Voronoi-Based Label Placement for Metro Maps

Hsiang-Yun Wu[1] * Shigeo Takahashi[1] † Chun-Cheng Lin[2] ‡ Hsu-Chun Yen[3] §

[1]Graduate School of Frontier Sciences, The University of Tokyo, Japan

[2]Department of Industrial Engineering and Management, National Chiao Tung University, Taiwan

[3]Department of Electrical Engineering, National Taiwan University, Taiwan

ABSTRACT

Metro maps with thumbnail photographs serve as common travel guides for providing sufficient information to meet the requirements of travelers in the cities. However, conventional methods attempt to minimize the total distance between stations and labels while maximizing the number of the labels rather than further taking into account the overall balance of the spatial distribution of labels. This paper presents an entropy-based approach for effectively annotating large annotation labels sufficiently close to the metro stations. Our idea is to decompose the entire labeling space intro regions bounded by the metro lines, and then further partition each region into Voronoi cells, each of which is reserved for a station to be annotated. This is accomplished by incorporating a new genetic-based optimization, while the fitness of the decomposition is evaluated by the entropy of the relative coverage ratios of such Voronoi cells. We also include several design examples to demonstrate that the proposed approach successfully distributes large labels around the metro network with minimal user intervention.

1 INTRODUCTION

Placing image annotation labels on metro maps is one of the promising ways to enhance the readability of the map contents especially for guiding travelers in transit networks of unknown cities. However, such annotated metro maps including those published in commercially available guidebooks have been manually designed by map illustrators, and cannot usually meet specific requirements demanded by each traveler. Automatically annotating user-specified landmark stations would facilitate interactive customization of annotated metro maps as needed. Although a variety of annotation techniques are studied so far for this purpose, it is still difficult to place annotation labels sufficiently close to the landmark stations while maximally enhancing their space coverage of the map domain.

The key challenge for achieving such well-balanced distribution of annotation labels is to spare enough space for the labels around the metro network as uniformly as possible while minimizing the distances between the labels and their corresponding stations. However, this usually incurs space competition between labels, especially in the central downtown area of the city, where densely arranged metro lines intersect with each other at many interchange stations. In practice, conventional approaches only focus on placing annotation labels close to their corresponding landmarks in a greedy fashion, thus usually results in a waste of labeling space since prepositioned labels are likely to kick out other labels even when their corresponding stations are near by. Effectively embedding image annotation labels onto available space has become an important technical problem especially for designing useful travel maps because they can provide travelers with more information about the transportation networks in the city.

This paper presents an approach for effectively embedding a maximum number of labels in proportion to the labeling area that is bounded by the metro lines. Our idea is to first decompose the map domain into several labeling areas bounded by metro lines, and then further partition the labeling area into a smaller cell for each landmark station, by employing a variant of weighted Voronoi tessellation with respect to the positions of such landmark stations. This is accomplished by incorporating genetic-based optimization that seeks an optimal distribution of labeling cells for annotation labels within the available labeling space, while employing an entropy-based measure to evaluate the optimality of such labeling space distribution. Our approach aligns the position of labels by referring to a regular grid pattern, and thus allows us to seek the best compromise between space coverage ratio of the annotation labels and their aesthetic layout over the map domain. We can also interactively edit the placement of annotation labels by turning on or off the annotation at each station using our prototype system.

Our design scenario can be described as follows: First of all, we distribute annotation labels for a set of selected stations randomly to the labeling areas confined by the metro lines. We then compute the Voronoi tessellation of such labeling areas by maximizing the entropy of the space distribution through genetic-based optimization. Finally, we customize the layout of annotation labels according to our preference with minimal user intervention, in order to finalized the design of annotated metro maps.

The remainder of this paper is structured as follows: Section 2 briefly summarizes previous approaches related to ours. Section 3 presents our entropy-based measure for evaluating the uniformity of labeling space distribution and then describes how we can optimize the entropy using genetic algorithms. Several design examples are provided in Section 4, which is followed by conclusion and future work in Section 5.

2 RELATED WORK

Incorporating image labels to metro maps is a common strategy for guiding travelers in cities [1]. However, conventional maps such as topological and schematic metro maps [10, 15] only provide names and general information of the stations, and thus limit the expressive power of the maps even when they are customized according to the user preference. Recently, image annotation has allowed travelers to emphasize important landmarks on the maps, which significantly improves the readability of the map content from a perceptual point of view [1].

The annotation problems contain two main topics, including the internal and external labeling problems. *Internal labeling* facilitates us to provide sufficient information especially in cartographic maps, by placing textual annotations closely to the landmark points called *sites* without overlaps with other landmarks. Christensen et al. [7] presented an empirical survey comparing several heuristic algorithms for internal labeling.

On the other hand, *external labeling* is another method for assigning relatively large-sized labels to their corresponding sites,

*e-mail: yun@visual.k.u-tokyo.ac.jp

†e-mail: takahashis@acm.org

‡e-mail: cclin321@nctu.edu.tw

§e-mail: yen@cc.ee.ntu.edu.tw

where each label is connected to its site through a line called *leader* in order to clarify the correspondence between the sites and labels. *Boundary labeling* [5] is a representative technique for the external labeling problems. Indeed, the problems for boundary labeling have often been mathematically formulated as constrained optimization problems, where the annotation labels are constrained to be placed outside the rectangular map domain.

Many extended techniques have been introduced for solving variations of this type of problem, including many-to-one correspondence [14, 13], non-straight line leader styles [3, 13, 19], and so forth. Employing both internal and external annotation techniques leads to another challenging problem while provides an effective means of annotating various types of maps and illustrations. Hartmann et al. [11] extracted several design metrics for functional requirements on label placement, as well as aesthetic constraints for guaranteeing the readability of the text annotations. Bekos et al. [4] also combined traditional internal labeling techniques together with boundary labeling, while taking into account possible restrictions on the marginal space for external label placement. Another technique known as *excentric labeling* [8, 6, 9] is an external labeling method that allows users to dynamically annotate labels around a movable lens.

The most relevant work was done by Wu et al. [18], which allocated available spaces individually for internal and external labels by partitioning the map domain according to the distance from the metro lines on the map. The proposed approach can be considered as a variant of this method [18], in the sense that we try to predefine almost equally distributed space for each label in the vicinity of its corresponding landmark station, and then actually place relatively large annotation label to fit its allocated space.

3 Entropy-Based Label Placement

Having the layout of a metro network as input (see Figure 1(a)), we can notice that a small labeling area is often fully confined by the metro lines as a closed face. This inspires us to traverse all the closed and open faces of the entire metro network initially, and then fit large annotation labels compactly to the available labeling areas in order to associate sufficient information such as text and images with landmark stations. Here, we first assign an annotation label to one of the labeling areas separated by the metro network in the map domain, and then partition the area into Voronoi-like cells so that we can prepare the space for uniquely placing each label. In our approach, we find a spatially efficient placement of annotation labels by defining an entropy-based measure for evaluating the uniformity of labeling space distribution and then optimizing it using genetic algorithms.

3.1 Label distribution

As mentioned earlier, the placement of lines in a metro network defines a group of several closed and open faces, which can be considered as available areas for placing annotation labels. As for the closed faces, we can just employ them as labeling areas without any modification. On the other hand, the open faces should be extended to the boundary of the map domain so that they can also cover regions around the boundary. This is done by drawing a virtual line segment from every dead-end vertex of the metro network to the boundary, which allows us to partition the marginal labeling space into several labeling areas. Figure 1(b) shows such an example where each partitioned labeling area is rendered in a different color. Here, we align each virtual line segment to one of the eight octilinear directions by approximating the orientation of the vector emanating from the center of the map domain to that dead-end vertex. This successfully allows us to introduce a radial pattern to partition the map domain into labeling areas.

In our setup, we assume that a subset of vertices is selected as important landmarks, each of which is associated with one annotation label beforehand. We then greedily assign every annotation label to one of the labeling areas that are incident to the corresponding vertex. This is because we want to place the annotation label in the vicinity of that vertex so that we can fully minimize the length of the leader that connects the label and vertex. For example, a degree-two vertex has two incident labeling areas and thus its label can be assigned to one of the two areas, while a vertex at which two metro lines intersect with each other enjoys four incident areas. We will optimize the distribution of annotation labels later in order to achieve uniform allocation of labeling spaces to the selected landmark vertices.

3.2 Partitioning labeling areas into Voronoi cells

Suppose that we have already distributed annotation labels to the labeling areas surrounding the metro network. We then partition each labeling area appropriately so that we can spare a smaller segmented region uniquely for the label of each selected vertex. This means that we have to divide a labeling area into multiple cells if it has more than one assigned label. In our approach, we compute the Voronoi tessellation within each labeling area and assign each Voronoi cell to its corresponding landmark vertex (see Figures 1(c) and (e)).

Nonetheless, one exception to this is a dead-end vertex of the metro network, where its label is associated with the combination of its two incident labeling areas in our implementation. We use this scheme because we would like to make the labels of the dead-end vertices to stay along the border between adjacent labeling area so as to not disturb cells allocated for other vertices. For this purpose, we compute a weighted Voronoi partitioning in each labeling area, where we assign a smaller weight to the dead-end vertices so as to suppress their influence on the segmentation of the map domain. Here, the weights for the dead-end vertices are empirically set to be 0.6 while other weights are 1.0. Figure 1(c) exhibits such a variant of Voronoi partitioning of the map domain where the Voronoi cells are rendered in different colors. Note that the figure also reveals that segmented cells assigned to dead-end vertices stay around the boundary of the map domain.

In our implementation, this labeling area partitioning is accomplished by employing a hardware-assisted algorithm for computing weighted Voronoi diagrams [12]. For rendering the Voronoi tessellation, we draw a 3D cone in a different color while its top coincides with the position of the corresponding selected vertex. Within each labeling area, in practice, we use the stencil buffer to apply a binary mask to limit our Voronoi partitioning of the map domain. We then evaluate the smallest depth value at each pixel in order to identify which Voronoi cell covers that pixel by referring to the corresponding pixel color. This computation of labeling area partitioning is quite fast and thus useful especially for interactive editing of annotation label placement.

3.3 Entropy-based fitness function

Before optimizing the distribution of annotation labels to the labeling areas confined by the metro network, we have to find an appropriate objective function that evaluates the fitness of map domain partitioning. Since we would like to maximize the number of annotation labels to be embedded into the available labeling space, we try to assign segmented cells to the landmark vertices as uniformly as possible. In our approach, we introduce the Shannon entropy measure to evaluate such uniformity of the labeling space allocation.

As described earlier, we can render segmented cells for the annotation labels in different colors in our implementation. Thus, we can calculate the relative coverage ratio of each cell with respect to the entire map domain easily by counting the number of pixels having the corresponding color in the color buffer. This is inspired by the computation of viewpoint entropy measure [16], where the pro-

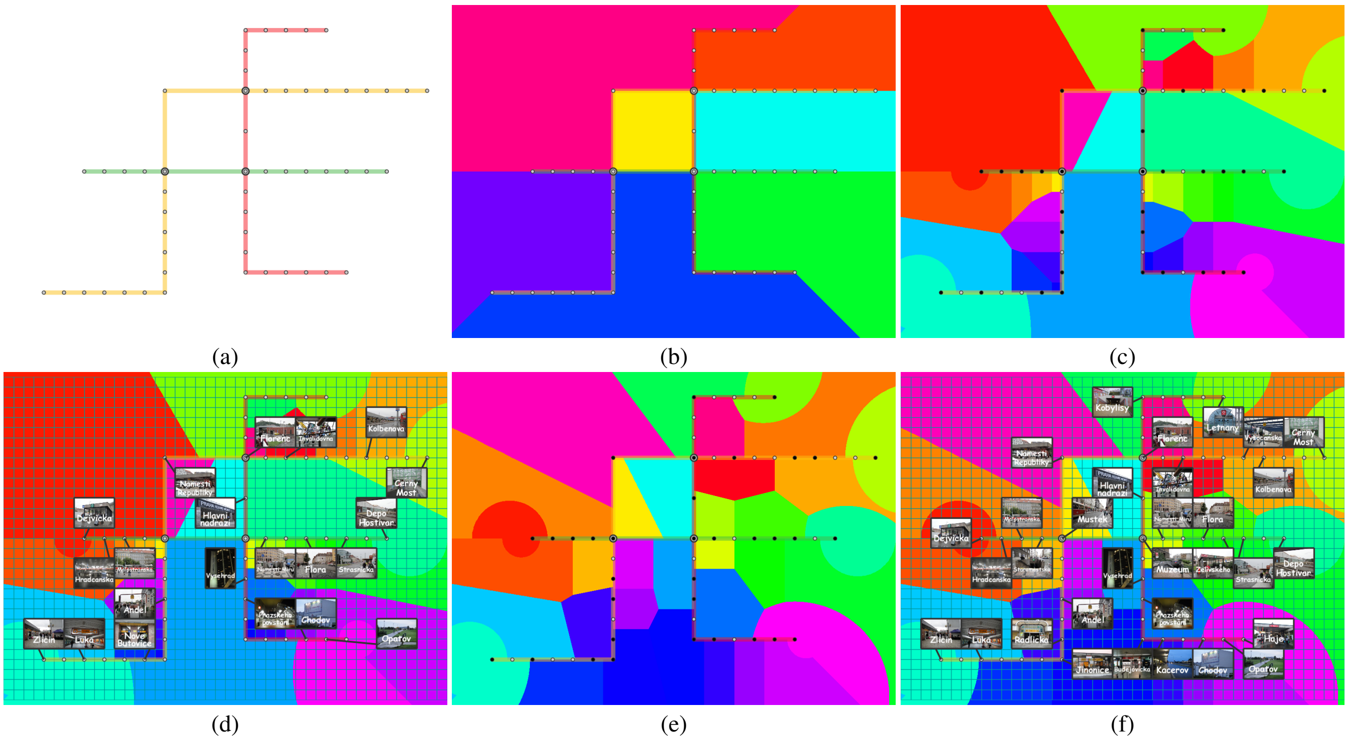


Figure 1: Partitioning the map domain for embedding annotation labels in Prague metro maps. (a) Original input layout and (b) its labeling areas. (c) Voronoi-like partitioning with landmark stations marked in black and (d) its corresponding label placement. (e) Optimized partitioning with landmark stations marked in black and (f) its corresponding label placement.

jected area of each face is computed by summing up the pixels that belong to that face [2]. Considering the relative coverage ratios as the probabilities of the segmented cells provides us with the means of evaluating the uniformity of the labeling space distribution using the Shannon entropy formulation. Let us denote by p_i the area coverage ratio of the cell for the i-th label with respect to the entire map domain, where $i \in \mathbb{N}$. We can define the entropy-based measure E as

$$E = \sum_{i \in \mathbb{N}} (-p_i \log p_i). \tag{1}$$

This measure enables us to evaluate the uniformity of the labeling space distribution by referring to the color buffer, once we can render the Voronoi cells over the map domain.

3.4 Genetic-based optimization

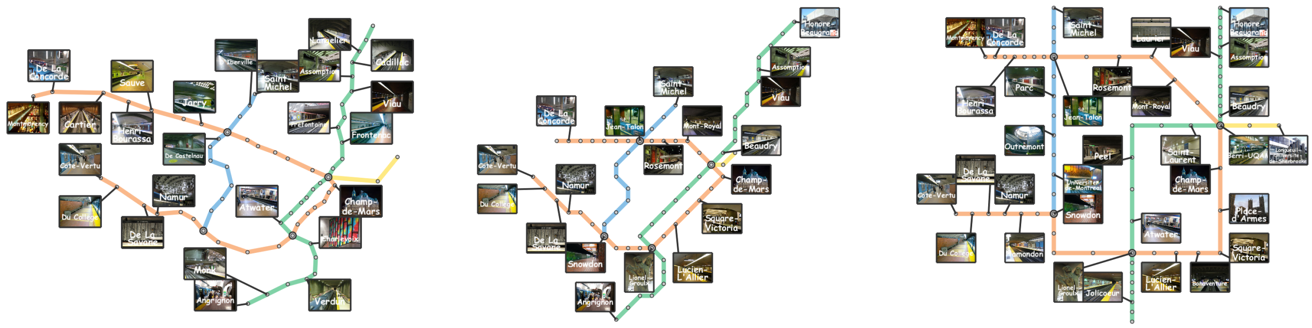
With the above objective function, we are now ready to optimize the distribution of annotation labels to the labeling areas around the metro network. This is accomplished by employing the genetic algorithm where we encode the ID of the labeling area assigned to each selected vertex as a gene, while a sequence of genes now represents how we distributed the annotation labels of the station vertices to the labeling area bounded by the metro lines. We then apply conventional crossover and mutation operations to the population of chromosomes and constitute an improved population as the next generation by evaluating the objective function in Eq. (1). This process of evolving the population is repeated a predefined number of times in order to find a fully optimized chromosome, which leads us to the best partitioning of the map domain for placing annotation labels. This genetic-based optimization is justified due to the fact that we can retain good local sequences of label distributions if we reflect the choice of labeling areas along a metro line in the local sequence of genes in a chromosome.

Note that when applying the mutation operations, some gene is randomly selected from the chromosome and replaced with another alternative. Here, the set of alternatives should be the IDs of labeling areas incident to the corresponding landmark vertex. This means that we have to first collect the IDs of incident labeling areas for each station vertex, and then choose one of them as the gene of that vertex in the encoded chromosome. From our experiments, we set the probability of the mutation operation to be 0.02 to appropriately replace each gene in the chromosome while that of the crossover operation to be 0.05 to fully retain good local sequences of genes. This genetic-based optimization successfully gives us a well-balanced partitioning of the map domain, and thus we can place more annotation labels by finding sufficient labeling space for each of them. Figure 1(e) shows the map domain partitioning obtained through the genetic-based optimization.

3.5 Greedy label placement

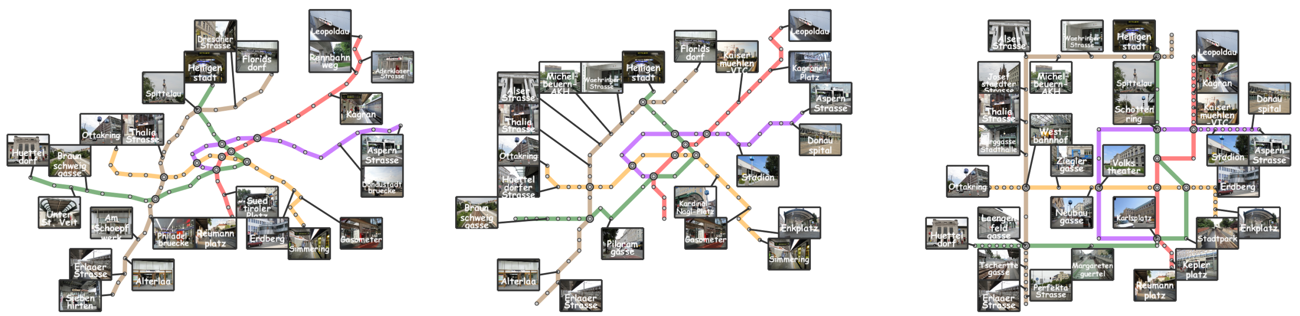
When embedding annotation labels within the map domain, we employ a regular grid to guide the position of annotation labels, in a similar way to [18]. This successfully limits the degrees of freedom in optimizing the positions of annotation labels in a way that the labels are more likely to be consistently aligned if they are close to each other.

During the actual placement of annotation labels, we employ a two-step greedy approach: We first compute acceptable positions for each label on the condition that the label is completely contained in the corresponding segmented cell. The optimal position is then selected among them in the sense that the corresponding leader length is minimal. After having searched for as many as possible positions of annotation labels, we delve into the second greedy step where we relax the condition so that we can allow the labels to partially overlap with their corresponding segmented cells. This additional search for the label positions effectively lets us fill in the



(a) (b) (c)

Figure 2: Montreal annotated metro maps based on (a) geographical layout, (b) octilinear layout, and (c) nearly-orthogonal layout with our approach, respectively.



(a) (b) (c)

Figure 3: Vienna annotated metro maps based on (a) geographical layout, (b) octilinear layout, and (c) nearly-orthogonal layout with our approach, respectively.

gaps between the labels that were already embedded in the previous step. Figure 1(f) presents a final annotated metro map where the label placement is conducted with the optimized distribution of annotation labels. As compared with the result in Figure 1(d) with a random distribution of annotation labels, we can significantly increase the number of labels that can be embedded in the labeling area around the metro network while preserving their closeness to the corresponding landmark vertices.

We again equip our design system with an interface for interactively adjusting the arrangement of the annotation labels for fine tuning. The system also enables us to freely activate or deactivate the label annotation of each vertex and change the labeling area to which the vertex belongs to, by clicking the corresponding station vertices on the map.

4 Results and Discussion

Our prototype system was implemented on a desktop PC with two Quad-Core Intel Xeon CPUs (2.4GHz, 12MB cache) and 8GB RAM, and the source code has been written in C++ using OpenGL for drawing metro layouts, and OpenCV for handling images. The implementation of the genetic-based computation in our system has been based on the GAlib package [17].

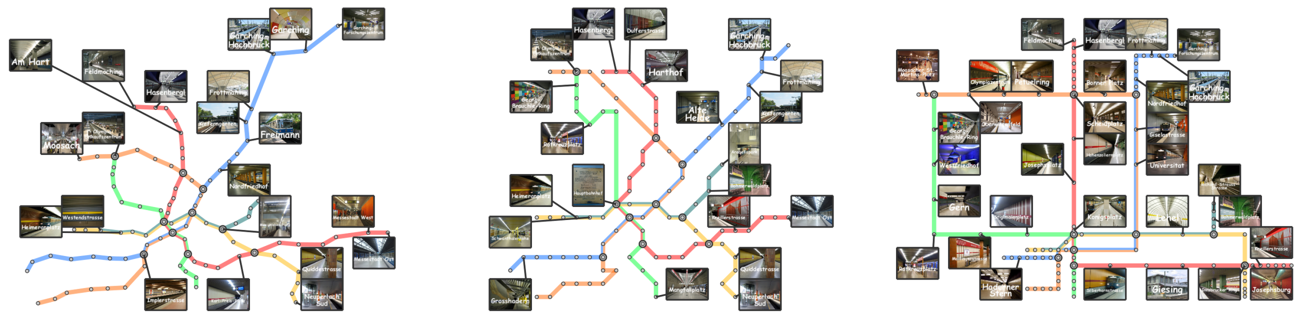
Figure 2 represent several design examples of annotated Montreal metro maps. Figures 2(a)-(c) correspond to the geographical, octilinear, and nearly-orthogonal layouts, while more annotation labels are embedded effectively if the shape of the layout is close to an orthogonal structure. All these maps are annotated by optimizing our entropy-based measure for evaluating the uniform distribution

Table 1: Statistics on label placement. Length indicates the average leader length in pixel units, while #{L} represents the number of labels embedded in the entire domain for each metro map.

City	Geographical		Octilinear		Nearly-orthogonal	
	Length	#{L}	Length	#{L}	Length	#{ L}
Prague	57.70	21	38.44	21	28.26	33
Montreal	36.90	25	50.30	19	23.07	33
Vienna	44.06	25	47.50	23	25.63	31
Munich	50.75	21	42.88	21	21.94	27

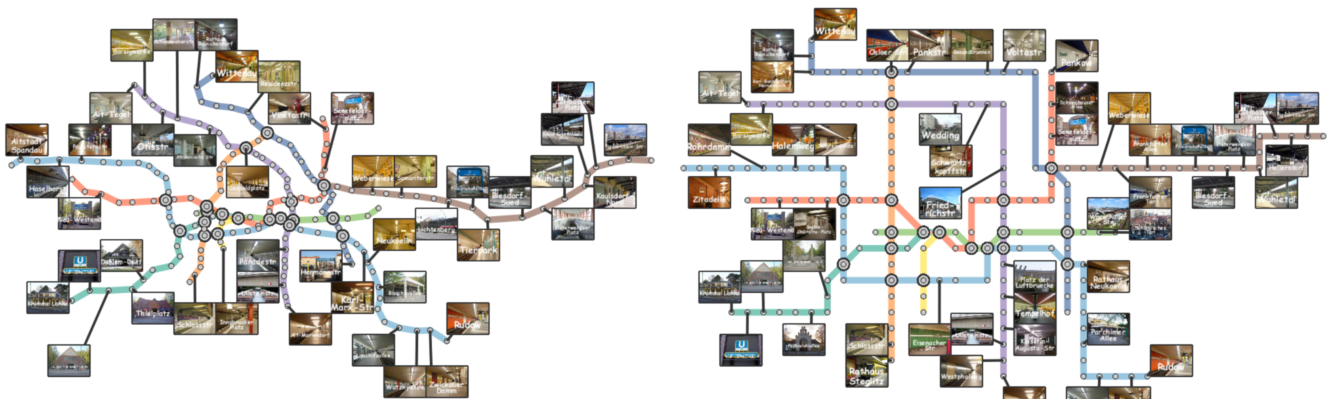
of Voronoi labeling cells. Figures 3 and 4 show annotated maps of Vienna and Munich metro networks, respectively. Statistics on label placement are shown in Table 1, where each entry corresponds to a pair consisting of the average leader length in pixel units, and the number of embedded labels in the entire map domain. Figure 5 provides a comparison between the annotated maps of geographical and nearly-orthogonal layouts of Berlin metro, which has 170 stations in the network system. Figure 6 shows another comparison with the convetional approach for greedy placement of annotation labels [18], which clarifies that our approach makes it possible to embed a larger number of annotation labels into the map domain while effectively minimizing the associated total leader lengths. Note that computation times for generating these design examples are all within 1 minute.

Although our approach provides powerful means of evenly allocating labeling space in the areas, its effectiveness generally de-



(a) (b) (c)

Figure 4: Munich annotated metro maps based on (a) geographical layout, (b) octilinear layout, and (c) nearly-orthogonal layout with our approach, respectively.



(a) (b)

Figure 5: Berlin annotated metro maps based on (a) geographical layout and (b) nearly-orthogonal layout with our approach, respectively.

pends on the geometry of the target metro network. As shown in Table 1, more labels can be embedded if the input layout has more orthogonal edges than diagonal ones. We can also increase the coverage ratio of the annotation labels by adjusting this grid interval, while excessively reducing the grid interval results in too small annotated labels (see Figure 7). The systematic control of the grid interval together with the size of annotation labels will be another important factor to improve the aesthetics of the annotated map layout.

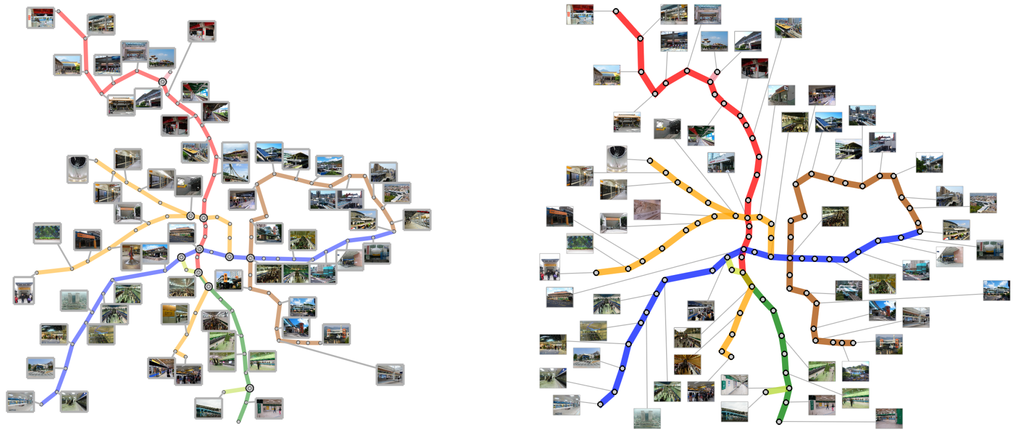
We also collected feedback of our prototype system both from ordinary map users and cartographers. Ordinary map users were basically satisfied with the performance of our system in that they could effectively allocate additional space for placing labels with minimal user intervention. They further reported that the entropy-based optimization significantly improved the compactness of the label arrangement and fully minimized the associated leader lengths. On the other hand, cartographers highly evaluated an interface for activating/deactivating label annotation and changing the choice of incident labeling areas for each station vertex in our system. Nonetheless, they also requested that they can newly create a sufficient space for placing labels for a specific set of station vertices especially when they are involved in the dense downtown areas. A systematic way for applying an appropriate deformation to the layout of the metro network helps us to spare such additional space for placing annotation labels, which has been left as future work.

5 Conclusion

This paper has presented an approach to designing well-balanced distribution of annotation labels on a metro map while fully minimizing the associated total leader length. A variant of Voronoi tessellation has been introduced to partition the entire map domain into small cells, each of which is prepared for placing an annotation label of the corresponding landmark station. Such arrangement of Voronoi cells has been also elaborated through the genetic-based optimization, where the area distribution to each label is adequately equalized by maximizing the associated Shannon entropy measure of the relative coverage ratios of the cells. We also provide an interface for interactively editing annotation labels so that the users can design their own customized metro maps according to their preference. Our future work includes the incorporation of a more powerful interface and the implementation on mobile devices, as well as the development of hybrid design of the network layout itself together with improved placement of annotation labels.

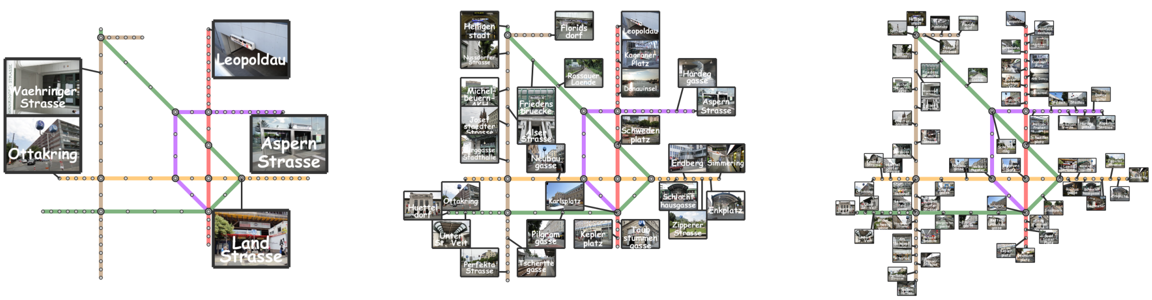
Acknowledgements

This work has been partially supported by JSPS under Grants-in-Aid for Scientific Research (B) No. 24330033 and challenging Ex-



(a) (b)

Figure 6: Taipei annotated metro maps with (a) the proposed approach (67 image labels) and (b) the previous approach [18] (62 image labels).



(a) (b) (c)

Figure 7: Vienna annotated metro maps based on (a) large, (b) middle, and (c) small grid interval with our approach, respectively.

ploratory Research No. 23650042, and NSC 101-2628-E-009-025-MY3 and NSC 100-2221-E-002-132-MY3, Taiwan.

References

[1] Visitors guide and bus route map of London. Transport for London, 2012. http://www.tfl.gov.uk/assets/downloads/visitor-guide.pdf.

[2] P. Barral, G. Dorme, and D. Plemenos. Scene understanding techniques using a virtual camera. In *Proc. Eurographics '00 Short Presentations*, 2000.

[3] M. A. Bekos, M. Kaufmann, M. Nöllenburg, and A. Symvonis. Boundary labeling with octilinear leaders. *Algorithmica*, 57(3):436–461, 2010.

[4] M. A. Bekos, M. Kaufmann, D. Papadopoulos, and A. Symvonis. Combining traditional map labeling with boundary labeling. In *Proc. 37th International Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM2011)*, volume 6543 of *Springer LNCS*, pages 111–122, 2011.

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

[6] E. Bertini, M. Rigamonti, and D. Lalanne. Extended excentric labeling. *Computer Graphics Forum*, 28(3):927–934, 2009.

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

[8] J.-D. Fekete and C. Plaisant. Excentric labeling: Dynamic neighborhood labeling for data visualization. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems (CHI 1999)*, pages 512–519, 1999.

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

[10] K. Garland. *Mr. Beck's Underground Map: A History*. Capital Transport, 1994.

[11] K. Hartmann, T. Gotzelmann, K. Ali, and T. Strothotte. Metrics for functional and aesthetic label layouts. In *Proc. 5th International Symposium on Smart Graphics (SG2005)*, volume 3638 of *Springer LNCS*, pages 115–126, 2005.

[12] K. E. Hoff III, T. Culver, J. Keyser, M. Lin, and D. Manocha. Fast computation of generalized voronoi diagrams using graphics hardware. In *Proc. SIGGRAPH '99*, pages 277–286, 1999.

[13] C.-C. Lin. Crossing-free many-to-one boundary labeling with hyperleaders. In *Proc. IEEE Pacific Visualization Symposium 2010 (PacificVis 2010)*, pages 185–192, 2010.

[14] C.-C. Lin, H.-J. Kao, and H.-C. Yen. Many-to-one boundary labeling. *Journal of Graph Algorithm and Appplications*, 12(3):319–356, 2008.

[15] M. J. Roberts. *Underground Maps After Beck*. Capital Transport, 2003.

[16] P.-P. Vázquez, M. Feixas, M. Sbert, and W. Heidrich. Viewpoint selection using view entropy. In *Proc. Vision Modeling and Visualization Conference (VMV01)*, pages 273–280, 2001.

[17] M. Wall. GAlib: A C++ library of genetic algorithm components. http://lancet.mit.edu/ga/.

[18] H.-Y. Wu, S. Takahashi, C.-C. Lin, and H.-C. Yen. A zone-based approach for placing annotation labels on metro maps. In *Proc. 11th International Conference on Smart Graphics (SG2011)*, volume 6815 of *Springer LNCS*, pages 91–102, 2011.

[19] H.-Y. Wu, S. Takahashi, C.-C. Lin, and H.-C. Yen. Travel-route-centered metro map layout and annotation. *Computer Graphics Forum*, 31(3):925–934, 2012.