Consistency-Aware Map Generation at Multiple Zoom Levels Using Aerial Image

Linwei Chen, Zheng Fang, and Ying Fu, *Senior Member, IEEE*

***Abstract*—The multilevel tiled map service is widely used and serves as a kind of digital infrastructure. These map tiles are usually rendered from vector data, whose update needs to walk or drive with professional equipment to check every point of interest. This leads to inconvenience and expensive cost in timely updating maps. Compared with vector data, aerial images are much easier and cheaper to obtain. In this article, we propose a novel multilevel map (MLM) generation framework that can automatically generate accurate and consistent maps with multiple zoom levels from aerial images. It consists of a level-aware map generator and a consistency-aware map generator. The level-aware map generator is able to generate accurate initial maps with realistic details for each zoom level. The consistency-aware map generator regards the initial maps at each zoom level as a sequence and builds the connection between them, so as to guarantee content consistency between maps at different zoom levels. Furthermore, we collect a large-scale high-quality dataset called MLM for map generation at multiple zoom levels. Experiments on our MLM dataset show that our method outperforms the previous state-of-the-art map generation methods on both comprehensive quantitative metrics and perceptual quality.**

***Index Terms*—Aerial image, GANs, image-to-image translation, remote sensing, semantic segmentation.**

I. Introduction

ONLINE multilevel map (MLM) service (e.g., Google Maps), as an important role in our lives, not only provides convenience for daily travel, but also serves as an important infrastructure for shared bike services, delivery services, logistics industry, transportation industry, and etc.

Since MLMs make it much easier for the users to gather cartographic information effectively by switching between different zoom levels, the tiled map service always provides maps with multiple zoom levels. They are usually rendered from vector data, whose update needs to walk or drive with professional equipment to check every point of interest. Although this ensures accuracy, it leads to expensive cost and cannot update timely in a crisis, such as an earthquake. Compared with vector data, aerial images are much easier and cheaper to obtain. They can be collected automatically and updated timely by remote sensing instruments, e.g., airplanes, satellites, and drones. Aerial images are widely used for earth observation, they inherently contain rich information about the ground surface that can be utilized for map generation [1], [2]. Therefore, the MLMs based on aerial images can be quickly updated even in extreme conditions, such as earthquakes, floods, mudslides, and other natural disasters, and provide the rescuers with important information about the ground surface in time to save people's lives.

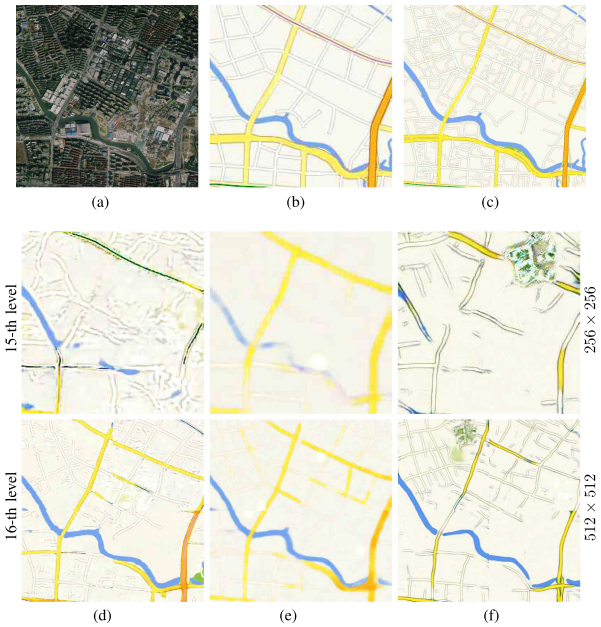


Fig. 1. MLM generation is at different zoom levels. (a) Aerial image. (b) and (c) Corresponding ground truth map at the 15th and 16th levels, respectively. (d)–(f) Results of applying the existing single-level map generation methods on multilevel generation task. (d) Pix2PixHD. (e) SMAPGAN. (f) SelectionGAN. It can be seen that these results are inconsistent in content, blurring, and have severe artifacts.

Generating MLMs from aerial images can be regarded as an image-to-image translation task [3], i.e., extracting cartographic information from an aerial image and render it as a series of RGB map images at different zoom levels, e.g., from (a) to (b) and (c) in Fig. 1. Currently, there are some methods, such as Pix2Pix [3], Pix2PixHD [4], CycleGAN [5], GeoGAN [6], and SMAPGAN [7], that can automatically generate images in map style from aerial images. But they do not explicitly understand the aerial images at pixel level and only consider map generation at a single zoom level. As shown in Fig. 1(d)–(f), these methods

Manuscript received 13 December 2021; revised 17 February 2022; accepted 8 April 2022. Date of publication 27 April 2022; date of current version 1 August 2022. This work was supported by the National Natural Science Foundation of China under Grant 62171038, Grant 62088101, and Grant 61827901. *(Corresponding author: Ying Fu.)*

The authors are with the Beijing Laboratory of Intelligent Information Technology, School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China (e-mail: chenlinwei@bit.edu.cn; fangzheng@bit.edu.cn; fuying@bit.edu.cn).

Dataset is [Online]. Available: https://github.com/ying-fu/MLM.
Digital Object Identifier 10.1109/JSTARS.2022.3170591

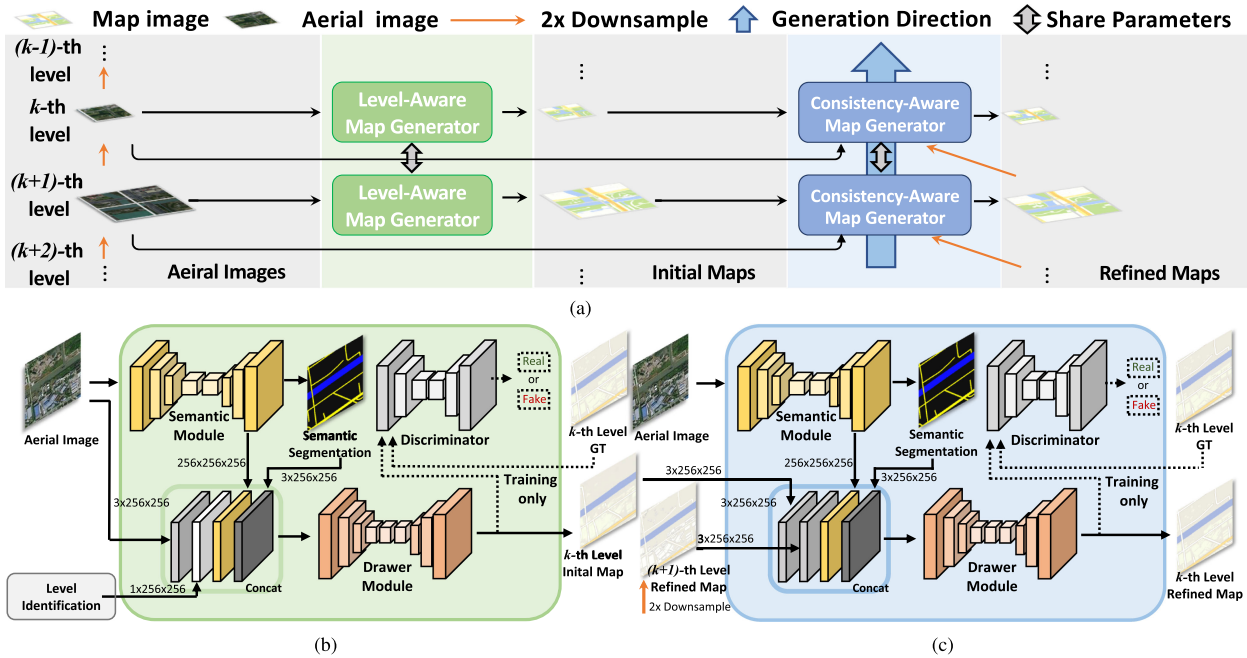


Fig. 2. Illustration of our proposed MLM generation framework. (a) Overview of our pipeline. The LAMG first generates accurate initial maps for each zoom level, and then the CAMG takes in the sequence of initial maps at different zoom levels and builds a connection between maps at different zoom levels in order to keep the content consistent and refine maps at each zoom level. (b) Details of level-aware map generator. (c) Details of consistency-aware map generator.

easily result in inaccurate map generation, blur, artifacts, and severe content inconsistency between different zoom levels. Thus, it is necessary to design a MLM generation method to avoid these problems. Besides, there is still no specialized MLM dataset, which should provide paired samples of aerial images and corresponding map images at multiple zoom levels. This hinders the development and evaluation of MLM generation methods.

In this article, we present a novel MLM generation method that can generate accurate and consistent MLM from aerial images. The proposed framework consists of a level-aware map generator (LAMG) and a consistency-aware map generator (CAMG), as shown in Fig. 2. To generate accurate initial maps with realistic details for each zoom level, the LAMG extracts semantic information in aerial images pixelwisely. It learns to generate initial maps in an adversarial way [8] and helps to keep the topological relationship among the geographical elements. To guarantee content consistency between the maps at different zoom levels, the CAMG refines initial maps at each zoom level iteratively by building the connection between them.

Furthermore, we collect a large-scale high-quality MLM dataset. It provides 18 700 paired samples of aerial images and corresponding maps from two representative cities (Shanghai and Rio de Janeiro) and with four zoom levels (15, 16, 17, and 18). Experiments on our MLM dataset show that our method outperforms the previous state-of-the-art map generation methods on both comprehensive quantitative metrics and perceptive quality.

In summary, our main contributions are as follows.

1) We present a novel MLM generation method, which is able to generate accurate and consistent MLMs from the aerial images.
2) We design an LAMG and CAMG, of which the former can generate accurate initial maps for each zoom level and the latter can guarantee content consistency between the maps of different zoom levels.
3) We collect a large-scale high-quality MLM dataset. To our best knowledge, it is the first specialized dataset for MLM generation.

II. Related Work

In this section, we review the traditional cartographic process, map generation methods, and the progress of datasets for map generation.

A. Traditional Cartographic Process

As an effective tool for humans to explore and recognize the world, the map has been widely involved in our daily life for a long time. Online maps have immense influence for they can provide a realistic view of the world to millions of web users [9]–[11]. Their production and update are done mainly in two steps, i.e., collecting vector data and visualizing the data in an appropriate way.

The collection of vector data usually relies on surveying [12], e.g., the data collectors need to walk or drive with professional equipment to check every point of interest and collect relevant data, which is labor-intensive and time-consuming. This process is easily limited by weather, ground surface condition of the city, and data collectors themselves, which may lead to inaccurate or even impossible data collection. After the vector data is collected or updated, a team of cartographic experts is needed to process the raw data, solve the problems that cannot be done automatically [13]–[16], and finally convert it into the latest map

tiles at multiple zoom levels with professional digital tools [17], [18]. In summary, the traditional cartographic process ensures the accuracy of MLMs, but the high cost both in labor and time makes it hard to update timely and limits its applications.

B. Generating Maps from Aerial Images

Aerial images have found wide applications in human life [1], [2], [19], [20]. They inherently contain rich information about the ground surface. Compared with vector data, aerial images are much easier and cheaper to obtain, e.g., airplanes, satellites, and drones are able to collect aerial images and keep them up-to-date.

Existing works, such as dense-global-residual network [21], bias U-Net [22], and road-extractor model [23], focus on extracting the road from aerial images. They are based on semantic segmentation and can achieve decent performance, but they are unable to generate maps. By formulating map generation as an image-to-image translation task [3], there are also many methods, such as Pix2Pix [3], Pix2PixHD [4], CycleGAN [5], GeoGAN [6], and SMAPGAN [7], that can generate images in map style from aerial images based on generative adversarial networks [8]. These methods treat the task of map generation as a simple image translation task, trying to generate maps from aerial images. These methods focus on generating images with diverse and realistic details but do not explicitly understand the content in aerial images, which may result in inaccurate map generation.

Besides, some researchers have presented map generation methods (e.g., GeoGAN [6] and SMAPGAN [7]) on single zoom level based on image-to-image translation techniques [3], [5]. Ganguli *et al.* [6] utilize conditional GAN [24] with reconstruction and style loss to convert aerial images to human-readable maps while Chen *et al.* [7] combine the supervised and unsupervised image-to-image translation methods [3], [5] to improve the map generation quality with limited training samples. These methods only consider generating maps on single zoom level instead of widely used MLMs which are more effective in practical usage, and do not consider the connection between the maps at different zoom levels. These easily cause content inconsistency when directly applied to MLM generation.

C. Datasets for Map Generation

The development of the map dataset is still in its early stage. Although all online tiled map services provide tiled maps with multiple zoom levels instead of single zoom level for better usage, there is no specialized dataset for MLM generation. And there seldom exist datasets [3], [25] that are related to map generation. The dataset in [25] provides maps but has no corresponding aerial images. The dataset in [3] only provides paired samples of aerial images and maps at a single zoom level. To push the development of MLM generation forward, we collect and produce the first MLM dataset.

III. Method

In this section, we first formulate the problem and introduce the motivation, and then we describe the LAMG and CAMG in our framework. Finally, the learning details of our method are provided.

A. Formulation and Motivation

MLM generation is a task that generates map tiles at multiple zoom levels from an aerial image pyramid, which are obtained from a single high-resolution aerial image by a series of $2\times$ downsampling that corresponds to scales of map zoom levels. Although the map at each zoom level is different from each other (e.g., a map at higher zoom level has higher resolution and is able to show more roads in detail), they keep their content consistent because they represent the same corresponding area.

In MLMs, the world map at the kth zoom level usually consists of $2^k \times 2^k$ map tiles, i.e., the number of map tiles at the kth zoom level decreases exponentially when k lowers. Each map tile usually has 256×256 pixels, thus the world map at the $(k+1)$th zoom level has a double resolution of that at the kth zoom level. In this way, the map at a higher zoom level shows more details and the map at a lower zoom level provides wider view.

Previous map generation methods, such as Pix2Pix [3], Pix2PixHD [4], CycleGAN [5], GeoGAN [6], and SMAPGAN [7], mainly focus on generating photo-realistic images or images in map style. They do not consider MLM generation and fail to generate accurate and consistent MLMs. To solve this, we propose a MLM generation framework consisting of an LAMG and CAMG. The former generator can generate accurate initial maps for each zoom level and the latter one can guarantee content consistency between maps at different zoom levels. The overview of our proposed method is illustrated in Fig. 2(a).

B. Model Structure

As shown in Fig. 2, our framework consists of two generators LAMG and CAMG. They share the same model structure, i.e., a semantic module and a drawer module.

In this section, we first introduce the semantic module, and then the details of the drawer module are described. In addition, we also introduce the structure of the discriminator that is utilized for adversarial learning.

1) Semantic Module: The semantic module based on DeepLabv3+ [26] and Xception-65 [27] serves as the backbone. Apart from its semantic segmentation results, we also take its $4\times$ upsampled final feature maps as the input of the drawer module for providing more semantic information.

2) Drawer Module: We design the drawer module with a coarse-to-fine encoder–decoder structure [4]. The drawer module consists of a local subnetwork and a global subnetwork. The global subnetwork focuses on the whole image and the local subnetwork focuses on the details of the image. The structure of the drawer module is shown in Fig. 4. We input the original image and the $2\times$ downsampled image to the local subnetwork and global subnetwork, respectively, and their features are fused by pixelwise addition. The detailed configurations of global subnetwork and local subnetwork are tabulated in Tables I and II, respectively.

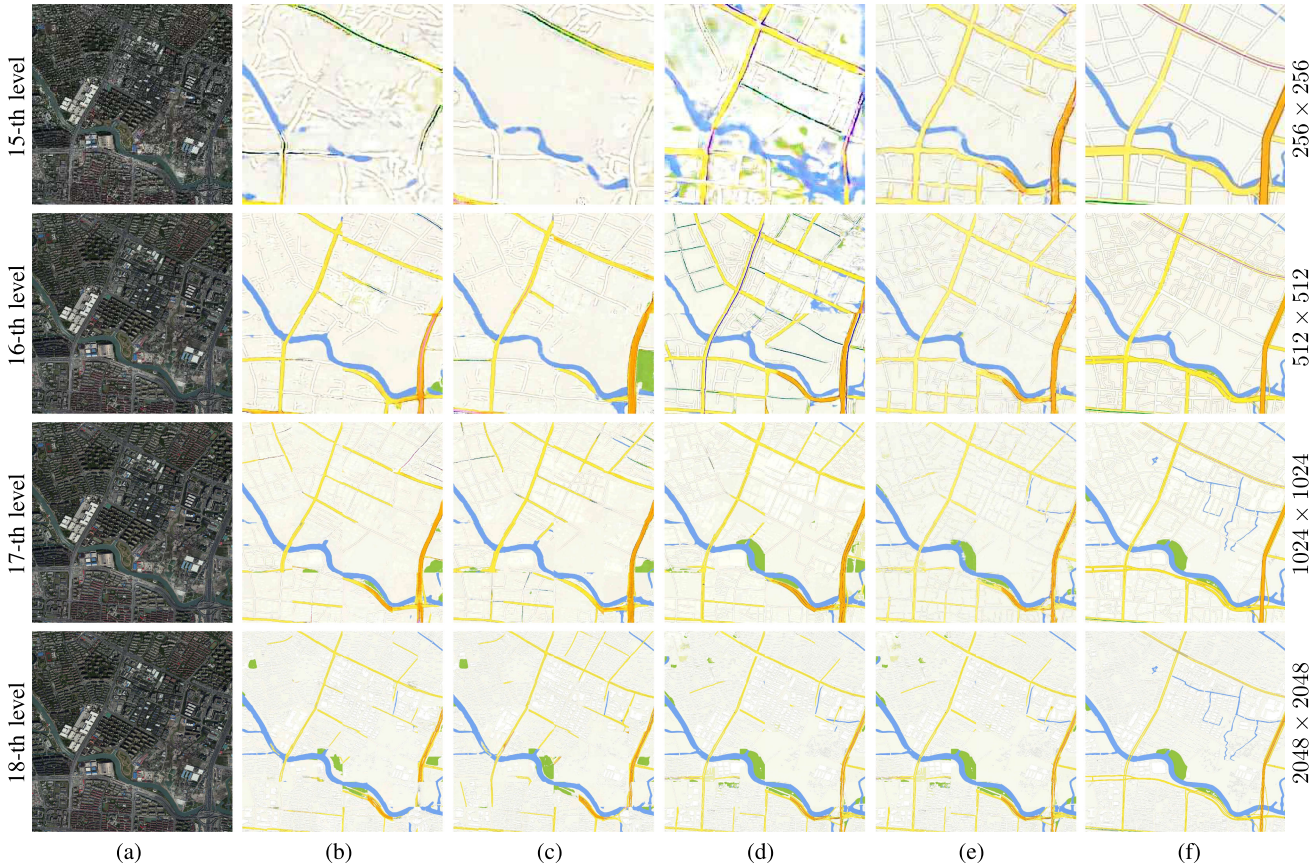


Fig. 3. Visualized results for each part in our pipeline. It can be seen that the accuracy and content consistency gradually improves from column (b) to column (e). LAMG indicates the level-aware map generator and CAMG indicates the consistency-aware map generator. (a) Aerial image. (b) Drawer module. (c) Drawer module + level ID. (d) Drawer module + level ID + semantic module (LAMG). (e) Drawer module + level ID + semantic module + CAMG. (f) Ground truth.

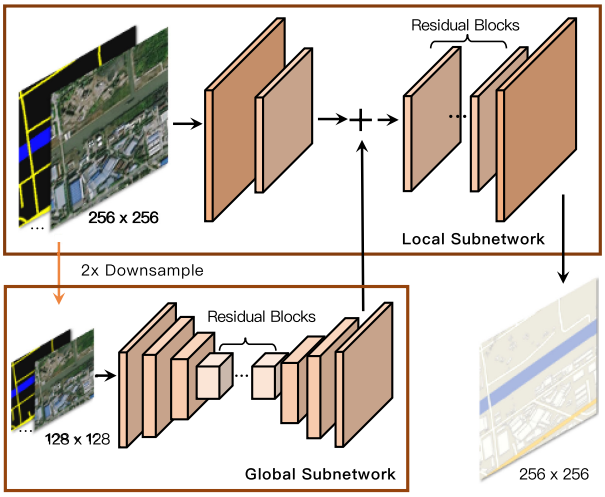


Fig. 4. Illustration of drawer module. "+" indicates pixelwise addition. More details are provided in Tables I and II.

3) Discriminator: We adopt a multiscale discriminator [4] to enhance the ability to differentiate real and synthesized maps at different zoom levels. It consists of three discriminators, which share the same structure, as shown in Table III. We refer to these discriminators as D1, D2, and D3, which operate at different image scales. D1 takes the original images as input, and D2 and D3 take $2\times$ and $4\times$ downsampled images as input, respectively. In this way, discriminators can have different receptive fields and focus on details at different scales that benefits the MLM generation.

C. Level-Aware Map Generator

As shown in Fig. 2(b), the LAMG consists of two modules, i.e., the semantic module and drawer module. To ensure accuracy, we employ a semantic module to explicitly extract pixel-wise semantic information from aerial images, and the drawer module is designed for drawing maps with realistic details.

To avoid the performance degradation caused by domain gap between maps at different zoom levels, we input level identification along with an aerial image to help the generator be level-aware, so as to generate high-quality maps for each zoom level. Thus, the LAMG can be represented by

$$y_k = g_\phi(x_k, f_\theta(x_k), k) \tag{1}$$

where x_k and y_k represent aerial image and initial generated map at the kth zoom level, respectively, and g_ϕ and f_θ indicate semantic module and drawer module of LAMG, respectively.

1) Semantic Module: To display a realistic view of the world and correct cartographic information, the high-quality generated

TABLE I
DETAILED CONFIGURATION OF GLOBAL SUBNETWORK IN DRAWER MODULE

Layer	Layer Information	Norm	Activation	Input → Output Shape
Conv	Conv(K7×7, S1, P3)	IN	ReLU	$(\frac{h}{2}, \frac{w}{2}, \mathrm{N}) \rightarrow (\frac{h}{2}, \frac{w}{2}, 128)$
Conv	Conv(K3×3, S2, P1)	IN	ReLU	$(\frac{h}{2}, \frac{w}{2}, 128) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$
Conv	Conv(K3×3, S2, P1)	IN	ReLU	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{8}, \frac{w}{8}, 512)$
Conv	Conv(K3×3, S2, P1)	IN	ReLU	$(\frac{h}{8}, \frac{w}{8}, 512) \rightarrow (\frac{h}{16}, \frac{w}{16}, 1024)$
ResBlk	Conv(K3×3, S1, P1) Conv(K3×3, S1, P1)	IN IN	ReLU -	$(\frac{h}{16}, \frac{w}{16}, 1024) \rightarrow (\frac{h}{16}, \frac{w}{16}, 1024)$
ResBlk	Conv(K3×3, S1, P1) Conv(K3×3, S1, P1)	IN IN	ReLU -	$(\frac{h}{16}, \frac{w}{16}, 1024) \rightarrow (\frac{h}{16}, \frac{w}{16}, 1024)$
ResBlk	Conv(K3×3, S1, P1) Conv(K3×3, S1, P1)	IN IN	ReLU -	$(\frac{h}{16}, \frac{w}{16}, 1024) \rightarrow (\frac{h}{16}, \frac{w}{16}, 1024)$
ResBlk	Conv(K3×3, S1, P1) Conv(K3×3, S1, P1)	IN IN	ReLU -	$(\frac{h}{16}, \frac{w}{16}, 1024) \rightarrow (\frac{h}{16}, \frac{w}{16}, 1024)$
ResBlk	Conv(K3×3, S1, P1) Conv(K3×3, S1, P1)	IN IN	ReLU -	$(\frac{h}{16}, \frac{w}{16}, 1024) \rightarrow (\frac{h}{16}, \frac{w}{16}, 1024)$
ResBlk	Conv(K3×3, S1, P1) Conv(K3×3, S1, P1)	IN IN	ReLU -	$(\frac{h}{16}, \frac{w}{16}, 1024) \rightarrow (\frac{h}{16}, \frac{w}{16}, 1024)$
ResBlk	Conv(K3×3, S1, P1) Conv(K3×3, S1, P1)	IN IN	ReLU -	$(\frac{h}{16}, \frac{w}{16}, 1024) \rightarrow (\frac{h}{16}, \frac{w}{16}, 1024)$
ResBlk	Conv(K3×3, S1, P1) Conv(K3×3, S1, P1)	IN IN	ReLU -	$(\frac{h}{16}, \frac{w}{16}, 1024) \rightarrow (\frac{h}{16}, \frac{w}{16}, 1024)$
ResBlk	Conv(K3×3, S1, P1) Conv(K3×3, S1, P1)	IN IN	ReLU -	$(\frac{h}{16}, \frac{w}{16}, 1024) \rightarrow (\frac{h}{16}, \frac{w}{16}, 1024)$
DeConv	DeConv(K3×3, S2, P1)	IN	ReLU	$(\frac{h}{16}, \frac{w}{16}, 1024) \rightarrow (\frac{h}{8}, \frac{w}{8}, 512)$
DeConv	DeConv(K3×3, S2, P1)	IN	ReLU	$(\frac{h}{8}, \frac{w}{8}, 512) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$
DeConv	DeConv(K3×3, S2, P1)	IN	ReLU	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{2}, \frac{w}{2}, 128)$

Notes: Conv: convolutional layer, DeConv: deconvolutional layer, N: the number of input channels, K: kernel size, S: stride size, P: padding size, IN: instance normalization.

TABLE II
DETAILED CONFIGURATION OF LOCAL SUBNETWORK IN DRAWER MODULE

Layer	Layer Information	Norm	Activation	Input → Output Shape
Conv	Conv(K7×7, S1, P3)	IN	ReLU	$(h, w, \mathrm{N}) \rightarrow (h, w, 64)$
Conv	Conv(K3×3, S2, P1)	IN	ReLU	$(h, w, 64) \rightarrow (\frac{h}{2}, \frac{w}{2}, 128)$
ResBlk	Conv(K3×3, S1, P1) Conv(K3×3, S1, P1)	IN IN	ReLU -	$(\frac{h}{2}, \frac{w}{2}, 128) \rightarrow (\frac{h}{2}, \frac{w}{2}, 128)$
ResBlk	Conv(K3×3, S1, P1) Conv(K3×3, S1, P1)	IN IN	ReLU -	$(\frac{h}{2}, \frac{w}{2}, 128) \rightarrow (\frac{h}{2}, \frac{w}{2}, 128)$
ResBlk	Conv(K3×3, S1, P1) Conv(K3×3, S1, P1)	IN IN	ReLU -	$(\frac{h}{2}, \frac{w}{2}, 128) \rightarrow (\frac{h}{2}, \frac{w}{2}, 128)$
DeConv	DeConv(K3×3, S2, P1)	IN	ReLU	$(\frac{h}{2}, \frac{w}{2}, 128) \rightarrow (h, w, 64)$
Conv	Conv(K7×7, S1, P3)	-	Tanh	$(h, w, 64) \rightarrow (h, w, 3)$

Notes: Conv: convolutional layer, DeConv: deconvolutional layer, N: the number of input channels, K: kernel size, S: stride size, P: padding size, IN: instance normalization.

maps should not only be images in map style but also provide accurate semantic information (e.g., the locations of roads and rivers). Existing single-level map generation methods [6], [7] focus on generating images with realistic details and do not explicitly force the model to learn and understand the content of the input image pixelwisely, which may lead to inferior and erroneous map generation. To better keep the topological relationship among geographical elements and help the drawer module to generate maps with high accuracy, we design a semantic module that can learn to assign each pixel a semantic category (e.g., road, water area, and land background) and understand the aerial image pixelwisely. The semantic module is based on the semantic segmentation model DeepLabv3+ [26], we choose it for its stable performance and easy training procedure.

TABLE III
DETAILED CONFIGURATION OF DISCRIMINATOR IN MULTISCALE DISCRIMINATOR

Layer	Layer Information	Norm	Activation	Input → Output Shape
Conv	Conv(K4×4, S2, P2)	-	LeakyReLU	$(h, w, 6) \rightarrow (\frac{h}{2}+2, \frac{w}{2}+2, 64)$
Conv	Conv(K4×4, S2, P2)	IN	LeakyReLU	$(\frac{h}{2}+2, \frac{w}{2}+2, 64) \rightarrow (\frac{h}{4}+3, \frac{w}{4}+3, 128)$
Conv	Conv(K4×4, S2, P2)	IN	LeakyReLU	$(\frac{h}{4}+3, \frac{w}{4}+3, 128) \rightarrow (\frac{h}{8}+3, \frac{w}{8}+3, 256)$
Conv	Conv(K4×4, S1, P2)	IN	LeakyReLU	$(\frac{h}{8}+3, \frac{w}{8}+3, 256) \rightarrow (\frac{h}{8}+6, \frac{w}{8}+6, 512)$
Conv	Conv(K4×4, S1, P2)	-	-	$(\frac{h}{8}+6, \frac{w}{8}+6, 512) \rightarrow (\frac{h}{8}+9, \frac{w}{8}+9, 1)$

Notes: Conv: convolutional layer, DeConv: deconvolutional layer, K: kernel size, S: stride size, P: padding size, IN: instance normalization.

The semantic module can provide pixelwise semantic information whose role is similar to vector data in the traditional cartographic process. It improves the accuracy of the generated map and keeps the right topological relations of map elements, as shown in Fig. 3(c) and 3(d).

2) Drawer Module: The drawer module is the basic module of the generator. As shown in Fig. 3(b), it can learn to generate maps with realistic details independently by adversarial learning [3], [8]. To optimize the quality of map details, it also takes aerial images along with level identification as input [see Fig. 3(c)]. To improve the generation accuracy, it takes feature maps and semantic segmentation results from the semantic module as an input and translates them to map with accurate geographical information [see Fig. 3(d)].

Its role is similar to cartographic experts in the traditional cartographic process. On the basis of pixelwise semantic information of aerial images from the semantic module, the drawer module can learn to draw a realistic map with accurate information by taking semantic segmentation, zoom level, and aerial image into consideration.

3) Level Identification: Compared with the single-level map generation, MLM generation is more challenging because it has to generate maps from aerial images for multiple zoom levels, and keeps their content consistent. A naive solution is to learn a mapping model for each zoom level, but it is inconvenient and makes the training procedure complicated. Moreover, the amount of training samples for each zoom level decreases exponentially when the zoom level lowers, which results in insufficient samples for the training model at a low zoom level.

To solve this problem, we utilize level identification as guidance information and input it into the generator. The level identification is $\mathbb{R}^{1\times H\times W}$, which simply repeats the zoom level number k for $H \times W$ times. It guides the generator to generate an appropriate map for each zoom level, and all the samples from different zoom levels can be used for training the generator in a unified way, achieving better performance especially for generating maps at low zoom level, e.g., the river at the 15th level in Fig. 3(c) is clearer than that in Fig. 3(b).

D. Consistency-Aware Map Generator

LAMG can generate an appropriate and accurate initial map for each zoom level. But due to the lack of connection between the generated maps at different zoom levels, there may exist content inconsistency between maps at different zoom levels,

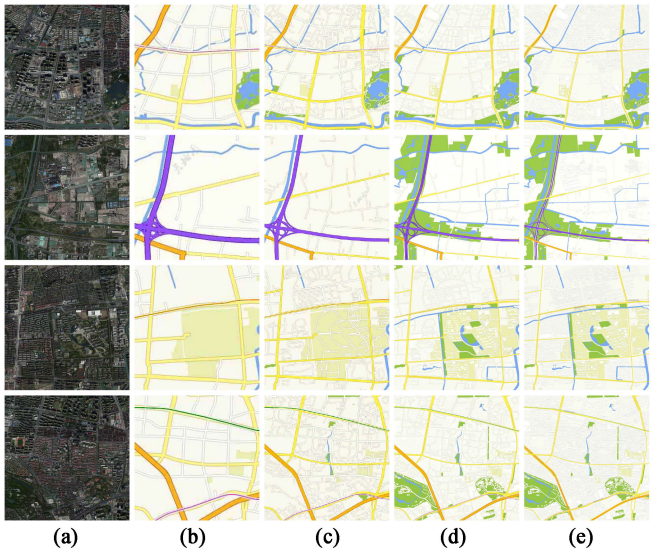


Fig. 5. Examples of our Multi-Level Map dataset from Shanghai. (a) Aerial image. (b) 15th level. (c) 16th level. (d) 17th level. (e) 18th level.

e.g., as shown in Fig. 1(d) and 1(e), the generated map at the 16th zoom level shows a river while a land shows up in the same area of generated map at the 15th zoom level.

Therefore, we employ the CAMG in our framework and build the connection between the generated maps at different zoom levels. As shown in Fig. 2(a), we view the initial generated maps at K different zoom levels as sequential data $\{y_1, y_2, \ldots, y_K\}$. The "frame" in map sequence is different in shape, i.e., the height and width of the map at the $(k+1)$th zoom level is twice the size of the map at the kth zoom level. Considering that the higher resolution of a map at higher zoom level is able to provide richer information for refining initial map of each level, we optimize the generation and refine the MLM sequence from high to low zoom level in an iterative way as follows:

$$y_k^r = g'_{\phi'}(f'_{\theta'}(x_k), y_k, y_{k+1}^r) \tag{2}$$

where y_k^r and y_{k+1}^r are the refined maps at the kth and the $(k+1)$th zoom level, and $g'_{\phi'}$ and $f'_{\theta'}$ indicate semantic module and drawer module of the CAMG, respectively. Note that we regard the initial map at highest zoom level y_K as a refined map y_K^r for the refinement of y_{K-1}. The refinement defined in (2) builds a connection between the adjacent zoom levels. By applying it iteratively, we can build a connection across different zoom levels to ensure their consistency, e.g., the refined map y_{k-1}^r can be obtained from y_k and y_{k+1}^r as follows:

$$y_{k-1}^r = g'_{\phi'}(f'_{\theta'}(x_{k-1}), y_{k-1}, g'_{\phi'}(f'_{\theta'}(x_k), y_k, y_{k+1}^r)). \tag{3}$$

It largely improves the accuracy of the generated map and keeps the content consistency between different levels, as shown in Fig. 3(e).

The structure of the CAMG is the same as the LAMG, which also consists of a semantic module and a drawer module. The main difference is their input, i.e., the LAMG takes an aerial image and level identification as input, while the CAMG takes an aerial image and initial maps from the kth and the $(k+1)$th zoom level as input, as shown in Fig. 2(c).

TABLE IV
COMPARISON WITH THE EXISTING MAP DATASETS

Dataset	Resolution	Samples	Aerial	Map	Levels
Kang *et al.* [25]	256×256	10,500	×	√	2
Isola *et al.* [3]	256×256	2,194	√	√	1
MLM (ours)	256×256	18,700	√	√	4

Notes: MLM provides larger number of samples and in more zoom levels.

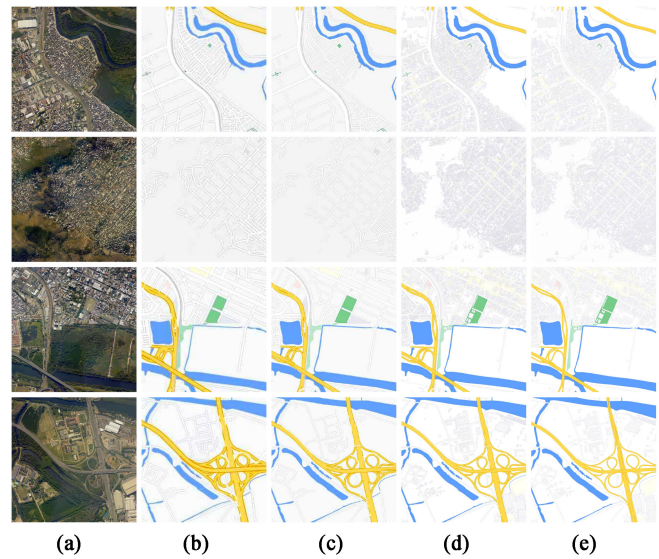


Fig. 6. Examples of our Multi-Level Map dataset from Rio de Janeiro. (a) Aerial image. (b) 15th level. (c) 16th level. (d) 17th level. (e) 18th level.

TABLE V
DETAILS OF OUR MLM DATASET

Zoom Level	Train Samples SH | RJ	Test Samples SH | RJ	Train + Test SH | RJ	Spatial Resolution
15	96 | 80	24 | 20	120 | 100	4.8 m/pixel
16	384 | 320	96 | 80	480 | 400	2.4 m/pixel
17	1,536 | 1,280	384 | 320	1,920 | 1,600	1.2 m/pixel
18	6,144 | 5,120	1,536 | 1,280	7,680 | 6,400	0.6 m/pixel
Total	8,160 | 6,800	2,040 | 1,700	10,200 | 8,500	**All 18,700**

Notes: It consists of samples from Shanghai (SH) region and Rio De Janeiro (RJ) region.

E. Learning Details

In this section, we first introduce the loss function and then describe the implementation details. Note that the LAMG and CAMG utilize the same loss function.

1) Losses for Semantic Module: The semantic module is trained to learn to understand the aerial images pixelwisely, we follow [26], [39], [40], and [41] to use pixelwise cross-entropy loss for training. And to better adapt to hard samples, we adopt focal loss [42] version of it

$$\mathcal{L}_{SM} = \frac{1}{H \times W} \sum_{i=1}^{H \times W} \mathrm{FL}(\mathrm{p_i}, \mathrm{y_i^{seg}})$$

$$\mathrm{FL}(p_i, y_i^{seg}) = \begin{cases} -\alpha_t (1-p_{it})^{\gamma} \log(p_{it}) & f y_i^{\mathrm{seg}} = t \\ -\alpha_t \, p_{it}^{\gamma} \log(1-p_{it}) & \text{otherwise} \end{cases} \tag{4}$$

TABLE VI
DETAILS OF EVALUATION METRICS

	Introduction	Details
FID↓	FID [28] is robust to noise and consistent with human perception. It uses the Fréchet distance between two multivariate Gaussians $$\mathrm{FID}(\mathbb{P}_r,\mathbb{P}_g)=\lVert\mu_r-\mu_g\rVert^2+\mathtt{Tr}(\mathtt{C}_r+\mathtt{C}_g-2(\mathtt{C}_r\mathtt{C}_g)^{\frac{1}{2}}).$$	$\mathbb{P}_r,\mathbb{P}_g$ indicate the distribution of real samples and generated samples respectively. $\mathtt{C}$ is the covariance matrix of feature vector. $\mathtt{Tr}$ is the trace of matrix. We follow [28] to use pre-trained InceptionV3 [29] network to extract features.
KID↓	KID [30] can be seen as an improved version of FID [28], its estimates are unbiased and asymptotically normal $$\begin{aligned}\mathrm{KID}(\mathbb{P}_r,\mathbb{P}_g)=&\mathbb{E}_{\substack{\mathtt{x}_r,\mathtt{x}'_r\sim\mathbb{P}_r,\\ \mathtt{x}_g,\mathtt{x}'_g\sim\mathbb{P}_g}}[k(I(\mathtt{x}_r),I(\mathtt{x}'_r))\\ &-2k(I(\mathtt{x}_r),I(\mathtt{x}_g))+k(I(\mathtt{x}_g),I(\mathtt{x}'_g))].\end{aligned}$$	$I(\mathtt{x}_r)$, $I(\mathtt{x}_g)$ are features obtained from real samples and generated samples respectively. k indicates the polynomial kernel function, $k(a,b)=(\frac{1}{d}a^{\mathrm{T}}b+1)^3$, where d is the representation dimension. We follow [30] to use pre-trained InceptionV3 [29] network to extract features.
KMMD↓	KMMD [31] uses fixed kernel function to measure the dissimilarity between real samples and generated samples $$\begin{aligned}\mathrm{MMD}^2(\mathbb{P}_r,\mathbb{P}_g)=&\mathbb{E}_{\substack{\mathtt{x}_r,\mathtt{x}'_r\sim\mathbb{P}_r,\\ \mathtt{x}_g,\mathtt{x}'_g\sim\mathbb{P}_g}}[k(\mathtt{x}_r,\mathtt{x}'_r)\\ &-2k(\mathtt{x}_r,\mathtt{x}_g)+k(\mathtt{x}_g,\mathtt{x}'_g)].\end{aligned}$$	$\mathtt{x}_r$ and $\mathtt{x}_g$ is the sample feature from true distribution and generated distribution. k is the gaussian kernel function. Results in [32] show KMMD appears to be good metric in terms of discriminability, robustness and efficiency with pre-trained ResNet [33]. Thus, we use pre-trained ResNet [33] for feature extraction.
IS↑	IS [34] has a reasonable correlation with human judgment of image quality and is able to show the diversity of generated images $$\mathrm{IS}(\mathbb{P}_g)=\exp(\mathbb{E}_{\mathtt{x}\sim\mathbb{P}_g}[KL(p_{\mathcal{M}}(y\vert\mathtt{x})\vert\vert p_{\mathcal{M}}(y))]).$$	$p_{\mathcal{M}}(y\vert\mathtt{x})$ is the label distribution of $\mathtt{x}$ as predicted by model $\mathcal{M}$, and $p_{\mathcal{M}}(y)=\int_{\mathtt{x}}p_{\mathcal{M}}(y\vert\mathtt{x})\mathrm{d}\mathbb{P}_g$. KL indicates Kullback-Leibler divergence. We follow [34] to use InceptionV3 [29] network to extract the probability distribution of each category.
PSNR↑	PSNR is commonly used to measure the quality of image by comparing samples with original images pixel-wisely $$\mathrm{PSNR}=10\times\log_{10}(\frac{MAX_I^2}{MSE}).$$	MAX_I is the maximum possible pixel value of the image. MSE is mean-square error.

Notes: Up arrow means higher is better and down arrow means lower is better.

where H and W indicate the height and width of the image, respectively, y^{seg} is the ground truth semantic segmentation label, $p_{it}\in[0,1]$ is the predicted confidence value of pixel i for category t, and α_t and γ are the hyperparameters of focal loss [42], whose values are 1.0 and 2. The ground truth semantic segmentation label is obtained from the real map by mapping different colors of the map to semantic categories, e.g., blue indicates rivers, yellow indicates roads, etc. We mainly focus on segmenting three semantic categories, i.e., roads, water area, and land.

2) Losses for Drawer Module: To generate high-quality maps with realistic details, we adopt content loss, adversarial loss, and perceptual loss for the training drawer module.

The content loss [3], [4] improves the similarity between generated map and real map

$$\mathcal{L}_{\text{con}}=\frac{1}{H\times W}\sum_{i=1}^{H\times W}\lVert G(x)-y^{\text{map}}\rVert_1 \tag{5}$$

where y^{map} is ground truth map and G is the generator.

The adversarial loss [8], [24] is able to make the details of the image more realistic:

$$\begin{aligned}\mathcal{L}_{\text{adv}}&=\min_G\max_D V(D,G)\\ &=\mathbb{E}_{x,y^{\text{map}}}[\log D(x,y^{\text{map}})]\\ &\quad+\mathbb{E}_{y^{\text{map}}}[\log(1-D(x,G(x)))]\end{aligned} \tag{6}$$

where D is the discriminator, G is the generator, and x and y^{map} indicate the aerial image and map image. To enhance the discriminator's ability to differentiate real and synthesized maps at different zoom level, we adopt multi-scale discriminator [4].

The perceptual loss [4], [43] is utilized to improve the perceptual quality of the generated map as

$$\mathcal{L}_{\text{per}}=\sum_{i=1}^{L}\frac{1}{N_i}\lVert F^{(i)}(y^{\text{map}})-F^{(i)}(G(x))\rVert_1 \tag{7}$$

where F is feature extraction network, $F^{(i)}$ indicates the ith layer of feature extraction network, N_i is the number of pixels of the corresponding feature map, and L is the number of layers. We use a pretrained VGG model [44] as a feature extraction network F.

To better stabilize the training procedure [4], we adopt feature-matching loss for training. Specifically, we extract features from multiple layers of the discriminator, and learn to match these intermediate representations from the real and generated maps

$$\mathcal{L}_{\text{fm}}=\sum_{i=1}^{L}\frac{1}{N_i}\lVert D^{(i)}(x,y^{\text{map}})-D^{(i)}(x,G(x))\rVert_1 \tag{8}$$

where $D^{(i)}$ means the first i layers of the discriminator D, L means the total number of layers, and N_i denotes the number of elements in each layer.

Thus, the total loss function for drawer module is

$$\mathcal{L}_{\text{DM}}=\lambda_{\text{con}}\mathcal{L}_{\text{con}}+\lambda_{\text{adv}}\mathcal{L}_{\text{adv}}+\lambda_{\text{per}}\mathcal{L}_{\text{per}}+\lambda_{\text{fm}}\mathcal{L}_{\text{fm}} \tag{9}$$

TABLE VII
QUANTITATIVE RESULTS ON MULTILEVEL MAP SHANGHAI (MLM-SH) DATASET

Level	Method	FID↓	KID↓	KMMD↓	IS↑	PSNR↑
15	Pix2Pix [3]	380.3	0.4179	0.6770	2.057	19.98
	Pix2PixHD [4]	371.7	0.2914	0.6324	2.565	19.50
	CycleGAN [5]	345.5	0.3437	0.6641	2.178	19.65
	GeoGAN [6]	534.6	0.6836	1.0008	1.374	20.30
	SMAPGAN [7]	391.9	0.4281	0.7027	1.938	20.96
	SPADE [35]	516.7	0.5265	0.9095	1.496	19.91
	SelectionGAN [36]	328.4	0.2672	0.5678	2.139	19.39
	TSIT [37]	323.3	0.3488	0.6833	1.633	19.47
	LPTN [38]	411.2	0.4380	0.8249	1.977	17.64
	Ours	**239.2**	**0.1268**	**0.3823**	**3.019**	**21.75**
16	Pix2Pix [3]	291.0	0.3181	0.6035	2.250	20.24
	Pix2PixHD [4]	296.7	0.1839	0.5031	3.019	20.17
	CycleGAN [5]	232.7	0.2333	0.5325	2.784	19.91
	GeoGAN [6]	448.1	0.5827	0.9164	1.528	20.61
	SMAPGAN [7]	392.7	0.4228	0.6709	2.155	21.42
	SPADE [35]	375.0	0.4369	0.7665	1.717	20.16
	SelectionGAN [36]	262.7	0.2332	0.4724	2.705	19.91
	TSIT [37]	177.8	0.1775	0.4864	2.056	19.73
	LPTN [38]	322.4	0.3700	0.7045	2.299	17.12
	Ours	**144.3**	**0.1032**	**0.3726**	**3.251**	**21.55**
17	Pix2Pix [3]	254.9	0.2160	0.5068	2.627	20.33
	Pix2PixHD [4]	246.7	0.1138	0.3943	3.170	20.64
	CycleGAN [5]	206.0	0.1461	0.4657	3.029	19.99
	GeoGAN [6]	384.9	0.3543	0.7941	1.735	21.22
	SMAPGAN [7]	308.1	0.2754	0.5967	2.484	21.96
	SPADE [35]	321.2	0.2829	0.6625	2.031	20.13
	SelectionGAN [36]	236.8	0.1557	0.4183	3.019	20.64
	TSIT [37]	143.5	0.1111	0.3863	2.352	19.81
	LPTN [38]	276.6	0.2067	0.6000	2.574	16.73
	Ours	**116.0**	**0.0684**	**0.3008**	**3.322**	**22.05**
18	Pix2Pix [3]	92.3	0.0667	0.2770	3.590	21.71
	Pix2PixHD [4]	140.8	0.0850	0.3015	3.055	22.30
	CycleGAN [5]	68.4	0.0477	0.2455	2.787	21.14
	GeoGAN [6]	291.8	0.2541	0.6772	1.866	22.49
	SMAPGAN [7]	213.8	0.2118	0.4767	2.307	**23.79**
	SPADE [35]	156.9	0.1317	0.4232	2.285	21.69
	SelectionGAN [36]	112.3	0.0802	0.2662	3.101	22.04
	TSIT [37]	80.1	0.0596	0.2419	2.688	21.45
	LPTN [38]	219.5	0.1503	0.5142	3.065	17.55
	Ours	**51.1**	**0.0237**	**0.1273**	**3.656**	22.71
Avg.	Pix2Pix [3]	254.6	0.2547	0.5161	2.631	20.57
	Pix2PixHD [4]	264.0	0.1685	0.4578	2.952	20.65
	CycleGAN [5]	213.2	0.1927	0.4770	2.695	20.17
	GeoGAN [6]	414.9	0.4687	0.8471	1.626	21.16
	SMAPGAN [7]	326.6	0.3345	0.6117	2.221	**22.03**
	SPADE [35]	342.5	0.3445	0.6905	1.882	20.47
	SelectionGAN [36]	235.0	0.1841	0.4312	2.741	20.50
	TSIT [37]	181.2	0.1743	0.4495	2.182	20.11
	LPTN [38]	307.4	0.2913	0.6609	2.479	17.26
	Ours	**137.7**	**0.0805**	**0.2957**	**3.312**	22.01

Notes: "Avg." indicates results that average over 15–18 zoom levels and also shows consistency results.

TABLE VIII
QUANTITATIVE RESULTS ON MULTILEVEL MAP RIO DE JANEIRO (MLM-RJ) DATASET

Level	Method	FID↓	KID↓	KMMD↓	IS↑	PSNR↑
15	Pix2Pix [3]	280.6	0.2388	0.5640	1.961	22.90
	Pix2PixHD [4]	249.5	0.1790	0.5244	1.989	24.48
	CycleGAN [5]	319.7	0.2945	0.6151	1.686	23.39
	GeoGAN [6]	548.7	0.4809	0.9097	1.170	25.51
	SMAPGAN [7]	218.4	0.1031	0.3486	1.979	**25.81**
	SPADE [35]	358.3	0.3637	0.6665	2.068	24.22
	SelectionGAN [36]	282.3	0.2328	0.5261	1.797	24.83
	TSIT [37]	286.7	0.2486	0.5245	1.628	24.25
	LPTN [38]	379.7	0.3364	0.7346	1.753	21.48
	Ours	**172.2**	**0.0644**	**0.2548**	**2.212**	25.73
16	Pix2Pix [3]	243.5	0.1677	0.5587	1.896	24.27
	Pix2PixHD [4]	174.0	0.0608	0.2714	2.824	26.14
	CycleGAN [5]	277.1	0.2354	0.5900	1.881	24.73
	GeoGAN [6]	491.4	0.3908	0.8700	1.307	27.16
	SMAPGAN [7]	181.5	0.0670	0.2398	2.774	**27.88**
	SPADE [35]	323.5	0.2766	0.6547	2.228	25.61
	SelectionGAN [36]	217.5	0.1142	0.3756	2.759	26.02
	TSIT [37]	247.0	0.1713	0.5434	2.003	25.63
	LPTN [38]	356.1	0.2440	0.7114	1.796	25.40
	Ours	**167.7**	**0.0652**	**0.2371**	**3.333**	27.52
17	Pix2Pix [3]	104.6	0.0670	0.3668	2.059	24.93
	Pix2PixHD [4]	54.9	0.0177	0.1516	2.557	24.81
	CycleGAN [5]	53.4	0.0203	0.2244	2.372	25.09
	GeoGAN [6]	380.1	0.3367	0.8179	1.374	26.49
	SMAPGAN [7]	303.5	0.2495	0.5861	2.674	**26.72**
	SPADE [35]	145.1	0.1070	0.4700	2.372	25.03
	SelectionGAN [36]	84.6	0.0450	0.2362	**2.728**	24.75
	TSIT [37]	97.5	0.0734	0.3369	2.196	25.10
	LPTN [38]	241.7	0.1519	0.5968	1.879	25.26
	Ours	**52.1**	**0.0153**	**0.1177**	2.527	25.14
18	Pix2Pix [3]	52.0	0.0204	0.1912	1.985	26.91
	Pix2PixHD [4]	42.1	0.0126	0.1245	2.528	27.02
	CycleGAN [5]	44.0	0.0139	0.1395	2.181	26.75
	GeoGAN [6]	407.6	0.2494	0.8428	1.405	28.33
	SMAPGAN [7]	317.1	0.1600	0.6077	2.109	**28.88**
	SPADE [35]	116.3	0.0705	0.3956	2.373	26.69
	SelectionGAN [36]	78.3	0.0451	0.2046	2.440	26.64
	TSIT [37]	95.3	0.0517	0.3493	2.312	26.79
	LPTN [38]	200.4	0.0916	0.5360	2.043	26.08
	Ours	**32.7**	**0.0061**	**0.0762**	**2.719**	26.92
Avg.	Pix2Pix [3]	170.2	0.1235	0.4202	1.975	24.75
	Pix2PixHD [4]	130.1	0.0675	0.2680	2.475	25.61
	CycleGAN [5]	173.6	0.1410	0.3922	2.030	24.99
	GeoGAN [6]	457.0	0.3644	0.8601	1.314	26.87
	SMAPGAN [7]	255.1	0.1449	0.4456	2.384	**27.32**
	SPADE [35]	235.8	0.2044	0.5467	2.260	25.39
	SelectionGAN [36]	165.7	0.1093	0.3356	2.431	25.56
	TSIT [37]	181.6	0.1363	0.4385	2.035	25.44
	LPTN [38]	294.5	0.2060	0.6447	1.868	24.56
	Ours	**106.2**	**0.0378**	**0.1715**	**2.698**	26.33

Notes: "Avg." indicates results that average over 15–18 zoom levels and also shows consistency results.

where we let $\lambda_{\text{con}} = \lambda_{\text{per}} = \lambda_{\text{fm}} = 10$ and $\lambda_{\text{adv}} = 1$ for loss balancing.

3) Implementation Details: We first train the LAMG, then fix it to help to optimize the CAMG. The training procedures of the two generators are the same. The input size of paired map tile is 256×256.

We train the semantic module and drawer module jointly with Adam optimizer [45]. We set the initial learning rate of the semantic module, backbone of the semantic module, and the drawer module to 7×10^{-4}, 7×10^{-5}, and 2×10^{-4}, respectively, for balancing the learning of each part, and the batch size is 4. Both generators are trained for 100 epochs on a GeForce RTX 3090. The semantic module adopts a polynomial learning rate decay schedule [26] while the learning rate of the drawer module is fixed for the first 50 epochs, and linearly decays to 0 for the rest 50 epochs.

IV. MULTILEVEL MAP DATASET

To develop and evaluate learning-based MLM generation methods, a dataset consisting of paired samples of aerial images and maps with multiple zoom levels is necessary needed. Thus, we collect and produce a large-scale high-quality MLM dataset for map generation at multiple zoom levels. The data comes from Google Maps and Tianditu. It has the following distinctive characteristics.

1) *Region diversity:* As shown in Figs. 5 and 6, our MLM dataset provides data for two regions with different geographical features, i.e., Shanghai and Rio de Janeiro,

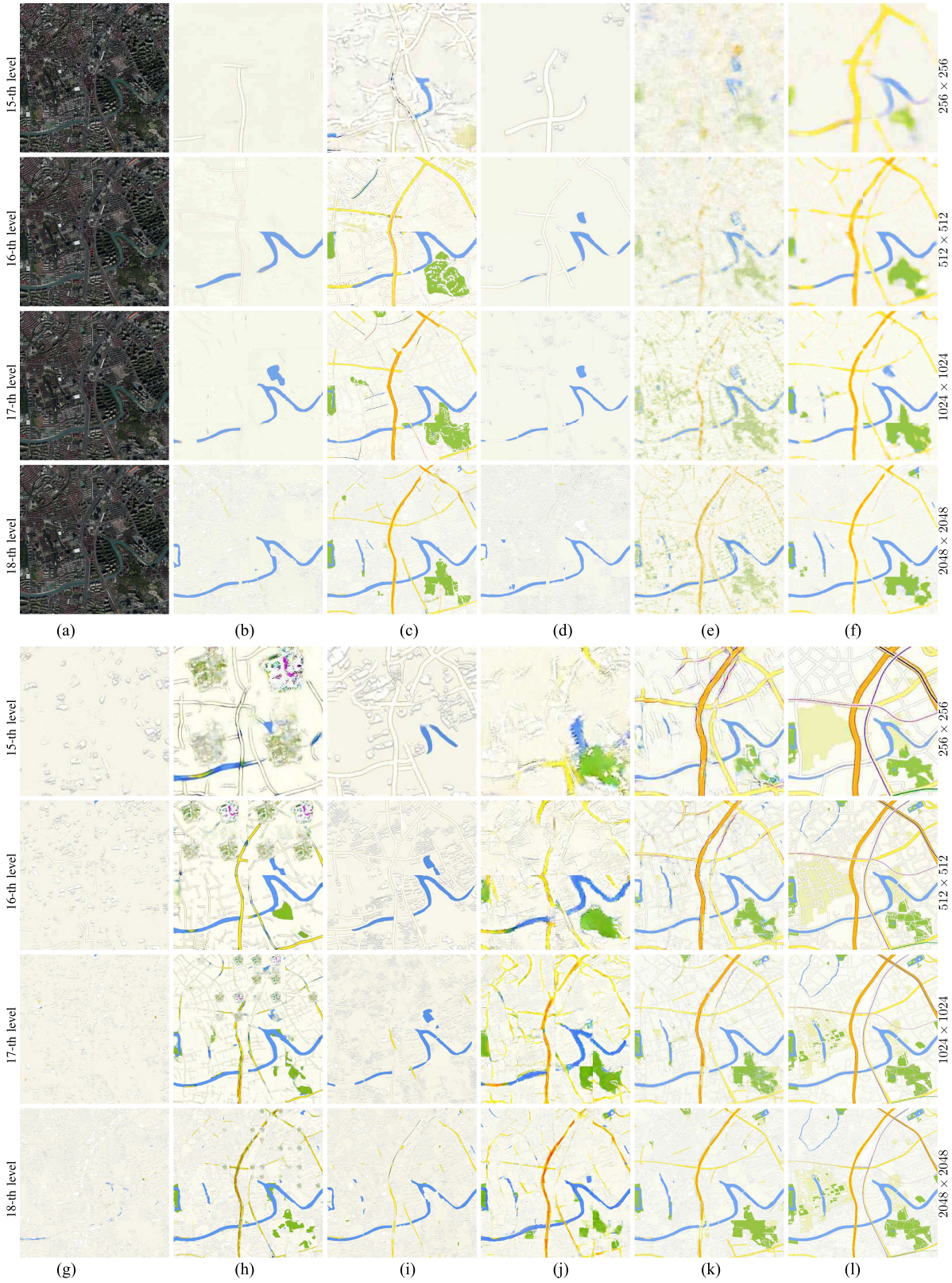


Fig. 7. Visualized results of MLM-SH generated by our method and counterparts. Our method shows satisfactory accuracy and content consistency especially for rivers and roads in all zoom levels. (a) Aerial image. (b) Pix2Pix. (c) Pix2PixHD. (d) CycleGAN. (e) GeoGAN. (f) SMAPGAN. (g) SPADE. (h) SelectionGAN. (i) TSIT. (j) LPTN. (k) Ours. (l) GT.

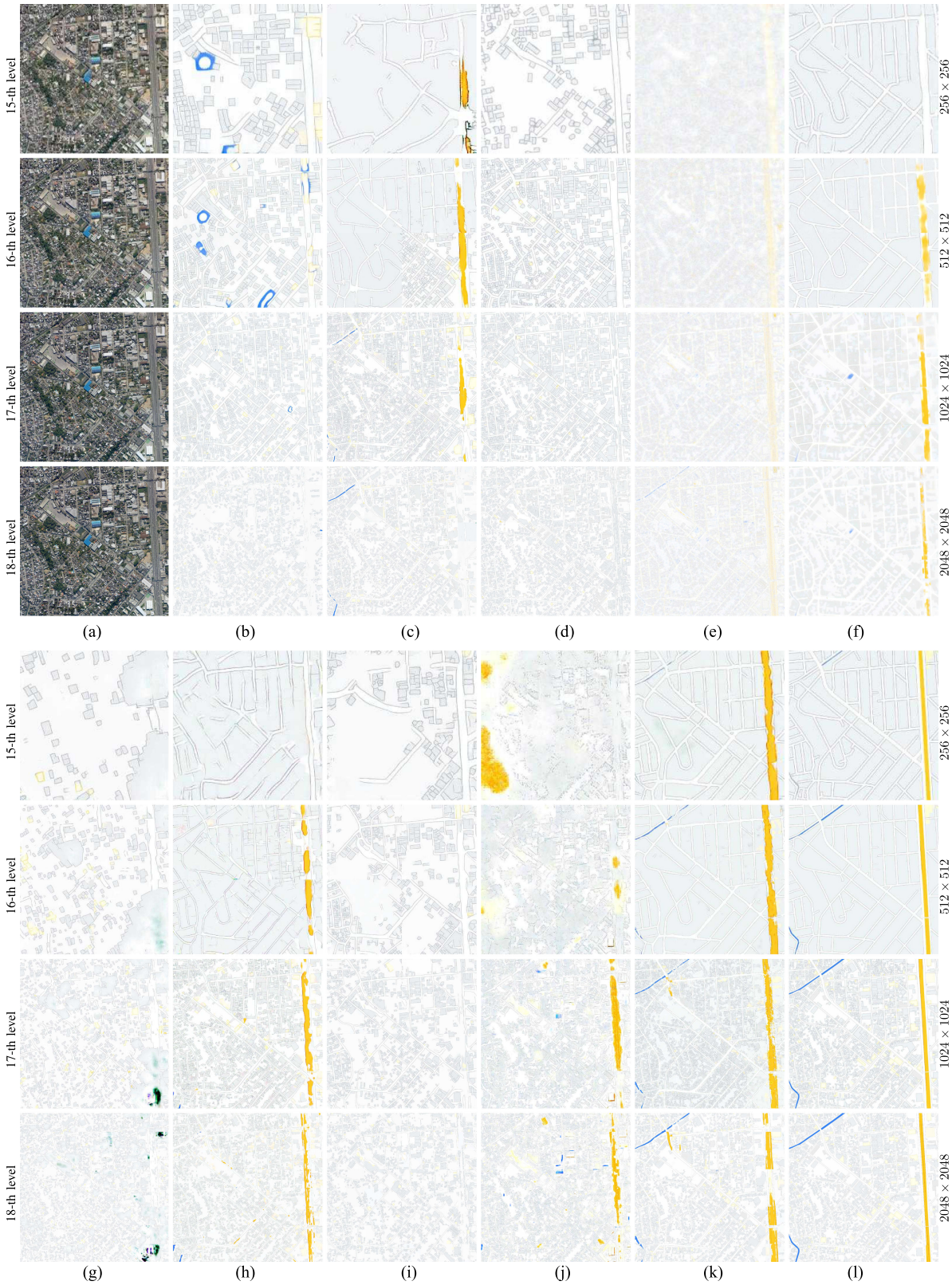


Fig. 8. Visualized results of MLM-RIO generated by our method and counterparts. Our method shows satisfactory accuracy and content consistency, especially for rivers and roads in all zoom levels. (a) Aerial image. (b) Pix2Pix. (c) Pix2PixHD. (d) CycleGAN. (e) GeoGAN. (f) SMAPGAN. (g) SPADE. (h) SelectionGAN. (i) TSIT. (j) LPTN. (k) Ours. (l) GT.

which are the representative cities from the Northern Hemisphere and the Southern Hemisphere. For brevity, we indicate the subset of the samples in Shanghai as MLM-SH and the subset of samples in Rio de Janeiro as MLM-RJ.

2) *Multilevel:* As shown in Table V, the MLM dataset provides paired samples in four different zoom levels, i.e., 15, 16, 17, and 18, that cover the central region of the two cities. The samples at zoom levels lower than 15 are too few due to the area size and high distance per pixel ratio, while those at zoom level higher than 19 are hard to access due to usage limitations. All of the samples form a complete MLM, as shown in Figs. 5 and 6.
3) *Large scale:* The MLM dataset has 18 700 high-quality paired samples, i.e., 10 200 samples of Shanghai and 8500 samples of Rio de Janeiro. It is much larger than the existing map dataset in [25] and dataset in [3], as shown in Table IV.
4) *High-quality:* During data collection, we ensure that the image pairs at different levels show the same corresponding geographic area. Each map image is semantically aligned with the corresponding aerial image.

V. Experiments

The proposed method is systematically evaluated on our MLM-SH and MLM-RJ datasets. We use their train set for training and test set for evaluation. In this section, the evaluation metrics are first introduced. Then, we quantitatively and qualitatively evaluate our method. Finally, the ablation studies are performed to analyze our proposed method.

A. Evaluation Metrics

1) Evaluation Metrics for Accuracy: To evaluate the accuracy and the similarity between the generated maps and real maps for each zoom level, we use Fréchet inception distance (FID) [28], kernel inception distance (KID) [30], kernel maximum mean discrepancy (KMMD) [31], and inception score (IS) [34] as metrics. FID and KID are shown to correlate well with the human judgment of visual quality and are often used to evaluate the quality of samples of GANs [28], [30], KMMD is able to identify generative or noisy images from the real images, and IS is a popular metric for image generation quality evaluation. In addition, we also adopt the classical peak signal-to-noise ratio (PSNR) as our metric. The lower FID, KID, and KMMD and higher IS and PSNR indicate better results. The relevant details are shown in Table VI.

2) Evaluation Metrics for Consistency: To evaluate the content consistency in a simple way, we compare the generated maps with real maps at each zoom level. Considering real maps keep their content consistent at different zoom levels, if the generated maps at all zoom levels are accurate and close to the real maps, they should also keep good content consistency, thus we evaluate the consistency by

$$\text{Metric}^c = \frac{1}{K}\sum_{i=1}^{i=K} \text{Metric}(y_i, y_i^{\text{map}}) \tag{10}$$

TABLE IX
Quantitative Results of Different Methods Training on Pix2pix Dataset and MLM-SH Dataset

Level	Method	Train on	FID↓	KID↓	KMMD↓	IS↑	PSNR↑
15	TSIT [37]	Pix2pix	362.3	0.3058	0.6371	**2.079**	21.53
		MLM-SH	**282.8**	**0.2877**	**0.5879**	1.791	**23.81**
	SelectionGAN [36]	Pix2pix	454.2	0.4004	0.7490	1.585	19.57
		MLM-SH	**433.8**	**0.3495**	**0.6784**	**2.262**	**24.94**
	LAMG	Pix2pix	435.0	0.3877	0.7856	1.445	16.02
		MLM-SH	**409.2**	**0.3062**	**0.5618**	**2.883**	**20.83**
16	TSIT [37]	Pix2pix	323.3	0.2608	**0.5685**	**2.301**	21.87
		MLM-SH	**263.6**	**0.2168**	0.5771	1.874	**24.56**
	SelectionGAN [36]	Pix2pix	414.9	0.3236	0.6914	2.136	20.34
		MLM-SH	**395.8**	**0.2765**	**0.6094**	**2.200**	**26.01**
	LAMG	Pix2pix	416.1	0.3255	0.7443	1.531	16.06
		MLM-SH	**296.0**	**0.1562**	**0.4338**	**2.781**	**24.60**
17	TSIT [37]	Pix2pix	290.1	0.3177	0.5628	**2.346**	21.34
		MLM-SH	**194.6**	**0.1464**	**0.4613**	2.113	**24.21**
	SelectionGAN [36]	Pix2pix	341.0	0.2905	0.6476	2.258	19.93
		MLM-SH	**338.4**	**0.2809**	**0.6395**	**2.367**	**24.79**
	LAMG	Pix2pix	336.4	0.3042	0.7351	1.576	15.99
		MLM-SH	**267.5**	**0.1940**	**0.5526**	**2.792**	**24.14**
18	TSIT [37]	Pix2pix	264.8	0.2454	0.6394	**2.747**	20.73
		MLM-SH	**175.7**	**0.1182**	**0.5180**	2.164	**24.75**
	SelectionGAN [36]	Pix2pix	373.5	0.2422	0.6720	**2.537**	19.08
		MLM-SH	**339.5**	**0.1891**	**0.6027**	2.448	**25.38**
	LAMG	Pix2pix	362.6	0.2443	0.7301	1.625	16.15
		MLM-SH	**225.1**	**0.0999**	**0.4521**	**2.859**	**25.24**
Avg.	TSIT [37]	Pix2pix	310.1	0.2824	0.6019	**2.368**	21.37
		MLM-SH	**229.2**	**0.1923**	**0.5361**	1.986	**24.33**
	SelectionGAN [36]	Pix2pix	395.9	0.3142	0.6900	2.129	19.73
		MLM-SH	**376.9**	**0.2740**	**0.6325**	**2.319**	**25.28**
	LAMG	Pix2pix	387.5	0.3154	0.7488	1.544	16.06
		MLM-SH	**299.4**	**0.1891**	**0.5001**	**2.829**	**23.70**

Notes: "Avg." indicates results that average over 15–18 zoom levels and also shows consistency results.

where y_i and y_i^{map} are the generated map and real map at the ith zoom level, respectively, Metric indicates the metric we use for accuracy evaluation, i.e., FID, KID, KMMD, IS, and PSNR, and Metric^c is the result of consistency. In this way, the result of consistency can also show the overall accuracy.

B. Quantitative and Qualitative Evaluation

We compare our method with several previous state-of-the-art methods for single-level map generation including Pix2Pix [3], Pix2PixHD [4], CycleGAN [5], GeoGAN [6], SMAPGAN [7], SPADE [35], SelectionGAN [36], TSIT [37], and LPTN [38]. As for the MLM generation method, there is still no relevant method for comparison. We note that all methods are trained with the same setting on the training set of the MLM dataset and evaluated on the testing set for fairness. We only need to crop the aerial images into 256×256 patches as inputs, and do not need other preprocessing for the training or testing. The quantitative and qualitative results show that our method outperforms the prior works on the MLM dataset.

As illustrated in Tables VII and VIII, our method outperforms all counterparts by a large margin on average results and most single-level results of FID, KID, KMMD, and IS metrics. Note that IS uses an ImageNet pretrained InceptionV3 [29] to calculate the realism of the generated images but does not compare the generated map images with the ground truth map images [34], so its result is not as reliable as FID, KID, and KMMD. Nevertheless, we provide the IS result due to its popularity and

TABLE X
RESULTS OF LAMG AND CAMG ON MLM-SH DATASET FOR ABLATION STUDY

Level	LAMG			CAMG	FID↓	KID↓	KMMD↓	IS↑	PSNR↑
	Drawer Module	Level ID	Semantic Module						
Avg.	√	×	×	×	265.2	0.1685	0.4578	2.952	20.65
	√	√	×	×	264.8	0.1654	0.4489	3.134	20.53
	√	√	√	×	174.0	0.0970	0.3042	3.244	20.26
	√	√	√	√	**137.7**	**0.0805**	**0.2957**	**3.312**	**22.01**

for a more comprehensive comparison. As for PSNR, we also achieve better results than including Pix2Pix [3], Pix2PixHD [4], CycleGAN [5], SPADE [35], SelectionGAN [36], TSIT [37], and LPTN [38]. Although GeoGAN [6] and SMAPGAN [7] achieve similar or better results than our method in PSNR, their visual quality is much worse than ours, as shown in Figs. 7 and 8. The reason may be that GeoGAN [6] and SMAPGAN [7] tend to minimize the mean square error (MSE), but the ability of MSE (and PSNR) to capture perceptually relevant differences (e.g., high texture detail) is very limited as they are defined based on the pixelwise image differences [46]–[48]. A series of research works have shown that a higher PSNR does not necessarily reflect perceptually better visual results [49].

The Figs. 7 and 8 show that the visual quality of the proposed method outperforms all counterparts in two aspects. First, the semantic module explicitly provides the pixelwise understanding of the input aerial images, so our method is able to generate clear and accurate maps that are similar to the real maps, while the visual results of the counterparts tend to be blurred and inaccurate at each zoom level. Second, our method largely keeps the content consistency across all of the zoom levels with the help of the CAMG, whereas the MLM generated by the counterparts suffer from severe content inconsistency. Nonetheless, our method still has room for improvement to distinguish different types of roads (indicated as white, yellow, and orange in map images). These roads have the same semantic features and look very similar in aerial images, but they need to draw in different colors, which makes it very challenging.

These results show that the proposed method outperforms all counterparts on accuracy and consistency by a large margin, quantitatively and qualitatively in MLM generation task, and demonstrate the effectiveness of the proposed method.

C. Ablation Study

We perform the ablation studies on MLM-SH for evaluating the effectiveness of each part in our method. Considering that the drawer module is the basic module of the generator and the semantic module cannot generate maps independently, we take the performance of the drawer module as a baseline and evaluate the effectiveness of level identification, semantic module, and CAMG.

1) Level Identification: Level identification provides the information of zoom level, which is helpful for the generator to optimize the output map at each zoom level. As shown in Table X, it leads to improvements on FID, KID, KMMD, and IS metrics. The accuracy of the generated maps with level identification is better than the baseline in Fig. 3(c) (e.g., the river in the 15th zoom level is more clear).

2) Semantic Module: The semantic module is designed to provide pixelwise understanding of aerial images and semantic information so as to keep the topological relationship among the geographical elements. It can be seen in Table X that the semantic module largely improves FID, KID, KMMD, and IS results. Fig. 3(d) shows that the semantic module increases the accuracy of the generated maps, especially for lower zoom levels. This shows the effectiveness of the semantic module.

3) Consistency-Aware Map Generator: CAMG cannot only build the connection between maps at different zoom levels, but also repair and refine the initial generated map with the information provided by the maps at higher zoom levels. After we employ the CAMG, the overall accuracy and content consistency of the generated maps are largely improved, as shown in Table X and Fig. 3(e).

4) Dataset for Multilevel Map Generation: In order to demonstrate the effectiveness of our dataset for MLM generation, we perform a series of experiments with our MLM dataset and an aerial-to-map dataset provided by Pix2Pix [3]. The Pix2Pix dataset [3] is a well-known high-quality aerial-to-map dataset, which validates the feasibility of translating the aerial images to single-level maps.

For efficiency, we choose TSIT [37], SelectionGAN [36], and our LAMG for experiments. The reason why LAMG was chosen instead of our entire model is that the Pix2pix dataset does not include multilevel data and cannot be used to train our entire model. For fairness, we train these models on the Pix2pix dataset and the MLM-SH dataset, respectively, and we evaluate them on the MLM-RJ dataset. Note that the maps in MLM-SH and MLM-RJ have a very different style (i.e., domain gap), as shown in Figs. 5 and 6. Therefore, the MLM-SH, MLM-RJ, and Pix2Pix dataset [3] can be regarded as three independent datasets.

As shown in Table IX, the models trained on the MLM-SH dataset achieve the best results on most metrics at each level. These results show that our dataset is effective and has a better potential for MLM generation task, whereas the existing single-level map dataset cannot handle the needs of this task.

VI. CONCLUSION

In this article, we propose a novel method for MLM generation from aerial images. By understanding aerial images pixelwisely and building connections between the maps at different zoom levels, it cannot only generate accurate maps for each zoom

level, but also keep the content consistency between the maps at different zoom levels. The quantitative and qualitative results both show that our method achieves the best performance compared with the previous state-of-the-art map generation methods. In addition, we collect and produce a large-scale high-quality dataset called MLM. The MLM dataset provides 18 700 samples of aerial images and maps from two representative cities in four zoom levels. It is able to serve as a benchmark and support the future research of MLM generation.

In the future, we plan to collect and produce a MLM dataset with a larger scale, more cities, and more zoom levels. Considering that the development of MLM generation is still in its early stage, it is worth investigating the better model structure, loss function, and evaluation metric specially designed for this task. We hope our dataset and preliminary work can serve as a foundation and open new chances for future research.

References

[1] L. Chen, Y. Fu, S. You, and H. Liu, "Efficient hybrid supervision for instance segmentation in aerial images," *Remote Sens.*, vol. 13, no. 2, 2021, Art. no. 252.

[2] Y. Fu, S. Liang, D. Chen, and Z. Chen, "Translation of aerial image into digital map via discriminative segmentation and creative generation," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–15, 2022, Art. no. 4703715.

[3] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit*., 2017, pp. 5967–5976.

[4] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional GANs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit*., 2018, pp. 8798–8807.

[5] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 2242–2251.

[6] S. Ganguli, P. Garzon, and N. Glaser, "GeoGAN: A conditional GAN with reconstruction and style loss to generate standard layer of maps from satellite images," 2019, *arXiv:1902.05611*.

[7] X. Chen *et al.*, "SMAPGAN: Generative adversarial network-based semisupervised styled map tile generation method," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4388–4406, May 2021.

[8] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst*., 2014, pp. 1–9.

[9] S. Hu and T. Dai, "Online map application development using Google Maps API, SQL Database, and ASP. NET," *Int. J. Inf. Commun. Technol.*, vol. 3, no. 3, pp. 102–110.

[10] B. Veenendaal, M. A. Brovelli, and S. Li, "Review of web mapping: Eras, trends and directions," *ISPRS Int. J. Geo-Inf.*, vol. 6, no. 10, 2017, Art. no. 317.

[11] A. Skopeliti and L. Stamou, "Online map services: Contemporary cartography or a new cartographic culture?," *ISPRS Int. J. Geo-Inf.*, vol. 8, no. 5, 2019, Art. no. 215.

[12] W. Shi, S. Shen, and Y. Liu, "Automatic generation of road network map from massive GPS, vehicle trajectories," in *Proc. 12th Int. IEEE Conf. Intell. Transp. Syst*., 2009, pp. 1–6.

[13] P. Haunold and W. Kuhn, "A keystroke level analysis of manual map digitizing," in *Proc. Eur. Conf. Spatial Inf. Theory*., 1993, pp. 406–420.

[14] S. V. Ablameyko, B. S. Beregov, and A. N. Kryuchkov, "Computer-aided cartographical system for map digitizing," in *Proc. Int. Conf. Doc. Anal. Recognit*., 1993, pp. 115–118.

[15] M.-J. Kraak, "The cartographic visualization process: From presentation to exploration," *Cartogr. J.*, vol. 35, no. 1, pp. 11–15, 1998.

[16] B. P. Buttenfield and R. B. McMaster, *Map Generalization: Making Rules for Knowledge Representation*. London, U.K.: Longman Sci. Tech., 1991.

[17] M. P. Peterson, *Interactive and Animated Cartography*. Hoboken, NJ, USA: Prentice Hall, 1995.

[18] M.-J. Kraak and F. Ormeling, *Cartography: Visualization of Geospatial Data*. Boca Raton, FL, USA: CRC Press, 2020.

[19] Y. Zou, Y. Fu, Y. Zheng, and W. Li, "CSR-Net: Camera spectral response network for dimensionality reduction and classification in hyperspectral imagery," *Remote Sens.*, vol. 12, no. 20, 2020, Art. no. 3294.

[20] Y. Fu, Z. Liang, and S. You, "Bidirectional 3D quasi-recurrent neural network for hyperspectral image super-resolution," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 2674–2688, 2021.

[21] Q. Wu, F. Luo, P. Wu, B. Wang, H. Yang, and Y. Wu, "Automatic road extraction from high-resolution remote sensing images using a method based on densely connected spatial feature-enhanced pyramid," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 14, no. 1, pp. 3–17, Apr. 2020.

[22] Z. Chen, C. Wang, J. Li, N. Xie, Y. Han, and J. Du, "Reconstruction bias U-net for road extraction from optical remote sensing images," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 14, no. 1, pp. 2284–2294, Jan. 2021.

[23] D. Pan, M. Zhang, and B. Zhang, "A generic FCN-based approach for the road-network extraction from VHR remote sensing images-using openstreetmap as benchmarks," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 14, no. 1, pp. 2662–2673, Feb. 2021.

[24] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*.

[25] Y. Kang, S. Gao, and R. E. Roth, "Transferring multiscale map styles using generative adversarial networks," *Int. J. Cartogr.*, vol. 5, pp. 115–141, 2019.

[26] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 833–851.

[27] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit*., 2017, pp. 1800–1807.

[28] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local nash equilibrium," in *Proc. Neural Inf. Process. Syst*., 2017, pp. 1–38.

[29] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit*., 2016, pp. 2818–2826.

[30] M. Bińowski, D. J. Sutherland, M. Arbel, and A. Gretton, "Demystifying MMD GANs," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–36.

[31] A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and A. Smola, "A kernel method for the two-sample-problem," in *Proc. Neural Inf. Process. Syst*., 2007, pp. 1–8.

[32] Q. Xu *et al.*, "An empirical study on evaluation metrics of generative adversarial networks," 2018, *arXiv:1806.07755*.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit*., 2016, pp. 770–778.

[34] T. Salimans *et al.*, "Improved techniques for training GANs," in *Proc. Neural Inf. Process. Syst*., 2016, pp. 1–10.

[35] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu, "Semantic image synthesis with spatially-adaptive normalization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit*., 2019, pp. 2332–2341.

[36] H. Tang, D. Xu, N. Sebe, Y. Wang, J. J. Corso, and Y. Yan, "Multi-channel attention selection gan with cascaded semantic guidance for cross-view image translation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit*., 2019, pp. 2412–2421.

[37] L. Jiang, C. Zhang, M. Huang, C. Liu, J. Shi, and C. C. Loy, "TSIT: A simple and versatile framework for image-to-image translation," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 206–222.

[38] J. Liang, H. Zeng, and L. Zhang, "High-resolution photorealistic image translation in real-time: A laplacian pyramid translation network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit*., 2021, pp. 9387–9395.

[39] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.

[40] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang, "DenseASPP for semantic segmentation in street scenes," in *Proc. Int. Conf. Comput. Vis.*, 2018, pp. 3684–3692.

[41] J. Fu *et al.*, "Dual attention network for scene segmentation," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 3141–3149.

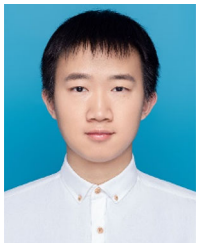
[42] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 2999–3007.

[43] J. Johnson, A. Alahi, and F.-F. Li, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 694–711.

[44] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent*., 2015, pp. 1–14.
[45] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent.*, 2014, pp. 1–15.
[46] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[47] P. Gupta, P. Srivastava, S. Bhardwaj, and V. Bhateja, "A modified PSNR metric based on HVS for quality assessment of color images," in *Proc. Int. Conf. Commun. Ind. Appl.*, 2011, pp. 1–4.
[48] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. Asilomar Conf. Signals, Syst. Comput*., 2003, pp. 1398–1402.
[49] C. Ledig *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit*., 2017, pp. 105–114.

Linwei Chen received the B.S. degree in mechanical engineering and automation from the China University of Geosciences, Beijing, China, in 2019, and the M.S. degree in software engineering from the Beijing Institute of Technology, Beijing, China, in 2021.

His research interests include image segmentation, object detection, and remote sensing.



Zheng Fang received the B.S. degree in computer science and technology from the Ocean University of China, Qingdao, China, in 2020. He is currently pursuing the M.S. degree in computer science and technology at Beijing Institute of Technology, Beijing, China.

His research focuses on the application of image translation and remote sensing.

Ying Fu (Senior Member, IEEE) received the B.S. degree in electronic engineering from Xidian University, Xi'an, China, in 2009, the M.S. degree in automation from Tsinghua University, Beijing, China, in 2012, and the Ph.D. degree in information science and technology from the University of Tokyo, Tokyo, Japan, in 2015.

She is currently a Professor with the School of Computer Science and Technology, Beijing Institute of Technology, Beijing, China. Her research interests include physics-based vision, image and video processing, and remote sensing.