

AUTOMATIC LABEL PLACEMENT OF AREA-FEATURES USING DEEP LEARNING

Y. Li^{1,*}, M. Sakamoto¹, T. Shinohara¹, T. Satoh¹

¹ PASCO CORPORATION, 4-9-6 Aobadai, Meguro-ku, Tokyo 153-0042, Japan
(yiomn_3951, moittu9191, taarkh6651, tuoost7017)@pasco.co.jp

Commission IV, WG IV/3

KEY WORDS: Label Placement, Area-feature, Deep Learning, Key-point Detection, Heatmap, Stacked Hourglass Networks

ABSTRACT:

Label placement is one of the most essential tasks in the fields of cartography and geographic information systems. Numerous studies have been conducted on the automatic label placement for the past few decades. In this study, we focus on automatic label placement of area-feature, which has been relatively less studied than that of point-feature and line-feature. Most of the existing approaches have adopted a rule-based algorithm, and there are limitations in expressing the characteristics of label placement for area-features of various shapes utilizing handcrafted rules, criteria, objective functions, etc. Hence, we propose a novel approach for automatic label placement of area-feature based on deep learning. The aim of the proposed approach is to obtain the complex and implicit characteristics of area-feature label placement by manual operation directly and automatically from training data. First, the area-features with vector format are converted into a binary image. Then a key-point detection model, which simultaneously detect and localize specific key-points from an image, is applied to the binary image to estimate the candidate positions of labels. Finally, the final label placement positions for each area-feature are determined via simple post-process. To evaluate the proposed approach, the experiments with cadastral data were conducted. The experimental results show that the ratios of the estimation errors within 1.2 m (corresponding to one pixel of the input image) were 92.6% and 94.5% in the center and upper-left placement style, respectively. It implies that the proposed approach could place the labels for area-features automatically and accurately.

1. INTRODUCTION

Label placement is one of the essential functions in the fields of cartography (Imhof, 1962; Yoeli, 1972; Monmonier, 1982) and geographic information systems (GIS; Freeman, 1991). A number of studies have been conducted on automatic label placement (Wolff and Strijk, 1996; Oeltze-Jafra et al., 2014). However, most studies adopted a rule-based approach that requires careful design of the specific rules, criteria, and objective functions for the given label placement style.

Label placement can be categorized into point, line, and area feature label placement. Compared to the automatic label placement of point-feature and line-feature, the automatic area-feature label placement is more challenging due to the complexity and variety of the shapes of area-features and there are relatively few studies (Roessel 1989; Edmondson et al., 1996; Dörschlag et al., 2003).

Figure 1 shows two different results of label placement on the center of area-feature conducted by GIS software and manual operation, respectively. GIS software places the label on the geometric center of the area-feature, but the result significantly differs from the ideal label position placed by manual operation. When the shape of area-feature becomes more complicated, it is extremely difficult to estimate the ideal label position based on handcrafted rules, criteria, and objective functions. In addition, in the rule-based approaches, when the required style of label placement changed, those rules, criteria, and objective functions need to be redesigned.

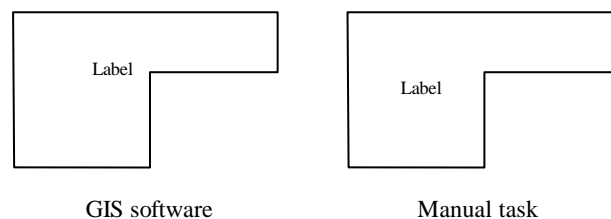


Figure 1. Label placement on the center of area-feature by GIS software and manual operation

In this study, we focused on deep learning, which is a specific kind of machine learning. Compared to the rule-based approach, deep learning does not require the human design of rules and features. Using deep learning, it is possible to automatically acquire potential rules and features of the training data through end-to-end learning. In recent years, deep learning has greatly outperformed traditional machine learning and widely applied in a variety of fields. In particular, in the image processing field where deep learning was commonly utilized first, numerous deep learning models for different tasks have already been proposed, i.e. classification model, regression model, object detection model, semantic segmentation model, key-point detection model, etc. It is important to adopt the appropriate model that is suitable for one's own task.

In this study, we proposed a novel approach for automatic label placement of area-feature based on key-point detection model,

* Corresponding author

which simultaneously detect and localize specific key-points from an input image. The aim of the proposed approach is to acquire the complex and implicit characteristics of manual label placement automatically, through training a deep learning model using a dataset with ground truth generated by human operation. To our knowledge, there have been no studies introduced by applying deep learning in automatic label placement of area feature.

2. RELATED RESEARCHES

Over the past few decades, a number of studies about label placement, especially about automatic label placement, were conducted (Wolff and Strijk, 1996). As the pioneering study, Imhof (1962, 1975) has thoroughly described placement rules of different labels for point, line, and area features, and the criteria for evaluating label positions as well. His studies are considered as the base of all followed researches on label placement. Another important work has been published by Yoeli (1972), who described his computer program for automatic label placement and defined recommended positions and their priorities, for point-feature label placement.

The label placement of area-feature (or polygon in the GIS field) is more challenging due to the complexity of shapes, comparing with that of point-feature and line-feature. About automatic label placement of area-feature, Roessel (1989) has proposed an algorithm to circumvent the problems caused by the approach of placing the labels simply on the centroid of polygons. In his algorithm, after computing a number of horizontal candidate labelling rectangles (boxes) within a polygon, the most suitable candidate is selected for the final label placement. Pinto and Freeman (1996) have developed a method that employs a feedback approach, wherein the placement result is evaluated and progressively modified from an initial placement position and reduced the deviation from the ideal position. Each placement is evaluated according to five criteria, i.e. length, clearance from boundaries, symmetry, horizontal placement, and conformity, which are derived from generally accepted standards for manual label placement. For the placement of curved label, Barrault (2001) has introduced a new measure to evaluate the fitness of a circular arc with respect to the boundary of the area being labelled. Furthermore, a near-real time method for curved label placement has been proposed based on the approach developed by Barrault (2001), through improving the efficiency of the evaluation process utilizing the skeleton of polygon (Krumpe and Mendel, 2020).

However, the most challenging issue of the rule-based approaches is to design appropriate evaluating criteria for label placement of area-features with complex shapes. A deep learning based approach is an alternative to avoid this problem since in deep learning the characteristics of ideal label placement can be obtained automatically and directly from the training data.

3. METHODOLOGY

Figure 2 shows the processes of the proposed approach. In order to utilize the wealth of knowledge on deep learning

accumulated in the image processing field, firstly the area-features with vector format in certain region is converted into an image. Then the candidate label position of each area-feature in the input image is estimated utilizing the key-point detection model, the stacked hourglass networks specifically. Finally, the ideal label position for each area-feature is determined through simple rule-based post-process. The details about the proposed approach are explained below.

3.1 Rasterization

In the rasterization process, a certain region of area-feature data in vector format is converted to a binary image. In other words, instead of individual area-feature, the multiple area-features contained in a certain region are converted into a single image. When converting vector data into an image, it is very important to properly set the image size, the image resolution, and the boundary line thickness of area-feature.

First, the size of the input image is set to 512 x 512 pixels due to the memory limitation of the GPU. Next, the image resolution is set to 12.5 cm/pixel in consideration of the maximum size of area-feature which can be fully contained in the input image. Specifically the actual size of the input image is 64m x 64m, and those area-features with a size larger than that of the input image cannot be treated properly. Finally, the thickness of the boundary line is set to 2 pixels in consideration of two aspects. One is the minimum size of the area-feature which can present the shape correctly in the input image, and the minimum size is 25cm x 25cm according to the settings. Another aspect is the easiness of identification for individual area-feature. It is easy to imagine that the thicker the boundary line, the easier it is to identify each area-feature. However, the boundary thickness is set to a relatively small value because the high-resolution feature maps are shared to the end of the network by skip connections in the stacked hourglass networks.

3.2 Production of heatmap

As shown in Figure 2, the stacked hourglass networks (Newell et al., 2016; Newell et al., 2017), a key-point detection model based on deep learning proposed for human pose estimation, is applied to the input image for producing a heatmap. The heatmap consists of pixel-by-pixel detection scores, with peaks displayed at each estimated label position of the area-feature.

At the beginning of the stacked hourglass networks, a feature map with a resolution of 1/4 of the input image is generated through convolution and pooling layers. Then the network comprised of multiple stacked hourglasses produces the heatmap with the same resolution as the feature maps. Each hourglass is an encoder-decoder network with skip connections. Stacking multiple hourglasses enables repeated bottom-up and top-down inference to produce a more accurate final prediction. In this study, two hourglasses were stacked in consideration of GPU memory and processing time.

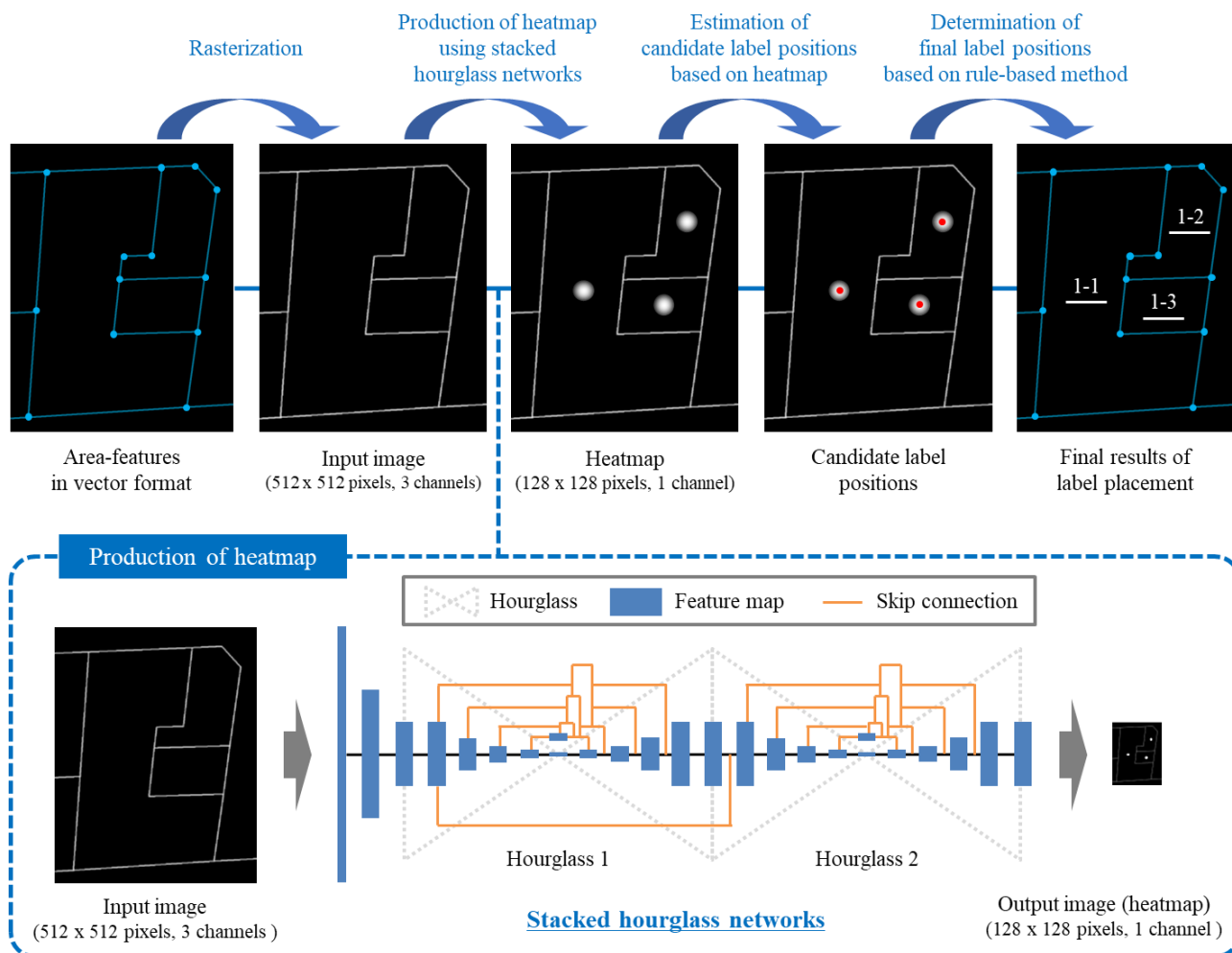


Figure 2. Process of proposed approach for area-feature label placement

During training phase, a detection loss is imposed on the output heatmap. The detection loss computes mean square error between each predicted detection heatmap and its ground truth heatmap which consists of a 2D Gaussian (with a standard deviation of σ) at each key-point location. The σ was set to 1 pixel in this study.

3.3 Estimation of candidate label positions

The candidate label positions are estimated by performing non-maximum suppression and threshold on the heatmap. The threshold was set to a relatively low value for leaving more possible candidates, and the over-detected candidates are discarded on the following determination process.

3.4 Determination of final label positions

In the post-process, after converting the estimated label positions of candidates on input image to the coordinates on vector data space, each candidate is assigned to the area-feature including that individual candidate. When there are multiple candidates included in a same area-feature, the candidate with the highest detection score is finally adopted.

As described above, the proposed approach has the following two features. One feature is that we treated the label placement

problem as a key-point detection task by converting the area-feature data into images. Another one is that the label positions of multiple area-features can be estimated simultaneously. Comparing to the label placement approaches for one area-feature at a time, the proposed approach can achieve faster processing time.

4. EXPERIMENTAL RESULTS

4.1 Dataset

In the experiments, the cadastral data (Figure 3), which contains the shape information of land lots and the label placement results by manual operation, were used for evaluating the proposed approach. The data of land lot boundary line in shapefile format is an open data published by Shizuoka Prefecture in Japan. Table 1 shows the details of the dataset.

To prepare input images, first we converted the land lot boundary line data in vector format into one large binary image with a resolution of 12.5cm. Then we extracted multiple image patches from the large image at a certain size and interval. The image patches for training and test were prepared from different regions of the dataset respectively. The label positions placed manually were used as ground truth. In the experiments,

two different styles of label placement were considered, i.e. center placement and upper-left placement, which are commonly adopted for label placement on cadastral maps in Japan.



"H28 Land lot boundary line"
© Shizuoka Prefecture (Licensed under CC BY 4.0)
<https://dataset.city.shizuoka.jp/dataset/shisanzei20160701-001>

Figure 3. Partial area of dataset (yellow points indicate the ground truth of label placement by manual operation)

Items	Training data	Test data
Area of region (km ²)	1.46	2.65
Number of area-features	6,700	8,900
Resolution of image (cm/pixel)	12.5	
Size of image patch (pixel)	512 x 512	
Interval of patch extraction (pixel)	128	256
Number of image patches	6,800	2,400

Table 1. Details of dataset

4.2 Training

Table 2 shows the settings of hyper-parameters for training. As shown in Figure 4, the training loss was reduced steadily and eventually converged around 0.0005.

Parameters	Values
Input image size (pixel)	512 x 512
Output heatmap size (pixel)	128 x 128
Learning rate	2e-4
Batch size	12
Maximum iterations (epoch)	100

Table 2. Settings of hyper-parameters

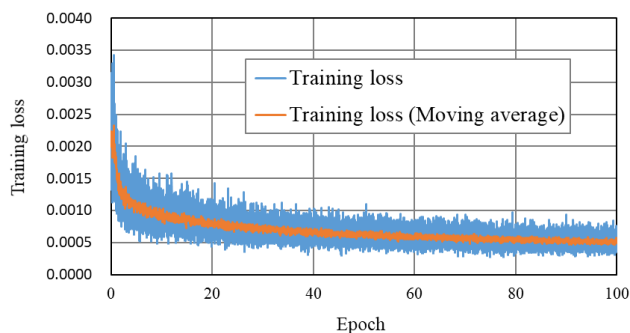


Figure 4. Training loss

4.3 Test

To evaluate the effectiveness of the proposed approach, the estimated label positions were compared with the ground truth. As shown in Figure 5, the proposed approach could estimate the accurate label positions of area-features, even though the area-features had complicated shapes.

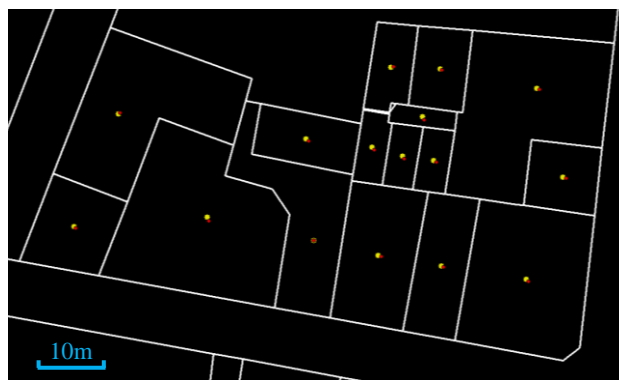


Figure 5. Comparison of estimated positions (red points) and ground truth (yellow points).

Furthermore, for quantitative evaluation, the distance between the estimated position and the corresponding ground truth was calculated in each area-feature as estimation error. The area-features that do not contain any estimated result were regarded as failures. According to the experimental results, the mean estimation error was 0.60 m, which corresponds to about half-pixel of the input image. As shown in Figure 6, the estimation errors are distributed mostly in the range of 0.2 m to 0.6 m and accounted for about 80% of the total. In addition, the ratio of estimation errors within 1.2 m was 92.8%, and the ratio of failed cases was 2.5%.

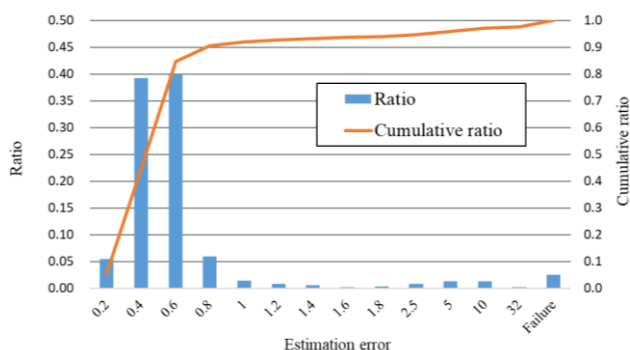


Figure 6. Distribution of estimation error

In contrast, in the experiment of the upper-left label placement, the mean estimation error was 0.83 m, the ratio of estimation errors within 1.2 m was 94.5%, and the failure rate was 0.4%.

5. DISCUSSION

5.1 The features of produced heatmap

From the example of a heatmap shown in Figure 7, it can be confirmed that single peak appeared in each area-feature except for one case. It implies that the trained stacked hourglass networks can identify individual area-feature from the input image containing multiple area-features. The area-features completely contained in the input image show a trend to have sharp peaks, while those partially included area-features tend to have relatively weak peaks. There was one area-feature where two peaks appear, and the position with the highest detection score was adopted as final label position in this case.

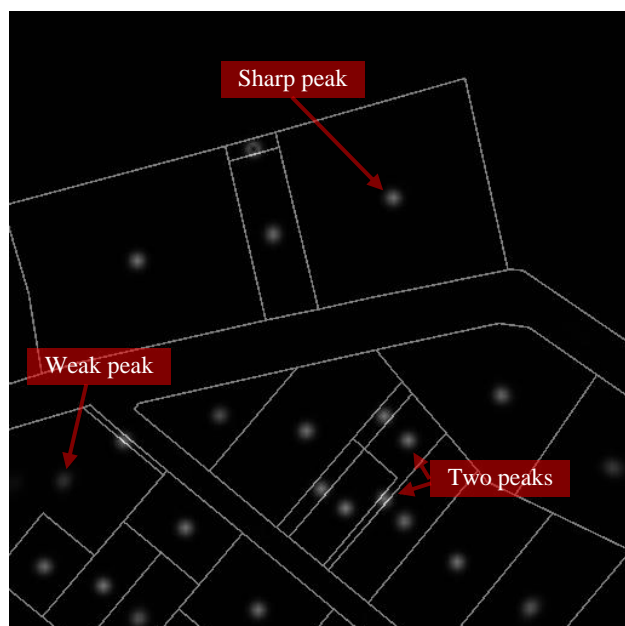


Figure 7. Example of output heatmap (128 x 128 pixels) by trained key-point detection model

5.2 Ablation study

The impacts of three factors on the accuracy of label placement were analyzed. The three factors include the type of input

image (binary and color images), the number of hourglasses in the stacked hourglass networks, and the value of σ for generating ground truth heatmap. Table 3 shows the label placement accuracy by four different models. Model 1 is the base model and model 2 to 4 are those models which changed one factor from the base model. Model 4 was used in the experiment as described in section 4.

Model	Input image	Number of hourglass	σ of heatmap	Ratio of errors within 1.2 m
1	Binary	2	2.0 pixel	92.2%
2	Color	2	2.0 pixel	92.0%
3	Binary	4	2.0 pixel	92.9%
4	Binary	2	1.0 pixel	92.8%

Table 3. Comparison of four different models

First, with respect to the effect of input image type, two different types of images were considered as shown in Figure 8. The left image is the binary image described in section 3, and the right image is a color image created by assigning each area-feature with a random color. The aim of adopting the color image is to improve the easiness of identification for individual area-features in the stacked hourglass networks. However, as we can see from the comparison of model 1 and model 2, the accuracy of the color image is 0.2% lower than that of the binary image. One reason is that the boundary lines are sufficient to identify individual area-features. And another possible reason is that the network was confused by the various colors, which have no specific meanings.

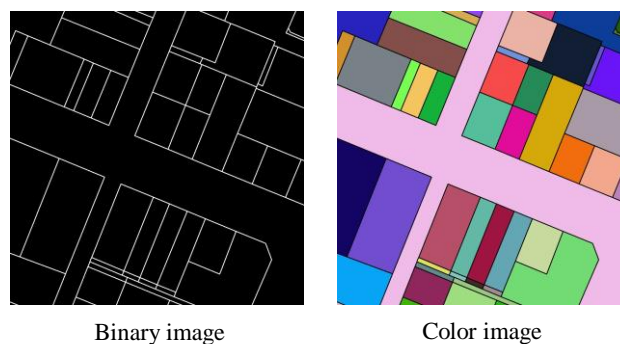


Figure 8. Two different types of input image

Next, the effects of the number of hourglasses were analyzed by comparing model 1 and model 3. It was found that the accuracy improved by 0.7% in the four-hourglass model compared to the two-hourglass model. In other words, there is a tendency for the accuracy improvement as the network size increases. However, increasing the number of hourglasses has the disadvantages of increasing the training and inference time, as well as GPU memory usage.

Finally, we analyzed the effect of the σ for generating ground truth heatmap on label placement accuracy. The value of σ in the original paper was set to 2 pixels (heatmap size / 64). In contrast, we tested a smaller value (1 pixel) for the purpose of

more accurate position estimation. As can be seen from the comparison between model 1 and model 4, the heatmap with smaller σ improved the accuracy by 0.6%. The reason is that the smaller σ produced sharper peaks on the ground truth heatmap, which led to improved accuracy.

6. CONCLUSIONS

The contributions of this study are as follows. 1) We proposed a novel approach for automatic label placement of area-feature based on deep learning. To our best knowledge, it is the first time to apply a deep learning based method to the automatic area-feature label placement. 2) The proposed approach estimates the label positions simultaneously for multiple area-features included in an input image. This improves the processing time significantly compared to processing one area-feature at a time. 3) We evaluated the effectiveness of our proposed approach through the experiments with real cadastral data. The experimental results showed that the proposed approach automatically estimates the positions of area-features accurately and effectively, even if the area-features have complicated shapes. The future works include further improvement of the estimation accuracy of label positions, and the consideration of the label size.

REFERENCES

- Barrault, M., A methodology for placement and evaluation of area map labels, 2001. *Computers, Environment and Urban Systems*, 25(1), 33-52.
- Dörschlag, D., Petzold, I., Plümer, L., 2003. Placing Objects Automatically in Areas of Maps. *Proc. 23rd Internat. Cartographic Conf. (ICC'03)*, 269-275.
- Edmondson, S., Christensen, J., Marks, J., Shieber, S., 1996. A General Cartographic Labelling Algorithm. *Cartographica*, 33(4), 13-24.
- Freeman, H., 1991. Computer name placement, ed. by Maguire, D. J., Goodchild, M. F. and Rhind, D.W., *Geographical Information Systems: Principles and Applications*, Longman, London, 445-456.
- Imhof, E., 1962. Die Anordnung der Namen in der Karte. *International Yearbook of Cartography*, 93-129.
- Imhof, E., 1975. Positioning Names on Maps. *The American Cartographer*, 2(2), 128-144.
- Krumpe, F. and Mendel, T., 2020. Computing Curved Area Labels in Near-Real Time, *arXiv:2001.02938*.
- Monmonier, M. S. 1982. *Computer-Assisted Cartography: Principles and Prospects*, Prentice-Hall, Inc., Englewood Cliffs, NJ.
- Newell, A., Huang, Z., Deng, J., 2017. Associative Embedding End-to-End Learning for Joint Detection and Grouping. *NIPS*.
- Newell, A., Yang, K., Deng, J., 2016. Stacked hourglass
- Oeltze-Jafra, S., Preim, B., 2014. Survey of Labeling Techniques in Medical Visualizations. *Visual Computing for Biomedicine (VCBM'14)*, Eurographics Association, 199-208.
- Pinto, I. and Freeman, H., 1996. The feedback approach to cartographic areal text placement, *In Advances in Structural and Syntactical Pattern Recognition Lecture Notes in Computer Science*, Springer-Verlag, New York, 1121, 341-350.
- Roessel, W., 1989. An algorithm for locating candidate labeling boxes within a polygon, *The American Cartographer*, 16, 201-209.
- Wolff, A., Strijk, T., 1996. The Map-Labeling Bibliography. <https://i11www.itk.edu/~awolff/map-labeling/bibliography/>. Accessed: 2020-02-03.
- Yoeli, P., 1972. The Logic of Automated Map Lettering. *The Cartographic Journal*, 9(2), 99-108.