International Journal of Geo-Information

MDPI

Article

An Assessment of the Map-Style Influence on Generalization with CycleGAN: Taking Line Features as an Example

Heng Yu [1,2,3], Haoxuan Chen [1,2,3] and Ling Zhang [1,2,3,*]

[1] Key Laboratory of Virtual Geographic Environment (Nanjing Normal University), Ministry of Education, Nanjing 210023, China; 10210339@njnu.edu.cn (H.Y.); 231301018@njnu.edu.cn (H.C.)
[2] State Key Laboratory Cultivation Base of Geographical Environment Evolution (Jiangsu Province), Nanjing 210023, China
[3] Jiangsu Center for Collaborative Innovation in Geographical Information Resource Development and Application, Nanjing 210023, China
* Correspondence: lingzhang@njnu.edu.cn

Abstract: As the complexity of GIS data continues to increase, there is a growing demand for automated map generalization. As end-to-end generative models, GAN models offer new solutions for automated map generalization. This study explores the impact of different map symbolization configurations on generative models, specifically using CycleGAN for line feature generalization. The quality of the generated results was assessed by constructing various symbolization datasets (line width, type, and color) and evaluating CycleGAN's performance using metrics such as the MSE, SSIM, and PSNR. The results indicate that moderate line widths (0.5–1) yield better detail preservation, and different line types (framed lines and dashed lines) can highlight feature boundaries and enhance visual perception. By contrast, high-contrast color schemes enhance feature differentiation but increase pixel-level errors. This study concludes that generative models can maintain the geometric structure and spatial distribution of line features, but it is crucial to choose more suitable line features for different scenarios to meet detail requirements, ensuring high-quality outputs under diverse configurations.

Keywords: map generalization; linear features; CycleGAN; symbolization configurations; GIS

check for updates

Citation: Yu, H.; Chen, H.; Zhang, L. An Assessment of the Map-Style Influence on Generalization with CycleGAN: Taking Line Features as an Example. *ISPRS Int. J. Geo-Inf.* **2024**, *13*, 418. https://doi.org/10.3390/ijgi13120418

Academic Editors: Florian Hruby and Wolfgang Kainz

Received: 5 August 2024
Revised: 8 November 2024
Accepted: 19 November 2024
Published: 21 November 2024

Copyright: © 2024 by the authors. Published by MDPI on behalf of the International Society for Photogrammetry and Remote Sensing. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Map generalization is a crucial process in multi-scale spatial representation related to map design. Automated map generalization is a technology that simplifies complex geographic data into map representations suitable for specific purposes and scales using computer algorithms. As geographic information systems (GISs) advance and data complexity increases, the need for automated map generalization has become more critical. This process typically involves identifying and extracting key geographic features while abstracting and simplifying the spatial information as necessary while maintaining the accuracy of essential geographic features and spatial relationships [1,2]. Traditional methods for map generalization rely on expert systems, multi-agent systems, and early machine learning or optimization techniques [3]. While these methods have succeeded in specific applications, they often require extensive manual intervention and specialized knowledge. Map generalization is people's cartographic behavior driven by perceptual responses, simplifying the shapes of things, reducing details, and highlighting features. Thus, it is the product of visual perception that can realize map generalization in real-time. The success of deep learning in computer vision makes it a promising paradigm for map generalization [4].

Machine learning, especially deep learning, has been increasingly explored in the field of map generalization, with researchers investigating its potential to automate and improve the generalization process [5]. The question of whether deep learning could be the new

agent for map generalization has been raised, highlighting the promising perspectives and potential issues that could be addressed with this technology [4]. During the initial stages of incorporating deep learning for map generalization, the primary emphasis was on the generalization of specific map features, such as creating small-scale maps using machine learning; data-rich automated road selection methods [6], DT algorithms [7], K-NN algorithms [8], and PCA algorithms [9] have been used to model settlement selection; BPNN, GAN, and FCN have been used for simplifying building edges [10–12]; and the U-net network has been used for line feature simplification [13–15]. Significantly, these endeavors entail the utilization of U-Net or GAN architectures, which are customary in the realm of computer vision. This involves creating input data through rasterizing spatial vector data [16]. The results of these efforts reveal substantial benefits in employing well-adapted deep learning models for image processing in the realm of map generalization. While this approach shows promise, it is important to acknowledge that there are also some noteworthy drawbacks associated with image-based learning [12]. One drawback of machine learning models is their unpredictable learning process. These models solve classification tasks using principles that may not align with good cartographic practices. They often lack an explicit representation of cartographic knowledge and the understanding of how to perform specific cartographic operations [3]. Researchers are exploring various ways to encode input for models. Some use multi-dimensional tensors [17], others use graphs [18], and another method involves transforming a line into a vector for processing via an autoencoder [19]. These techniques have shown promise in preserving maps' spatial and contextual characteristics, managing complex geographic information effectively, and facilitating efficient generalization across varying scales [20]. However, these methods are applied in vector data format. Another drawback is that supervised learning methods need to use a large amount of training data. Still, GAN methods [21,22], especially the application of Cycle-Consistent Generative Adversarial Networks (CycleGANs) [23], are a new solution for performing end-to-end image-based map generalization for small sample training, such as using CycleGAN and pix2pix models to automate the generalization of urban areas in topographic maps [15], using CycleGAN for coarse-to-fine semantic road segmentation [24], and encoding cartographic knowledge into deep learning models to perform map generalization tasks [25].

Overall, the literature suggests that deep learning holds significant potential for map generalization tasks, with researchers exploring various architectures and techniques to improve automation and accuracy. Real-time maps often need to adapt stylistically to different use cases, such as navigation, tourism, or emergency response. If CycleGAN can successfully generalize maps while varying symbolization parameters, the model could potentially switch styles dynamically, tailoring map outputs to different user needs on the fly [26]. Many researchers have pointed out that cartographical knowledge should also be trained in such models [27], which may bring better-generalized maps. Although the research trend has gradually shifted to vector-based deep learning methods, raster image-based map generalization is still a research hotspot. The author believes that map symbols are also a form of cartographic knowledge, so this study takes line features as an example to explore how different symbolization parameters (such as line width, line type, and color) affect the transformation of large-scale images into small-scale generalized images in the CycleGAN model.

This article is organized as follows: The next section presents the research methods used in this study, detailing the application of CycleGAN and the construction of different symbolization datasets. Section 3 describes the evaluation metrics of the experimental results and their significance. Section 4 provides a comprehensive assessment of the generative results of different line features based on the preset metrics. Finally, Section 5 concludes this study by summarizing the findings and proposing future research directions.

2. Methods

To investigate the impact of different symbolization methods on the quality of generative generalization for line features, this study adjusted three common symbolization parameters—line width, line type, and color; applied generative adversarial networks for map generalization; and evaluated the quality of the generalized results. This section details CycleGAN and the construction methods of different symbolization datasets. It discusses which aspects of each configuration have an impact on the generation results. The specific research framework is shown in Figure 1.

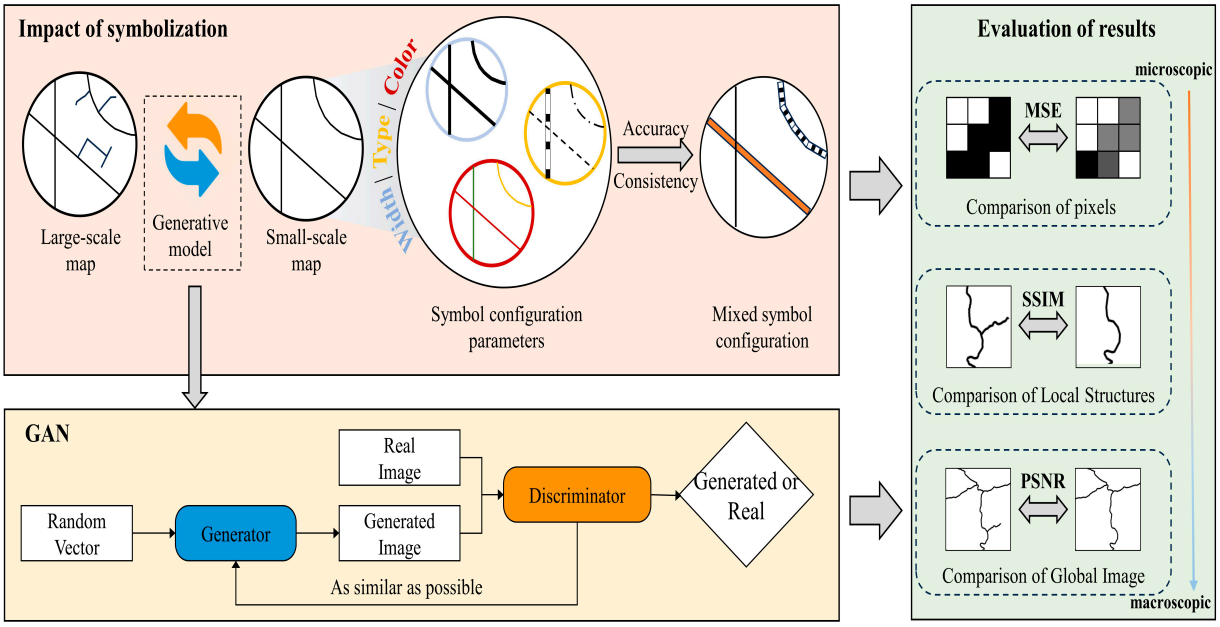


Figure 1. The research framework of symbolization impact on generative linear feature generalization.

The method in this paper is divided into three parts: (1) Symbolization Configuration. Line features are set according to different symbolization parameters (line width, line type, color) to construct generalized sample data. (2) Generative Generalization. The sample data are input into a generative adversarial model for training, and test experiments are conducted to generate generalized data. (3) Result Evaluation. Quality assessment of the generated results is conducted at microscopic, local, and macroscopic levels using metrics such as MSE, SSIM, and PSNR. The specific method is explained in detail below.

2.1. Generative Models

In recent years, generative models have developed rapidly in computer vision and image processing. They are primarily utilized to create realistic-looking images. They are used in map simplification by training them with pre- and post-generalization data to generate generalized images. This work primarily used the CycleGAN model for testing, which performs well in image-to-image generation tasks and is widely used.

CycleGAN is an unsupervised generative model consisting of two generators and two discriminators. The generators are responsible for converting images from one style to another, while the discriminators judge whether the generated images are authentic. The total loss of CycleGAN consists of two parts, the adversarial loss and the cycle consistency loss, which is formulated as follows:

$$\mathcal{L}_{GAN} = \mathcal{L}(G, D_Y, X, Y) + \mathcal{L}(F, D_x, X, Y) \quad (1)$$

$$\mathcal{L}_{CycleGAN} = \mathcal{L}_{GAN} + \lambda \mathcal{L}_{Cycle}(G, F) \quad (2)$$

where the $\mathcal{L}_{GAN}$ term ensures the co-evolution of the generator and discriminator, thereby ensuring that the generator produces more realistic images. By contrast, the $\mathcal{L}_{Cycle}$ term ensures that the output images of the generator are only stylistically different from the input images while the content remains the same. By introducing cycle consistency loss, it maintains the consistency of image style transformations, thereby improving the credibility of the generated results. When applied to map generalization, it ensures that the generated images are at a smaller scale while the feature content remains consistent with pre-generalization. The specific model architecture is shown in Figure 2.

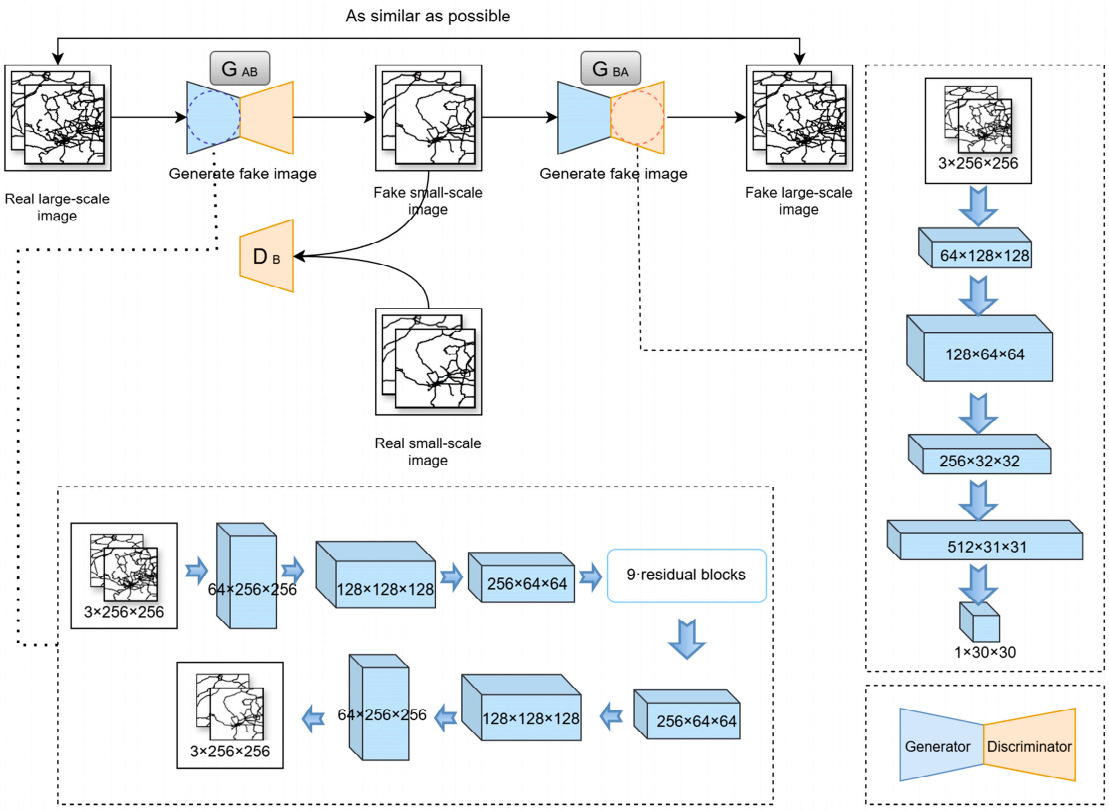


Figure 2. The model architecture of CycleGAN.

In this work, the generator was designed using a ResNet architecture with nine residual blocks. The discriminator followed a PatchGAN architecture with a 70 × 70 receptive field, allowing it to classify whether overlapping image patches are real or generated [23]. The generator utilized the ReLU activation function in its internal layers for non-linear transformations and the Tanh activation function in the output layer to generate images in the $[-1, 1]$ range. In contrast, the discriminator employed the LeakyReLU activation function, which helps to avoid dead neurons by allowing a small gradient for negative inputs.

In studying the process of generalizing linear features, the training data included image tiles of various scales representing the linear features before and after generalization. The test data consisted of small-scale image tiles, and the model automatically produced large-scale images as output. This work used the CycleGAN model to compare the effects of different symbolization configurations on linear features in the map generalization task, thereby assessing the impact of symbolization configurations on the results of generative models. This provides a basis for exploring the feasibility of applying generative models in map generalization.

2.2. Symbolization Configurations

Symbolization configurations play a crucial role in cartography. Different symbolization parameters significantly affect the performance of generative models in map general-

ization. This work investigated how symbolization parameters such as line width, line type, and color impact the generative effects of CycleGAN. It also proposed related experiments to evaluate their roles. The following describes a preliminary qualitative analysis and the selection of experimental symbols.

2.2.1. Line Width

Line width is one of the critical parameters in symbolization configurations. Different line widths can affect the visualization of linear features and the feature extraction capability of the model. For this study, line width categories were defined with specific ranges: Thin (<0.5 mm), Medium (0.5–1 mm), Thick (1–1.5 mm), and Very Thick (≥1.5 mm). These ranges were chosen to reflect common cartographic practices, ensuring consistency and reproducibility. Thinner lines on a map help maintain clarity when there are many linear features, but they make it more difficult for the model to extract features due to their subtle appearance. Conversely, thicker lines provide higher visual prominence, making them easier for the model to recognize and generate. However, if lines are excessively thick, they can obscure features and lead to overlap in dense areas, reducing detail preservation and spatial accuracy. The characteristics of different line widths are shown in Table 1.

Table 1. The characteristics of multi-width linear features.

Width	Visual Perception	Model Recognition Difficulty	Detail Preservation Level
Thin (<0.5 mm)	Clear details	High	High
Medium (0.5–1 mm)	Clear overall	Medium	Medium
Thick (1–1.5 mm)	Highlights main features	Low	Low
Very Thick (≥ 1.5 mm)	Clear spatial distribution	Very Low	Very Low

To explore the role of line width settings in generative models, this work constructed datasets of linear features with various widths. This study investigated the function of line width settings in generative models by analyzing and evaluating the impact of different line widths on the generation results. Data samples are shown in Figure 3.

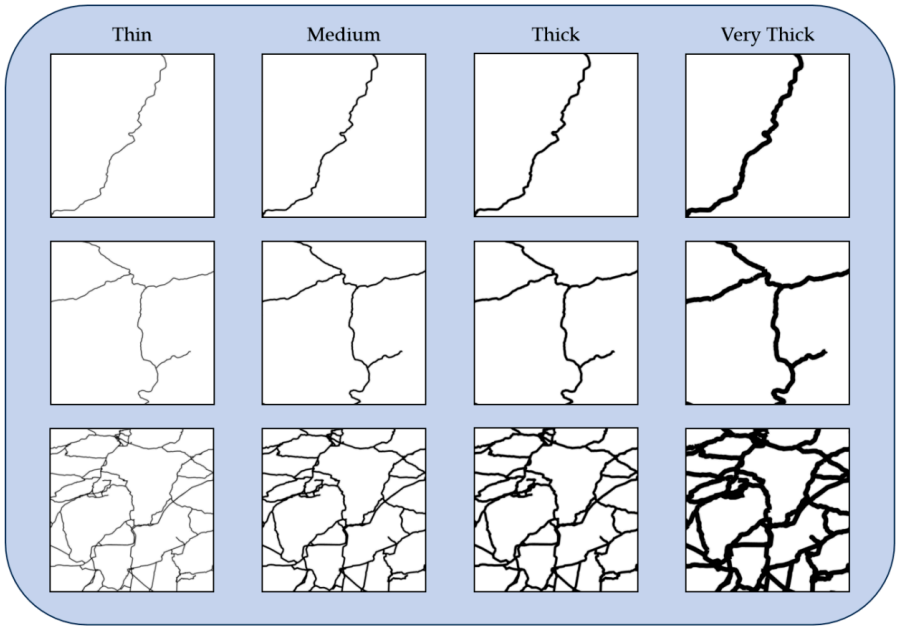


Figure 3. Example of multi-width linear feature dataset.

2.2.2. Line Type

Line types describe different linear features, and different line types can represent various geographic characteristics. Various line types play a crucial role in cartography. In

this study, line types were categorized and assigned specific visual and structural properties: dashed lines represent intermittent connections and, due to their segmented nature, are relatively easy to recognize and generate, thus, typically exhibiting lower structural complexity; framed lines are continuous and emphasize major roads or boundaries on maps, making them visually clear and contributing to spatial continuity; and striped lines are used to represent unique features and have higher structural complexity due to their segmented and varied line segments. The "structural complexity" mentioned in this study is defined based on the visual characteristics of the line types and the difficulty encountered by the generative model in processing them. These line types are common in cartographic symbolization, and Table 2 shows the characteristics of these line types.

Table 2. The characteristics of multi-type linear features.

Type	Model Recognition Difficulty	Visual Continuity	Structural Complexity
Dashed lines	Medium	Discontinuous	Low
Framed lines	Low	Continuous	Medium
Striped lines	Medium	Discontinuous	High

Therefore, this work constructed datasets with various line types to evaluate the sensitivity and performance of the generative model to different line types; thus, exploring the impact of line type configurations on the model's feature extraction and generation effects. Sample data are shown in Figure 4.

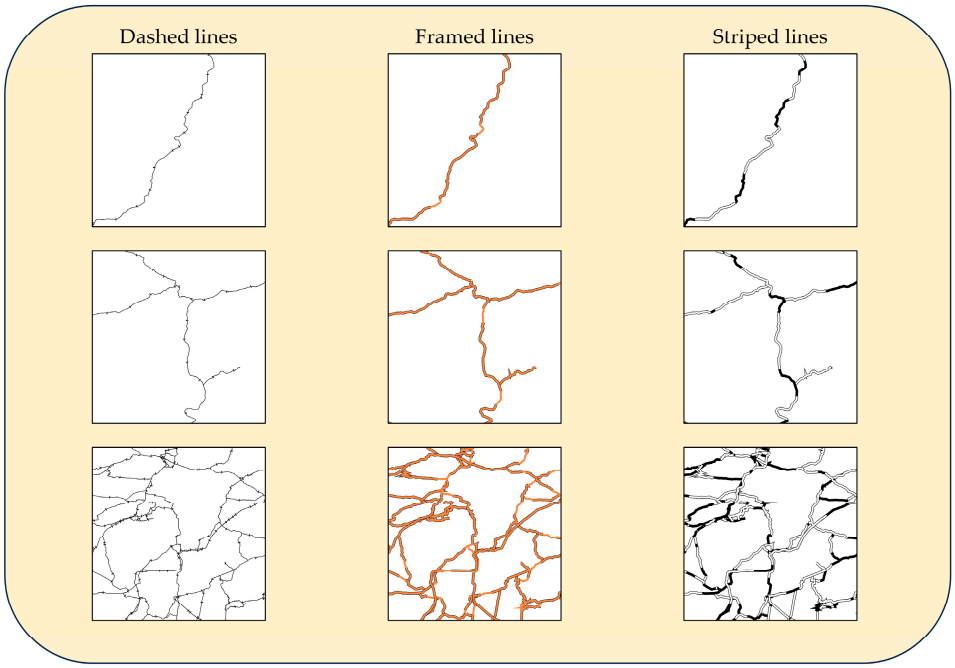


Figure 4. Example of multi-type line feature dataset.

2.2.3. Line Color

Color is an essential parameter in symbolization configurations, directly affecting the generative model's feature extraction and result generation. Color is crucial for conveying information, distinguishing features, and significantly impacting the model's ability to recognize images. This study divided color configurations into grayscale and high-contrast color schemes to assess their impact on model performance. The grayscale dataset was used to analyze the model's feature extraction capability without color information, emphasizing the model's reliance on shape and structure, which increases recognition difficulty. This configuration helps in understanding the model's baseline performance without color cues.

In contrast, the high-contrast color dataset was used to evaluate the impact of rich color variations in the generation results and model adaptability. Introducing more complex visual information made distinguishing features more accessible for the model, thereby reducing recognition difficulty, but the higher contrast could also increase processing complexity. Table 3 shows the characteristics of these configurations.

Table 3. The characteristics of multi-color linear features.

Color	Visual Perception	Model Recognition Difficulty	Contrast
Grayscale	Single color	Medium	Low
High-contrast color	Rich colors	Low	High

This work explored the impact of different color configurations on the generation results. By constructing grayscale and high-contrast color datasets, this study evaluated the influence of color configurations on the generative model. Sample data are shown in Figure 5.

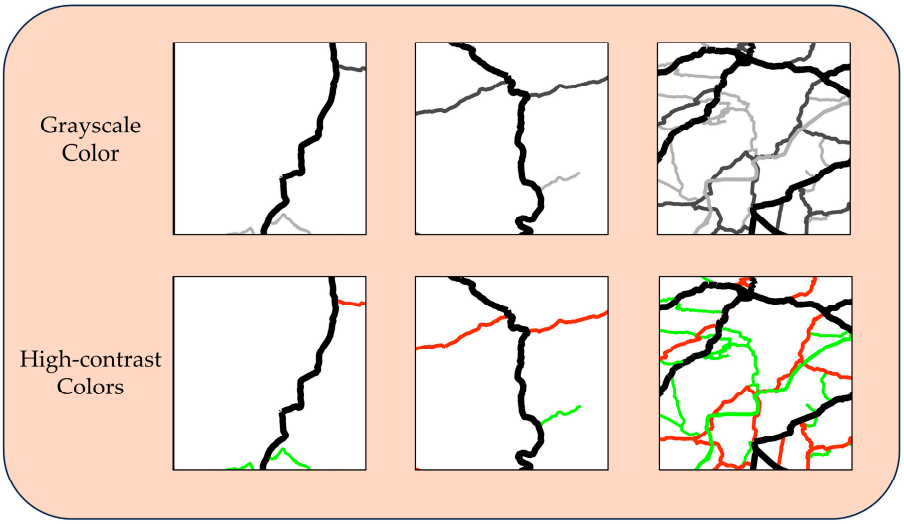


Figure 5. Example of multi-color line feature dataset.

3. Evaluation Method of Generated Results

To assess the impact of different symbolization configurations on the generative effects of CycleGAN, this work selected three evaluation metrics: MSE (Mean Squared Error), SSIM (Structural Similarity Index), and PSNR (Peak Signal-to-Noise Ratio). These metrics measure the quality of the generated images from different dimensions, providing a comprehensive evaluation. The following is a detailed description of each evaluation metric and its application in this study.

3.1. MSE

MSE measures the difference between two images, calculated as the average squared differences between each corresponding pixel. The lower the MSE value, the closer the generated image is to the reference image. The formula for MSE is as follows:

$$MSE = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}[I(i,j) - K(i,j)]^2 \quad (3)$$

where i and j represent the row and column indices, respectively; I(i, j) is the pixel value of the corresponding pixel in image *I*; K(i, j) is the pixel value of the corresponding pixel in image K; m is the total number of pixels in image I; and n is the total number of pixels in image K.

This study used MSE to evaluate the pixel-level error between the generated and reference images. A lower MSE value indicates that the generated image has a minor difference from the reference image at the pixel level. Specifically, in the task of linear feature generalization, an MSE value below 100 generally indicates good model performance. In contrast, higher values may signify significant pixel-level deviations that could affect detail retention. MSE provides an accurate numerical measure of the difference, helping to analyze the fidelity of the generated image at the pixel level.

3.2. SSIM

SSIM is a metric for assessing the similarity between two images, primarily considering brightness, contrast, and structural information. SSIM effectively evaluates the fidelity of local structures in images and is a good indicator of subjective image quality. The formula for SSIM is as follows:

$$SSIM(I,K) = \frac{(2\mu_I\mu_K + C_1)(2\sigma_{IK} + C_2)}{(\mu_I{}^2 + \mu_K{}^2 + C_1)(\sigma_I{}^2 + \sigma_K{}^2 + C_2)} \tag{4}$$

where μ_I and μ_K are the average brightness values of images I and K, respectively; $\sigma_I{}^2$ and $\sigma_K{}^2$ are the standard deviations of the pixel values of images I and K, respectively; σ_{IK} is the covariance of the pixel values between the two images; and C1 and C2 are constants used to prevent the denominator from being zero.

In this study, SSIM was used to evaluate the similarity of structural information between the generated image and the reference image. Given the critical importance of structural information in linear features of maps, SSIM reflects the fidelity of the generated results in terms of details and local structures, determining whether the generative model can accurately reproduce these features. Typically, an SSIM value above 0.9 indicates good image quality, while a range between 0.8 and 0.9 is considered acceptable. Values below 0.8 signify poor image quality, suggesting that the model struggles to maintain structural fidelity.

3.3. PSNR

PSNR is a pixel-based error measurement method to assess an image's quality and noise level, and it is measured in decibels (dB), with higher values indicating better image quality. The formula for PSNR is as follows:

$$PSNR = 10 * log_{10}\left(\frac{MAX_I^2}{MSE}\right) \tag{5}$$

where MAX_I is the maximum pixel value of the image I.

PSNR was primarily used to evaluate the overall quality and noise level of the generated images. A high PSNR value in map generalization indicates that the generated image has higher overall quality and less noise than the reference image. This is crucial for assessing the generative model's overall performance when handling different symbolization configurations, ensuring consistency in the global visual effect of the generated images. Typically, a PSNR value above 40 dB signifies excellent image quality, while values between 30 and 40 dB indicate an acceptable range with some distortion. Values below 30 dB suggest poor image quality, indicating significant noise or discrepancies that may affect the visual integrity.

These metrics comprehensively evaluate the impact of different symbolization configurations on the generative effects. MSE assesses pixel-level errors, SSIM reflects the fidelity of the generated images in terms of local structures, and PSNR measures the images' overall quality and noise level. The combination of these metrics evaluates the quality of the generated images from multiple dimensions, including global quality, local structure, and pixel differences, providing quantitative evaluation criteria for exploring the role of symbolization configurations in generative models for map generalization.

4. Experiments and Results

4.1. Study Area and Data Preparation

The data selected for this experiment came from the vector tile data on the Mapbox website (https://www.mapbox.com accessed on 16 May 2024). Specifically, three level-6 tiles at a scale of 1:5,000,000 and forty-eight level-8 tiles at a scale of 1:500,000, covering the same spatial extent, were used. These tiles cover the longitude range of 61.77°–73.22° and 78.58°–84.55°, with a latitude range of 36.67°–31.88° for both, encompassing a total area of approximately 373,588 km^2. Regarding spatial distribution, the area includes densely populated road sections and areas with fewer roads. Regarding geometric features, it contains both continuous curves and longer straight segments. Therefore, selecting this area for experimentation provides a certain diversity in sample construction, making it easier to assess the model's generative performance across different types of roads.

The tiles are encoded and stored in a triplet format, which includes the tile level, row number, and column number. For example, the tile labeled 6-43-25 indicates that it is at the 6th level, with a column number of 43 and a row number of 25. The linear features were stored as vector format files and converted to shp format files. Road features were selected through the attribute table. This resulted in 82 linear road features at level 6 and 579 at level 8. The symbolization configurations were implemented using ArcGIS 10.8 software, where multiple sets of symbolized linear feature data were configured according to the preset schemes and exported as images. Because current models cannot directly process vector data, the configured linear feature data were exported as images for experimentation on raster datasets. Each image was then expanded in data volume through 5 × 5 tiling and clockwise 90-degree rotation. After removing blank images and invalid data containing only a few linear features, 2496 training images were obtained, with 104 images for each level and each symbolization configuration. During the model training phase, images numbered 1–77 from each level were used as the training set and those numbered 78–104 were used as the test set. As shown in Figure 6, the red dividing lines represent the expansion of the dataset by dividing each image into 5 × 5 blocks, and the green dividing lines represent the correspondence in a quadtree encoding, where each 6th-level tile has 16 corresponding 8th-level tiles.

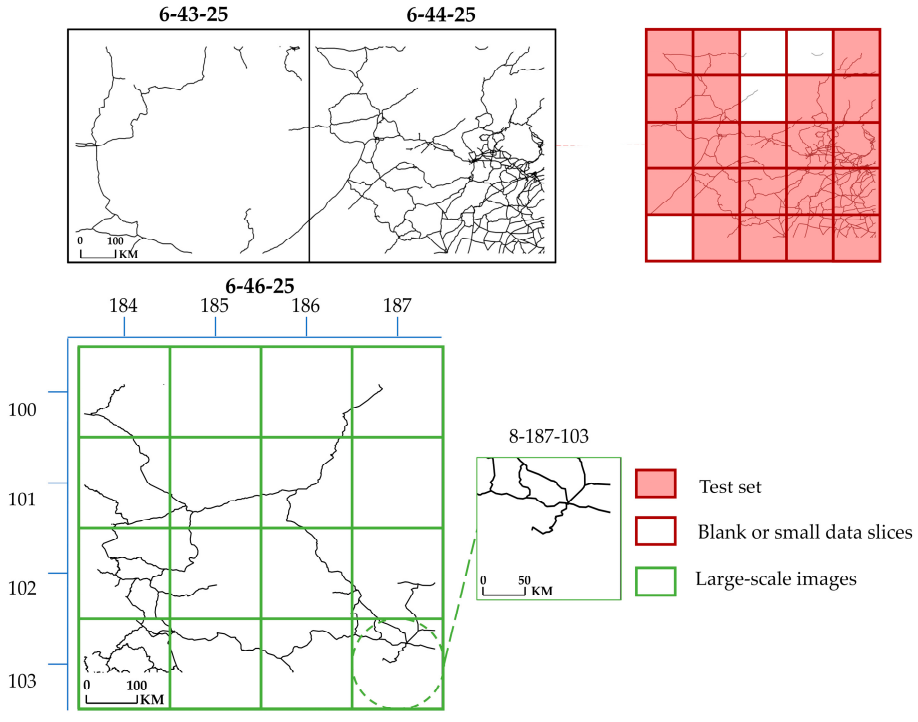


Figure 6. Study area and data samples.

4.2. Generalization of Different Symbolized Linear Features

4.2.1. Generalization of Multi-Width Linear Features

In exploring the impact of line width on the generation of map linear features using generative models, the experimental results show that the model exhibits a certain degree of stability across different line widths, with consistent spatial distribution and connectivity outcomes. The model can identify geometric features of the lines in the image, such as length and connectivity, and based on these features, retain or remove certain minor roads, especially those that are dead-ends and not connected to other roads, as shown in Figure 7.

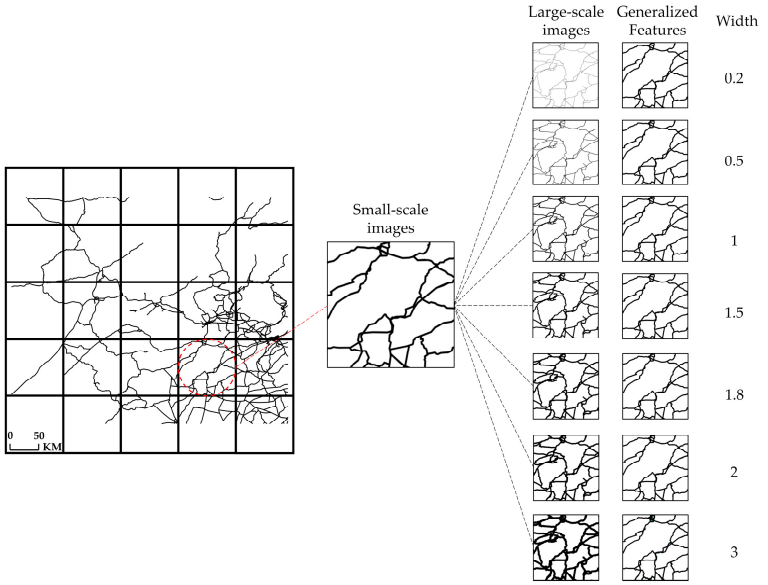


Figure 7. Generation results of multi-width linear features.

Calculations based on the predefined evaluation metrics reveal that line width has a minimal impact on the structural similarity between the generated image and the reference image, with both maintaining high similarity across different line widths. However, when the line width ranges from 0.5 to 1, the average PSNR value of the generated images gradually increases, improving the model's overall quality. Meanwhile, the average MSE value decreases, signifying reduced pixel-level errors in the generated images. This suggests that within this range, the line width configuration better preserves the details and structure of the map's linear features, making the generated results closer to the reference images. However, when the line width exceeds 1, the average SSIM and PSNR values begin to decline, and the average MSE value increases. This phenomenon indicates that extensive lines must be improved for the model to recognize and generate images, resulting in decreased detail and overall quality. Overly broad lines can cause the linear features to merge and create chaotic expressions in high-density areas, affecting the accuracy and consistency of the generated images, as shown in Figure 8.

Among the slices with poor evaluation metrics, which indicate areas with suboptimal generative results, most contain many roads. When the line width increases to 1.5, the model's processing capability is limited, failing to maintain the gaps between adjacent roads. Consequently, closely spaced but unconnected line segments are erroneously connected, and continuous curved roads tend to become straight. This suggests that larger line widths may reduce sensitivity to subtle differences between lines, neglecting the details and structural differences, resulting in structural distortions in the generated images, as shown in Figure 9.

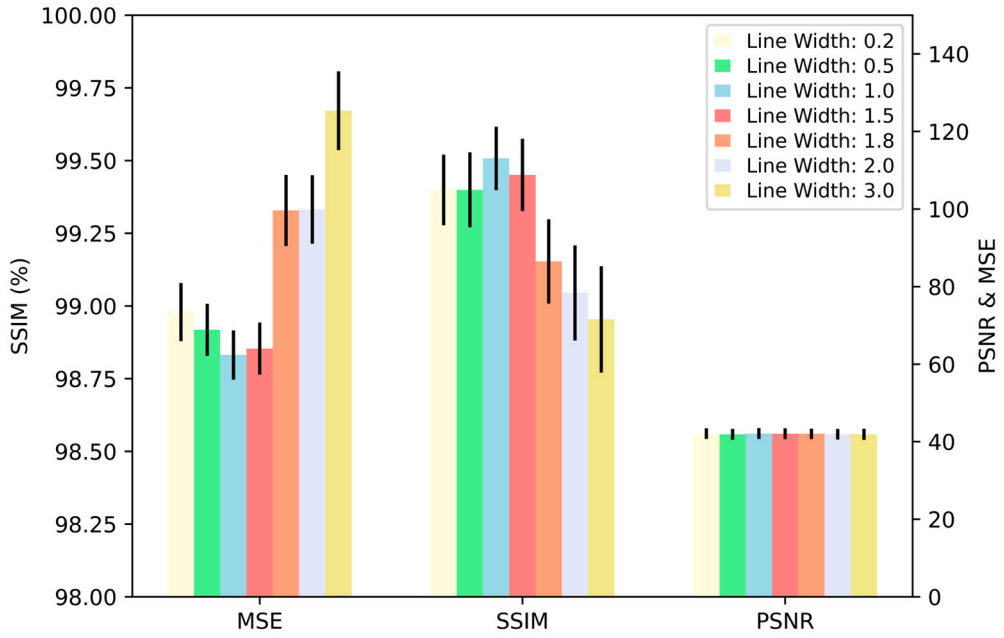


Figure 8. Evaluation metrics for the generative results of multi-width features.

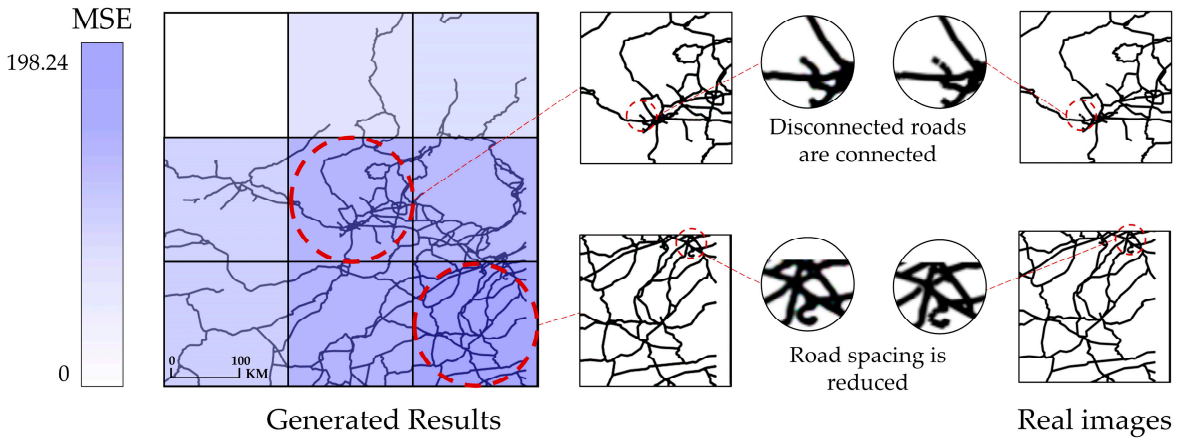


Figure 9. Issues in the generative results when the line width is thicker.

4.2.2. Generalization of Multi-Type Linear Features

In this section, the impact of different line types on generative models is explored through experiments using three line features: dashed lines, spaced lines, and framed lines. These three lines have various structural and color complexities, but all enhance visual perception in specific ways, strengthening the visualization of related lines. The particular information of the three linear features is shown in Table 4.

Table 4. The structural and color complexity of different line types.

Type	Example	Structural Complexity	Color Complexity
Dashed lines		Medium	Low
Framed lines		Medium	Medium
Striped lines		High	High

The generated results were quantitatively evaluated based on the metrics proposed in Section 3, and the results show that all three have similar global quality, with little difference in PSNR values. However, the SSIM index of the dashed line outperforms those

of the framed line and the striped line, while the framed line has a higher pixel-level error, as indicated by its higher MSE value, as shown in Figure 10.

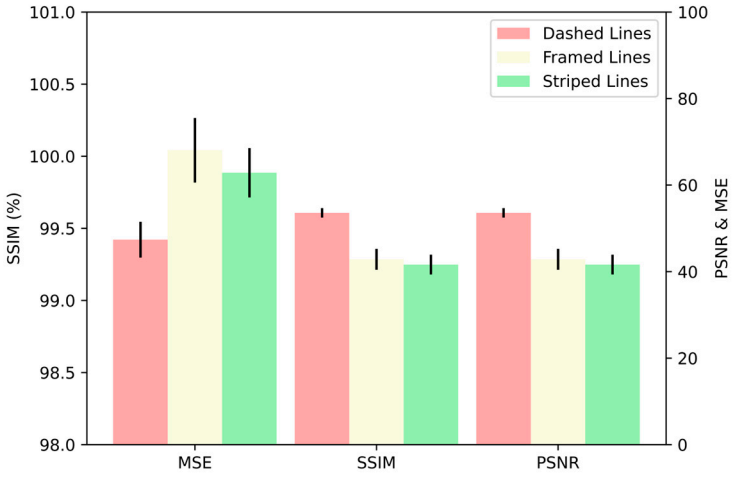


Figure 10. Evaluation metrics for the generative results of multi-type features.

1. Dashed lines

The generative results of linear features expressed with dashed symbols show good overall structure, maintaining the expected spatial structure and distribution characteristics of the linear features. This indicates that the model performs well in preserving the alignment and connectivity of the roads, as shown in Figure 11. This is also reflected in the evaluation metrics, where the dashed lines show superior results compared to the other two types of linear features.

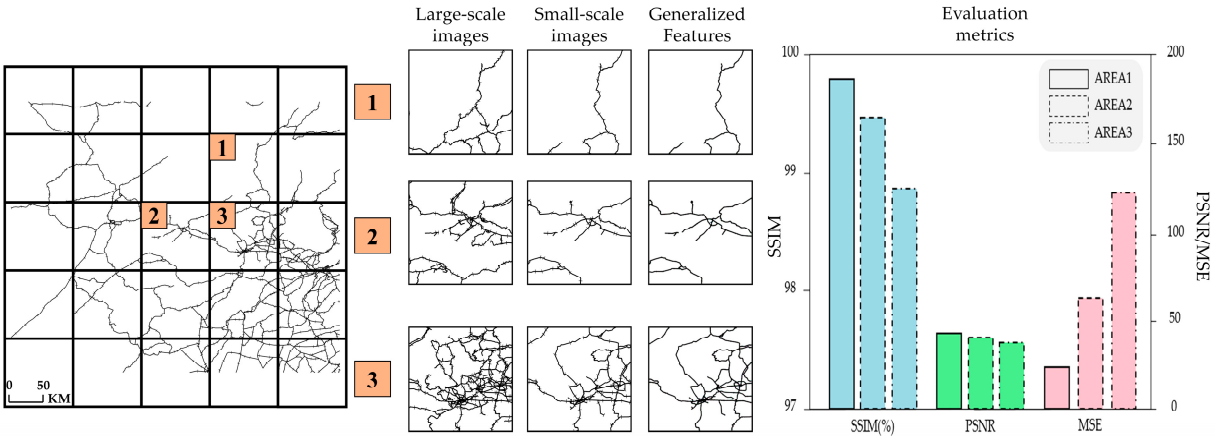


Figure 11. Example of generative results for dashed lines.

However, there is a significant issue with the generative results of the dashed symbols: the model's linear features need to be expressed using the specified symbolization method. The final images transform the linear features into ordinary black lines, as shown in Figure 12. Since the model is more accustomed to processing continuous features, it encounters difficulties with the intermittent structure of the dashed lines. Additionally, the vertical segments perpendicular to the main line are concise, negatively impacting the model's ability to capture these features.

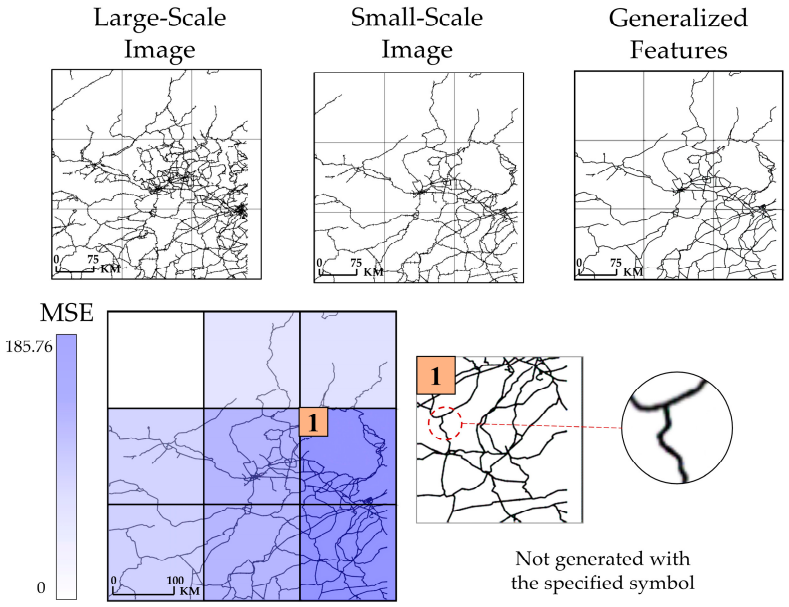


Figure 12. Disadvantages of the generative results for dashed lines.

2. Framed lines

Compared to simple black linear features, the model can generally recognize and retain the shape characteristics of framed lines. The design of framed lines enhances the visual resolution of the linear features, making it easier for the model to capture and reproduce this symbolic structure during the generation process, resulting in overall generative effects similar to the reference images, as shown in Figure 13.

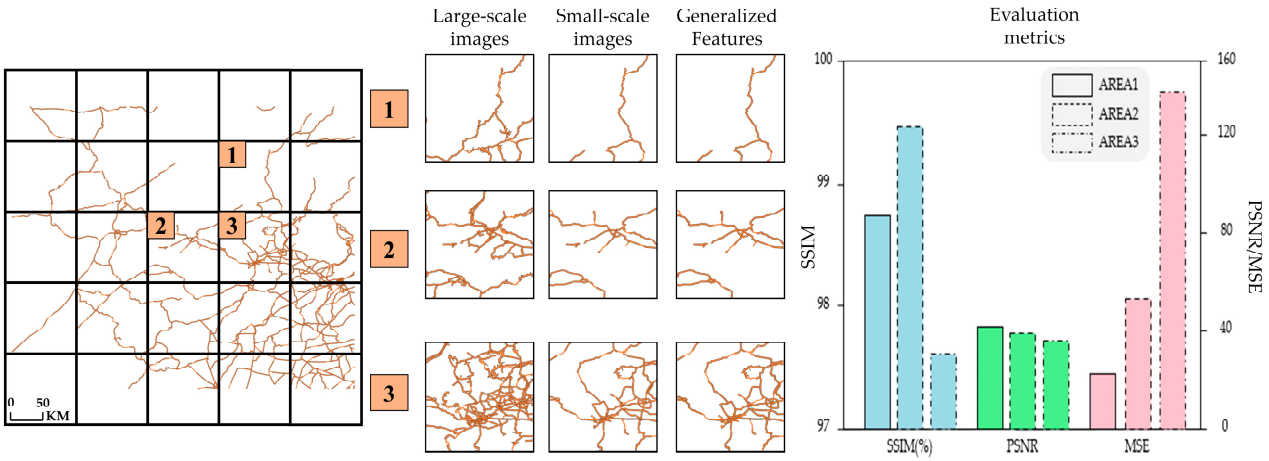


Figure 13. Example of generative results for framed lines.

However, the model exhibits some errors in the details of the generated results for certain connecting roads. These connecting roads do not have black borders. They are represented only by orange lines, making their symbolic features less distinct than the main roads and thus, making it more difficult for the model to recognize and retain. Consequently, some generated connecting roads appear lighter in color and narrower in width. Additionally, when multiple roads intersect, the model's generated results become blurred, especially at complex intersections, where the model needs help to accurately capture and reproduce the details of each road, resulting in discontinuities or overlaps at the intersections. These issues indicate that, although framed lines enhance visual perception effectively, the model still requires further optimization to improve the accuracy

and consistency of generating detailed connecting roads and intersections, as shown in Figure 14.

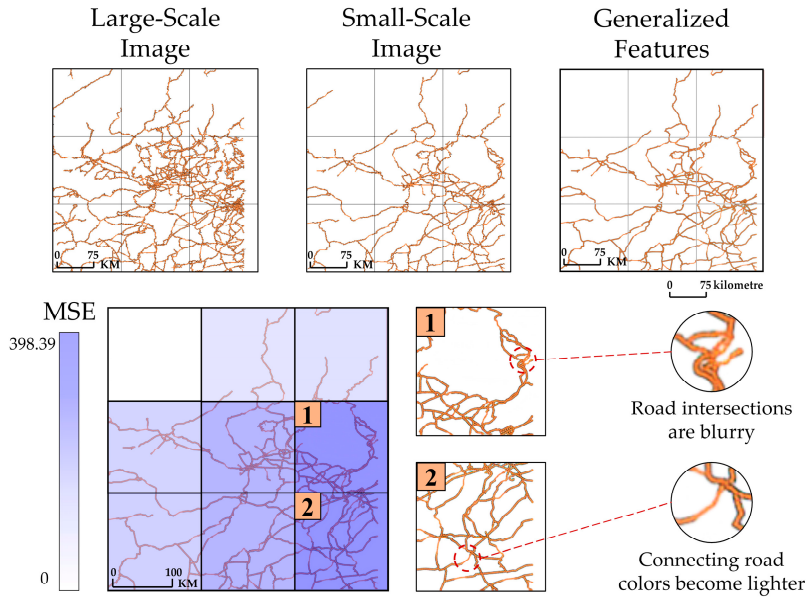


Figure 14. Disadvantages of the generative results for framed lines.

3. Striped lines

When generating maps using black-and-white spaced lines (with black borders around the white parts), the results indicate that the model can recognize and process these visually distinctive linear features, maintaining the basic structure and geometric characteristics of the lines in the generated maps, while correctly deleting some unimportant roads, as shown in Figure 15.

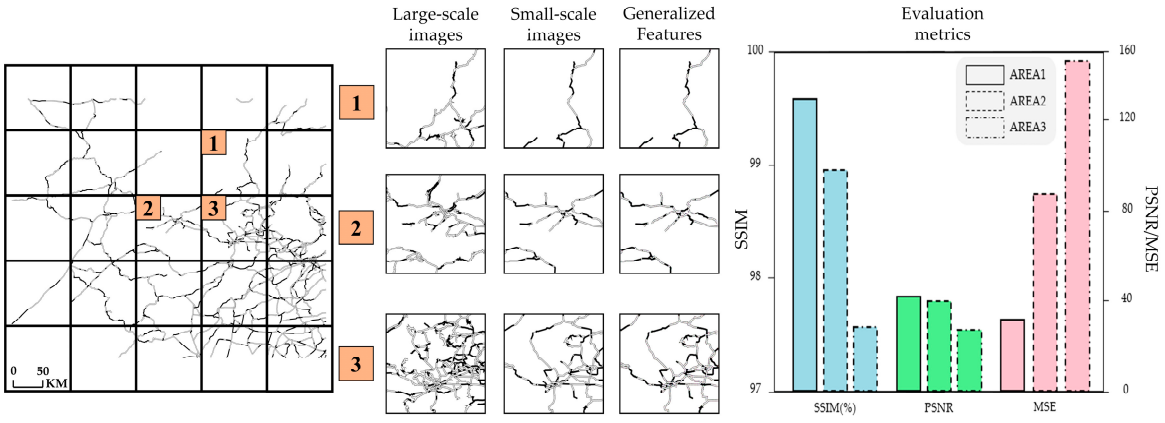


Figure 15. Example of generative results for striped lines.

However, there is a significant amount of blurring at the edges of the generated images and black bleeding at the junctions of adjacent straight lines. This phenomenon may be related to the limitations of the generative model in handling high-contrast and complex boundaries, especially at the junctions between the white parts and black borders of the striped lines. When the model attempts to transition these high-contrast areas smoothly, it negatively affects the generation results, as shown in Figure 16.

The experiments discussed above illustrate the impact of line type on generative models. The model's generation capability and accuracy are significantly limited when handling lines with sparse visual features. Therefore, in designing and optimizing linear

feature symbolization configurations, avoiding overly simplified features is crucial for improving the model's performance in complex line generation tasks.

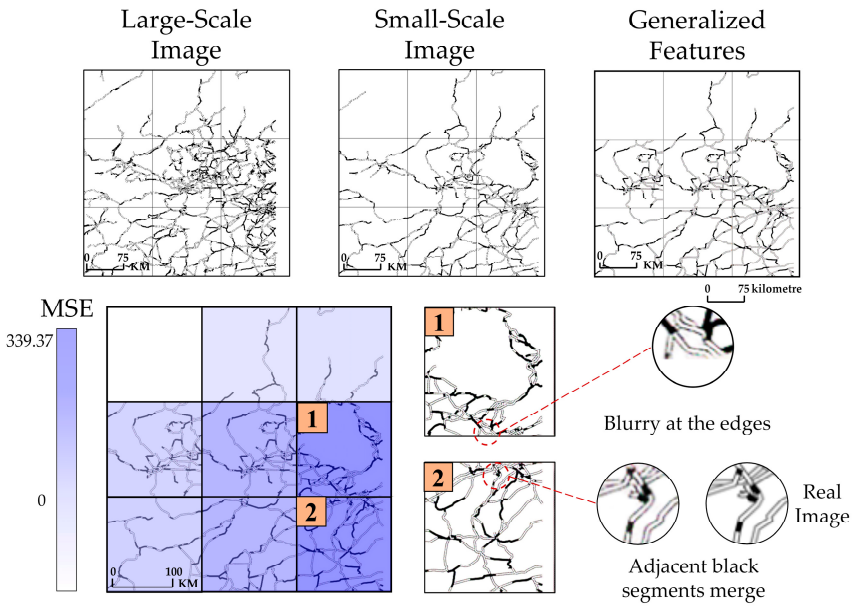


Figure 16. Disadvantages of the generative results for striped lines.

4.2.3. Generalization of Multi-Color Linear Features

This section experimentally investigates the impact of different color configurations on the generative results of linear features. The PSNR values of the generated results from the two color schemes do not differ significantly, and both are greater than 40, indicating high overall generation quality. Regarding SSIM values, grayscale color schemes are higher than high-contrast color schemes. This may be because the grayscale color scheme reduces color interference, making the model more stable in capturing image structures and details. The MSE value for high-contrast color schemes is much higher than that for grayscale. Due to the richness of colors and high contrast in high-contrast color schemes, the generative model is more prone to deviations and errors when handling slight color differences. These color differences are amplified in the MSE calculation, which is very sensitive to pixel changes, indicating that the complexity of high-contrast color schemes increases the model's requirement for color accuracy during generation, thus, leading to higher MSE values, as shown in Figure 17.

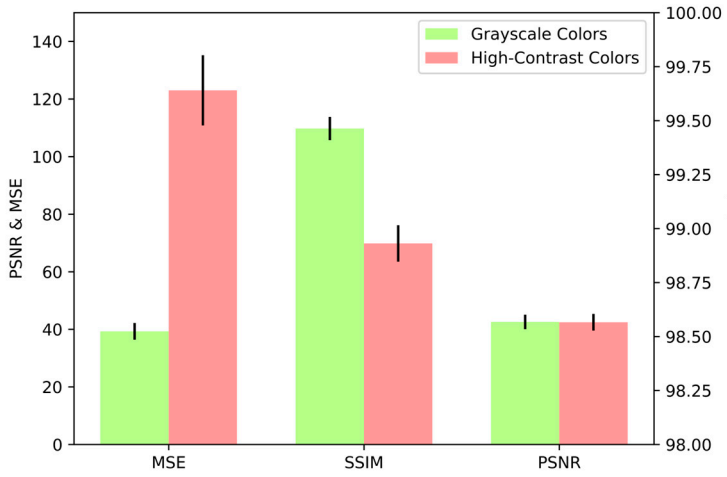


Figure 17. Evaluation metrics for the generative results of multi-color features.

Visually, using multiple colors to represent various linear features makes the image more vivid, enhancing the model's ability to distinguish and maintain features. Consequently, the generated results generally maintain the structure and spatial distribution of real-world features. However, the different color configurations significantly affect the model's performance. When using a grayscale color configuration, it is prone to color fading or disconnection issues. For example, the generated images show color deviations where dark gray roads become lighter and light gray roads almost disappear. Because the background is white, the model struggles to distinguish similar light gray tones, leading to a blend of colors visually or insufficient resolution to maintain the continuity of linear features. This indicates that the model needs help to accurately capture and reproduce the subtle differences between different gray values under grayscale color conditions, leading to detailed deficiencies and local structure in the generated results. At the same time, some continuously curved roads tend to become straight. This may be due to the lack of distinct color contrast in the grayscale scheme, making the model lack sufficient visual cues when capturing and reconstructing complex curve features. This results in an inaccurate processing of road curvature, with the model tending to generate more superficial straight structures to reduce complexity and processing difficulty, as shown in Figure 18.

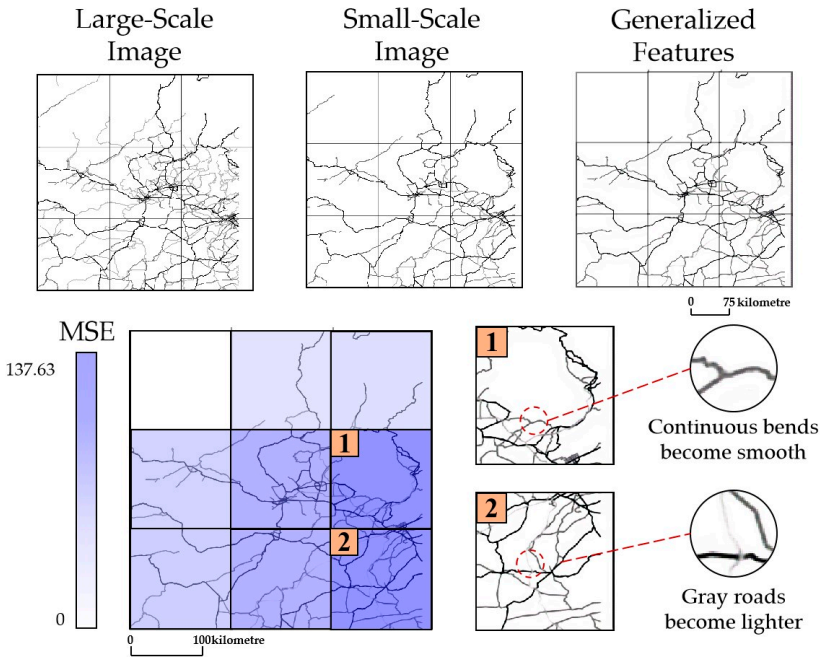


Figure 18. Examples and disadvantages of generative results for grayscale features.

When the line features use high-contrast colors, the model can more effectively distinguish which lines need to be deleted or retained based on color, thus, providing better overall generative results. However, there are significant differences between the generated features and the reference image at the pixel level. The "MSE" map shows areas with higher error, indicating substantial discrepancies between the generated results and the small-scale reference image used as the reference data. For example, in region "1", a green road in the reference image is incorrectly displayed as black in the generated results, indicating errors in road type recognition. This discrepancy may be due to the high saturation and contrast in the original color scheme, making it difficult for the model to accurately reproduce colors. Additionally, overlapping roads in the large-scale image are disrupted in the generated results. In region "2", shorter road segments overlapping with longer roads are often omitted, affecting spatial relationships and road connectivity. This may be because the model prioritizes longer road features under high-contrast conditions, leading to the omission of shorter, overlapping roads. Therefore, while the model performs well in generating major road structures, it struggles to maintain accurate connectivity and type

recognition in complex areas, especially under high-contrast color conditions, as shown in Figure 19.

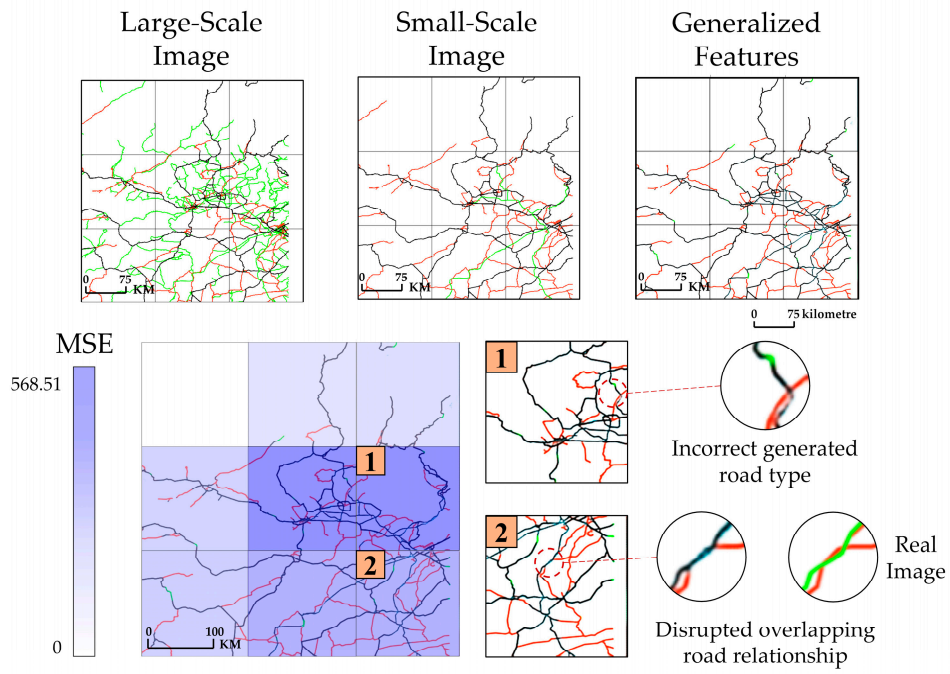


Figure 19. Examples and disadvantages of generative results for high-contrast features.

4.3. Applicability of Different Symbolized Linear Features

In Section 4.2, detailed comparisons of line width, line type, and color configurations were conducted to determine the advantages and disadvantages of each configuration in the generative results. This section selects four optimal symbol schemes to evaluate the applicability of these optimal configurations for comparative analysis of their applicability: line width of 1, framed lines, dashed lines, and high-contrast colors.

The dataset with a line width of 1 shows excellent detail retention and overall structural consistency during generation. Due to simple black lines, this configuration avoids the detail interference and generative errors of complex symbol structures, allowing the model to exhibit stability and accuracy when processing complex linear features. This configuration is particularly suitable for expressing detailed roads in urban road networks. The appropriate line width and simple linear feature symbol structure balance detail and overall visual effect, ensuring the clarity and consistency of roads in the generated images, as shown in Figure 20.

Framed lines and dashed lines have advantages in emphasizing significant roads. The presence of a frame enhances visual perception, allowing the model to identify and reproduce the features of substantial roads more clearly. Dashed symbols, with their unique visual intermittency and regularity, help the model more effectively identify and handle different linear features in complex backgrounds. These configurations are suitable for map tasks highlighting major traffic routes, ensuring that these essential features are accurately reproduced in the generative results. However, the model needs further optimization in high-density areas to address edge-blurring issues, as shown in Figure 21.

The dataset with high-contrast colors shows good generative results and is suitable for map tasks requiring solid visual contrast. Its advantage lies in highlighting different road categories, making it ideal for generalization tasks involving multiple types of linear features. The rich color information allows the model to effectively distinguish and handle complex color combinations, making it suitable for maps with abundant geographical features and strong color contrasts. However, high-contrast colors also lead to higher pixel-level errors, requiring attention for precise color matching and transitions during generation, as shown in Figure 22.

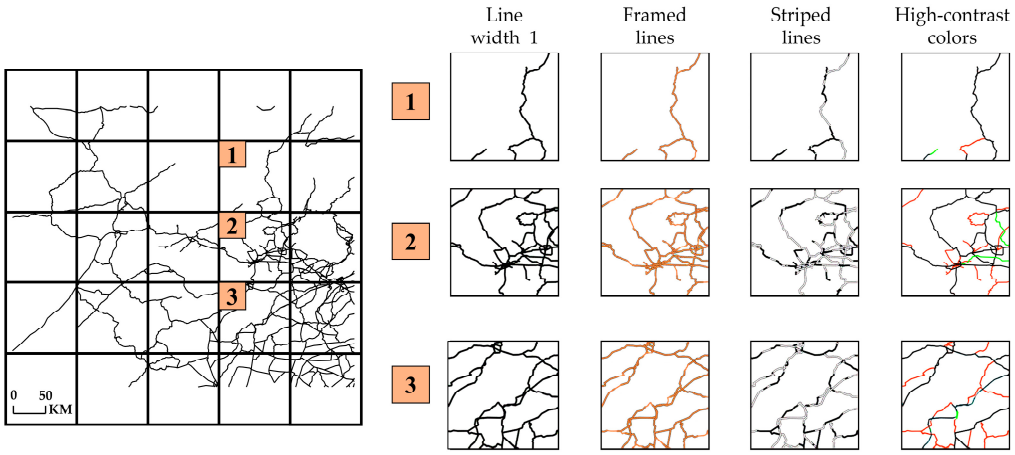


Figure 20. Consistency of generative results for linear features at appropriate widths.

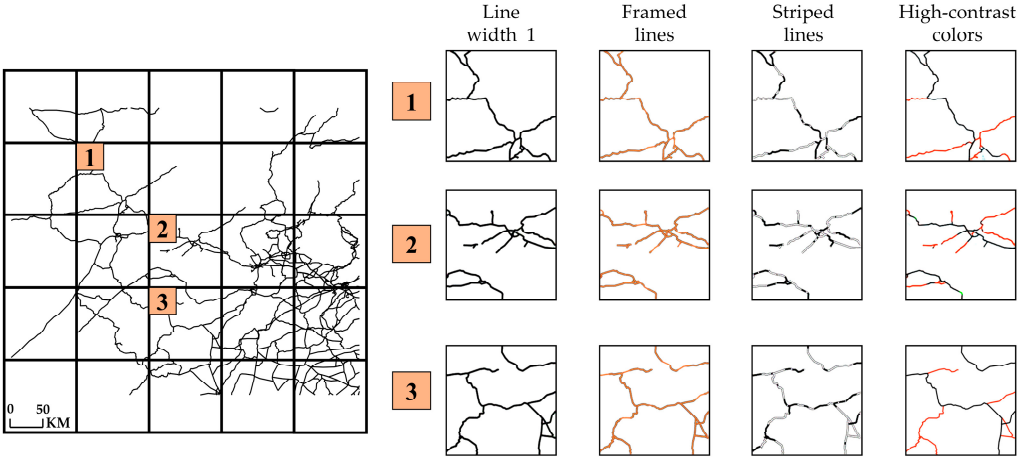


Figure 21. The highlighting effect of framed lines and dashed lines on features.

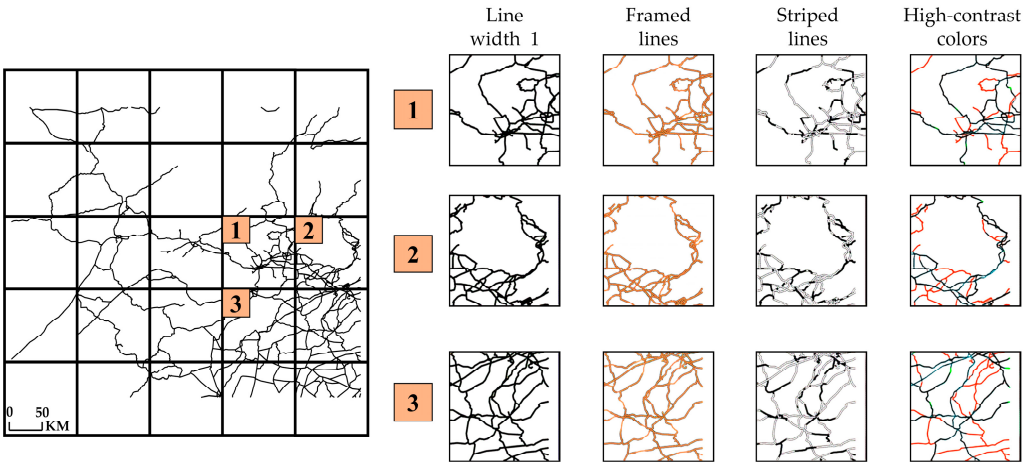


Figure 22. The distinguishing effect of high-contrast colors on multiple types of features.

In summary, these four optimal symbol schemes have advantages in different applications. A line width of 1 suits tasks balancing detail and overall effect; framed lines and dashed symbols are ideal for emphasizing major roads and boundaries. High-contrast colors are appropriate for tasks needing a clear distinction between multiple types of linear features. Future research should further optimize the generative models for these appli-

cations to improve the quality and consistency of the generative results under different symbol configurations.

4.4. Generalization of Mixed Symbolized Linear Features

Linear features in real maps typically include a combination of various widths, types, and colors to convey complex geographical information. Mixed symbolization configurations can simulate the complexity of actual maps, helping to evaluate the model's ability to handle multiple symbol parameters simultaneously and its robustness when faced with complex and diverse inputs. This includes the accuracy and consistency of generating features with various line widths, the detail retention ability when handling complex line structures, and the feature extraction and generation capabilities with mixed colors. Based on the above experimental results, this section uses a dataset constructed with a line width of 1 and high-contrast multi-type line features to conduct experiments, aiming to explore the advantages and limitations of the model when dealing with complex symbolization in commonly used maps, as shown in Figure 23.

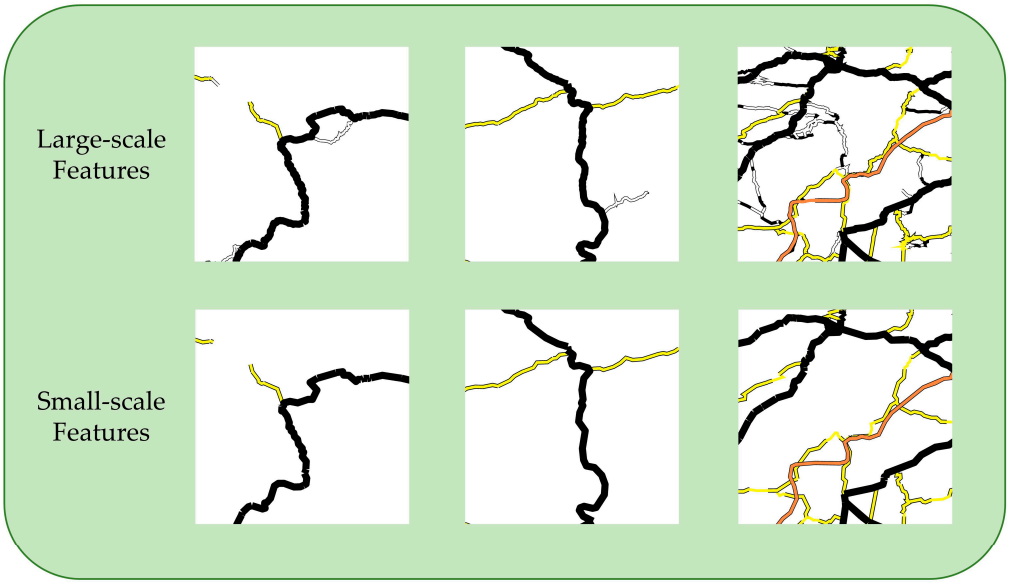


Figure 23. Example of mixed symbolized line feature dataset.

The results show that, in the experiments with mixed symbolization configurations, the model demonstrates strong recognition and generation capabilities when handling a combination of multiple symbol parameters. The generated images effectively retain the necessary lines based on color, and the overall spatial structure and distribution are generally correct, as shown in Figure 24.

Although the model's overall performance could be better, there are still some issues, including the following: (1) At intersections of linear features, the generated results appear blurry, especially with a tendency for the color at the edges of the generated features to lighten. This may be due to the model's lack of precision in generating details when handling complex intersection structures, leading to blurriness at the edges. (2) When there are roads with similar colors, the model struggles to distinguish between them, resulting in roads with comparable colors in the generated image being indistinguishable, affecting the accuracy of the results. When handling lines with similar colors, the model may need clarification due to the ambiguity of color information, reducing the fidelity of the generated image's details. (3) When generating framed lines, the border effects appear blurry. The model struggles to balance the clarity of the border and the integrity of the lines when handling complex edge structures, resulting in the borders in the generated results needing to be clearer, as shown in Figure 25.

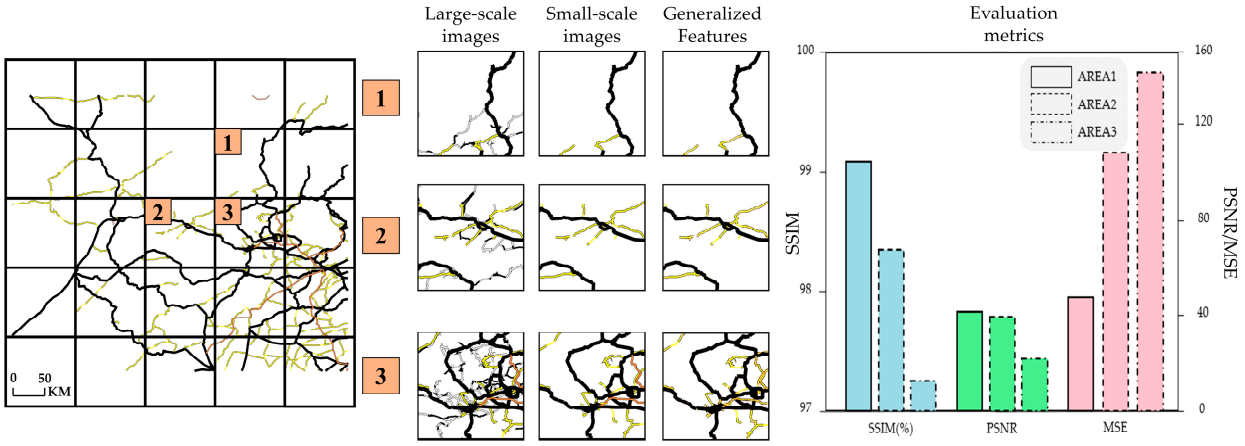


Figure 24. Evaluation metrics for the generative results of mixed symbolized features.

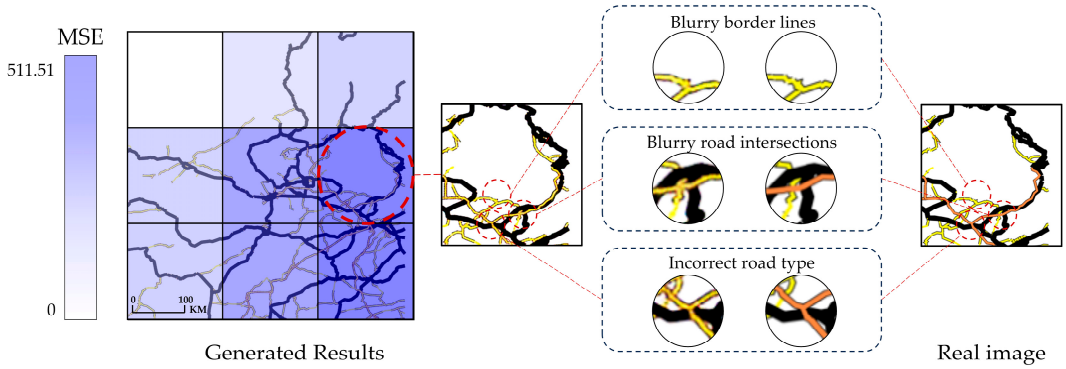


Figure 25. Issues in generated images with mixed symbolization configurations.

In summary, the experimental results of mixed symbolization configurations indicate that the generative model can adequately retain the overall spatial structure and color information when handling multiple symbol parameters. However, there are shortcomings in detail handling and edge clarity. Future research should address these issues by further optimizing the model's detail generation capabilities and color differentiation abilities to improve the accuracy and consistency of the generated results.

To further understand the capabilities of generative models in handling complex symbolization configurations, this section compares the performance of the CycleGAN model with the GCGAN (Geometrically Consistent Generative Adversarial Network) in generalizing mixed symbolized line features. The analysis focuses on how each model performs when faced with diverse symbolization parameters, such as varying line widths, types, and colors and by examining the similarities and differences in their generative outputs, aiming to identify the strengths and limitations of each model in maintaining spatial consistency, feature detail, and overall image quality. The results are shown in Figure 26.

The results indicate that both models effectively retain essential line features and maintain the overall spatial structure and distribution of the generated images. However, differences in the quality of the outputs are evident. The CycleGAN model performs better in preserving line clarity and accurately simplifying line types. It effectively removes roads at the same level, though minor errors in type recognition and edge blurring occur in dense areas. In contrast, the GCGAN model shows more significant limitations: road removal at the same level is less precise, resulting in incomplete elimination and blurred edges, especially at intersections. Additionally, GCGAN struggles to correctly generate border lines, sometimes adding borders where they should not appear or producing faint lines in complex segments, as shown in Figure 27.

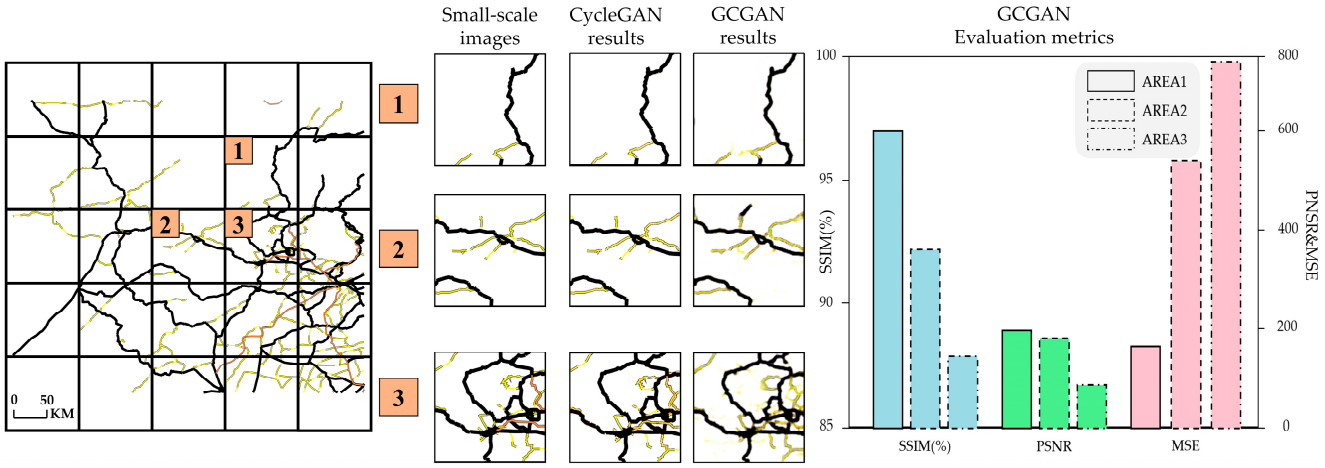


Figure 26. Evaluation metrics for the GCGAN results of mixed symbolized features.

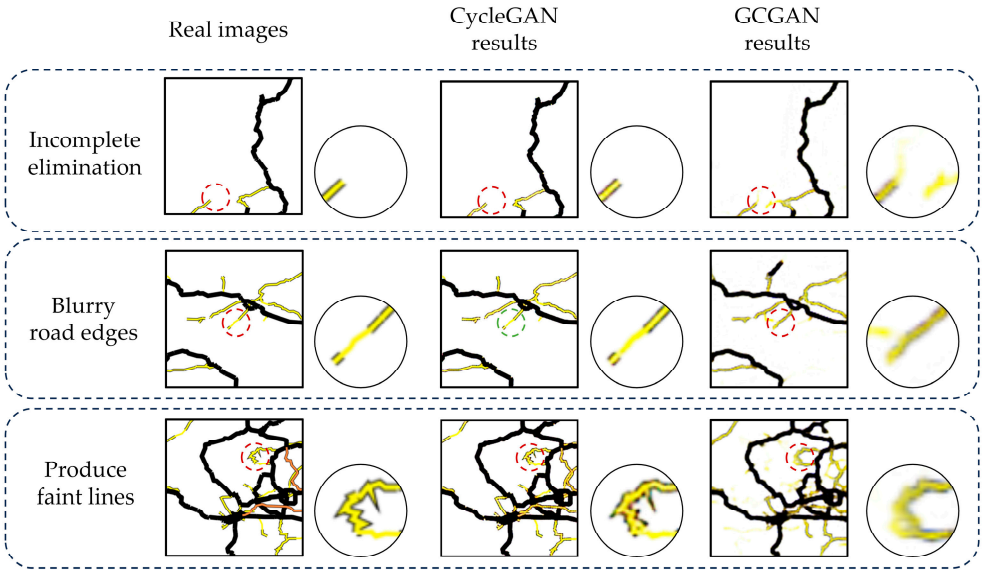


Figure 27. Comparison of results between CycleGAN and GCGAN.

GCGAN incorporates geometric consistency constraints to improve the quality of generated geometric shapes. However, these constraints can make it difficult for the model to balance geometric consistency and the expression of detailed image features when handling complex geometric structures (such as intersections and intricate road connections). This can ultimately lead to detail loss and blurred edges, making the overall quality of handling various symbolized line features slightly inferior to that of the CycleGAN model.

5. Conclusions

This study comprehensively investigates the impact of various symbolization configurations on the performance of generative models in map generalization, explicitly focusing on linear features. Utilizing the CycleGAN model, this research explores how different configurations of line width, line type, and color affect the generative results.

The experiments demonstrate that line width significantly influences the generative outcomes, with moderate widths (0.5 to 1) providing better detail preservation and structural consistency. More comprehensive lines, while more straightforward for the model to recognize and generate, tend to cause issues with detail retention and spatial continuity, leading to errors in high-density areas. Line types also play a crucial role, with framed and striped lines showing better overall performance. The presence of a frame enhances visual perception, aiding the model in accurately capturing and generating major road features.

Striped lines, with their distinct visual markers, allow the model to maintain the basic structure and geometric features of the linear features while eliminating less significant roads. Color configurations further highlight the model's ability to distinguish and process complex visual information. High-contrast color schemes enhance the model's ability to differentiate between various types of roads, making it particularly suitable for tasks requiring the integration of multiple line features. However, the complexity introduced by these color schemes can lead to higher pixel-level errors.

In the experiments with mixed symbolization configurations, the model demonstrated strong recognition and generation capabilities, effectively preserving the necessary features based on color and maintaining the overall spatial structure and distribution. However, some challenges remain. The main issue is that the mixed configuration of multiple symbols results in a large amount of feature information, which is difficult for the model to process, especially when handling intersections of linear features and distinguishing between roads with similar colors. The comparison analysis between CycleGAN and GCGAN shows that although both models can maintain the overall spatial structure and distribution of line features over larger spatial extents, the GCGAN model causes more significant disruptions to road continuity in finer detail and exhibits limitations in handling complex intersections and generating clear boundaries. In this task, CycleGAN outperforms GCGAN.

Overall, this study highlights the critical role of symbolization parameters in optimizing the generative performance of models for map generalization. While this research primarily examines the effects of line width, type, and color using the CycleGAN model, it does not assess the impact of different image resolutions—a limitation that future studies should address to understand model performance under various scales. Although the chosen symbolization schemes significantly influenced the outcomes, challenges remain in handling complex configurations, such as high-contrast colors and intricate line types. Enhancing model robustness in these scenarios is essential for achieving consistent and accurate results. Future work should focus on refining generative models to better adapt to complex symbolization settings, contributing to more reliable and versatile automated map generalization techniques.

Author Contributions: Conceptualization, Ling Zhang; methodology, Heng Yu and Haoxuan Cheng; writing—original draft preparation, Heng Yu, Haoxuan Chen, and Ling Zhang; writing—review and editing, Ling Zhang; funding acquisition, Ling Zhang. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (No. 42271451).

Data Availability Statement: The data that support the findings of this study are available in Github at https://github.com/CTRL25/An-Assessment-of-Map-Style-Influence-in-Generalization-with-CycleGAN-Data.git.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Li, Z. *Algorithmic Foundation of Multi-Scale Spatial Representation*; CRC: Boca Raton, FL, USA, 2007.
2. Sester, M. Cartographic Generalization. *J. Spat. Inf. Sci.* **2020**, *21*, 5–11. [CrossRef]
3. Harrie, L.; Touya, G.; Oucheikh, R.; Ai, T.; Courtial, A.; Richter, K.F. Machine Learning in Cartography. *Cartogr. Geogr. Inf. Sci.* **2024**, *51*, 1–19. [CrossRef]
4. Touya, G.; Zhang, X.; Lokhat, I. Is Deep Learning the New Agent for Map Generalization? *Int. J. Cartogr.* **2019**, *5*, 142–157. [CrossRef]
5. Sahu, M.; Ohri, A. Vector map generation from aerial imagery using deep learning. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2019**, *4*, 157–162. [CrossRef]
6. Karsznia, I.; Adolf, A.; Leyk, S.; Weibel, R. Using Machine Learning and Data Enrichment in the Selection of Roads for Small-Scale Maps. *Cartogr. Geogr. Inf. Sci.* **2024**, *51*, 60–78. [CrossRef]
7. Lisiewicz, M.; Karsznia, I. Verifying and Exploring Settlement Selection Rules and Variables for Small-Scale Maps Using Decision Tree-Based Models. *Geogr. Anal.* **2021**, *53*, 560–587. [CrossRef]
8. Xie, L.; Qian, H.; He, H.; Liu, C.; Duan, P. A Habitation Selection Method by Using Case-Based Reasoning. *Acta Geod. Cartogr. Sin.* **2017**, *46*, 1910. [CrossRef]

9. Hu, H.; Qian, H.; He, H.; Wang, X.; Chen, J. Auto-Selection of Areal Habitation Based on Analytic Hierarchy Process. *Acta Geod. Cartogr. Sin.* **2016**, *45*, 740. [CrossRef]
10. Cheng, B.; Liu, Q.; Li, X.; Wang, Y. Building Simplification Using Backpropagation Neural Networks: A Combination of Cartographers' Expertise and Raster-Based Local Perception. *GIScience Remote Sens.* **2013**, *50*, 527–542. [CrossRef]
11. Sester, M.; Feng, Y.; Thiemann, F. Building Generalization Using Deep Learning. *ISPRS Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2018**, *XLII–4*, 565–572. [CrossRef]
12. Courtial, A.; Touya, G.; Zhang, X. Generative Adversarial Networks to Generalise Urban Areas in Topographic Maps. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. ISPRS Arch.* **2021**, *43*, 15–22. [CrossRef]
13. 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.* **2022**, *37*, 4158–4180. [CrossRef]
14. Courtial, A.; Touya, G.; Zhang, X. Deriving Map Images of Generalised Mountain Roads with Generative Adversarial Networks. *Int. J. Geogr. Inf. Sci.* **2023**, *37*, 499–528. [CrossRef]
15. Viore, M.; Kesidis, A.L.; Krassanakis, V. Deep Learning for Map Generalization: Experimenting with Coastline Data at Different Map Scales. *Abstr. Int. Cartogr. Assoc.* **2024**, *7*, 180. [CrossRef]
16. Touya, G.; Lokhat, I. Deep Learning for Enrichment of Vector Spatial Databases. *ACM Trans. Spat. Algorithms Syst. TSAS* **2020**, *6*, 21. [CrossRef]
17. Courtial, A.; Touya, G.; Zhang, X. Representing Vector Geographic Information As a Tensor for Deep Learning Based Map Generalisation. *AGILE GIScience Ser.* **2022**, *3*, 32. [CrossRef]
18. Yu, H.; Ai, T.; Yang, M.; Huang, L.; Yuan, J. A Recognition Method for Drainage Patterns Using a Graph Convolutional Network. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *107*, 102696. [CrossRef]
19. Yu, W.; Chen, Y. Data-driven Polyline Simplification Using a Stacked Autoencoder-based Deep Neural Network. *Trans. GIS* **2022**, *26*, 2302–2325. [CrossRef]
20. Xiao, T.; Ai, T.; Yu, H.; Yang, M.; Liu, P. A Point Selection Method in Map Generalization Using Graph Convolutional Network Model. *Cartogr. Geogr. Inf. Sci.* **2024**, *51*, 20–40. [CrossRef]
21. Chen, X.; Chen, S.; Xu, T.; Yin, B.; Peng, J.; Mei, X.; Li, H. SMAPGAN: Generative Adversarial Network-Based Semisupervised Styled Map Tile Generation Method. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 4388–4406. [CrossRef]
22. Kang, Y.; Gao, S.; Roth, R.E. Transferring Multiscale Map Styles Using Generative Adversarial Networks. *Int. J. Cartogr.* **2019**, *5*, 115–141. [CrossRef]
23. Zhu, J.Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In Proceedings of the IEEE International Conference on Computer Vision 2017, Venice, Italy, 22–29 October 2017; pp. 2242–2251. [CrossRef]
24. Zohourian, F.; Pauli, J. Coarse-to-Fine Semantic Road Segmentation Using Super-Pixel Data Model and Semi-Supervised Modified CycleGAN. *J. Image Graph.* **2022**, *10*, 132–144. [CrossRef]
25. Kang, Y.; Rao, J.; Wang, W.; Peng, B.; Gao, S.; Zhang, F. Towards Cartographic Knowledge Encoding with Deep Learning: A Case Study of Building Generalization. In Proceedings of the AutoCarto, Online, 20–22 May 2020; pp. 1–6.
26. Weibel, R.; Burgardt, D. On-the-Fly Generalization. In *Encyclopedia of GIS*; Springer: New York, NY, USA, 2008; pp. 339–344.
27. Scheider, S.; Richter, K.F. Pragmatic GeoAI: Geographic Information as Externalized Practice. *KI Kunstl. Intell.* **2023**, *37*, 17–31. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.