# Scale-Aware Cartographic Displacement Based on Constrained Optimization

Ken Maruyama*, Shigeo Takahashi*, Hsiang-Yun Wu†, Kazuo Misue‡, and Masatoshi Arikawa§
* University of Aizu, Aizu-Wakamatsu, Japan
† TU Wien, Vienna, Austria
‡ University of Tsukuba, Tsukuba, Japan
§ Akita University, Akita, Japan

***Abstract*—The consistent arrangement of map features in accordance with the map scale has recently been technically important in digital cartographic generalization. This is primarily due to the recent demand for informative mapping systems, especially for use in smartphones and tablets. However, such sophisticated generalization has usually been conducted manually by expert cartographers and thus results in a time-consuming and error-prone process. In this paper, we focus on the displacement process within cartographic generalization and formulate them as a constrained optimization problem to provide an associated algorithm implementation and its effective solution. We first identify the underlying spatial relationships among map features, such as points and lines, on each map scale as constraints and optimize the cost function that penalizes excessive displacement of the map features in terms of the map scale. Several examples are also provided to demonstrate that the proposed approach allows us to maintain consistent mapping regardless of changes to the map scale.**

***Index Terms*—Cartographic generalization, displacement, constrained optimization, scale-aware mapping**

## I. Introduction

Recent development of mobile devices such as smartphones and tablets has facilitated the efficient exploration of digital map contents. With such mobile devices, users interactively change the map scale to focus their attention on areas of interest with various map scales according to their preference. In general, reducing the map scale inevitably concentrates existing map features such as roads and buildings within a small map area, while their minimal size should be retained. Figure 1 shows such an example, in which the roads and buildings are more likely to overlap with each other as the scale decreases, because we retain the width of the road with respect to the size of the map domain. Solving this problem usually requires a process of properly displacing the map features to avoid mutual geometric conflicts.

In practice, this *displacement* process plays an important role in so-called *cartographic generalization*, which allows us to adjust the configuration of map contents in accordance with the map scale. Cartographic generalization often poses technically involved problems in conventional map editing. In the process, experienced cartographers manually adjust the positions of map features to avoid visual clutter arising from their geometric conflicts, especially when they are too crowded in a limited space. This always results in a tedious and time-consuming task that is prone to error. Automating this task enhances the readability of the digital map contents on mobile devices and accelerates the manual process of editing maps on various scales.

This paper presents a novel approach to automating displacement on arbitrary map scales to ensure an aesthetic configuration of map components. In this approach, the displacement process is formulated as a constrained optimization problem, by taking advantage of linear programming techniques. Our formulation employs several aesthetic criteria as constraints for preserving the appearance of the map content, and minimizes the displacement and distortion of map features. We also develop the consistent selection of map features with respect to the map scale, in such a way that the map features avoid having unexpected pop-ups and sudden changes in the positions across the scale. Several experimental results will also be included to demonstrate the feasibility of our proposed approach.

Our technical contributions are summarized as follows:

- The displacement process is formulated as a constrained optimization problem in which hard and soft constraints are explicitly formulated.
- The entire map layout is optimized by minimizing the displacement and distortion of the associated map features while keeping their relative positions.

The remainder of this paper is organized as follows. Section II surveys the previous work related to the proposed approach. Section III provides an overview of our approach. Our major technical contributions are described in the subsequent three sections. Computing the optimal displacement of map features on individual scales is detailed in Section IV, while controlling the consistent placement of the map features in accordance with the map scales is described in Section V. After having presented several experimental results together with the discussion on the proposed approach in Section VI, we conclude this paper together with possible future extensions of our work in Section VII.

## II. Related Work

Cartographic generalization has been considered as a process of extracting important map features including lines and boundaries while reducing minor ones, selecting and combining map features, schematizing the shapes of features as their simplified version, resolving geometric con-

2375-0138/19/$31.00 ©2019 IEEE
DOI 10.1109/IV.2019.00022

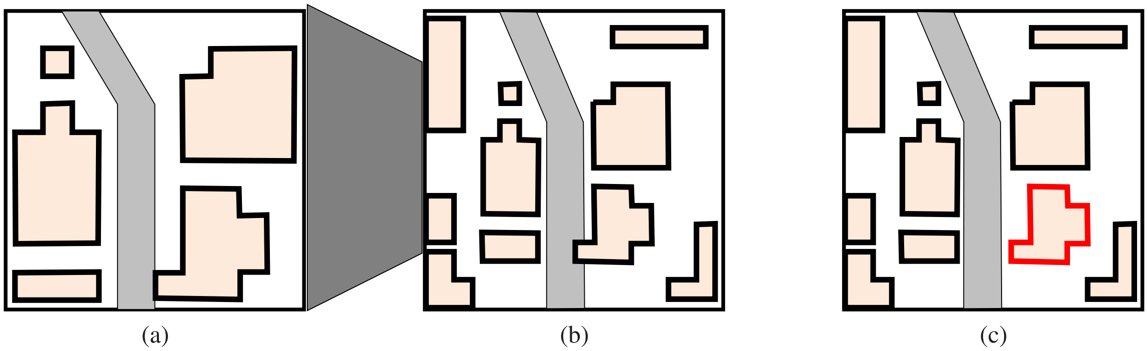


Fig. 1. Maps on various scales: Roads and buildings are more likely to overlap with each other as the scale decreases: (a) Map at a large scale; (b) Map at a small scale; (c) Displacement of a building to avoid overlaps.

flicts among features and so on when placing the map components within the limited map domain at the small scale. The conventional manual generalization processes have been formulated as a set of specific operations so far [1], [2] and the associated principles are detailed in the literature [3]. Map generalization mechanisms and examples are also exhibited in several books [4], [5].

Among the generalization processes, the simplification operation has been studied rather intensively in the community. For example, the shapes of geometric entities, such as buildings, are also abstracted by level-of-detail simplification based on image-processing techniques, which are based on the scale-space theory [6] and a set of optimization techniques that include least-square adjustment, self-organizing maps and neural networks [7].

Algorithmic formulation of such simplification operation starts with line schematization for reducing the geometric complexity of the rectangular cartograms [8], and then extends to more complicated polygonal shapes by employing subdivision simplification [9] and area preservation [10]. Schematization techniques have also been applied to various types of map styles and often the associated operations are formulated as constrained optimization processes. Examples include schematic metro maps [11]–[13] road networks [14], [15] and urban area maps [16]–[18].

Aggregation of geometric components are successfully formulated by taking advantage of Gestalt-based clustering rules [19]. Selection operations have also been successfully incorporated to schematize map contents. In particular, automatically composing route maps of high readability through selection of important road segments will facilitate travelers navigating with mobile devices [20]–[22].

Displacement operations can avoid visual clutter arising from overlapping map features. Ware and Jones [23] proposed an interactive approach to avoid geometric conflicts of map features by incorporating the steepest gradient descent and simulated annealing techniques, whereby the cost function is incorporated to penalize such undesirable conflicts. Liu et al. [24] also solved this cartographic displacement problem by introducing the combination of constrained Delaunay triangulation and deformation based on an elastic beam truss [25].

Finally, it is important to implement the consistent placement of map features in accordance with changes to map scales. Been et al. [26] formulated consistent labeling in a dynamic environment in which they optimized the dynamic label placement by maximizing the total sum of active range within the scale domain and further explored several computational aspects of this labeling problem [27]. We also employed this formulation in order to maximize the total stay of map features with respect to the map scale.

## III. Overview

In our approach, we model a road network as a set of connected polylines. Thus, we consider each road polyline as a single map feature in this formulation. This means that we consider the proper arrangement of the map feature by displacing its component vertices to avoid unwanted overlap between the map features. Further, we encode the basement of each building as a closed polygon and again search for its conflict-free placement by optimizing the displacement of the building.

In this problem setup, we draw each road segment with a fixed width with respect to the screen size. This implies that the road segment is more likely to have conflicts with other building polygons as well as road segments as the map scale becomes smaller, since the map domain contains more geographic components. We solve this problem by making enough space around the road segments so that we can avoid visual clutter arising from conflicts between roads and buildings that are close to each other. In our approach, we formulate this problem as a constrained optimization problem by imposing constraints derived from several aesthetic criteria on spatial relationships among the map features. In practice, we introduce linear constraints with respect to the geometric positions of the map features together with the additional variables, and then solve the optimization problem by employing conventional linear programming techniques. In the sections that follow, we detail the formulations for displacing map features and consistently placing them across multiple map scales.

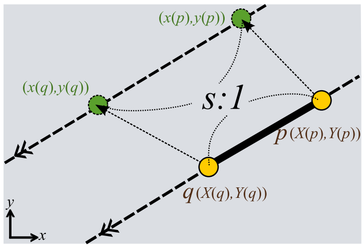
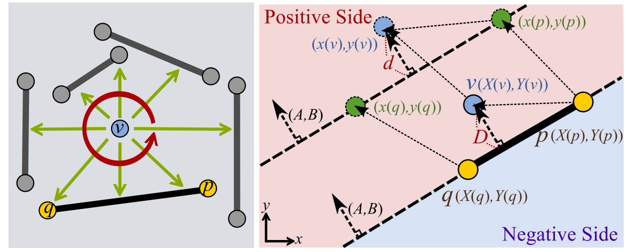



Fig. 2. Fixed line orientation constraint for keeping the orientation of line features during the displacement process.




(a) (b)

Fig. 3. Relative position constraint for preserving the spatial relationship between a sample point and a line segment.

## IV. Displacement

It is natural to avoid geometric overlaps among map features by properly translating them with minimum displacement, as shown in Figure 1(c). Nevertheless, we still have to maximally keep the visual appearance of the map features together with their spatial relationships, so that we can maintain the inherent configuration of the given maps. In our approach, inspired by Hirono et al. [18], we achieved this goal by introducing three aesthetic criteria in the configuration of map features: fixed line orientation, relative position and scale limits.

### A. Fundamental Constraints for Displacement

The *fixed line orientation* constraints allow us to strictly fix the orientation of each line feature, while its length can be adjusted within the predefined scale range. Suppose that, as shown in Figure 2, the line segment $\overline{pq}$ will be translated to a new segment through the displacement process, where $(X(p), Y(p))$ and $(X(q), Y(q))$ are the original coordinates of $p$ and $q$, respectively, and $(x(p), y(p))$ and $(x(p), y(q))$ are those of $p$ and $q$ after the displacement, respectively. The orientation of the line segment can be retained by imposing the following equations:

$$\left\{ \begin{array}{lcl} x(p) - x(q) & = & s\,(X(p) - X(q)) \\ y(p) - y(q) & = & s\,(Y(p) - Y(q)) \end{array} \right. , \tag{1}$$

where $s$ signifies the scale factor of the line segment in terms of the original length. Since $X(p)$, $Y(p)$, $X(q)$ and $Y(q)$ are constant, equations in Eq. (1) are linear constraints with respect to the variables $x(p)$, $y(p)$, $x(q)$ and $y(q)$. In our formulation, the fixed orientation constraints are imposed on all the road and building segments to maximally maintain the local shapes of these map features.

On the contrary, the *relative position* constraints let us preserve the spatial relationships among map features. This is especially important in maintaining the accuracy of the map contents, because with the constraints we can avoid cases in which two buildings are swapped or a building goes beyond a road. This is accomplished by preserving the relative positions of a vertex (i.e., a sample point) and an edge (i.e., a line segment) that are next to each other, as shown in Figure 3(a). In our implementation, we search for the neighboring edges of each vertex by looking around it. This lets us sort out edges in the vicinity of the vertex by their distance from the vertex at the respective reference angle in the polar coordinate system. Collecting such combinations of vertices and edges can be conducted as a preprocess, and the associated results can be stored on the disk storage prior to the map optimization if necessary.

Now suppose that we have the vertex $v$ on the left side of the line segment $pq$, as shown in Figure 3(b), and employ the same notation as in Figure 2 to express the coordinates of these vertices before and after the displacement process. Since the coordinates of the original positions of $pq$ are given, we can easily calculate a unit vector normal to the line segment $pq$ as $(A, B)$ and thus represent the equation of the line $L$ along the line segment as $Ax + By + C = 0$. Now we are ready to obtain the signed distance of $v$ from the line segment $pq$, by calculating the inner vector $\overrightarrow{pv}$ and the unit normal vector $(A, B)$ as:

$$D = A(X(v) - X(p)) + B(Y(v) - Y(p)). \tag{2}$$

This implies that the equation of $L$ along $\overline{pq}$ splits the entire 2D domain into a positive side ($Ax + By + C > 0$) and a negative side ($Ax+By+C < 0$), which facilitates keeping the relative position $v$ with respect to $\overline{pq}$ by constraining $v$ within the positive side of $\overline{pq}$. This constraint should be satisfied, even after the displacement process as:

$$d = A(x(v) - x(p)) + B(y(v) - y(p)). \tag{3}$$

Note that the orientation of the line segment $\overline{pq}$ has been already fixed, and thus the coefficients $A$ and $B$ remain constant throughout the displacement process.

This formulation finally allows us to compose a linear inequality of the relative position constraint as:

$$\begin{cases} A(x(v) - x(p)) + B(y(v) - y(p)) \geq H & \text{if } D > 0 \\ A(x(v) - x(p)) + B(y(v) - y(p)) \leq -H & \text{if } D < 0 \end{cases} \tag{4}$$

Here, $H$ represents the possible spacing from the line segment. In particular, as described earlier, we associate the width of each road segment and $H$ represents the half-width of the road on the corresponding map scale. Let us denote the inverse of the map scale $S$ by $R = 1/S$. Actually, $H$ is proportional to $R$ and can be written as $H = R \cdot C$, where C is a constant value. If the line segment is on the boundary of a building basement, we can simply set $H = 0$.

The *scale limits* constraint, which is the last of the aforementioned three types of constraints, merely restricts the

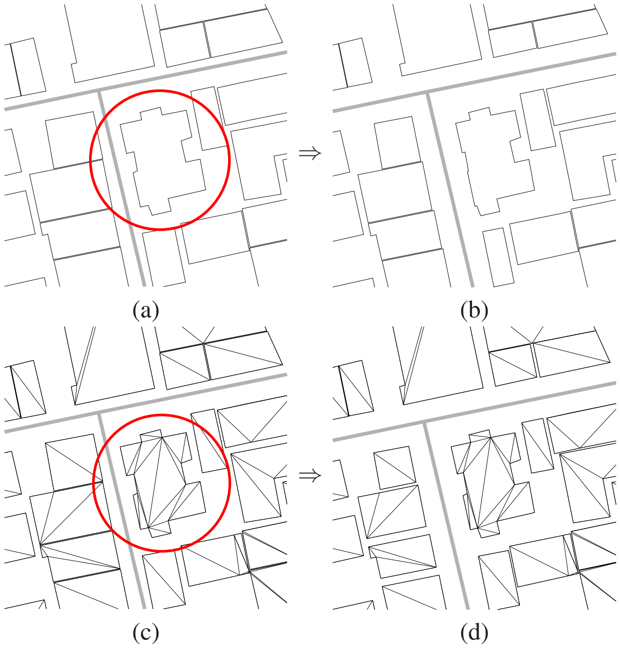



Fig. 4. Unexpected shape distortion of a building basement polygon: (a) Original polygon shape; (b) Optimized shape containing unwanted distortions; (c) Triangulated polygon shape; and (d) Optimized shape while maximally retaining the orientations of internal diagonals.

available range of the scale factor $s$, so that we can avoid an excessive shrinkage or expansion of the map features after the displacement process. On all the scale factors, we impose the following constraints in our implementation:

$$\frac{1}{10} \leq s \leq 10. \tag{5}$$

We also have to define a cost function to find an optimized layout of the map features under these types of constraints. For this purpose, we simply penalize the overall amount of displacement to maximally retain the original configuration of the map features. This can be achieved by evaluating the absolute difference between the positions of each vertex before and after the displacement as:

$$\begin{cases} -\delta_x(v) \leq x(v) - X(v) \leq \delta_x(v) \\ -\delta_y(v) \leq y(v) - Y(v) \leq \delta_y(v) \end{cases}, \tag{6}$$

where $\delta_x(v)(\geq 0)$ and $\delta_y(v)(\geq 0)$ are variables that are newly introduced to represent the displacement of the vertex $v$ along the $x$- and $y$-axes, respectively. Our cost function is given by the sum of all the displacements as:

$$\mathrm{cost}_{\mathrm{displacement}} = W_d \sum_{v \in V} (\delta_x(v) + \delta_y(v)), \tag{7}$$

where $W_d$ corresponds to the weight value associated with the displacement process.

### B. Constraints for Preserving Shapes

It is sometimes the case that we cannot fully retain the shapes of building basements even with the aforementioned

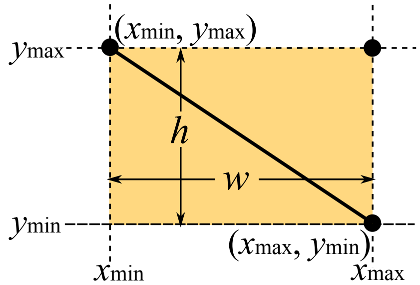



Fig. 5. Preserving the orientation of a line segment by bounding the aspect ratio of the associated bounding box.

constraints. Figure 4 shows such an example in which we optimized the layout of the map features. Unfortunately, in this case, we introduced unwanted distortions around the small protruding parts of the building basement (Figure 4(b)) when compared with the original shape (Figure 4(a)). To alleviate this problem, we first triangulate the inside of the building polygons by inserting diagonals, as shown in Figure 4(c), and then preserving the orientation of such diagonals, as shown in Figure 4(d). More specifically, we take advantage of an ear clipping algorithm [28] for triangulating concave polygons and impose additional constraints to restrict the orientation of the diagonal segments that are newly introduced into the inside of the polygon.

The new formulation for preserving the polygonal shapes has been inspired by the work that respects mental maps in schematic railway diagrams [29] and can be summarized as follows. Suppose that, as shown in Figure 5, the bounding boxes that confines a diagonal line segment before and after the map optimization are represented by $[X_{\min}, X_{\max}] \times [Y_{\min}, Y_{\max}]$ and $[x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}]$, respectively. We can restrict the change in the orientation of the line segment by bounding the aspect ration of this bounding box as:

$$\frac{W}{(1+E)H} \leq \frac{x_{\max} - y_{\min}}{y_{\max} - y_{\min}} \leq \frac{(1+E)W}{H}, \tag{8}$$

where $W = X_{\max} - X_{\min}$, $H = Y_{\max} - Y_{\min}$ and $E$ indicates the error tolerance and set to be $0.05$ by default. Note that the above formulation can be reduced to linear inequality constraints, and thus permits the further preservation of the shapes of the map features, as presented in Figure 4(d). Moreover, our present formulation can be solved using the linear programming technique and requires less computation times.

## V. Scale-Aware Consistent Mapping

So far, we have assumed that the optimization of the map layout is conducted at the present map scale on demand. This means that we have to adjust the layout of the map features every time we change the map scale, which may prevent us from interactively exploring the map contents at arbitrary scales. Optimizing map layouts on individual scales also causes unexpected pop-up effects of map features and sudden leaps in their position as the map scale changes.

This leads us to the idea of precomputing the optimized map layouts within the specific range of the map scale, to ensure consistency in the placement of map features irrespective of the change to the map scale.

We implemented this functionality by first sampling a specific number of scale values $S^{[l]}$ $(l = 1, \dots)$ in the available scale range and optimizing the total sum of the cost functions at the individual scales. As for the position of each vertex $v$ within the scale range, we interpolate optimized positions on the vertex $v$ with respect to the map scale. Assume that the current scale $S$ falls into the range $[S^{[l]}, S^{[l+1]}]$, that is, $S \in [S^{[l]}, S^{[l+1]}]$, and the coordinates of $v$ on the scale $S^{[l]}$ and $S^{[l+1]}$ are obtained as $(x^{[l]}(v), y^{[l]}(v))$ and $(x^{[l+1]}(v), y^{[l+1]}(v))$, respectively. Linearly interpolating between the two coordinates allows us to obtain the coordinates of $v$ at the scale $S$ as:

$$(1-t)(x^{[l]}(v), y^{[l]}(v)) + t(x^{[l+1]}(v), y^{[l+1]}(v)), \tag{9}$$

where $t = (S - S^{[l]})/(S^{[l+1]} - S^{[l]})$. This allows us to interactively explore the map contents at arbitrary scales once we have precomputed the optimized layout within the available range of the map scale.

## VI. Results and Discussion

We have implemented our prototype system on a laptop PC with Intel Core i7 CPU with 2 cores (3.3GHz, 256KB L2 Cache per core and 4MB L3 Cache), 16GB RAM and an Intel Iris Graphics 550 GPU (1536MB VRAM). The source code has been written in C++ using Qt for the interface, OpenGL for drawing map contents, CGAL for geometric computation and IBM ILOG CPLEX for solving constrained optimization problems through linear programming techniques. Table I provides computation times and numbers of variables and linear constraints, which are required to optimize the map layouts in the experimental results.

As described earlier, the optimized layouts of a residential area using multiple-scale displacement are demonstrated in Figure 6 and Figure 7, respectively. In particular, the scale-aware optimization framework allows us to precompute the optimized layout of the map features by sampling the available scale range, and fully facilitates interactively exploring the digital map contents on arbitrary map scales.

## VII. Conclusion

This paper has presented a novel formulation for displacement techniques as effective map-editing processes in cartographic generalization. We formulated the aesthetic displacement of the map features as constrained optimization problems, in which we employed linear programming techniques for finding appropriate solutions. Furthermore, our proposed formulation is scale-aware in the sense that it retains the consistent placement of map features such as roads and buildings by taking into account the space spanned by the map domain and scale axis. Experimental results demonstrate that our formulation can precompute visually plausible layouts of map features within an acceptable time frame, which fully facilitates the exploration of digital map contents on arbitrary scales on computers and mobile devices.

TABLE I
*Computation times (in seconds)*

| Case | Computation times | No. of variables | No. of constraints |
|---|---|---|---|
| Figure 6 | 35.9 | 162,244 | 386,368 |
| Figure 7 | 54.7 | 196,852 | 471,680 |

Our future work includes integrating existing formulations such as simplification, aggregation, and selection into our approach. Inferring semantics of map features to adaptively adjust penalty weights still remains to be tackled. More sophisticated displacement can also be explored by seeking collaboration with domain experts. It is also interesting to extend this formulation to 3D architectural map [18] in which 3D objects are displaced and deformed consistently as the scale changes.

## Acknowledgments

This work has been partially supported by JSPS KAKENHI (grant numbers 19H04120, 16H02825, and 26730061). This work has also been conducted under the support by Joint Research Program No. 398 at CSIS, the University of Tokyo.

## References

[1] K. S. Shea and R. B. McMaster, “Cartographic generalization in a digital environment: When and how to generalize,” in *Proceedings Auto-Carto 9*, 1989, pp. 56–67.
[2] D. Lee, “Automation of map generalization: The cutting-edge technology,” Environmental Systems Research Institute, Inc. (ESRI), Tech. Rep., 1996.
[3] A. H. Robinson, J. L. Morrison, P. C. Muehrcke, A. J. Kimerling, and S. C. Guptill, *Elements of Cartography*, 6th ed. John Wiley & Sons, Inc., 2006.
[4] R. B. McMaster and K. S. Shea, *Generalization in Digital Cartography*. Association of American Geographers, 1992.
[5] Z. Li, *Algorithmic Foundation of Multi-Scale Spatial Representation*. CRC Press, 2006.
[6] H. Mayer, “Scale]Spaces for generalization of 3D buildings,” *International Journal of Geographical Information Science*, vol. 19, no. 8-9, pp. 975–997, 2005.
[7] M. Sester, “Optimization approaches for generalization and data abstraction,” *International Journal of Geographical Information Science*, vol. 19, no. 8-9, pp. 871–897, 2005.
[8] B. Speckmann, M. van Kreveld, and S. Florisson, “A linear programming approach to rectangular cartograms,” in *Progress in Spatial Data Handling: 12th International Symposium on Spatial Data Handling*, 2006, pp. 529–546.
[9] K. Buchin, W. Meulemans, and B. Speckmann, “A new method for subdivision simplification with applications to urban-area generalization,” in *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2011, pp. 261–270.
[10] ——, “Area-preserving c-oriented schematization,” in *Abstracts of the 27th European Workshop on Computational Geometry*, 2011, pp. 163–166.
[11] M. Nöllenburg and A. Wolff, “Drawing and labeling high-quality metro maps by mixed-integer programming,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 5, pp. 626–641, 2011.
[12] Y.-S. Wang and M.-T. Chi, “Focus+context metro maps,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2528–2535, 2011.
[13] H.-Y. Wu, S. Takahashi, D. Hirono, M. Arikawa, C.-C. Lin, and H.-C. Yen, “Spatially efficient design of annotated metro maps,” *Computer Graphics Forum*, vol. 32, no. 3, pp. 261–270, 2013.

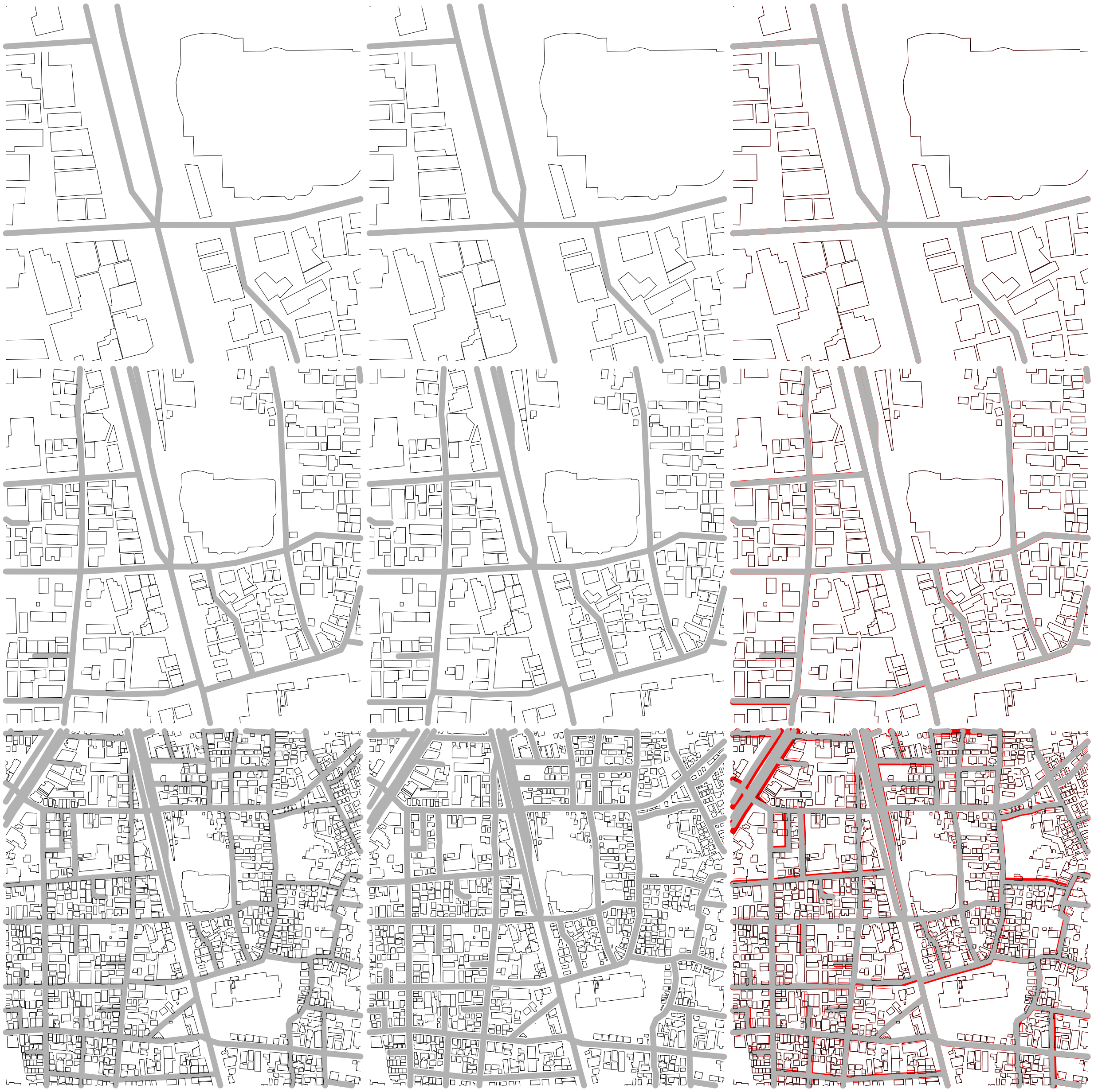


Fig. 6. Scale-aware consistent placement of map features in the area around the Omori Sta. Left: original layout. Middle: optimized layout. Right: Difference between the two layouts. From top to bottom, we monotonously decrease the map scale. Red features correspond to the effects obtained by the displacement processes.


[14] J.-H. Haunert and L. Sering, "Drawing road networks with focus regions," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2555–2562, 2011.

[15] M. Chimani, T. C. van Dijk, and J.-H. Haunert, "How to eat a graph: Computing selection sequences for the continuous generalization of road networks," in *Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2014, pp. 243–252.

[16] F. Grabler, M. Agrawala, R. W. Sumner, and M. Pauly, "Automatic generation of tourist maps," *ACM Transactions on Graphics*, vol. 27, no. 3, 2008, article No. 100.

[17] H. Qu, H. Wang, W. Cui, Y. Wu, and M.-Y. Chan, "Focus+Context route zooming and information overlay in 3D urban environments," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 6, pp. 1547–1554, 2009.

[18] D. Hirono, H.-Y. Wu, M. Arikawa, and S. Takahashi, "Constrained optimization for disoccluding geographic landmarks in 3D urban maps," in *Proceedings of the 6th IEEE Pacific Visualization Symposium (PacificVis 2013)*, 2013, pp. 17–24.

[19] Z. Liqiang, D. Hao, C. Dong, and W. Zhen, "A spatial cognition-based urban building clustering approach and its applications," *International Journal of Geographical Information Science*, vol. 27, no. 4, pp. 721–740, 2013.

[20] M. Agrawala and C. Stolte, "Rendering effective route maps: Improving usability through generalization," in *Proceedings of ACM SIGGRAPH 2001*, 2001, pp. 241–250.

[21] F. Schmid, "Knowledge-based wayfinding maps for small display cartography," *Journal of Location Based Services*, vol. 2, no. 1, pp. 57–83,

Fig. 7. Scale-aware consistent placement of map features in the area around the Koujiya Sta. Left: original layout. Middle: optimized layout. Right: Difference between the two layouts. From top to bottom, we monotonously decrease the map scale. Red features correspond to the effects obtained by the displacement processes.


2008.

[22] J. Kopf, M. Agrawala, D. Bargeron, D. Salesin, and M. Cohen, "Automatic generation of destination maps," *ACM Transactions on Graphics*, vol. 29, no. 6, 2010, article No. 158.

[23] J. M. Ware and C. B.Jones, "Conflict reduction in map generalization using iterative improvement," *GeoInformatica*, vol. 2, no. 4, pp. 383–407, 1998.

[24] Y. Liu, Q. .Guo, Y. Sun, and X. Ma, "A combined approach to cartographic displacement for buildings based on skeleton and improved elastic beam algorithm," *PLoS ONE*, vol. 9, no. 20, 2014.

[25] M. Bader, M. Barrault, and R. Weibel, "Building displacement over a ductile truss," *International Journal of Geographical Information Science*, vol. 19, no. 8-9, pp. 915–936, 2005.

[26] K. Been, E. Daiches, and C. Yap, "Dynamic map labeling," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 5, pp. 773–780, 2006.

[27] K. Been, M. Nöllenburg, S.-H. Poon, and A. Wolff, "Optimizing active ranges for consistent dynamic map labeling," *Computational Geometry: Theory and Applications*, vol. 43, no. 3, pp. 312–328, 2010.

[28] H. ElGindy, H. Everett, and G. Toussaint, "Slicing an ear using prune-and-search," *Pattern Recognition Letters*, vol. 14, no. 9, pp. 719–722, 1993.

[29] S. Takahashi, K. Maruyama, T. Kawagoe, H.-Y. Wu, K. Misue, and M. Arikawa, "Mental map preservation for progressively labeling railway networks," *International Journal of Art, Culture and Design Technologies*, 2019, accepted.