and vertical leader lines for the eight new label positions out of 12 effectively helps avoid unnecessary conflicts with existing annotation labels when composing *scale-aware* consistent label placement. Furthermore, these IDs indicate the order of preference for the 16 label positions at each point feature, which means that the smaller ID corresponds to a better label position. Based on this design principle, our algorithm will search for the position of each label in this preference order to arrange the label as close as possible while maximally keeping it to the top right of the point feature. Of course, this search process is conducted under the assumption that all the labels are placed without mutual overlap within the map domain at the respective scales.

Nonetheless, avoiding overlapping among the labels at multiple map scales inevitably results in inconsistent placement of labels as shown in Figure 1. To solve this problem, we avoid such overlapping across the scale by considering the 3D space spanned by the 2D map domain and scale, by following the conventional ARO formulation (Been et al., 2006, 2010). In cartography, the map scale s often refers to the degree of magnification of the specific map region, which also indicates that the larger *cartographic scale* provides more detailed contents of the map. However, in this paper, we employ the *zooming scale* r (Been et al., 2006), which is defined as the inverse of the cartographic scale s (i.e., $r = 1/s$), as the map scale for later convenience. This implies that the labels tend to easily collide with each other once the zooming scale becomes larger as illustrated in Figure 4(a), and thus, the leader lines provide a powerful means of alleviating such conflicts among the labels. To the best of our knowledge, our work is the first to introduce leader lines in the context of scale-aware consistent labeling based on ARO. The idea permits us to utilize available space around the individual point features and encourages us to improve the associated map readability by maximizing the number of labels placed on the map.

4 Consistent Label Placement with Leader Lines

In this section, we describe how we extended Been et al.'s conventional ARO approach for implementing consistent placement of map labels with their leader lines across the scale. We explain our approach in three subsections: genetic-based optimization of active ranges, occlusion-free placement of labels and leader lines, and fitness evaluation of their spatial layout.

4.1 Genetic-based optimization of active ranges

As described previously, we employ the ARO technique (Been et al., 2006, 2010) to maintain the consistency in the label placement across multiple scales. For this purpose, we introduce genetic algorithms to optimize the placement of pyramidal active range volumes in the 3D space while avoiding unwanted overlap among labels at individual scales. In our formulation, we first encode a specific sequence
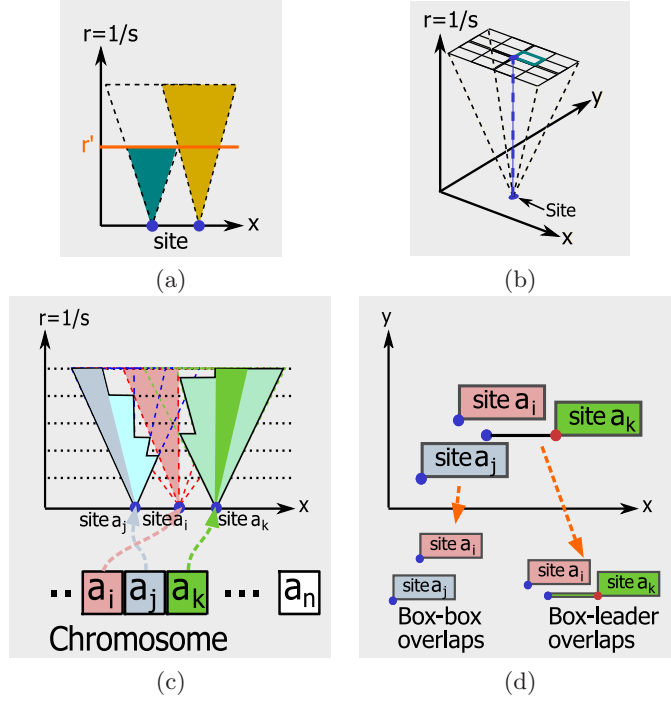
Fig. 4. Active ranges in the 3D space spanned by the map domain (x, y) and the zooming scale r. (a) The cross section of the pyramidal active range volume. (b) The top face of the active range consists of the set of possible label positions, and its center corresponds to the feature point. (c) Greedy placement of active ranges by referring to a specific chromosome. (d) Possible conflicts among labels and leader lines in the active range optimization.

of label IDs as a chromosome, then try to greedily place the active range volumes in this order (see Section 4.2), and finally evaluate the fitness of the chromosome by referring to the resultant label placement (see Section 4.3).

In this optimization, we initialize a set of chromosomes by randomizing the individual sequences of label IDs and prepare it as the population of the first generation. During the optimization phase, we try to update the population with the next generation by evolving the individual chromosomes. This is usually conducted by applying the crossover and mutation operations and replacing the existing chromosomes with superior ones to finally compose a set of elite chromosomes.

4.2 Occlusion-free placement of labels and leader lines

Now we are ready to consider how we can find the actual placement of pyramidal active ranges in the 3D space by referring to chromosomes, each of which encodes an individual order of label IDs. Note that the top face of each pyramidal active

range in the 3D space contains the 16 label positions while the center of the face corresponds to the point feature, as shown in Figure 4(b). Suppose that we currently have a specific order of label IDs as a chromosome and have already finished placing the i-th label a_i, as shown in Figure 4(c). In this case, we first investigate how the pyramidal active range of the next j-th label a_j overlaps with the set of existing active ranges, by properly sampling the map scale. The light blue region indicates the available space for the label to be placed next. Among possible positions of the label, we select the one that has the largest active range in the map scale, which implies that we employ the active range in blue from the available 3D space. Furthermore, we also have to record the 3D space that is already occupied because this is no longer available for the labels to be placed later.

We also have to avoid unexpected overlap between the labels and leader lines when we select one of the newly introduced 12 label positions as shown in Figure 4(d). For this purpose, we approximate the shape of each leader line as a bounding box that encloses the line. For example, we employ a thin rectangle as the bounding box if the leader line is exactly horizontal or vertical. If the leader line is slanted, we insert a normal rectangle as the leader line's bounding box in such a way that the diagonal of the rectangle coincides with the leader line. This inevitably increases the number of bounding boxes we have to maintain at each sampled map scale while effectively facilitates implementation of consistent label placement with leader lines by taking advantage of existing data structures.

4.3   Fitness evaluation of the label placement

The final task is to formulate the fitness function for each chromosome, so that we can systematically improve the set of chromosomes during the genetic-based evolution. In our approach, we employ the following function

$$\{\text{Weight}\} \times (M - \{\text{Label position ID}\}) \times \{\text{Active range}\}, \tag{1}$$

to evaluate the fitness of an individual label. Here, {Weight} is the weight assigned to the label, {Label position ID} is the ID among the 16 label positions, and {Active range} is the size of the corresponding active range on the map scale. Moreover, we set $M = 20$ in Eq. (1) so that we can assign larger fitness values for the smaller label position IDs. We sum up the fitness values for all the labels to evaluate the overall label placement of each chromosome. During the chromosome evolution in the optimization, we maximize this fitness function to find a satisfactory solution for the scale-aware label placement.

5   Results

We employed a dataset that has $3,887$ point features over Japan as a demonstration example. Figure 5 shows several screenshots of label placements around Tokyo in Japan. The results were generated using our genetic-based optimization

Fig. 5. Label placement around Tokyo, Japan. Left: Naïve inconsistent placement. Right: Scale-aware consistent placement. Different colors are assigned to the names according to their administrative levels; the prefecture, city, county, and town names are in red, orange, green, and gray, respectively.

Fig. 6. Label placement around Osaka, Japan. Left: Consistent placement with four label positions. Right: Consistent placement with 16 label positions. Different colors are assigned to the names according to their administrative levels; the prefecture, city, county, and town names are in red, orange, green, and gray, respectively.
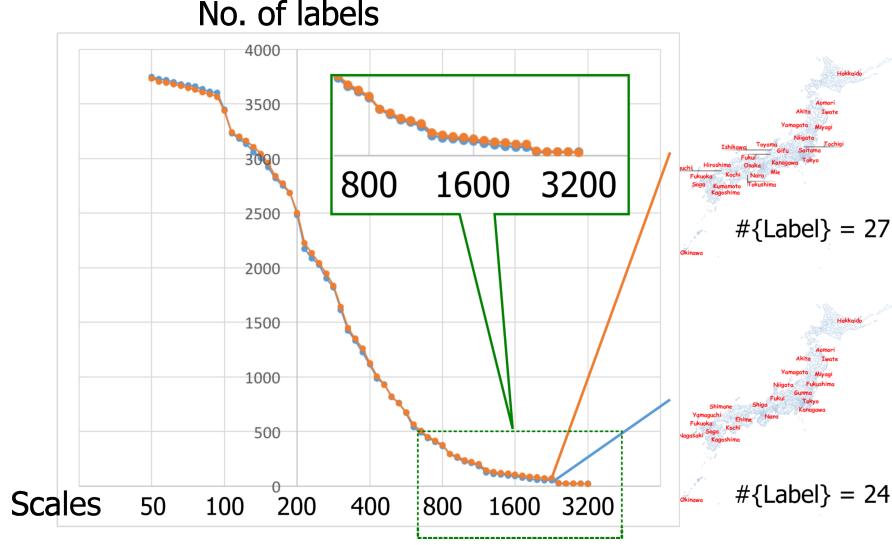
Fig. 7. Comparison between the label placement with the conventional four label positions and the new 16 label positions. The chart summarizes the number of labels that can be embedded in the entire map of Japan (the vertical axis) for the respective scales (i.e., distances of the sides of the map domain (km; the horizontal axis)).

approach, where from top to bottom we zoomed in on the map by monotonously reducing the zooming scale. Note that the images on the left were generated using a naïve approach, which optimized the label placement individually at the respective scales and thus could increase the number of visual labels at the cost of inconsistency in the layout of labels and leader lines across multiple scales. However, the image on the right reveal scale-aware consistent placement of labels synthesized using our approach, which successfully avoids the unexpected pop-up effects of labels and leader lines and retains their positions and drawing styles. For example, the text label "Chosei" (in green on the bottom right of the map), a county name, changes its position in the naïve inconsistent label placement (on the left of Figure 5), while it naturally appears as the zooming scale decreases in the consistent label placement (on the right of Figure 5). The consistent placement is much more reasonable because labels with low priority naturally disappear as we zoom out on the map. Note that here we draw the leader lines together with the corner of the corresponding label boxes by following the styles in Meng et al. (2015).

Figure 6 presents the optimal label placement around Osaka in Japan, where the left and right columns show the label placement using the conventional four label positions and the proposed 16 label positions, respectively. Here, both placements avoid unwanted popping effects and sudden leaps of the labels and successfully maintain consistency across successive scales. Nonetheless, the results in the right column clearly demonstrate that our approach maximally uti-

lizes the space outside the boundary of Japan for accommodating more labels to be placed especially in densely annotated regions. Furthermore, our new approach allows us to avoid distracting conflicts between the labels and the leader lines while keeping the overall layout as tight as possible.

Figure 7 shows the comparison between the results using the conventional four label positions and the proposed 16 label positions, where we summarize the numbers of labels that can be embedded in the entire map of Japan at the respective scales. Note that the horizontal axis of this chart represents the approximate distance of each side of the map square domain, which is proportional to the power of the corresponding scale. The comparison shows that we can increase the number of labels placed on the map especially when the zooming scale is large, to optimize the use of available map space when we zoom out on the map.

6 Conclusion and Future Work

In this paper, we presented an approach for consistently placing labels across multiple scales by introducing leader lines for the conventional active range optimization problem. The approach allows us to incorporate more annotation labels into the map domain without unwanted artifacts, such as overlapping among labels and distracting popping effects across multiple scales. The visibility of the annotation labels is maximized using genetic algorithms by optimizing the order of the labels in which they are embedded in the map domain in a greedy fashion.

Our future directions include accelerating our genetic-based computation by grouping labels into several clusters of reasonable size, as well as combining them together to form the entire label placement. Other potential extensions include investigating additional aesthetic constraints for the label placement in terms of country borders, administrative districts, ponds and rivers, as well as the layout of transportation systems.

Acknowledgments

References

Been, K., Daiches, E., & Yap, C. (2006). Dynamic map labeling. *IEEE Transactions on Visualization and Computer Graphics*, *12*(5), 773–780.

Been, K., Nöllenburg, M., Poon, S.-H., & Wolff, A. (2010). Optimizing active ranges for consistent dynamic map labeling. *Computational Geometry: Theory and Applications*, *43*(3), 312–328.

Bekos, M. A., Cornelsen, S., Fink, M., Hong, S.-H., Kaufmann, M., Nöllenburg, M., ... Symvonis, A. (2015). Many-to-one boundary labeling with backbones. *Journal of Graph Algorithms and Applications*, *19*(3).

Bekos, M. A., Kaufmann, M., Symvonis, A., & Wolff, A. (2007). Boundary labeling: Models and efficient algorithms for rectangular maps. *Computational Geometry: Theory and Applications*, *36*(3), 215–236.

Chiang, Y.-Y., Leyk, S., & Knoblock, C. A. (2014). A survey of digital map processing techniques. *ACM Computing Surveys*, *47*(1), 1:1–1:44.

Christensen, J., Marks, J., & Shieber, S. (1995). An empirical study of algorithms for point-feature label placement. *ACM Transactions on Graphics*, *14*(3), 203–232.

Edmondson, S., Christensen, J., Marks, J., & Shieber, S. (1997). A general cartographic labeling algorithm. *Cartographica*, *33*(4), 13–23.

Fekete, J.-D., & Plaisant, C. (1999). Excentric labeling: Dynamic neighborhood labeling for data visualization. In *Proceedings of the sigchi conference on human factors in computing systems* (pp. 512–519).

Fink, M., Haunert, J., Schulz, A., Spoerhase, J., & Wolff, A. (2012). Algorithms for labeling focus regions. *IEEE Transactions on Visualization and Computer Graphics*, *18*(12), 2583–2592.

Gemsa, A., Niedermann, B., & Nöllenburg, M. (2013). Trajectory-based dynamic map labeling. In *Proceedings of the 24th international symposium on algorithms and computation (isaac 2013)* (pp. 413–423).

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

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

Lin, C.-C. (2010). Crossing-free many-to-one boundary labeling with hyperleaders. In *Proceedings of 2nd ieee pacific visualization symposium (pacificvis 2010)* (pp. 185–192).

Luboschik, M., Schumann, H., & Cords, H. (2008). Particle-based labeling: Fast point-feature labeling without obscuring other visual features. *IEEE Transactions on Visualization and Computer Graphics*, *14*(6), 1237-1244.

MacEachren, A. M. (1995). *How maps work*. The Guilford Press.

Meng, Y., Zhang, H., Liu, M., & Liu, S. (2015). Clutter-aware label layout. In *Proceedings of the 7th ieee pacific visualization symposium (pacificvis 2015)* (p. 207-214).

Petzold, I., Gröger, G., & Plümer, L. (2003). Fast screen map labeling—data-structures and algorithms. In *Proceedings of the 23rd international cartographic conference (icc 2003)* (pp. 288–298).

Poon, S.-H., & Shin, C.-S. (2005). Adaptive zooming in point set labeling. In *Proceedings of the 15th fundamentals of computation theory (fct 2005)* (pp. 233–244).

Raidl, G. R. (1998). A genetic algorithm for labeling point features. In *Proceedings of the international conference on imaging science, systems and technology* (pp. 189–196).

Schwartges, N., Haunert, J., Wolff, A., & Zwiebler, D. (2014). Point labeling with sliding labels in interactive maps. In *Proceedings of the 17th agile conference on geographic information science* (pp. 295–310).

Wolff, A. (2009). *The map-labeling bibliography.* (http://i11www.iti.uni-karlsruhe.de/map-labeling/bibliography/)

Wu, H.-Y., Takahashi, S., Lin, C.-C., & Yen, H.-C. (2011). A zone-based approach for placing annotation labels on metro maps. In *Proceedings of the 11th international conference on smart graphics (sg 2011)* (Vol. 6815, pp. 91–102).

Yoeli, P. (1972). The logic of automated map lettering. *The Cartographic Journal*, *9*(2), 99-108.

Zhang, X., Poon, S.-H., Li, M., & C.S.Lee, V. (2015). On maxmin active range problem forweighted consistent dynamic map labeling. In *Proceedings of the 7th international conference on advanced geographic information systems, applications, and services* (pp. 32–37).

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