Scientific Reports

https://doi.org/10.1038/s41598-026-39096-0

Article in Press

A method for compiling satellite image map geographic objects based on vector map data via deep learning

Received: 19 May 2025

Accepted: 2 February 2026

Published online: 16 February 2026

Cite this article as: Du J., Zeng D., Cai K. *et al.* A method for compiling satellite image map geographic objects based on vector map data via deep learning. *Sci Rep* (2026). https://doi.org/10.1038/s41598-026-39096-0

Jiawei Du, Dexian Zeng, Kaijun Cai & Yiyang Qiu

A Method for Compiling Satellite Image Map Geographic Objects based on Vector Map Data via Deep Learning

Jiawei Du*, Dexian Zeng, Kaijun Cai, Yiyang Qiu
Space Engineering University, Beijing 100000, China

Jiawei Du (first author & corresponding author): Space Engineering University, whdxdjw@126.com
Dexian Zeng: Space Engineering University, zengdexian@139.com
Kaijun Cai: Space Engineering University, 527516943@qq.com
Yiyang Qiu: Space Engineering University, 1242149161@qq.com

Abstract: Map compilation is a fundamental component of cartography. Apart from vector maps, satellite image maps also require compilation for specific purposes, such as enhancing visual clarity, concealing sensitive information, and ensuring consistency with corresponding vector representations. This paper proposes an automated method for satellite image map compilation based on deep learning and guided by vector map data, which consists of several key steps. First, scale-matched vector maps and satellite image maps are aligned and partitioned to generate training samples in the form of paired vector tiles and satellite image patches. Second, a base satellite image generation model with an encoder-decoder-type deep learning architecture is constructed and trained on these sample pairs to learn the mapping from vector data to realistic satellite imagery. Third, geographic objects designated for compilation are identified, and transfer learning is performed to fine-tune the base model, which improves its sensitivity to regions requiring modification. Subsequently, specific compilation operations—deletion, insertion, distortion, and displacement—are defined; corresponding vector features are then edited using analogous transformations before being input into the trained model to generate updated satellite images reflecting the intended changes. The proposed method enables both selective and operation-diverse compilation of geographic objects in satellite image maps. Experiments conducted using real-world datasets verify the capability of the proposed method to compile linear and polygonal geographic objects through various operations.

Keywords: vector map; remote sensing cartography; generative adversarial network; transfer learning; deepfake cartography

1. Introduction

Satellite image maps produced through satellite remote sensing cartography represent regional geographical environments more realistically than vector maps, thereby helping users achieve intuitive and comprehensive spatial awareness. Satellite image maps are also widely used across various electronic map platforms. During cartographic production, map compilation is one of the most important processes[1]. Traditionally, while map compilation focuses on editing vector map data, the compilation of geographic objects in satellite image maps is also important. Satellite image maps have

long been perceived merely as mosaics of remote sensing images without compiled geographic objects. With the popularization of satellite image maps and the expanding scope of cartography, the compilation of geographic objects in satellite image maps has become necessary in certain circumstances[2-4]. For example, some buildings and roads in satellite image maps may need to be deleted, moved, or created to ensure the security of sensitive geographic information[5-7].

Unlike general image editing, satellite image map compilation necessitates various operations to edit geographic data selectively based on cartographic constraints, such as map context consistency, topology consistency, geographic consistency, and consistency across multiple data types. Moreover, the compilation of geographic objects in satellite image maps is far more complex and tedious than in vector maps. Geographic objects in vector maps are defined independently and represented by corresponding geometric units, whereas those in satellite image maps are blended with satellite image pixels. Moreover, geographic objects in vector maps can be compiled by simply editing their vector coordinates; however, to compile geographic objects in satellite image maps, different geographic objects must be separated and their pixel values adjusted individually, which is particularly tedious given the enormous number of pixels to be processed. Thus, manually compiling satellite image maps and appropriately connecting the geographic objects with their surroundings is exceedingly time-consuming and energy-intensive.

The compilation of geographic objects in satellite image maps can be automated by first detecting the target objects within satellite imagery and then editing them using digital image-processing techniques. Considering the growing accuracy and usability of remote-sensing-based target detection techniques, certain deep-learning-based algorithms[8-10] can be employed to automatically detect geographic objects for compilation in satellite image maps. For example, Mask RCNN has been used to detect and separate geographic objects from satellite image maps; moreover, Deep-Fill models or masked autoencoders have been used to learn the surroundings of geographic objects and replace them with artificially generated pixels to conceal the objects[11-12]. Additionally, deep-learning--based image-editing methods[13-15] that manipulate input representations (e.g., segmentations or masks) to produce modified images can improve such techniques further. However, the effectiveness of such methods depends on the accuracy of the geographic object detection model; additionally, only objects detected by the model can be processed, and all the detected objects compulsorily undergo processing. Thus, the target-detection-based strategy is not sufficiently flexible for compiling unidentified geographic objects or selectively processing detected objects. Additionally, as this strategy can only blend detected objects with their surroundings, it effectively enables only deletion operations and is incapable of other operations essential for compiling satellite image maps, e.g., insertion, displacement, or distortion.

Generative artificial intelligence (GenAI) has provided a breakthrough in image processing[16-17], enabling the batch creation and modification of diverse types of images. Thus, GenAI represents a promising approach to automating the compilation of geographic objects in satellite image maps. This paper proposes a novel strategy for compiling geographic objects in satellite image maps, which involves using transfer-learning-based GenAI to perform data transformation between vector maps and satellite

image maps. The proposed strategy supports various operations, e.g., deletion, insertion, and displacement, for compiling geographic objects in satellite image maps and can also generate deepfake satellite image maps with selectively represented geographic objects. The remainder of this paper is organized as follows: Section 2 reviews the related literature. Section 3 introduces the overall concept of the proposed strategy. Section 4 details the methodology of compiling geographic objects in a satellite image map according to a corresponding vector map using deep learning, including sample extraction from the vector map and satellite image map, the creation of the deep-learning-based satellite image generation model, and the use of transfer learning to compile the geographic objects in the generated satellite image map. Section 5 describes the experimental setup and discusses some use cases. Finally, Section 6 concludes the paper and outlines the scope for further research.

2. Related work

2.1. GenAI in map compilation

Considering the breakthroughs achieved in image translation, style transfer, and image creation[16-17], GenAI has become popular in map compilation research[18], as exemplified by GenAI-based remote sensing mapping[18-22], map generalization[24-28], multi-scale map representation[29], and map style transfer[30-32]. Specifically, in map compilation, converting remote sensing images to general map tiles aligns with the primary aim of remote sensing cartography and has been a key focus of map compilation research. Consequently, most GenAI-based map compilation studies have focused on general maps rather than satellite image maps, and most existing models are trained to convert remote sensing images to general map tiles. Examples of such models include certain generative adversarial networks[19-23]. With the expanding scope of cartography[33], the concept of "pan-maps"[34] has emerged. Apart from capturing the real world accurately, pan-maps are also intended to represent different types of virtual or fictional areas, such as in video game maps[35,36], deepfake geography[37], and even metaverse cartography[38], where remote sensing images need to be edited or created selectively. Most maps, including remote sensing maps, are intended to represent certain parts of the world selectively[2-4]. Therefore, in the pan-map era, deepfake remote sensing cartography—compiling remote sensing maps for selective representation using GenAI—is necessary. However, research on creating remote sensing images using GenAI, especially satellite image map compilation with GenAI, remains limited. For example, Cycle-GAN[39], which enables unpaired image-to-image translation, has been utilized for interconversion between remote sensing images and general map tiles. However, while this approach involves GenAI-based remote sensing image creation, it does not support the editing of geographic objects within remote sensing images—a capability that is essential for satellite image map compilation. Despite the potential and importance of such functionality in enabling GenAI-based satellite image map compilation, few studies have investigated this domain.

2.2. GenAI in remote sensing image creation

Existing GenAI-based methods for remote sensing map creation can be classified into two types: text-based and image-based. Text-based strategies involve creating or enhancing text-based learning models, training these models to fit the mapping from semantic information to remote sensing images, and then using the trained models to generate remote sensing images from semantic information. For example, the Hopfield neural network has been improved to realize text-based remote sensing image generation[40]. Under image-based strategies, style transfer models are first created or enhanced and are then trained to fit the mapping from other types of images to remote sensing images; subsequently, the trained models can transfer the styles of different images to generate remote sensing images. For example, CycleGAN has been used to learn the mapping from raster map tiles to remote sensing images[41], while CSEBGAN was designed to generate remote sensing images from semantic classification images[42]. Even though text-based and image-based strategies can be used to distort, move, or blur geographic objects in the generated images, these methods cannot compile geographic objects selectively or allow different operations (e.g., insertion, displacement, and distortion) to be performed on selected geographic objects. In fact, existing GenAI-based methods are designed to generate remote sensing images that closely replicate real images rather than to selectively compile geographic objects in the generated images. The research conducted thus far has verified the feasibility of fitting the mapping from general map data to remote sensing data. Accordingly, GenAI-based methods can adaptively generate large amounts of remote sensing images, thereby facilitating the convenient generation of satellite image maps.

3. Overall strategy

A deep convolutional neural network (CNN) is used to learn and fit the mapping from vector maps to satellite image maps, which serves as the basis for designing the method for compiling geographic objects in the satellite image map. Fig. 1 illustrates the overall strategy. First, based on the map scale and resolution, the satellite image map and vector data map are matched and divided into grid cells to extract samples (i.e., matched satellite images and vector tiles) suitable for the deep neural network. Second, a satellite image generation model is designed based on the deep CNN and is trained to fit the mapping from vector tiles to satellite images. Third, the geographic objects to be compiled are selected from the matched map data by indexing and confirmed, whereafter the samples containing these objects are segmented and extracted based on the same grid size as in the first step; these samples are regarded as region-specific data. Subsequently, through transfer learning, the satellite image generation model is fine-tuned on the region-specific data to enhance its ability to generate satellite imagery for the regions to be complied. Thus, the trained model can generate compiled satellite image maps from edited vector maps. Fig. 1 shows an example of the deletion of a building (delineated by the red boxes) in satellite images resulting from the deletion of the corresponding geometric feature (enclosed by the yellow box) in the vector data. Specifically, performing various operations (i.e., deletion, insertion, and displacement) to edit the geographic features in the vector map and feeding the edited vector tiles into the trained model yields satellite image maps that replicate the corresponding modifications to the geographic objects.

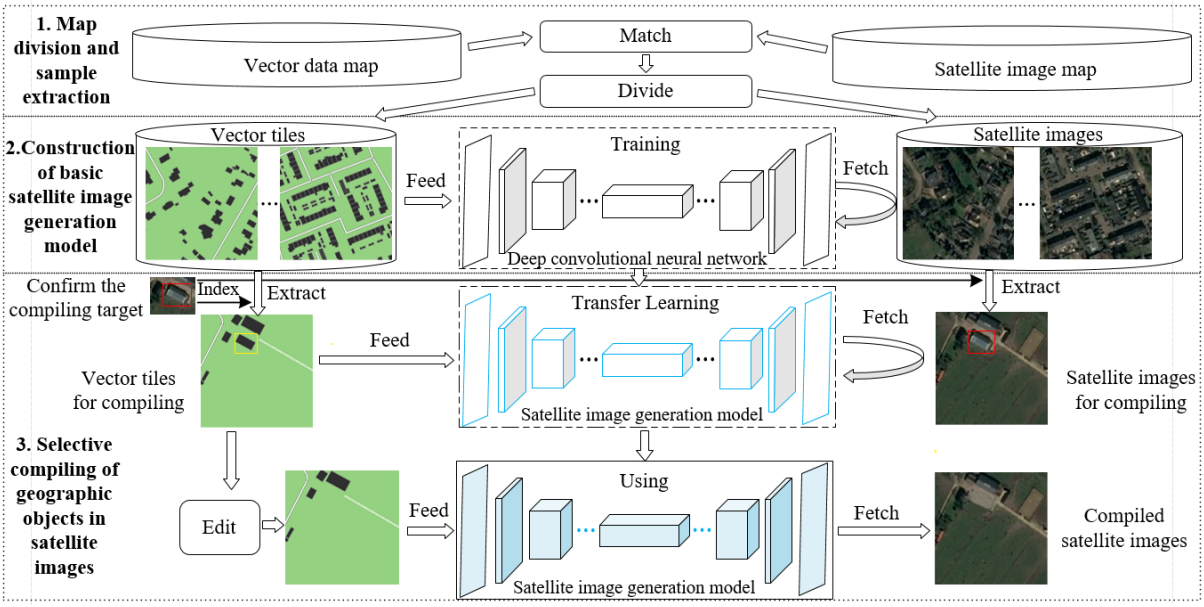


Fig. 1 Overall strategy to compile geographic objects in satellite image maps based on vector maps via deep learning (the original vector tiles and satellite images were crawled from OpenStreetMap (https://www.openstreetmap.org/) and Bing Maps (https://cn.bing.com/maps), respectively, based on ArcGIS 10.2 (http://www.esri.com), the compiled satellite images were modified by the authors and are copyright-free).

4. Methodology

4.1. Map division and sample extraction

Large-scale vector map data are often acquired from high-resolution satellite remote sensing images; such vector maps share a correspondence with the source satellite image maps. If the scales of such vector maps and satellite maps are consistent, then the geographical objects in the vector map can be considered equivalent to those in the satellite map. The relationship between the vector map's scale and the satellite image map's resolution can be expressed as follows:

$$\lambda \cdot (svo/scale) \le pix < svo/scale,\ 0<\lambda<1, \tag{1}$$

where *pix* refers to the actual distance represented by one pixel in the satellite image, which quantifies the satellite image map's resolution[43]; *scale* refers to the vector map's scale; *svo* refers to the smallest visible distance[44] on the vector map; and the ratio *svo*/*scale* denotes the smallest object size that can be represented in the real world based on the vector map's scale. Additionally, λ signifies the adjustment parameter for controlling the range of scale similarity. To match similar-scale vector maps and satellite image maps, data extracted from such maps should be unified in the same projection system using suitable projection conversion methods[45]. Then, the matched maps to be compiled must be divided into tiles for sample extraction, as outlined in the following steps, where *VecMap* and *ImgMap* refer to the vector map data and satellite image map data, respectively.

Step 1. Construction of division grid: Fixed grid cells are constructed to segment *VecMap* and *ImgMap*. The edge length of each grid cell is denoted by l. Let $l = pix \cdot n_{pix}$,

where $n_{pix} \in Z^+$. Additionally, n_{pix} represents the number of extracted image-edge pixels. The constructed grid cells are denoted by $\{G_i | I \in [1, n_{grid}]\}$, where $n_{grid} \in Z^+$.

Step 2. Sample extraction: $G_{i'}$ is assumed to represent any $G_i \subseteq (VecMap \cap ImgMap)$, and *VecMap* and *ImgMap* are divided according to the boundary of $G_{i'}$. The portions of *VecMap* and *ImgMap* contained within $G_{i'}$ are rasterized and normalized to $vm_{i'}$ and $im_{i'}$, respectively, where *pix* represents the rasterization resolution. The matrix size of $vm_{i'}$ or $im_{i'}$ is denoted by $[n_{pix}, n_{pix}, n_{ch}]$, where n_{ch} represents the image channels. By repeating this process while traversing $\{G_i | i \in [1, n_{grid}]\}$, all samples are extracted, i.e., $\{\{vm_{i'}, im_{i'}\} | i' \in [1, n_{grid}]\}$, where a pair of $vm_{i'}$ and $im_{i'}$ constitutes one sample.

Step 3. Construction of sample sets: $\{\{vm_{i'}, im_{i'}\} | i' \in [1, n_{grid}]\}$ is divided into two non-overlapping sets in a certain proportion, i.e., the training set and the testing set, denoted as $VI_1 = \{\{vm_{i1'}, im_{i1'}\} | i_1' \in [1, n_1]\}$ and $VI_2 = \{\{vm_{i2'}, im_{i2'}\} | i_1' \in [1, n_2]\}$, respectively. Additionally, if necessary, more samples are extracted by moving the grid cells to augment the data in the training and testing sets[27].

4.2. Construction of basic satellite image generation model

The encoding and decoding[27] process is somewhat similar to map cognition and satellite cartography. Thus, deep CNNs with an encoder-decoder structure can be used to construct a generative model to fit the mapping from vector tiles to satellite images. The encoder can extract various features from vector tiles through multiple convolution and pooling layers, while the decoder can decode these features to generate satellite images through multiple convolution and unpooling layers. A classic deep CNN with an encoder–decoder structure is used to construct a satellite image generation model. As an example, Fig. 2 shows the convolution, pooling, and unpooling functions in this network, denoted by **C**, **P**, and **U**, respectively. The encoder and decoder are denoted by **E** and **D**, respectively, while the loss function is denoted by **L**. Additionally, residual blocks, jump connections, adversarial structures, attention mechanisms, and other deep learning elements[46] can be integrated into this classic model to construct various generative models with better learning and prediction capabilities. The proposed model is not limited to a specific encoder–decoder network.

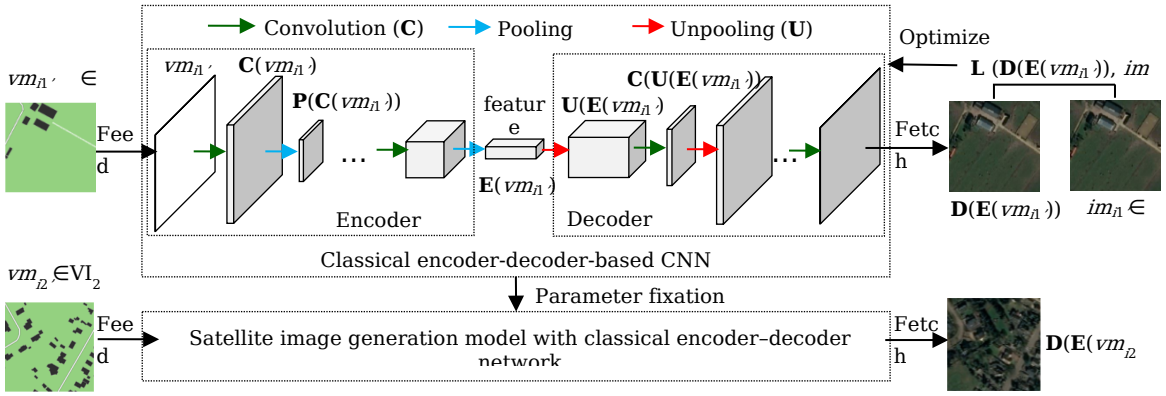


Fig. 2 Satellite image generation model based on classical encoder–decoder-type CNN (the original vector tiles and satellite images were crawled from OpenStreetMap (https://www.openstreetmap.org/) and Bing Maps (https://cn.bing.com/maps), respectively,

based on ArcGIS 10.2 (http://www.esri.com); the compiled satellite images were modified by the authors and are copyright-free).

The general process of constructing a satellite image generation model is as follows:

Step 1. An encoder-decoder-based CNN is constructed, denoted by $DE_{(W, B)}$, where W and B represent the neural network parameters, which are initialized randomly.

Step 2. The CNN is trained with VI_1. Batches of samples from VI_1 are fed to the network, and the network parameters are adjusted adaptively based on Equation (2). Specifically, let $\{\{vm_{i1'}, im_{i1'}\} | i_1' \in [j, j + \eta]\} \in VI_1$ represent a batch of samples, where $j \in [1, n_1]$, and η represents the number of sample batches. Furthermore, assuming $j = 1$, $L(DE_{(W, B)}(\{vm_{i1'}\}), \{im_{i1'}\})$ is calculated, which constitutes the basis for adjusting the neural network parameters using the optimizer. The adjusted parameters are denoted by W' and B'. Let $W = W'$ and $B = B'$, and let $j = j + \eta + 1$. This step is repeated until $j \geq n_1$ (i.e., all samples have been learned). Then, assuming $j = 1$, the network is trained over several epochs to achieve perfect fitting. Thereby, the network parameters become fixed.

$$\mathrm{Min}(L(DE_{(W, B)}(vm_{i'}), im_{i'} | \{vm_{i'}, im_{i'}\} \in VI_1)). \quad (2)$$

Step 3. The trained $DE_{(W, B)}$ is tested on VI_2. After feeding $vm_{i2'}$ into the trained network, $DE_{(W, B)}(vm_{i2'})$ is fetched. Then, the similarity between $DE_{(W, B)}(vm_{i2'})$ and $im_{i2'}$ is evaluated in terms of precision, recall, accuracy, F1 score, and other metrics[46]. Based on the results, the network with the highest testing precision can be used to generate satellite images. Specifically, for any $vm_{i'}$, $DE_{(W, B)}(vm_{i'})$ is calculated, and $DE_{(W, B)}(vm_{i'})$ is denormalized to generate satellite images.

4.3. Selective compiling of geographic objects in satellite images

For clarity, the compiled geographic object in the satellite image map is denoted as ob', and the compiled geographic feature corresponding to ob' in the vector map is denoted as ob. Additionally, ob comprises a series of coordinates, i.e., $ob = \{(x_k, y_k) | k \in [1, n_{ob}]\}$, where x_k and y_k represent the x-coordinate and y-coordinate, respectively, in the projection, and n_{ob} denotes the coordinate count.

4.3.1. Transfer learning using satellite image generation model for compiling region

Through transfer learning, the satellite image generation model is additionally trained on a small amount of vector map data and satellite image map data containing the compiling region to enhance the comprehension of that region. Specifically, the transfer learning process is divided into sample extraction and model training, which are detailed as follows:

- *Map division and sample extraction for compiling region*

The vector map and satellite image map, which contain ob and ob', respectively, are divided based on the following steps to extract raster samples. If all compiled geographic objects are contained within the extracted samples, the following steps are terminated; otherwise, the steps must be repeated.

Step 1. Construct a division grid set, denoted as Gri, and a square grid, denoted as $G_{i''}$, centered on the geometric center of ob, whose side length equals l, as mentioned in Section 4.1. Subsequently, add $G_{i''}$ to Gri.

Step 2. If *ob* is fully contained within *Gri*, proceed to the next step; otherwise, construct a new square grid, $G_{i''+1}$, adjacent to $G_{i''}$ in eight directions. If $G_{i''+1} \cap ob \neq \emptyset$, add $G_{i''}$ to *Gri* and repeat this step.

Step 3. For any $G_{i''} \in Gri$, divide *VecMap* and *ImgMap* according to the boundary of $G_{i''}$; rasterize the regions contained within $G_{i''}$, where the resolution ratio equals *pix*, as mentioned in Section 4.1; and normalize the rasterized tiles of *VecMap* and *ImgMap* to $vm_{i''}$ and $im_{i''}$, respectively. Repeat this step while traversing *Gri* to extract all compiling region samples, which are denoted as $\{\{vm_{i''}, im_{i''}\} | i'' \in [1, n'']\}$, where n'' represents the number of compiling region samples.

- *Training of satellite image generation model based on compiling region samples*

The instance-based transfer learning strategy[47] is adopted to adjust the parameters of the trained satellite image generation model. Specifically, the compiling region samples, i.e., $\{\{vm_{i''}, im_{i''}\} | i'' \in [1, n'']\}$, are used to train the satellite image generation model (which has already been trained as explained in Section 3.2) additionally to enhance its understanding of the compiling region; this additional training process is similar to that described in Section 4.2.

4.3.2. Selective compiling operations for geographic objects in satellite images

The compiling of geographic objects in satellite images using the satellite image generation model can be considered as the editing of the corresponding geographic features in the vector map. The proposed compiling method is detailed as follows:

First, the operations to edit the geographic features in the vector map corresponding to the geographic objects to be compiled are selected, and the rasterized tiles centered on the edited geographic features are extracted from the vector map.

Second, the edited vector map tiles are fed into the satellite image generation model trained via transfer learning, and the compiled satellite image tiles are generated.

Finally, the original satellite image tiles are replaced with the generated satellite image tiles to complete the compilation of geographic objects in the satellite image map.

The proposed method enables rapid batch retrieval of geographic features in the vector map based on their attributes, facilitating the swift compilation of geographic objects in the satellite image map. Furthermore, corresponding to the editing operations for the geographic features in the vector map, various compiling operations—such as deletion, insertion, distortion, and displacement—for the geographic objects in the satellite image map can be automated through the proposed strategy, as explained below.

- *Deletion*

After deleting *ob* from *VecMap*, *VecMap* is divided according to *Gri*, and the divided regions are extracted before being rasterized and normalized to $\{vm_{i''\_1}\}$. Then, $\{vm_{i''\_1}\}$ is traversed, and every $vm_{i''\_1}$ is fed into the satellite image generation model to generate satellite image tiles, which are denoted as $\{\mathrm{DE}_{(W, B)}(vm_{i''\_1})\}$. The regions of the satellite image map contained within *Gri* are replaced with $\{\mathrm{DE}_{(W, B)}(vm_{i''\_1})\}$, and *ob'* is then deleted in the compiled satellite image map. Fig. 3 presents an example of the deletion of a geographic object from the satellite image map.

- *Insertion*

After inserting a new *ob* into *VecMap*, *VecMap* is divided according to *Gri*, and the divided regions are extracted before being rasterized and normalized to $\{vm_{i''\_2}\}$. Then $\{vm_{i''\_2}\}$ is traversed, and every $vm_{i''\_2}$ is fed into the satellite image generation model to generate satellite image tiles, denoted as $\{DE_{(W,B)}(vm_{i''\_2})\}$. The regions of the satellite image map contained within *Gri* are then replaced with $\{DE_{(W,B)}(vm_{i''\_2})\}$, and a new *ob'* is accordingly inserted into the compiled satellite image map.

- *Distortion*

After reshaping *ob* in *VecMap* by adjusting its vertices, *VecMap* is divided according to *Gri*, and the divided regions are extracted before being rasterized and normalized to $\{vm_{i''\_3}\}$. Subsequently, $\{vm_{i''\_3}\}$ is traversed, and every $vm_{i''\_3}$ is fed into the satellite image generation model to generate satellite image tiles, denoted as $\{DE_{(W,B)}(vm_{i''\_3})\}$. The regions of the satellite image map contained within *Gri* are then replaced with $\{DE_{(W,B)}(vm_{i''\_3})\}$, whereby *ob'* in the compiled satellite image map becomes distorted to match the reshaped *ob* in *VecMap*. The foregoing distortion process can also be considered a combination of the deletion and insertion processes described earlier.

- *Displacement*

The displacement of geographic objects represents a combination of deleting geographic objects and reinserting the deleted objects at new locations. By combining the four main operations, i.e., deletion, insertion, distortion, and displacement, several other compiling operations for geographic objects in satellite images can be derived.

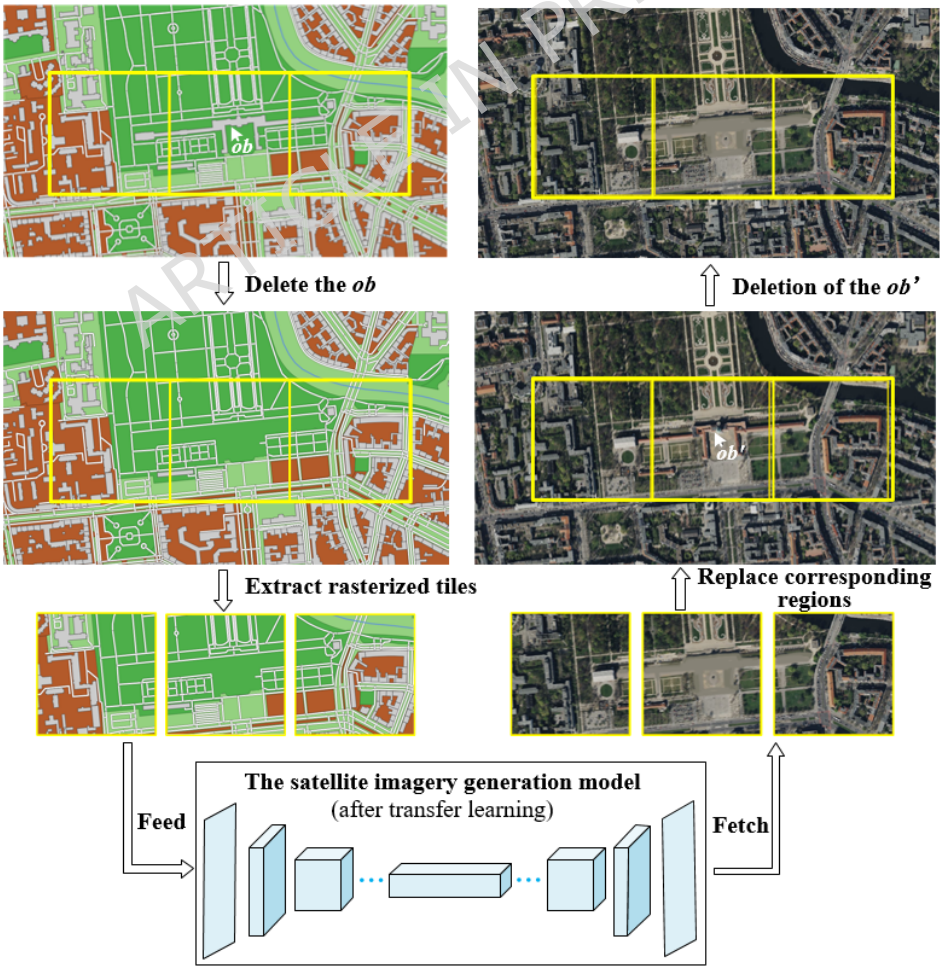
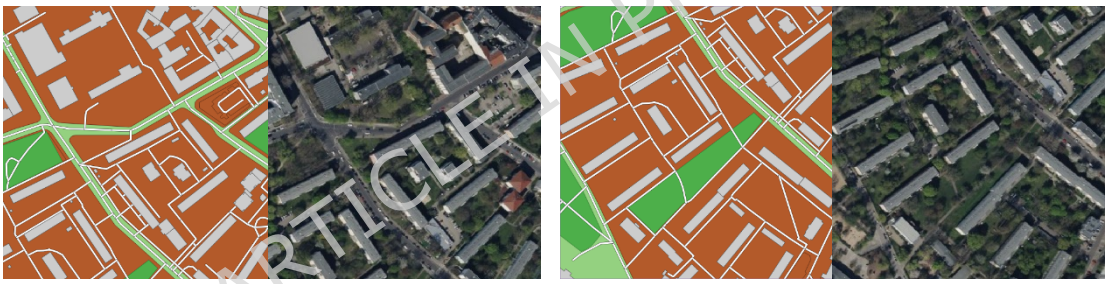


Fig. 3 Deletion of geographic object from satellite image map based on proposed strategy (the original vector tiles and satellite images were crawled from OpenStreetMap (https://www.openstreetmap.org/) and Bing Maps (https://cn.bing.com/maps), respectively, based on ArcGIS 10.2 (http://www.esri.com); the compiled satellite images were modified by the authors and are copyright-free).

5. Experiments and analyses

5.1. Experimental setup

The vector map and satellite image map of a region in Berlin, Germany, were extracted from OpenStreetMap (https://www.openstreetmap.org/) and Bing Maps (https://cn.bing.com/maps); features such as buildings, roads, and rivers exhibit certain similarities between these two maps. The resolution of the extracted satellite image map was about 0.5 m, the scale of the extracted vector map was inferred to be 1:3,000 according to a prior study[48], and *svo* was set as 0.2 mm according to another study[44]; thus, Equation (1) was satisfied, which means that the scales of the extracted vector map and satellite image map were similar and that the map data were suitable for the compiling experiment. The vector data of the experimental region were downloaded first, and the matched satellite image map was cut and crawled based on the vector data region. Additionally, the regions to be compiled were selected randomly from the aforementioned data, as shown in Fig. 4.



(a) First region to be compiled (b) Second region to be compiled

Fig. 4 Vector data and satellite images of regions to be compiled (the original vector tiles and satellite images were crawled from OpenStreetMap (https://www.openstreetmap.org/) and Bing Maps (https://cn.bing.com/maps), respectively).

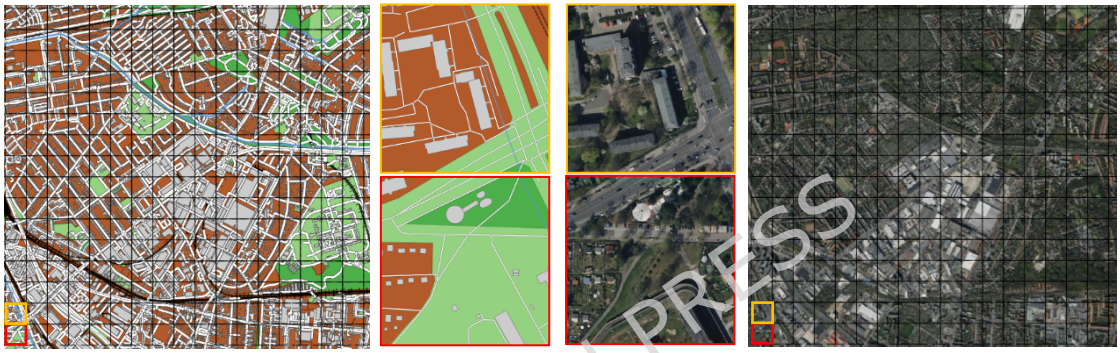
In our experiment, the processing of vector map data was automated using a secondary development based on ArcEngine 10.2 (http://www.esri.com), and deep learning was performed using Python and TensorFlow 1.12.0 (https://tensorflow.google.cn/), which were run on an Nvidia RTX 2070S 8 GB GPU. The map division and sample extraction, the construction of the satellite image generation model, and the selective compiling of geographic objects in the satellite images mentioned in Section 4 were automated step by step. Furthermore, to achieve better compilation, two popular encoder-decoder-structured models were compared, with the better model selected as the kernel learning model for the proposed method to verify the compilation effectiveness of geographic objects in satellite image maps. Several compilation operations were tested with the proposed method for both polygonal and linear objects. Moreover, the proposed method was compared with a method based on the same

encoder–decoder-structured model but without transfer learning to verify the proposed method's superiority and to support ablation analyses of the transfer learning process for the proposed compilation method.

5.2. Results and discussion

5.2.1. Extracted samples and sample sets

For map division, n_{pix} was set to 512, and 9,940 samples, i.e., 9,940 pairs of vector tiles and satellite images, were extracted from the experimental data, according to the process outlined in Section 4.1. Considering the complexity of the learning task (i.e., conversion from vector tiles to satellite images and from low-density data to high-density data), these samples were then divided into training and test sets in a 9:1 ratio (i.e., 8,946 training samples and 994 test samples)[25]. Fig. 5 presents examples of the divided region and extracted sample.

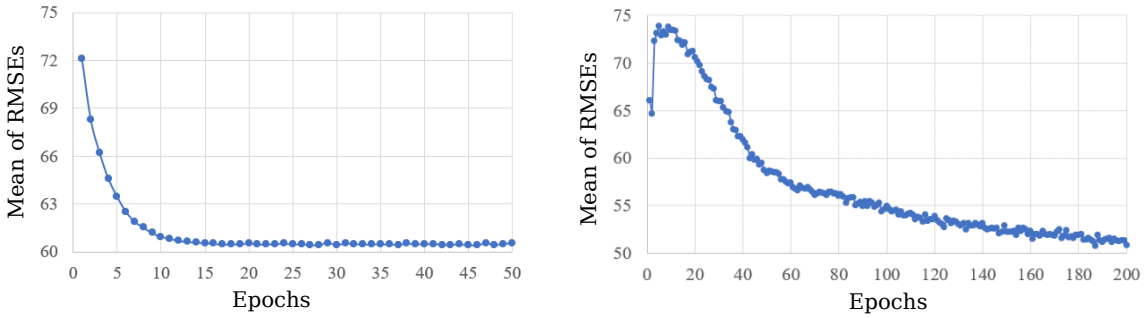


(a) Vector map division and tile extraction (b) Image map division and tile extraction

Fig. 5 Examples of map division and samples extracted from experimental area (the vector map and satellite imagery were downloaded from OpenStreetMap (https://www.openstreetmap.org/) and Bing Maps (https://cn.bing.com/maps), respectively; the satellite images and vector tiles were segmented by the authors using ArcGIS 10.2 (http://www.esri.com), and all modified maps are copyright-free).

5.2.2. Comparison and construction of basic satellite image generation model

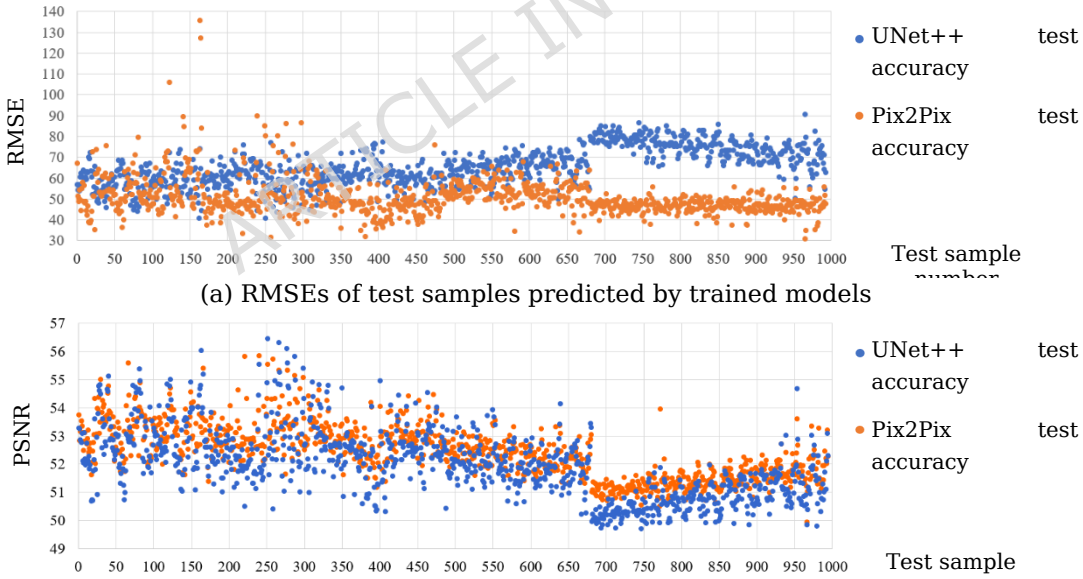
According to some studies[25, 27], UNet++[49] and Pix2Pix[50] perform better than other encoder–decoder-based neural networks in certain cartography applications. Hence, we constructed our basic satellite image generation models based on these two architectures. To compare the performance of these architectures, the same loss function (i.e., the cross-entropy function) and optimizer (i.e., Adam optimizer[51] with a learning rate of 0.001) were adopted for both models. For the training process, η was set to 8, and the two models were trained on all training samples over several epochs. The mean of the root-mean-square errors (RMSEs)[46] for the training samples was calculated and recorded after every epoch (Fig. 6). With UNet++, the mean of the RMSEs for the training samples became stable after 50 epochs, while Pix2Pix required 200 epochs (over 100 h) to achieve the same stability.



(a) Training accuracy variation for UNet++ (b) Training accuracy variation for Pix2Pix

Fig. 6 Fluctuations in means of RMSEs for training samples during training.

The two trained models were then tested on the test samples. Fig. 7 and Table 1 display the calculated RMSEs and peak signal-to-noise ratios (PSNRs, widely used to evaluate image reconstruction quality; the larger the better[52]) based on the test results. Statistically, more than 87% of the test samples produced RMSE values that were lower for Pix2Pix than for UNet++, while more than 80% of the samples produced PSNR values that were higher for Pix2Pix than for UNet++. Table 2 lists some additional test results. Based on these results, the images generated by the trained Pix2Pix model were evidently more similar to the real satellite images than those generated by the UNet++ model, not only in terms of the quantitative accuracy metrics but also from a qualitative perspective. Therefore, during the subsequent experiments, the Pix2Pix architecture was employed in the basic satellite image generation model.



(a) RMSEs of test samples predicted by trained models

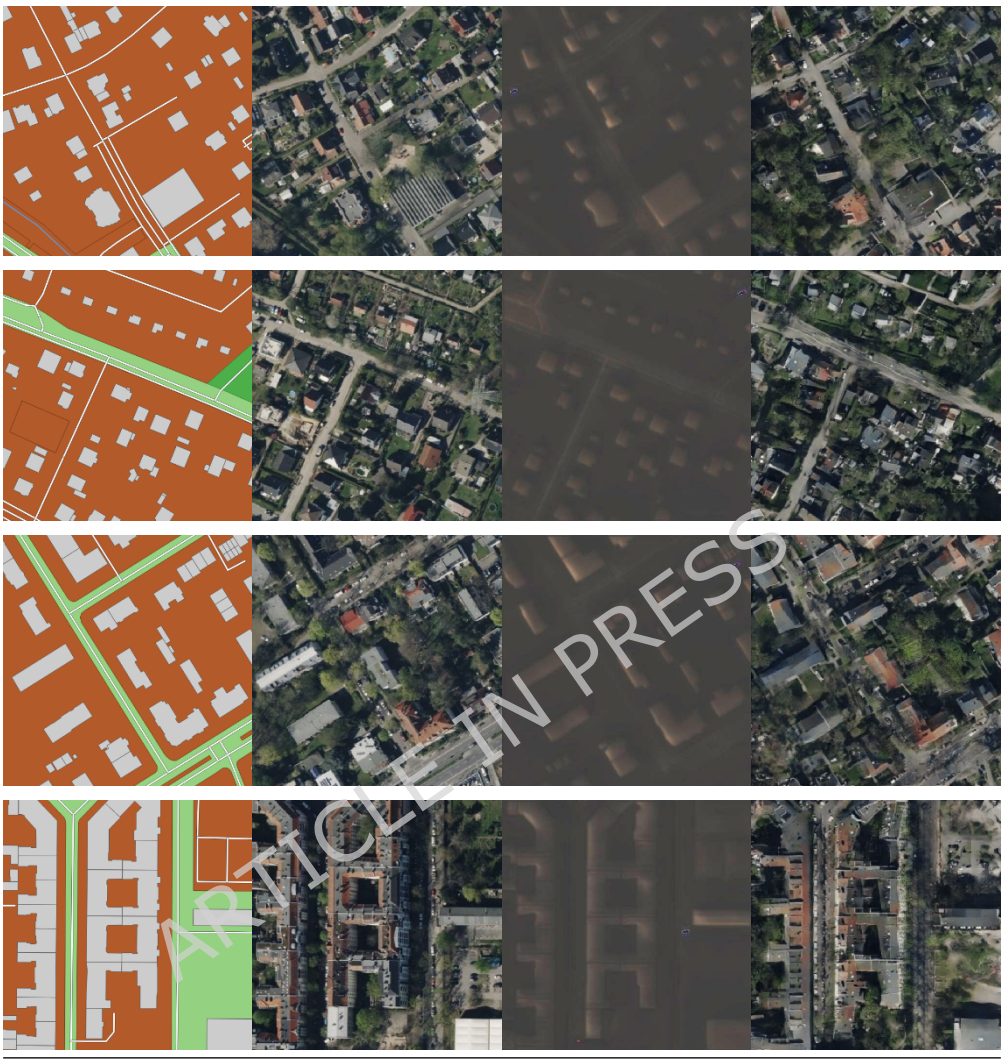
(b) PSNRs of test samples predicted by trained models

Fig. 7 RMSEs and PSNRs of test samples predicted by different trained models.

Table 1 Statistics of evaluation metrics for tested samples with different models.

Evaluation metric values	UNet++			Pix2Pix		
	Maximum	Minimum	Mean	Maximum	Minimum	Mean
RMSE	90.12	40.35	65.32	135.51	30.36	50.31
PSNR	49.72	56.46	51.97	49.95	55.85	52.44

Table 2 Examples of maps generated by trained models (the original vector tiles and satellite images were crawled from OpenStreetMap (https://www.openstreetmap.org/) and Bing Maps (https://cn.bing.com/maps), respectively; the compiled satellite images were modified by the authors using the proposed method and are copyright-free).

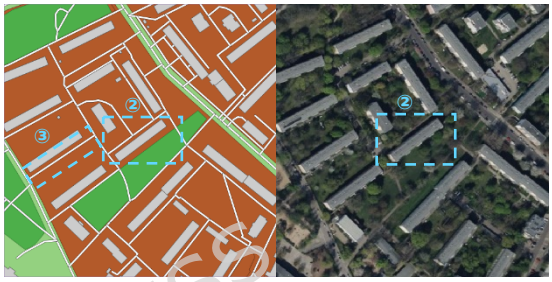
Vector tile	Satellite image	UNet++ image	Pix2Pix image
			

5.2.3. Compilation results

To enhance the learning and understanding of the regions to be compiled, the basic satellite image generation model was trained via transfer learning. Specifically, vector tiles and satellite images of the target regions (Fig. 4) were used to train the Pix2Pix-based satellite image generation model over 200 epochs, which required less than 3 min. Thereafter, the satellite image generation model was modified to fit the mapping from remote sensing images to general map tiles for the target regions; this allowed for selectively compiling geographic objects in the generated satellite images.

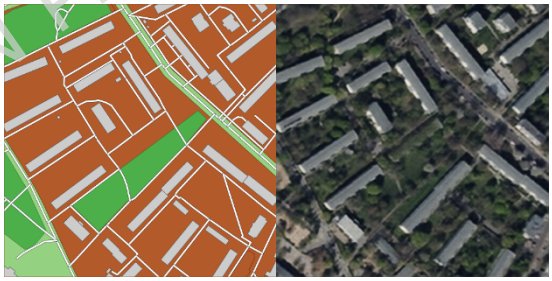
Based on the proposed method, the final trained model was used to automatically and selectively compile geographic objects in the generated satellite image map of the target regions, and the results were found to be consistent with the edited geographic features in the corresponding vector map. In these experiments, individual geographic objects as well as batches of geographic objects, both polygonal and linear, were compiled; moreover,

different compile operations, such as deletion, insertion, distortion, and displacement, were evaluated. The results of deletion, insertion, distortion, and displacement are shown in Figs. 8, 9, 10, and 11, respectively. Additionally, the geometric accuracy of compiled geographic objects was evaluated by measuring the differences between the positions of the compiled geographic features in the vector data and their corresponding positions in the generated satellite imagery. For polygonal geographic objects and features, the distance between the centroids was measured as the geometric error, while for linear geographic objects and features, the average distance between the intersections was measured. The smaller the geometric error, the higher the geometric accuracy and the better the compiled geographic objects. Except for the deletion operation, the geometric errors of the geographic objects compiled through all operations could all be calculated. Table 3 lists the calculation results.



(a) Marked geographic objects in 1st compiling region (b) Marked geographic objects in 2nd compiling region

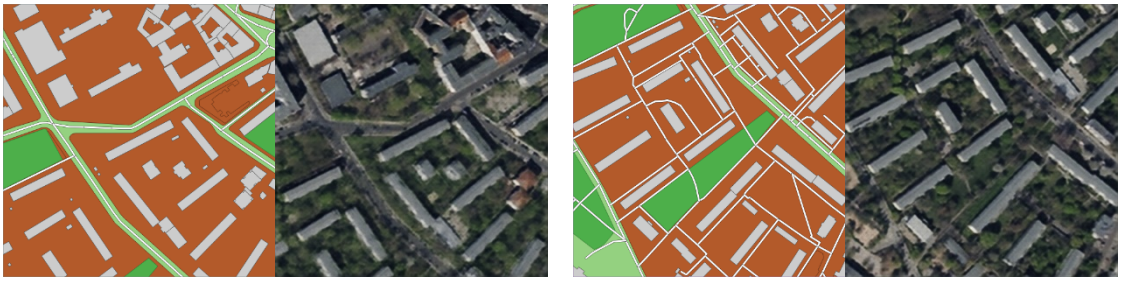
(c) Deletion of polygonal building ①



(d) Deletion of polygonal building ②

(e) Batch deletions of polygonal buildings around ①

(f) Batch deletions of polygonal buildings around ②



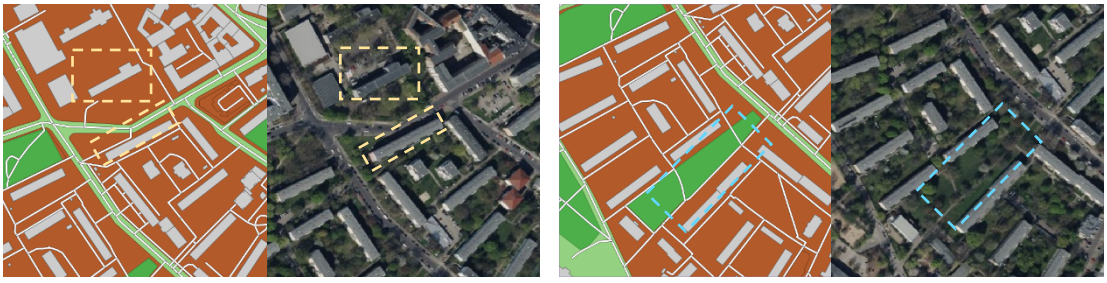
(g) Batch deletions of linear roads in the neighborhood (h) Deletion of linear road ③

Fig. 8 Deletion of geographic objects in satellite maps generated using proposed method (the original vector tiles and satellite images were crawled from OpenStreetMap (https://www.openstreetmap.org/) and Bing Maps (https://cn.bing.com/maps), respectively; the compiled satellite images were modified by the authors using the proposed method and are copyright-free).

(a) Marked geographic objects in 1st compiling region (b) Marked geographic objects in 2nd compiling region

(c) Insertion of polygonal building into marked region (d) Insertion of linear road into marked region

Fig. 9 Insertion of geographic objects into satellite maps generated using proposed method (the original vector tiles and satellite images were crawled from OpenStreetMap (https://www.openstreetmap.org/) and Bing Maps (https://cn.bing.com/maps), respectively; the compiled satellite images were modified by the authors using the proposed method and are copyright-free).



(a) Marked geographic objects in 1st compiling region (b) Marked geographic objects in 2nd compiling region

(c) Distortion of polygonal buildings in marked regions (d) Distortion of linear road in marked region

Fig. 10 Distortion of geographic objects in satellite maps generated using proposed method (the original vector tiles and satellite images were crawled from OpenStreetMap (https://www.openstreetmap.org/) and Bing Maps (https://cn.bing.com/maps), respectively; the compiled satellite images were modified by the authors using the proposed method and are copyright-free).

(a) Marked geographic objects in 1st compiling region (b) Marked geographic objects in 2nd compiling region

(c) Displacement of marked polygonal building (d) Displacement of marked linear road

Fig. 11 Displacement of geographic objects in satellite maps generated using proposed method (the original vector tiles and satellite images were crawled from OpenStreetMap (https://www.openstreetmap.org/) and Bing Maps (https://cn.bing.com/maps), respectively; the compiled satellite images were modified by the authors using the proposed method and are copyright-free).

Table 3 Geometric errors of compiled geographic objects.

Compilation operation	Average geometric errors of compiled buildings (m)	Average geometric errors of compiled roads (m)
Insertion	1.37	2.61

Distortion	1.49	2.48
Displacement	1.18	2.78

These compilation results confirm some of our hypotheses, as explained below:

The proposed method allows for compiling one or more specific geographic objects in the generated satellite image map. Fig. 8 shows examples of geographic objects deleted in the compiled satellite maps. Specifically, buildings ① and ②, marked in Fig. 8(a) and 8(b), respectively, were deleted in the vector data, as shown in Fig. 8(b) and 8(c), respectively; the corresponding buildings in the satellite images generated from the edited vector data were also hidden, as shown in Fig. 8(b) and 8(c). Additionally, Fig. 8(e) and 8(f) show the deletion of the vector buildings around the ① and ② regions marked in Fig. 8(a) and 8(b), respectively; the corresponding buildings in the generated satellite images were also hidden. Moreover, Fig. 8(g) and 8(h) show that using the proposed method, roads in the satellite image corresponding to the deleted vector data were deleted individually or in batches. This verifies that the proposed method can be used to delete one or more specific polygonal or linear objects in compiled satellite images. Additionally, all deleted geographic objects were appropriately hidden with respect to their surroundings in the compiled satellite image map, as shown in Fig. 8(c)–(h). This confirms that the proposed method can effectively automate object-level compiling in satellite image maps.

The proposed method can also automate various compiling operations (e.g., deletion, insertion, distortion, and displacement) on geographic objects in satellite image maps. Specifically, the marked regions in Fig. 9(a) and 9(b) were inserted into a vector building and a vector road, respectively; the corresponding building and road were appropriately observable in the generated satellite images, as seen in Fig. 9(c) and 9(d), respectively. Furthermore, the vector buildings and road marked in Fig. 10(a) and 10(b), respectively, were distorted, and the corresponding buildings and road in the compiled satellite images were equivalently distorted, as shown in Fig. 10(c) and 10(d), respectively. Additionally, the vector building and road marked in Fig. 11(a) and 11(b), respectively, were displaced, and this modification was accurately reflected in the compiled satellite images, as depicted in Fig. 11(c) and 11(d), respectively. Most of the inserted, distorted, and displaced buildings and roads in the compiled satellite image maps were clear and detailed, and all of them were consistent with the corresponding edited vector data and cohesive with their surroundings in the image maps. Additionally, analyzing the geometric errors in Table 3 from the perspective of the scale or resolution of the original data reveals that the geometric errors of the compiled polygonal objects were all smaller than 0.5 mm in the original scale and 3 pixels in the original resolution; additionally, the geometric errors of the compiled linear objects were all smaller than 1 mm in the original scale and 5 pixels in the original resolution. All of these geometric errors are exceedingly small with respect to the original scale or resolution and can therefore be tolerated in many cartography applications[53-54]. Thus, the proposed method is effective in implementing diverse modifications to geographic objects in satellite image maps.

To summarize, the proposed satellite map generation model uses a combination of deep learning with an encoder–decoder-based architecture and utilizes transfer learning

to convert edited polygonal and linear vector features to compiled geographic objects in satellite image maps. Thus, the compiling of geographic objects in the satellite image can be controlled through the editing of geographic features in the vector map. Accordingly, different operations on the vector data can be replicated in the satellite image map. Thereby, the proposed method facilitates precise and flexible remote sensing cartography.

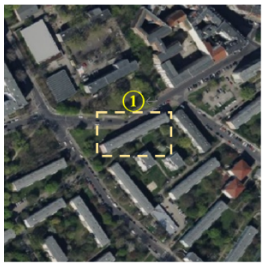
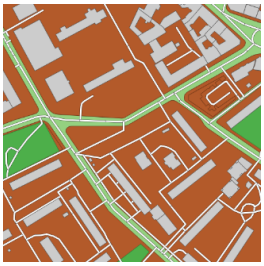
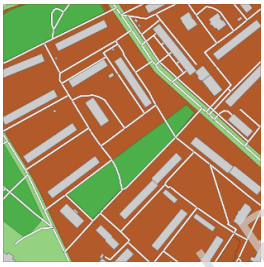
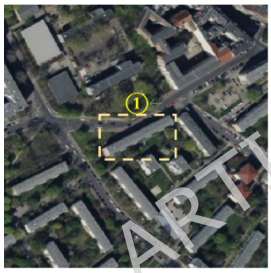
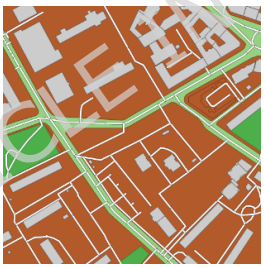
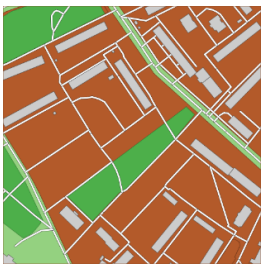
5.2.4. Discussion of compilation method

In principle, the sample extraction, basic model construction, and compilation stages are necessary for the proposed method of compiling geographic objects in satellite image maps. Naturally, the samples directly influence the model's learning accuracy and compilation effectiveness. Indeed, satellite image map compilation requires representative training samples, i.e., the extracted samples should possess similar texture characteristics to the geographic objects that need to be compiled. According to Tobler's first law[55], samples extracted from regions near the compiled geographic objects are representative; thus, the proposed method requires original map data containing the geographic objects to be compiled. According to empirical evidence[37], the textural characteristics of satellite images from the same city are similar; thus, in our experiment, the samples and compiled geographic objects were extracted from the same region of Berlin. Additionally, the basic satellite image generation model develops a general understanding of regional map data. The better the basic model learns, the more accurate the compilation is. Because the Pix2Pix-based model learned more textural characteristics of the selected region, it was selected as the basic model to perform various compilation operations on satellite images. This section highlights the necessity of the transfer learning process involved in the proposed method based on ablation experiments. Additionally, considering methods that focus on the compilation of satellite image maps, such ablation analyses can also be regarded as comparing results obtained from the basic model and those obtained from the improved model featuring transfer learning.

For the satellite image generation model without transfer learning, the mean of the RMSEs of the generated and original satellite images in the target region was 51.2; by comparison, for the model with transfer learning, the mean of the RMSEs was only 9.4. Thus, transfer learning improved the test accuracy of the model by 81.6%, thereby enhancing the quality of the compiled regions considerably. Apart from the quantitative similarity between the original and generated satellite images, the difference in quality between the compiled satellite images with and without transfer learning was also evident. Table 4 shows some compiled satellite images generated by the basic satellite image generation model without transfer learning and the improved model with transfer learning from the same vector tiles. Although the textures of the satellite images generated by the basic model were similar to those of the original satellite images, the geographic objects in these generated images were highly fuzzy and disordered. Additionally, the geometric and visual characteristics of the compiled geographic objects and their spatial relations could not be distinguished in the compiled satellite images generated by the basic model, which were substantially poorer than those generated by the improved model with transfer learning. Therefore, the transfer learning process strengthens the model's understanding of the target compilation regions, in addition to enhancing the accuracy of

the conversion of the target regions from the vector map to the satellite map. Thus, it enables the selective modification of geographic objects in satellite maps based on vector data, facilitating effective satellite image map compilation.

Table 4 Results obtained from basic model and improved model with transfer learning (the original vector tiles and satellite images were crawled from OpenStreetMap (https://www.openstreetmap.org/) and Bing Maps (https://cn.bing.com/maps), respectively; the output images were generated by the authors and are copyright-free).

Compilation operation	Original image	Input vector tile	Improved model's output	Basic model's output
Deletion of building ①				
Deletion of building ②				
Batch deletion of buildings around ①				
Batch deletion of buildings around ②				

Therefore, the transfer learning process improves the quality of the generated satellite image map, facilitating the compilation of geographic objects in the satellite image map. Thus, the transfer learning process is beneficial and necessary for the proposed method.

6. Conclusion and outlook

In this study, a satellite image generation model was constructed based on the encoder–decoder deep learning architecture, and transfer learning was then performed

to fit the mapping from vector data to satellite images for the compiling regions. Subsequently, the satellite image generation model was used to convert edited vector features to compiled geographic objects in the generated satellite image, which yielded object level–controllable satellite maps. Based on corresponding vector data modifications, the proposed method automates four types of operations—deletion, insertion, distortion, and displacement—to compile geographic objects in the satellite image map. Thus, the correspondence between vector data and satellite image maps, the convenience of indexing and editing vector data, and the powerful capability of deep learning and transfer learning to automate different compiling operations are fully leveraged to generate compiled geographic objects that are consistent with the modified vector features and cohesive with their surroundings in the generated satellite image. Accordingly, the proposed method not only reduces labor costs but also enhances the consistency and flexibility of the compilation of geographic objects in remote sensing cartography.

Additionally, the proposed method strengthens the theoretical foundation of deepfake remote sensing cartography by improving relevant cartographic representations. Furthermore, the method can be employed to produce clearer, multi-scale satellite image maps that preserve the security of important locations. For example, based on their vector data attribute, sensitive geographic objects in a satellite image map can be conveniently masked to ensure their security. Moreover, scenes involving combinations of real and virtual geographic data, such as maps in video games, animated videos, and the metaverse, can be created using the proposed method.

However, the method has certain limitations that must be addressed. For instance, the representation of linear geographic objects in the compiled satellite image map must be enhanced, prior knowledge must be incorporated to improve the accuracy of the compiled map, and the compilation of minute objects in the satellite image map must be achieved. Additionally, the effectiveness of the proposed method can be improved by adjusting the learning model and its hyperparameters, such as the model architecture, training optimizer, model learning rate, sample quantity, and even the division ratio of sample sets; such modifications will be tested in the future. Notably, the proposed method is not limited to any specific GenAI model. While only two encoder–decoder-based deep learning architectures were compared in this study, with the ongoing development of generative deep learning, other advanced models and training techniques can be used to enhance the accuracy of the proposed method in the future.

Acknowledgments: We thank our colleagues and the reviewers for their constructive comments and valuable suggestions, which have improved this paper substantially.

Funding: This work was supported by the Youth Fund of the National Natural Science Foundation of China [number 42301504].

Conflicts of interest/Competing interests: No potential conflict of interest is reported by the authors.

Data availability statement: The datasets generated and/or analyzed during the current study are available from the corresponding author upon reasonable request.

Authors' contributions: Du (first author & corresponding author) conceived of the proposed method and wrote the manuscript. Concrete algorithms were designed and implemented by Du and Qiu. Zeng and Cai helped Du estimate the proposed approach and test its applicability.

References

[1] Cui, Y. A Geospatial Frame Data Model to Simplify Digital Map compilation and Integration. British Columbia Ministry of Energy, Mines and Low Carbon Innovation, British Columbia Geological Survey Paper. Preprint at https://www.researchgate.net/publication/360013167 (2021).

[2] Timothy, J. P. Trust in Maps: What We Know and What We Need to Know. *Cartogr. Geogr. Inf. Sc.* **52**, 1-18 (2025**)**. https://doi.org/10.1080/15230406.2023.2281306.

[3] Monmonier, M. *How to Lie with Maps*, 3rd ed (2018). University of Chicago Press, Chicago, USA.

[4] Alexander, K. Trust Me, I'm a Cartographer: Post-truth and the Problem of Acritical Cartography. *Cartogr. J.* **54**, 193-195 (2017). https://doi.org/10.1080/00087041.2017.1376489.

[5] Yuan, G., Hao, Q. Digital Watermarking Secure Scheme for Remote Sensing Image Protection. *China Commun.* **17**, 88-98 (2020). https://doi.org/10.23919/JCC.2020.04.009.

[6] Tong, D., Ren, N., Zhu, C. Secure and Robust Watermarking Algorithm for Remote Sensing Images Based on Compressive Sensing. *Multimed. Tools Appl.* **78**, 16053-16076 (2019). https://doi.org/10.1007/s11042-018-7014-1.

[7] Chesney, B., Citron, D. Deep Fakes: a Looming Challenge for Privacy, Democracy, and National Security. *Calif. Law. Rev.* **107**: 1753-1819 (2019). https://scholarship.law.bu.edu/faculty_scholarship/640.

[8] Liu, X., Gong, W., Shang, L., Li, X., Gong, Z. Remote Sensing Image Target Detection and Recognition Based on YOLOv5. *Remote. Sens.* **15**, 4459 (2023). https://doi.org/10.3390/rs15184459.

[9] Zhao, Y., Sun, H., Wang, S. Small Object Detection in Medium-Low-Resolution Remote Sensing Images Based on Degradation Reconstruction. *Remote. Sens.* **16**, 2645 (2024). https://doi.org/10.3390/rs16142645.

[10] Wang, H., et al. A Remote Sensing Image Target Detection Algorithm Based on Improved YOLOv8. *Appl. Sci.* **14**, 1557 (2024). https://doi.org/10.3390/app14041557

[11] Lu, P., Xu, D., Ren, F., Xu, S., Qiu, T., Peng, R. Auto-detection and Hiding of Sensitive Targets in Emergency Mapping Based on Remote Sensing Data. *Geomatics and Information Science of Wuhan University*. **45**, 1263-1272 (2020). https://doi.org/10.13203/j.whugis20200131.

[12] Li, P., Bai, W. Automatic Hiding Method of Sensitive Targets in Remote Sensing Images Based on Transformer Structure. *Geomatics and Information Science of Wuhan University*. **47**, 1287-1297 (2022). https://doi.org/10.13203/j. whugis20220219.

[13] Boopathi, S., Pandey, B. K., Pandey, D. Advances in Artificial Intelligence for Image Processing: Techniques, Applications, and Optimization. In B. Pandey, D. Pandey, R. Anand, D. Mane, & V. Nassa (Eds.), *Handbook of Research on Thrust Technologies' Effect on Image Processing* (pp. 73-95). IGI Global Scientific Publishing. Preprint at https://doi.org/10.4018/978-1-6684-8618-4.ch006 (2023).

[14] Yang, J., Ruhaiyem, N. I. R. Review of Deep Learning-Based Image Inpainting Techniques. *IEEE Access*, **12**, 138441-138482 (2024). https://doi.org/10.1109/ACCESS.2024.3461782.

[15] Huang, Y. et al. Diffusion Model-Based Image Editing: A Survey. *T. Pattern. Anal.* **47**, 4409-4437 (2025). https://doi.org/10.1109/TPAMI.2025.3541625.

[16] Zala, K., Thumar, D., Thakkar, H. K., Maheshwari, U., Acharya, B. A Survey and Identification of Generative Adversarial Network Technology-based Architectural Variants and Applications in Computer Vision. *Int. J. Syst. Assur. Eng.* **15**, 4594-4615 (2024). https://doi.org/10.1007/s13198-024-02478-6.

[17] Dubey, S. R., Singh, S. K. Transformer-Based Generative Adversarial Networks in Computer Vision: A Comprehensive Survey. *IEEE Trans. Artif. Intell.* **5**, 4851-4867 (2024). https://doi.org/10.1109/TAI.2024.3404910.

[18] Kang, Y., Gao, S., Roth, R. E. Artificial Intelligence Studies in Cartography: a Review and Synthesis of Methods, Applications, and Ethics. *Cartogr. Geogr. Inf. Sc.* **51**, 599-630 (2024). https://doi.org/10.1080/15230406.2023.2295943.

[19] Li, J., Chen, Z., Zhao, X., Shao, L. MapGAN: an Intelligent Generation Model for Network Tile Maps. *Sensors*. **20**, 3119 (2020). https://doi.org/10.3390/s20113119.

[20] Zhang, Y., et al. An Enhanced GAN Model for Automatic Satellite-to-Map Image Conversion. *IEEE Access*. **8**, 176704-176716 (2020). https://doi.org/10.1109/ACCESS.2020.3025008.

[21] Chen, X., et al. SMAPGAN: Generative Adversarial Network-based Semi-supervised Styled Map Tile Generation Method. *IEEE T. Geosci. Remote.* **59**, 4388-4406 (2020). https://doi.org/10.1109/TGRS.2020.3021819.

[22] Chen, X., Yin, B., Chen, S., Li, H., Xu, T. Generating Multiscale Maps from Satellite Images via Series Generative Adversarial Networks. I*EEE Geosci. Remote. S.* **19**, 1-5 (2021). https://doi.org/10.1109/LGRS.2021.3129285.

[23] Wu, A. N., Biljecki, F. GANmapper: Geographical Data Translation. *Int. J. Geogr. Inf. Sci.* **36**, 1394-1422 (2022). https://doi.org/10.1080/13658816.2022.2041643.

[24] Touya, G., Zhang, X., Lokhat, I. Is Deep Learning the New Agent for Map Generalization? *Int. J. Cartography.* **5**, 142-157 (2019). https://doi.org/10.1080/23729333.2019.1613071

[25] Feng, Y., Thiemann, F. & Sester, M. Learning Cartographic Building Generalization with Deep Convolutional Neural Networks. *ISPRS Int. J. Geo-Inf.* **8**, 258-278 (2019). https://doi.org/10.3390/ijgi8060258.

[26] Du, J., Wu, F., Xing, R., Gong, X., Yu, L. Segmentation and Sampling Method for Complex Polyline Generalization Based on a Generative Adversarial Network. *Geocarto. Int.* **37**, 4158-4180 (2021). https://doi.org/10.1080/10106049.2021. 1878288.

[27] Du, J., Wu, F., Xing, R., Li, C., Li, J. Trial and Comparison of Some Encoder-Decoder Based Deep Learning Models for Automated Generalization of Buildings. *Geomatics and Information Science of Wuhan University.* **47**, 1052-1062 (2022). https://doi.org/10.13203/j.whugis20200143.

[28] Fu, C., Zhou, Z., Xin, Y., Weibel, R. Reasoning Cartographic Knowledge in Deep Learning-based Map Generalization with Explainable AI. *Int. J. Geogr. Inf. Sci*. **38**, 2061-2082 (2024). https://doi.org/10.1080/13658816. 2024.2369535.

[29] Zhu, W., Guo, Q., Yang, N., Tong, Y., Zheng, C. An Improved Generative Adversarial Network for Generating Multi-Scale Electronic Map Tiles Considering Cartographic Requirements. *ISPRS Int. J. Geo-Inf.* **13**, 398 (2024). https://doi.org/10.3390/ijgi13110398.

[30] Kang, Y., Gao, S., Roth, R. E. Transferring Multiscale Map Styles Using Generative Adversarial Networks. *Int. J. Cartography*. **5**, 115-141 (2019). https://doi.org/10.1080/23729333.2019.1615729.

[31] Wu, M., Sun, Y., Lv, G. Cartographic Style Transfer: Idea, Review and Envision. *Geomatics and Information Science of Wuhan University*. **47**, 2069-2084 (2022). https://doi.org/10.13203/j.whugis20220439.

[32] Christophe, S., Mermet, S., Laurent, M., Touya, G. Neural Map Style Transfer Exploration with GANs. *Int. J. Cartography.* **8**: 18-36 (2022). https://doi.org/10.1080/23729333.2022.2031554.

[33] Clarke, K. C., Johnson, J. M., Trainor, T. Contemporary American Cartographic Research: a Review and Prospective. *Cartogr. Geogr. Inf. Sc.* **46**: 196-209 (2019). https://doi.org/10.1080/15230406.2019.1571441.

[34] Chen, Y., et al. A Framework of Pan-maps: Facilitating a Unification of Maps and Map-likes. In Proceedings of the International Cartographic Association. Florence, Italy, 14-18 December 2021. Preprint at https://doi.org/10.5194/ica-proc-4-20-2021 (2021).

[35] Ying, S., Hou, S., Chen, Y., Su, J. Virtuality and Reality of Game Maps. *Geomatics and Information Science of Wuhan University*. **47**, 2085-2095 (2022).https://doi.org/10.13203/j.whugis20220406.

[36] Horbiński, T., Zagata, K. View of Cartography in Video Games: Literature Review and Examples of Specific Solutions. *KN - Journal of Cartography and Geographic Information*, **72**, 117-128 (2022). https://doi.org/10.1007/s42489-022-00104-8.

[37] Zhao, B., Zhang, S., Xu, C., Sun, Y., Deng, C. Deep Fake Geography? When Geospatial Data Encounter Artificial Intelligence. *Cartogr. Geogr. Inf. Sc.* **48**, 338-352 (2021). https://doi.org/10.1080/15230406.2021.1910075.

[38] Bidgoli, A. J. Investigating the Impact of the Metaverse on Design Discourse: A Cartography of Emerging Disciplines. Preprint at https://www.researchgate.net/publication/376799945_Investigating_the_Impact_of_the_Metaverse_on_ Design_ Discourse_A_Cartography_of_Emerging_Disciplines (2023).

[39] Zhu, J., Park, T., Isola, P., Efros, A. A. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. In Proceedings of 2017 IEEE International Conference on Computer Vision (ICCV). Venice, Italy, 22-29 October 2017. Preprint at https://arxiv.org/pdf/1703.10593 (2017).

[40] Xu, Y., Yu, W., Ghamisi, P., Kopp, M., Hochreiter, S. Txt2Img-MHN: Remote Sensing Image Generation from Text Using Modern Hopfield Networks. *IEEE. T. Image. Process.* **23**, 5737-5750 (2023). https://doi.org/10.1109/TIP.2023. 3323799.

[41] Xu, C., Zhao, B. Satellite Image Spoofing: Creating Remote Sensing Dataset with Generative Adversarial Networks. In the 10th International Conference on Geographic Information Science, Dagstuhl, Germany, 2018. Preprint at https://doi.org/10.4230/LIPIcs.GIScience.2018.67 (2018).

[42] Wang, C., Chen, B., Zou, Z., Shi, Z. Remote Sensing Image Synthesis via Semantic Embedding Generative Adversarial Networks. *IEEE T. Geosci. Remote.* **61**, 1-11 (2023). https://doi.org/10.1109/TGRS.2023.3279663.

[43] Wu, H., Li, Z. Scale Issues in Remote Sensing: A Review on Analysis, Processing and Modeling. *Sensor.* **9**, 1768-1793 (2009). https://doi.org/10.3390/s90301768.

[44] Li, Z., Openshaw, S. Algorithms for Automated Line Generalization Based on a Natural Principle of Objective Generalization. *Int. J. Geogr. Inf. Sci.* **6**, 373-389 (1992). https://doi.org/10.1080/02693799208901921.

[45] Esri. An Overview of the Projections and Transformations Toolset. preprint at https://doc.arcgis.com/en/allsource/latest/ analysis/geoprocessing-tools/data-management/an-overview-of-projections-and-transformations-toolset.htm (2024).

[46] Tan, J., Guo, Z., Li, J., Dian, S. *DeepLearning-500-questions.* Electronic Industry Press, Beijing, China. Preprint at https://github.com/scutan90/DeepLearning-500-questions (2020).

[47] Zhuang, F., et al. A Comprehensive Survey on Transfer Learning. *P. IEEE.* **109**, 43-76 (2021). https://doi.org/10.1109/ JPROC.2020.3004555.

[48] Touya, G., Reimer, A. Inferring the Scale of OpenStreetMap Features. In *OpenStreetMap in GIScience*, *Lecture Notes in Geoinformation and Cartography*; edited by Arsanjani, J. J., et al. Springer Cham, Cham, Switzerlan. Preprint at https://doi.org/10.1007/978-3-319-14280-7_5 (2015).

[49] Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., Liang, J. UNet++: Redesigning Skip Connections to Exploit Multiscale Features in Image Segmentation. *IEEE T. Med. Imaging.* **39**, 1856-1867 (2020). https://doi.org/10.1080/10.1109/ TMI.2019.2959609.

[50] Isola, P., Zhu, J., Zhou, T., Efros, A. A. Image-to-image Translation with Conditional Adversarial Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, HI, USA, 9 November 2017. Preprint at https://doi.org/10.1109/CVPR.2017.632 (2017).

[51] Kingma, D. P., Ba, J. Adam: a Method for Stochastic Optimization. In Proceedings of the International Conference on Learning Representations. San Diego, CA, USA, 7-9 May 2015. Preprint a https://arxiv.org/pdf/1412.6980 (2015).

[52] Anjum, U., Afzal, U., Hussain, A., Hussain, I., Shah, S. A. JPEG Image Compression Using Multiple Core Strategy in FPGA achieving High Peak Signal to Noise Ratios. In the International Congress of Advanced Technology and Engineering (ICOTEN 2021). Taiz, Yemen, 2021, pp. 1-6, Preprint at https://doi.org/10.1109/ICOTEN52080.2021. 9493460 (2021).

[53] Wang, J., Fan, Y., Han, T. Cartographic generalization theory of general map. Surveying and Mapping Press, Beijing, China, 1993.

[54] Qian, H. Z., Zhang, M., Wu, F. A New Simplification Approach Based on the Oblique-dividing-Curve Method for Contour Lines. *Int. J. Geogr. Inf. Sci.* **5**, 153-174 (2016). https://doi.org/10.3390/ijgi5090153.

[55] Tobler, W. R. A Computer Movie Simulating Urban Growth in the Detroit Region. Econ. Geogr. **46**, 234-240 (1970). http://dx.doi.org/10.2307/143141.