CLPSTNet: A Progressive Multi-Scale Convolutional Steganography Model Integrating Curriculum Learning

FengChun Liu[1], **Tong Zhang**[2], **Chunying Zhang**[3]

[1]Qianan College, North China University of Science and Technology, Tangshan, Hebei 063210, China
[2]School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing 100876, China
[3]College of Science, North China University of Science and Technology, Tangshan, Hebei 063210, China
`lnobliu@ncst.edu.cn, 2024010368@bupt.cn, hblg_zcy@126.com`

ABSTRACT

In recent years, a large number of works have introduced Convolutional Neural Networks (CNNs) into image steganography, which transform traditional steganography methods such as hand-crafted features and prior knowledge design into steganography methods that neural networks autonomically learn information embedding. However, due to the inherent complexity of digital images, issues of invisibility and security persist when using CNN models for information embedding. In this paper, we propose Curriculum Learning Progressive Steganophy Network (CLPSTNet). The network consists of multiple progressive multi-scale convolutional modules that integrate Inception structures and dilated convolutions. The module contains multiple branching pathways, starting from a smaller convolutional kernel and dilatation rate, extracting the basic, local feature information from the feature map, and gradually expanding to the convolution with a larger convolutional kernel and dilatation rate for perceiving the feature information of a larger receptive field, so as to realize the multi-scale feature extraction from shallow to deep, and from fine to coarse, allowing the shallow secret information features to be refined in different fusion stages. The experimental results show that the proposed CLPSTNet not only has high PSNR , SSIM metrics and decoding accuracy on three large public datasets, ALASKA2, VOC2012 and ImageNet, but also the steganographic images generated by CLPSTNet have low steganalysis scores.You can find our code at https://github.com/chaos-boops/CLPSTNet.

1 Introduction

With the digital development of Internet technology and information, people's lives can be convenient at the same time, the leakage of personal privacy, confidential data subject to malicious attacks and tampering and other problems are increasing, and the corresponding information security protection technology is developing rapidly. Information security protection technology mainly includes two kinds of information encryption and information hiding. Information encryption is a technology that utilizes a specific algorithm to transform the original plaintext information into unrecognizable ciphertext information, and then restores the ciphertext to plaintext through a decryption algorithm. Steganography is a technology that embeds secret information into a carrier in a covert way so as to protect the safe transmission of secret information. Steganography mainly researches how to embed secret information into other information carriers efficiently and securely, and conceal the existence of the information in the transmission process so as to guarantee the security of secret information transmission. Steganography embeds the secret information into the carrier through specific encoding algorithms, and then the receiver of the information realizes the extraction of the secret information through specific decoding algorithms.

Image steganography is the use of specific algorithms to embed secret information into digital images, and then use specific algorithms to extract the secret information from the digital images, generally used for the transmission of secret information, digital copyright authentication and other scenarios. For image steganography, the performance of an image steganography algorithm can be measured by embedding capacity, invisibility and security. With the

introduction of deep learning into the field of steganography, image steganography models based on various types of neural network models have emerged in large numbers, greatly expanding the steganographic capacity of image steganography algorithms, but such deep learning image steganography algorithms perform poorly in terms of invisibility and security.

As a result, researchers have focused on how to improve the invisibility and security of deep learning image steganography, such as the use of adversarial training to improve security [1], improve the model structure to improve the invisibility [2] and so on. Adversarial training in image steganography algorithms originates from generative adversarial networks, where steganography and steganalysis networks are used as opposites, and the two are trained against each other to improve the steganographic image's resistance to steganalysis. The earliest SteGAN based on encoding-decoding network was proposed by Hayes et al [3] for image steganography, which defines the tripartite adversarial of Alice, Bob and Eve, representing the process of image steganography - information extraction - steganalysis respectively, to improve the security of steganography algorithm.Zhang et al [4] proposed SteganoGAN which includes the three-partite adversarial of encoding-decoding-evaluating parties, which has become the mainstream deep learning-based deep learning adversarial training. and becomes the current mainstream deep learning-based adversarial image steganography modeling framework. Firstly, the embedded information is transformed into binary data with tensor size of and spliced with the image in depth, the encoding network encodes it into the natural image with size of, the information is reconstructed from it by the decoding network, and the evaluation network is used to evaluate the performance of the encoding network in order to generate a more realistic cryptographic image.

Research for image steganography model structure focuses on improving the network's ability to capture multi-scale features [5], mitigating the gradient vanishing problem [6], and new ways of embedding information [7]. For example, SteganoGAN [4] and FC-DenseNet [6], which choose dense connectivity module to mitigate the gradient vanishing problem; using fusion dilation convolution and dense connectivity module to extract multi-scale and multi-expansion rate image features [8], adaptively embedding more information in the redundant image region with rich texture; introducing Inception module [5] for fusing different sensory domain size feature maps to improve the network's ability to capture features at different scales. However, this type of model has the problems of color distortion and poor security in the generated steganographic images when embedding large amount of information.

The earliest curriculum learning method that appeared was a data-level sampling strategy, and with the gradual application of curriculum learning in various fields, the existing curriculum learning methods can be categorized into data-based, task-based, and network model-based curriculum learning according to the application object. Model-based curriculum learning algorithms make the network model obtain better performance by regularly transforming the network model during the training process. For example, gradually increasing the number of network layers [9], discarding neuron probabilities [10], etc. Inspired by curriculum learning algorithms, we propose an implicitly written network structure CLPSTNet (Curriculum Learning Progressive Steganography Network) that incorporates the idea of curriculum learning step-by-step, and design a progressive multi-scale convolution that incorporates the Inception structure and the dilated convolutional module. The shallow network starts from a convolution with a smaller convolution kernel and dilation rate, and gradually expands to a convolution with a larger convolution kernel and larger dilation rate, realizing feature extraction from shallow to deep and from fine to coarse. In summary, the main contributions of this paper include the following three parts:

1. A Progressive Multi-scale Convolution Block (PMCB), a multi-pathway convolution structure that incorporates Inception structure and dilation convolution, is designed to enhance the ability of the network to capture multi-scale features.

2. A progressive multi-scale image steganography framework containing densely connection module and PMCB module is designed, which enables shallow information features to be refined at different fusion stages.

3. Experiments on three large public datasets, ALASKA2, VOC2012, and ImageNet, show that the proposed steganography scheme has high steganographic quality metrics such as SSIM and PSNR, while the generated steganographic images have low steganalysis scores.

2 Related Work

In this section, we first present work on deep learning image steganography related to the study of model structure. In addition, we address curriculum learning algorithm concepts and related work.

2.1 Deep learning steganography

In 2014, GoodFellow et al [11] proposed Generative Adversarial Networks (GAN), in which the generative and discriminative networks play with each other to produce a fairly good output. Since then, researchers have used

Generative Adversarial Networks in the field of image steganography, and a large number of steganographic models based on various types of deep learning have emerged. The research on deep learning image steganography in terms of model structure improvement is mainly based on the introduction of multi-scale feature processing, the introduction of the attention mechanism, the introduction of the simulation of attack structure to improve the robustness and so on.

Research related to multi-scale feature processing mainly focuses on increasing the width of the network or introducing novel convolutions.Li et al [8] proposed HCISNet, which fuses dilation convolution and dense connectivity modules to achieve the fusion of multi-scale and multi-dilatancy image features for better access to structurally and visually redundant regions of the image to enhance the steganographic capacity.Zhang et al [5] and Wang et al [8] proposed to utilize the Inception module for fusing feature maps with different perceptual domain sizes to enhance the network's ability to capture features at different scales. Also by expanding the network width, the design of double convolution, including 1×1 convolution and 3×3 convolution, is added to the U-Net model in the study of Zeng et al [12], which are all designed to enhance the network's adaptability to features of different scales by enhancing the network's width.Zhang et al [13] propose cross-process comparative refinement of the CFCR and cross-process multiscale CPMS of the JAIS-Net for jointly adapting and refining images in the information hiding and recovery phases, where the differences between the current pairs of steganographic images at different scales are used to guide the next phase of the adapting and refining embedding process during the training process.

In order to enhance the invisibility of steganographic images, researchers have tried to introduce various types of attention mechanisms into the model structure improvement.Peng [14]et al. proposed multi-scale channel attention to generate channel attention for feature maps at different scales, and orthogonal fusion was used to integrate the channel information from different sensory fields to improve the decoding accuracy. In Yao et al.'s [15] study, multi-scale attention is generated by widening the network width, and branches with different convolutional kernel sizes are added to the base channel attention to obtain feature maps at different scales, which makes the model pay more attention to the features that are useful for improving the performance of information embedding. Swin Transformer has received much attention in the field of computer vision due to its excellent global modeling ability of multi-scale features from its self-attention module, and has likewise been applied in the field of image steganography.Kashif et al [16] proposed enhanced pixel privacy utilizing Swin transformer with multiple heads of attention in combination with CycleGAN (EPPGAN); Li et al [17] proposed to add Shuffle Linear layer to the base of Swin transformer module to enhance the inductive bias ability of self-attention module, and to enhance the extraction of local features by shuffling the channels of the feature map in order to ensure that the feature information flows across the channels in the subsequent convolutional layers. Ke et al [18] proposed that the original self-attention module only considers spatial information and ignores channel information, and proposed Channel Adaptive Transformer Block (CATB) to utilize the global information of each channel to adjust the difference between channels.

The earliest reversible neural network for image steganography task was proposed by Jing et al [7], which explicitly modeled image recovery as an inverse process of image hiding, and only needed to train the network once to obtain all the parameters of the hiding and recovery networks, which achieved state-of-the-art performances in image recovery and hiding invisibility. Subsequently, Xu et al [19] proposed flow-based reversible neural network for steganography task, which is easier to compute and adds structures such as content-aware noise projection to improve the robustness of steganographic images. In addition, researchers have improved the invisibility of steganography by combining reversible neural networks with Swin transformer [20], combined spatial channel attention with reversible neural networks to guide the embedding of secret information into more secure image regions [21], and utilized reversible neural networks to guide multi-image steganography [22].

2.2 Curriculum Learning

The basic idea of curriculum learning originates from curriculum education in human behavior, humans need to undergo a long period of training from birth to adulthood, and this training is highly organized, with different concepts introduced at different stages, corresponding to a gradual increase in the difficulty, which leads to a gradual mastery of the knowledge learned. The concept of curriculum learning was initially proposed by Bengio et al [23], where an easier subset of data is used for training in the early stage of model training, and the difficulty of the subset of data is gradually increased until the entire data set is utilized for training, claiming that this makes it easy for the model to find better local optima, while speeding up the training speed. With the gradual application of curriculum learning in various fields, many research results have emerged. Existing curriculum learning methods can be categorized into data-based, task-based, and network model-based curriculum learning based on the object they are applied to [24].

Data-based curriculum learning advocates training models starting with simple samples and progressing gradually to complex samples. For example, in medical image analysis tasks, e.g., starting training from heavy disease images (the more severe the image lesion i.e., the simpler it is) [25] and progressively transitioning to moderate and mild, or starting training from images with nodules [26] and unlabeled images containing high information content [27] are used to

balance out the problem of training bias in the medical image task because samples containing high information content have a higher probability of belonging to a minority class (rare cases). Task-based curriculum learning approaches tasks incrementally by focusing on the connections between tasks, where each subtask is a simplified version of the next, and each task uses previously learned knowledge of the task. For example, starting with simpler tasks [28], more relevant [29] task sets, etc., and gradually expanding to more difficult and less relevant tasks.

Model-based curriculum learning allows network models to achieve superior performance by regularly modifying them during the training process. For example, gradually increasing the number of network layers [9], controlling filters [30], discarding neuron probabilities [10], and increasing the capacity and strength of the discriminator [31, 32]. In the research for generative adversarial network models, Karras et al [9] used to start with a low resolution image, so that the model captures the contour information of the data from it, and gradually add new network layers dealing with higher resolution details during the subsequent training process, which are used to increase the detail information of the image; Sharma et al [33] proposed that by continuously enhancing the discriminator's discriminatory ability is used to find the generator the problem that the generator needs to progress under increasingly difficult curriculum tasks in order to deceive the discriminator and achieve high quality images.

3 CLPSTNet

In this section, we present the overall architecture of CLPSTNet, describing in detail the design of the progressive multiscale convolutional module, followed by the structure of the encoding and decoding networks, and the cryptographic analysis network. Finally, the definition of the loss function is given.

3.1 Overall framework

As shown in Figure 1, the overall framework of CLPSTNet includes encoding network, decoding network and steganalysis network. The encoding network accomplishes information hiding, the decoding network accomplishes information recovery, and the steganography analysis network is used to evaluate the performance of the encoding network. Encoding network Encoder takes original image and secret information as input.

Assuming that the inputs are the original image $X_{cover} \in R^{3 \times H \times W}$ and the secret information $Y_{secret} \in R^{D \times H \times W}$, the original image and the secret information are spliced on the channel as inputs to the encoding network $I \in R^{(3+D) \times H \times W}$:

$$I = Cat(X_{cover}, Y_{secret}) \tag{1}$$

After processing by the Encoder, the steganographic image $X_{container} \in R^{3 \times H \times W}$ is generated, and the process can be described as:

$$Encoder : (X_{cover}, Y_{secret}) \rightarrow X_{container} \tag{2}$$

The decoding network Decoder accepts the steganographic image as input and generates the recovered secret message $Y_{recovered} \in R^{D \times W \times H}$ from the steganographic image, the process can be described as:

$$Decoder : X_{container} \rightarrow Y_{recovered} \tag{3}$$

The main goal of the coding network is to create a steganographic image that keeps the visual appearance of the original image intact while preserving the secret information in the steganographic image from being discovered. The steganalysis network Critic takes the steganographic image and the original image as inputs and outputs a steganalysis score $s \in [0, 1]$. When the score is closer to 1, it indicates that the image is more likely to contain secret information:

$$Critic : (X_{cover}, X_{container}) \rightarrow s \tag{4}$$

3.2 PMCB Module

In the process of realizing the embedding and reconstruction of multi-channel secret information, the quality of the steganographic image and the reconstruction decoding rate decrease with the expansion of the embedding capacity, which is related to the inability of the encoding and decoding network to deal with feature information of different scales. Inspired by curriculum learning, it mimics the process of human learning, starting from smaller and simpler knowledge and gradually expanding to macroscopic and more complex knowledge. As shown in Figure 2.
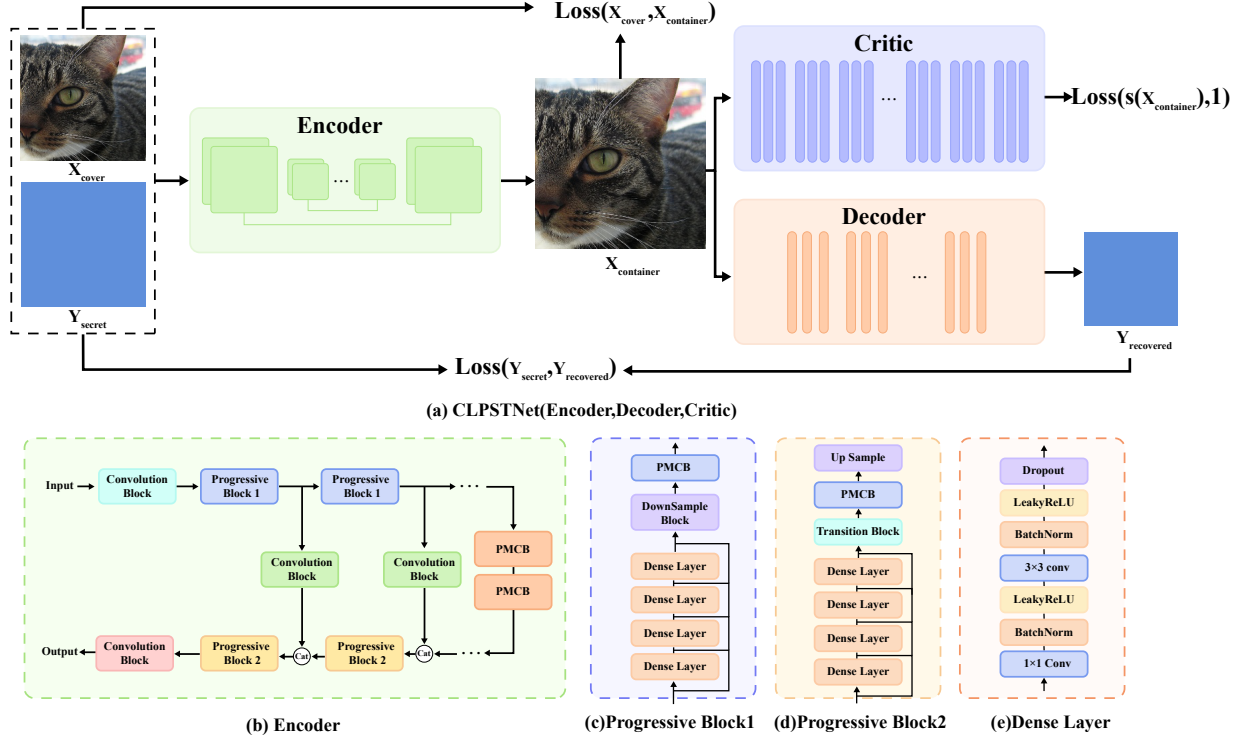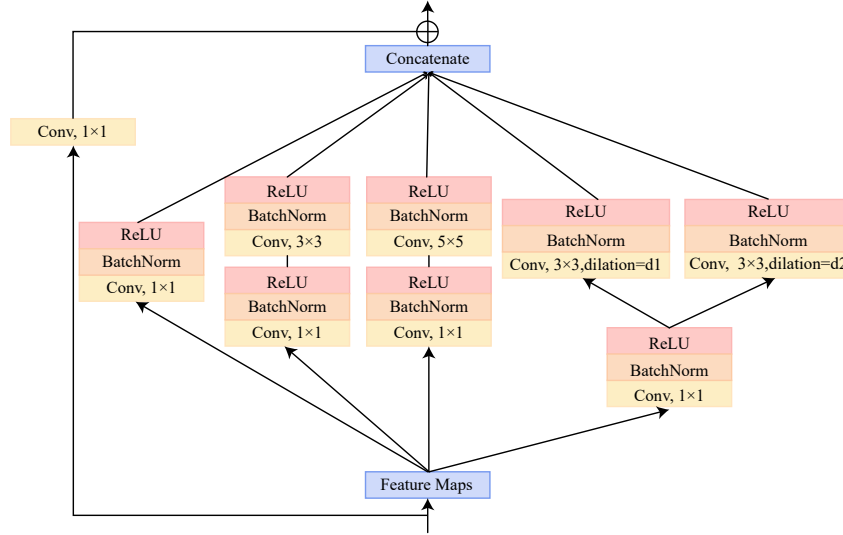
4

Figure 1: Framework of CLPSTNet

Figure 2: PMCB Module

Fusing the Inception structure and dilation convolution to construct a multi-pathway convolution structure, the Progressive Multi-scale Convolution Block (PMCB) is constructed, starting from a convolution containing a smaller convolution kernel and a dilation rate, to extract the most basic and localized feature information from the feature map. As the network deepens, it gradually expands to a convolution containing a larger convolution kernel and dilatation rate for sensing feature information of a larger range and larger sensory field, and extracts features from shallow to deep and from fine to coarse, with each layer observing different information features for each different convolution pathway to enhance the model's ability of extracting multi-scale information features.

A single PMCB module contains five branching pathways, including 1×1conv, 3×3conv, 5×5conv, 3×3 dilated convolution with dilation rate D1, and 3×3 dilated convolution with dilation rate D2. The 3×3 convolution kernel and 5×5 convolution kernel, and the two dilation convolution branching paths are preceded by 1×1conv added for dimensionality reduction to reduce the computational and parametric quantities due to the addition of large convolutions, and to increase the model's nonlinear expressive capability. The feature maps of the previous layer are input to the PMCB module, and are processed in parallel by the five branching paths, thus obtaining the multiscale coded information, and finally the results of the five branching paths are channel spliced in order to fuse the multiscale information. Compared with the original Inception module, the PMCB module adds the branching path of dilation convolution, which introduces dilation convolution processing to increase the receptive field and obtain denser feature information, which is especially important for image steganography to better obtain the structural and visual redundancy regions of the image to enhance the steganographic capacity.

Progressive Multiscale Convolution Module PMCB The progressive growth of scale in the coding network is reflected in the size of the convolution kernel, dilation rate. First, the shallow layer of the network starts after the PMCB1 module containing 1×1conv, 3×3conv, 5×5conv, dilation rate (Dilation=3) dilation convolution, and dilation rate (Dilation=6) dilation convolution, from which the network captures local details of a smaller scale and a smaller perceptual range, and the shallow network focuses more on the edge and detail features. It then expands to the PMCB2 module containing 1×1conv, 3×3conv, 5×5conv, dilation rate (Dilation=6) dilation convolution, and dilation rate (Dilation=12), which gradually captures feature information with larger sensory fields and more scales, and finally contains 1×1conv, 3×3conv, 5×5conv, dilation rate (Dilation=12) dilation convolution, and PMCB3 module with dilation rate (Dilation=18) to perceive feature information with larger sensory field and more scales, so that the secret information features and image information can be fused and refined at different fusion stages through the processing of PMCB modules with different scales, in order to enhance the feature processing capability of the network for the fusion and separation of secret information and images.

3.3 Encoder

As shown in Figure 1(b), the complete Encoder model mainly contains an initial convolution block, a downsampling stage, an upsampling stage and an output convolution block. First, the input of Encoder is processed by the initial convolution module, which contains four residual convolution modules, and the resolution of the output feature map is not changed. Each residual convolution module contains 3×3 conv, BatchNorm and LeakyReLU and a residual short-circuit join. Next, the feature maps here are processed by three Progressive Blocks1, each of which contains four layers of densely connected layers, downsampling layers, and PMCB modules, as shown in Figure 1(c).

The structure of each densely connected layer is shown in Figure 1(e), containing 1 × 1 conv, BatchNorm, LeakyReLU and Dropout, and the output of each layer is added to the subsequent network layer by channel splicing. The inputs of each layer consider the feature mapping of all previous layers. The dense connectivity module is used to alleviate the problem of gradient vanishing due to deepening of the network and to enhance feature propagation to realize the reuse of low-level feature information about edges, shapes, etc. in the shallow layer, so that the shallow secret information features and image features flow in different layers of the network. The resolution of the feature map output inside the dense connectivity module is kept constant.

The downsampling layer contains a 3×3 conv, BN, activation function LeakyReLU with a step size of 2. The resolution of the feature map output from this layer is halved. The internal construction in each PMCB module during the encoding process is not always the same, where the two branching paths of the dilation convolution use progressive growth, and the dilation rate parameter for this part uses [3,6,12,18,3,6,12,18], which is followed to ensure that the region of use under the sensory field is continuous. The output of each Progressive Block1 is fed individually to the Jump Connection Module for processing and is used to recover feature map resolution and supplemental details.

Subsequently, two PMCB modules are used as bottleneck layers for capturing multi-scale feature information at low resolution. Next, we use a U-Net structure similar to downsampling for feature reconstruction, and the decoding stage also contains three Progressive Blocks2, as shown in Figure 1(d). Each Progressive Block2 contains four layers of dense connectivity layer, PMCB module, Transition Block and upsampling layer. Each Progressive Block2 in the decoding stage accepts the processed feature maps from each Progressive Block1 in the encoding stage, and performs feature fusion by channel splicing after changing the number of channels via 1×1 conv, which is used to recover the feature map resolution and complementary details. The feature maps are then processed by the Dense Linking Module without changing the resolution of the feature maps. Then it is processed by the Transition Block module for adjusting the number of channels and reducing the computational effort, which contains BN, LeakyReLU, and 1×1conv. Subsequently it is processed by the PMCB module with different dilation rate settings, and the input is fed into the upsampling module, which contains an inverse convolutional module, BN, and LeakyReLU for recovering the feature map resolution. Finally after two residual convolution layers are used for output.

Table 1: Decoder model structure

Group/Layer	Progress	Output size
Input	/	$128 \times 128 \times 3$
Residual Block	Conv+BN+ReLU	$128 \times 128 \times 32$
PMCB	Dilation=(3,6)	$128 \times 128 \times 192$
PMCB	Dilation=(6,12)	$128 \times 128 \times 288$
PMCB	Dilation=(12,18)	$128 \times 128 \times 512$
Conv	/	$128 \times 128 \times D$

Table 2: Critic model structure

Group/Layer	Progress	Output size
Input	/	$128 \times 128 \times 3$
ConvBlock1	Conv+BN+ReLU+AvgPool	$64 \times 64 \times 8$
ConvBlock1	Conv+BN+Tanh+AvgPool	$32 \times 32 \times 16$
ConvBlock2	Conv+BN+ReLU+AvgPool	$16 \times 16 \times 32$
ConvBlock2	Conv+BN+ReLU+AvgPool	$8 \times 8 \times 64$
ConvBlock3	Conv+BN+ReLU	$4 \times 4 \times 128$
SPPBlock	/	3840×1
FC	/	128×1
FC	/	2×1

3.4 Decoder

Decoder accepts the output of Encoder as input and recovers the secret information from the steganographic image.Decoder consists of initial convolutional block, PMCB module and output convolutional block, the specific network structure and the shape of output tensor are shown in Table 1.

3.5 Critic

Critic accepts the output of Encoder and the original image together as input, and outputs a score representing the probability that the image contains secret information, the closer the score is to 1, i.e., the higher the probability that the image contains secret information, which means that the image is more likely to be detected by the steganalysis network.Critic chooses the XuNet-based steganalysis auxiliary network proposed by Zhang et al [5]. Critic uses XuNet based steganalysis assisted network proposed by Zhang et al [5]. The steganalysis network is used to evaluate the steganalysis resistance of the samples during the training process, and the encoder network-decoder network is formed with adversarial training to improve the steganalysis resistance of the encrypted image, the specific network structure and the shape of the output tensor of each layer are shown in Table 2.

3.6 Loss function

In order to improve the performance of the steganographic model, the objective function of the CLPSTNet model can be categorized into embedding loss, recovery loss and steganalysis loss, which can be defined as follows:

$$L_{total} = L_{encode} + a \times L_{decode} + b \times L_{stehsis} \tag{5}$$

where L_{total} is the overall loss function of the network, L_{encode} is the embedding loss for information hiding, L_{decode} is the recovery loss for information reconstruction, and $L_{stehsis}$ is the loss for steganalysis. α, β are the parameters used to balance the three components of the loss.

Embedding Loss: In order to ensure the visual consistency between the original image and the steganographic image, we consider several metrics for evaluating image similarity, including Mean Square Error (MSE), Structure Similarity Index Measure (SSIM), and Multi-Scale Structure Similarity Index (Multi-Scale Structure Similarity Index Measure, MSSSIM) [**?**]

$$L_{encode} = \lambda_1 \times MSE(x, y) + \lambda_2 \times (1 - SSIM(x, y)) + \lambda_3 \times (1 - MSSSIM(x, y)) \tag{6}$$

$\lambda_1 \lambda_2 \lambda_3$ are the parameters used to balance the three components of the embedding loss. MSE is chosen to measure the pixel-level difference between the original image and the steganographic image pair, and SSIM and MSSSIM are chosen to ensure the visual consistency of the image pairs on the human visual system.SSIM takes into account the sensitivity property of the human visual system to the structural information in terms of brightness, contrast and structural information of the images.SSIM takes the value of [0,1], and when the value of SSIM is closer to 1 The value of SSIM is [0,1], when SSIM value is closer to 1, it indicates that the two images are more similar in terms of brightness, contrast and structural information:

$$SSIM(x,y) = \frac{(2\mu_x\mu_y) + C_1}{\mu_x{}^2 + \mu_y{}^2 + C_1}{}^\alpha \cdot \frac{2\sigma_x\sigma_y + C_2}{\sigma_x{}^2 + \sigma_y{}^2 + C_2}{}^\beta \cdot \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}{}^\gamma \tag{7}$$

where μ_x, μ_y are the mean values of the original and loaded images respectively, σ_x, σ_y is the variance of the original and loaded images respectively, σ_{xy} is the covariance of the original and loaded images, C_1, C_2, C_3 is used to avoid parameters with denominators close to zero, $\alpha > 0, \beta > 0, \gamma > 0$ which are parameters used to adjust the importance of the three components.

MSSSIM is a variant of SSIM, a multiscale based SSIM metric that iteratively downsamples the image using a low-pass filter, with the original image having a scale of 1 and the highest scale of M. MSSSIM is obtained by calculating on different scales:

$$MSSSIM(x,y) = [l(x,y)]^{\alpha M} \times \prod_{j=1}^{M} [c_j(x,y)]^{\beta j} \times [s_j(x,y)]^{\gamma j} \tag{8}$$

Recovery loss: Binary cross entropy (BCE) is chosen for evaluating the accuracy of information recovery:

$$L_{decode} = BCE(Y_{secret}, Y_{recovered}) \tag{9}$$

4 Experiments

4.1 Experimental Platform and Datasets

The experiments in this paper were all conducted on a Linux operating system using the PyTorch 1.10.1 deep learning framework, with the system's GPU being the NVIDIA GeForce RTX 3090. The datasets used were the ALASKA2, Pascal VOC2012, and ImageNet, which are three publicly available large-scale datasets. The ALASKA2 is a public dataset from the ALASKA2 Image Steganalysis competition on the Kaggle platform. Within the ALASKA2 dataset, 10,000 original images from the "Cover" category were selected for the training set, 3,000 for the validation set, and 7,000 for the test set. ImageNet is a large-scale public computer vision dataset; 25,000 images were extracted from it, with 20,000 used for the training set and the remainder for testing. VOC2012 is a dataset used for object detection and semantic segmentation; 13,000 images were selected from it to form the training set, with the remaining 5,000 used for the test and validation sets. Due to computational power limitations, all original images from the datasets were processed through a Matlab program to be resized to 128×128 pixels.

4.2 Parameters

The experiments in this paper utilize the Adaptive Moment Estimation (Adam) algorithm provided by the PyTorch platform to optimize the encoding and decoding networks. The initial learning rate is set to 0.001, with momentum parameters (betas) configured as (0.9, 0.999). The Stochastic Gradient Descent (SGD) algorithm is selected to optimize the steganalysis adversarial network, with the initial learning rate (lr) set to 0.0001/3, and weight decay (weight decay) set to 1e-8, updating the steganalysis network parameters every 5 batches. The number of samples selected for each training session (Batch size) is 8, and the maximum number of iterations for the model (max-iter) is 120. The loss function comprises encoding loss, decoding loss, and steganalysis loss. The encoding loss incorporates the Structural Similarity Index (SSIM), Multi-Scale Structural Similarity (MS-SSIM), and Mean Square Error (MSE) as evaluation metrics, with corresponding proportional coefficients of 0.5:0.5:0.3. Both the decoding loss and the steganalysis loss employ binary cross-entropy. The ratio of the encoding loss to decoding loss to steganalysis loss in the loss function is 1:1:0.1. In the experiments, the steganographic capacity is D=1-6 bpp (that is, the hidden tensor in a 128×128 image is 128×128×D in size).

4.3 Experimental results

1) Test results of the model on multiple datasets. The CLPSTNet model is selected to be tested on the 1-6 bpp ALASKA2, VOC2012, and ImageNet datasets, and the experimental results are shown in Table 3. As can be seen from

Table 3: Performance of the CLPSTNet model in steganography and decoding accuracy

Dataset	D	SSIM	MSSSIM	PSNR	RMSE	Accuracy
ALASKA2	1	0.99932	0.99997	50.983	0.0028	0.98
	2	0.99889	0.99996	44.819	0.0059	0.87
	3	0.99518	0.99970	42.098	0.0092	0.60
	4	0.98896	0.99920	39.190	0.0116	0.66
	5	0.99184	0.99917	38.959	0.0319	0.67
	6	0.98991	0.99919	37.419	0.0162	0.64
VOC2012	1	0.99930	0.99994	50.215	0.0032	0.94
	2	0.99840	0.99993	46.085	0.0050	0.96
	3	0.99550	0.99946	38.554	0.0132	0.68
	4	0.99049	0.99943	39.399	0.0109	0.62
	5	0.98053	0.99730	34.065	0.0249	0.72
	6	0.96507	0.99732	33.874	0.0312	0.75
ImageNet	1	0.99933	0.99997	49.537	0.0034	0.95
	2	0.99862	0.99991	44.573	0.0062	0.83
	3	0.99819	0.99989	43.884	0.0067	0.83
	4	0.99486	0.99971	37.994	0.0147	0.65
	5	0.98880	0.99925	37.000	0.0207	0.67
	6	0.98810	0.99887	36.669	0.0178	0.66

Table 3, the CLPSTNet model performs well in all indicators of 1-6 bpp steganographic capacity on the two datasets, with the PSNR exceeding 40 under the 1-3 bpp indicator, and the SSIM indicator is close to 1, which indicates that the steganographic image generated by the CLPSTNet model is very similar to the original image.

2) PMCB module validation for coding and decoding effectiveness. In order to explore the effectiveness of PMCB module in encoding and decoding networks, the Conv model, which contains only the base convolutional module, is chosen as the baseline model, and the ProgressiveNet model, which contains only the PMCB module in the decoding network and only the base convolutional module in the encoding network, is chosen as a comparative model, to be compared with the CLPSTNet model, which contains the PMCB module in both the encoding and decoding networks. CLPSTNet model for comparison. The 1-6 bpp steganographic capacity on the ALASKA2 dataset was chosen for the experiments, and the experimental results are shown in Table 4.

As can be seen from Table 4, the CLPSTNet model outperforms the baseline model in 1-6 bpp steganographic capacity, and outperforms the baseline model in several metrics of steganographic image quality, such as SSIM, PSNR, and MSSSIM, while the decoding accuracy is the same or slightly lower than that of the baseline model. On the other hand, ProgressiveNet with PMCB selected for decoding network and basic convolutional structure selected for encoding network has lower performance than CLPSTNet in several metrics of steganographic image quality, while the decoding accuracy is slightly higher than that of CLPSTNet and the baseline model. It shows that the PMCB module is very effective in improving the decoding accuracy, while the CLPSTNet model, which adopts the PMCB module for both the encoding and decoding networks, is higher than PregressiveNet and the baseline model in several metrics of the steganographic quality assessment, indicating that the PMCB module is also effective in improving the quality of steganographic images.

3) PMCB module and Inception module comparison experiment. In order to further explore the effects of the basic Inception module and the PMCB module improved by the Inception module on the model performance, the steganographic network containing only the Inception module, the steganographic network containing the Inception module and the dense connectivity module, and the CLPSTNet containing the PMCB module and the dense connectivity module were selected for comparison. The 1-3 bpp steganographic capacity on the ALASKA2 dataset was chosen for the experiments, and the experimental results are shown in Table 5.

As can be seen from Table 5, the PMCB module improved by the Inception module outperforms the model containing only Incepiton or the model containing the Incetpion module and the Dense Connection module in several

9

Table 4: Impact of the PMCB Module on the Performance of the Encoding and Decoding

D	Model	SSIM	MSSSIM	PSNR	Accuracy
	Conv	0.99027	0.99806	34.260	0.74
1	ProgressiveNet	0.98063	0.99771	33.788	**0.99**
	CLPSTNet	**0.99932**	**0.99997**	**50.983**	0.98
	Conv	0.98694	0.99651	35.263	0.61
2	ProgressiveNet	0.98934	0.99847	34.324	**0.99**
	CLPSTNet	**0.99889**	**0.99996**	**44.819**	0.87
	Conv	0.98632	0.99714	34.183	0.57
3	ProgressiveNet	0.99489	0.99877	35.714	**0.87**
	CLPSTNet	**0.99518**	**0.99970**	**42.098**	0.60
	Conv	0.98452	0.99703	33.768	0.52
4	ProgressiveNet	**0.99267**	0.99715	33.334	**0.70**
	CLPSTNet	0.98996	**0.99920**	**39.190**	0.66
	Conv	0.97835	0.99329	31.010	0.52
5	ProgressiveNet	0.99136	0.99760	34.427	0.67
	CLPSTNet	**0.99184**	**0.99917**	**38.959**	0.67
	Conv	0.97987	0.99405	32.366	0.50
6	ProgressiveNet	0.98845	0.99733	34.384	0.64
	CLPSTNet	**0.98991**	**0.99919**	**37.419**	0.64

Table 5: Comparative experimental results of PMCB module and Inception module

D	Inception	PMCB	Dense Block	SSIM	MSSSIM	PSNR	RMSE	Accuracy
	√	×	×	0.98312	0.99625	28.286	0.0386	0.63
1	√	×	√	0.99850	0.99992	47.842	0.0040	0.79
	×	√	√	**0.99932**	**0.99997**	**50.983**	**0.0028**	**0.98**
	√	×	×	0.95990	0.99114	27.974	0.0400	0.64
2	√	×	√	0.99316	0.99962	40.577	0.0094	0.82
	×	√	√	**0.99889**	**0.99996**	**44.819**	**0.0059**	**0.87**
	√	×	×	0.92621	0.97698	23.670	0.0716	0.50
3	√	×	√	0.97997	0.99915	39.821	0.0110	0.50
	×	√	√	**0.99518**	**0.99970**	**42.098**	**0.0092**	**0.60**

Table 6: Influence of densely connected module and PMCB module on model performance

D	Dense Block	PMCB Block	SSIM	MSSSIM	PSNR	Accuracy
	×	×	0.99027	0.99806	34.260	0.74
1	√	×	0.99918	**0.99997**	46.299	0.95
	×	√	0.99916	0.99995	43.728	0.83
	√	√	**0.99932**	**0.99997**	**50.983**	**0.98**
	×	×	0.98694	0.99651	35.263	0.61
2	√	×	0.99839	0.99994	**47.565**	0.82
	×	√	0.99682	0.99972	39.228	0.81
	√	√	**0.99889**	**0.99996**	44.819	**0.87**
	×	×	0.98632	0.99714	34.183	0.57
3	√	×	0.99583	0.99979	**43.569**	0.69
	×	√	**0.99784**	**0.99985**	42.897	**0.83**
	√	√	0.99518	0.99970	42.098	0.60

Table 7: Influence of dilation rate parameters in PMCB module on model performance

Dilation	SSIM	MSSSIM	PSNR	RMSE	Accuracy
(3)	0.99941	0.99996	48.855	0.0036	0.94
(6)	0.99918	0.99996	48.991	0.0036	0.89
(3,6)	**0.99948**	**0.99998**	50.511	0.0030	0.93
CLPSTNet	0.99932	0.99997	**50.983**	**0.0028**	**0.98**

metrics of implicit writing evaluation, such as SSIM, PSNR, and Decoding Accuracy, indicating that the PMCB module is more effective in improving the model performance compared to the Inception module.

4) PMCB module and dense connectivity module validation. In order to investigate the effect of the internal structure of the proposed CLPSTNet model on the model performance, the dense connectivity module and the PMCB module, which are included in the coding network of the CLPSTNet model, are selected for ablation experiments. The test results of 1-3 bpp hidden writing capacity on the dataset ALASKA2 are shown in 6.

As can be seen from Table 6, the CLPSTNet models containing only the dense connectivity module, only the PMCB module, and both the PMCB module and the dense connectivity module outperform the baseline model in terms of image steganography quality and decoding accuracy, indicating the effectiveness of the dense connectivity module and the PMCB module. At 1 bpp steganographic capacity, the CLPSTNet model outperforms the rest of the compared models in all cases, indicating that the proposed CLPSTNet model structure has excellent steganographic performance and decoding accuracy.

5) Comparison of CLPSTNet progressive modules.The expansion rate of several PMCB modules in the CLPSTNet network is set to progressive growth.In order to explore the effectiveness of this progressive idea, the schemes that contain only the expansion rate of 3, 6, 3, and 6 parameters are selected to be compared, and the test results are shown in Table 7. As can be seen from Table 7, the scheme with the expansion rate parameter of the PMCB module set to progressive growth performs the best in all the indicators, including SSIM, PSNR and decoding accuracy, which are all better than the PMCB module scheme with only a single expansion rate set, indicating the effectiveness of the progressive PMCB module in CLPSTNet.

6) Comparison with other models. In order to more fully demonstrate the excellent performance of CLPSTNet, different structures of image steganography models are selected for experimental comparative analysis, including the classical image steganography network structures ResNet [34], SteganoGAN [4], and HCISNet [8], as well as the multiscale network models DenseASPP [35], FC-DenseNet [6], the as well as network structures that include attention mechanisms such as SENet [36], ECANe [37] and CBAM [38]. In addition, ProgressiveNet models that only decode networks containing PMCB modules were added for comparison, and the test results of these models and the CLPSTNet model for 1bpp hidden write capacity on the dataset ALASKA2 are shown in Table 8.

Table 8: Comparison experiments between CLPSTNet and other models

Model	SSIM	MSSSIM	PSNR	RMSE	Accuracy
Conv	0.98351	0.99771	33.788	0.020	**0.99**
ResNet	0.98993	0.99708	33.435	0.021	0.78
steganGAN	0.98970	0.99907	37.669	-	0.95
DenseASPP	0.99753	0.99933	36.008	-	0.95
SENet	0.99432	0.99881	37.131	0.014	0.96
ECANet	0.99438	0.99919	39.934	0.010	0.86
CBAM	0.99470	0.99869	37.878	0.012	0.92
FC-DenseNet	0.98179	0.99236	30.510	-	0.72
HCISNet	0.96532	0.99822	41.221	-	**0.99**
ProgressiveNet	0.98351	0.99771	33.788	0.020	**0.99**
CLPSTNet	**0.99932**	**0.99997**	**50.983**	**0.0028**	0.98

As can be seen from Table 8, the steganographic quality metrics MSSSIM and PSNR of CLPSTNet are higher than those of the other comparison models, which proves the excellent performance of the CLPSTNet model in image steganography quality. Under the same experimental conditions, the decoding accuracy of the ProgressiveNet model with only the decoding network containing the PMCB module has the best performance among the many comparison models, indicating the effectiveness of the PMCB module in improving the decoding accuracy.

7) Visual test of steganographic images. In order to further validate the quality of steganographic images for the CLPSTNet model, a part of images from ImageNet, ALASKA2, and VOC2012 datasets were selected for testing. This part of the images did not participate in the model training and verified the carrier-confidential images under 1-6 bpp steganographic capacity, and the test results are shown in Figure 3. As can be seen in Figure 11, the steganographic images generated by the CLPSTNet model with 1-6 bpp steganographic capacity are more similar to the original images in terms of color and brightness, which indicates that the model has superior image steganographic quality.
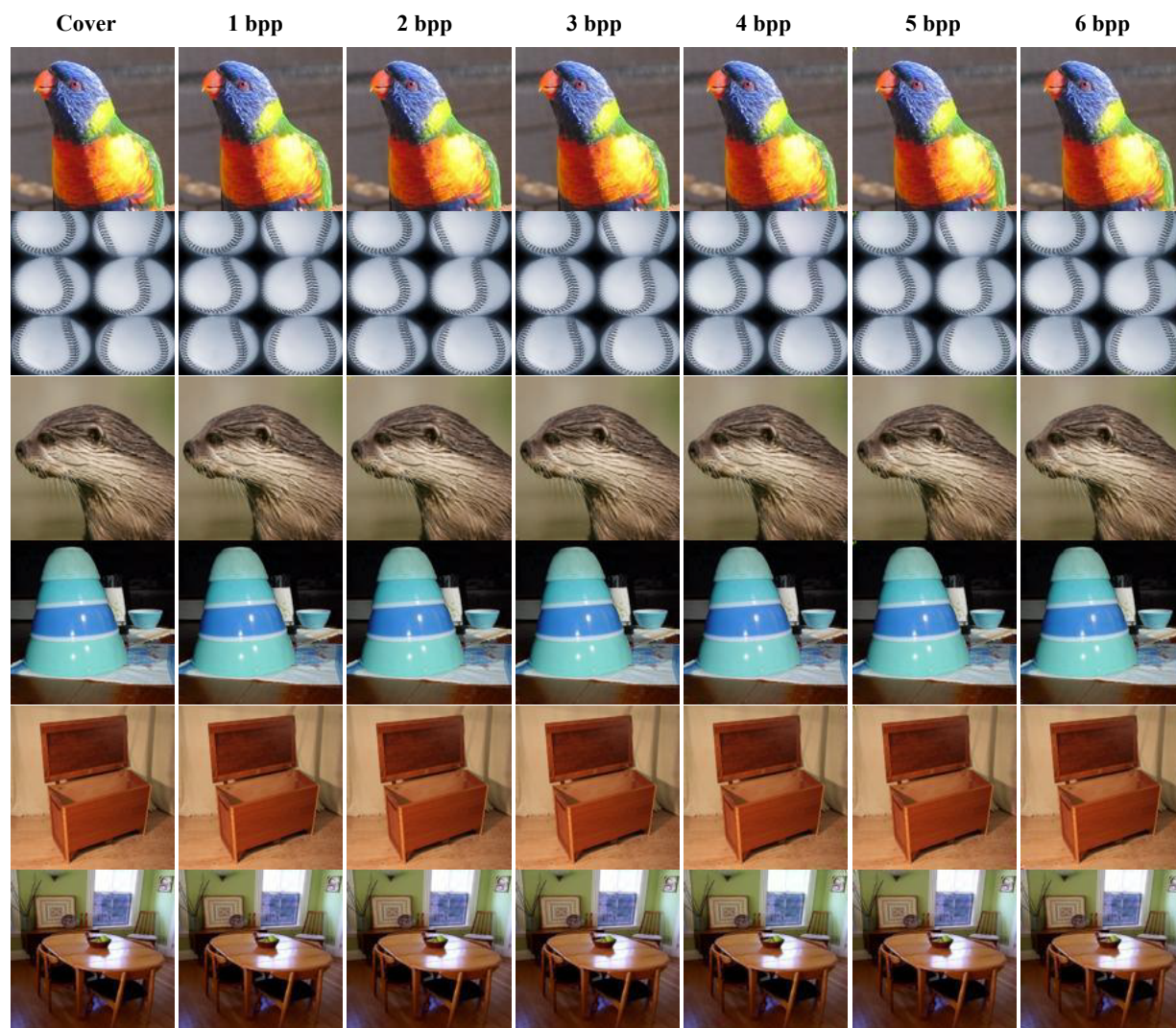
Cover	1 bpp	2 bpp	3 bpp	4 bpp	5 bpp	6 bpp

Figure 3: The original image and the stego image generated by CLPSTNet at 1-6 bpp

5   Conclusion

In this paper, a multi-scale convolutional steganography method, called CLPSTNet, is proposed.Based on GAN, encoding, decoding network, and steganographic analysis network are chosen as the information embedding, information recovery and evaluation parties. The progressive multi-scale convolutional module is used in the encoding network and decoding network, which integrates the idea of curriculum learning algorithm and combines the idea of curriculum learning with the model structure of steganography, which greatly improves the quality of steganographic images and the embedding capacity of images. We use multiple steganography evaluation metrics and steganography analysis network to analyze our proposed model, which realizes the invisibility and undetectability of information. However, the CLPSTNet model also has some weaknesses. First, CLPSTNet can only embed binary information, and cannot be embedded for other data such as images. In addition, the CLPSTNet model does not perform well enough in terms of information recovery accuracy, which appears to decrease with the expansion of information embedding capacity. In the future, we will explore the performance of progressive multi-scale convolutional modules for decoding.

References

[1] Junxue Yang and Xin Liao. Acgis: Adversarial cover generator for image steganography with noise residuals features-preserving. *Signal Processing: Image Communication*, 113:116927, 2023.

[2] Jingxuan Tan, Xin Liao, Jiate Liu, Yun Cao, and Hongbo Jiang. Channel attention image steganography with generative adversarial networks. *IEEE transactions on network science and engineering*, 9(2):888–903, 2021.

[3] Jamie Hayes and George Danezis. Generating steganographic images via adversarial training. *Advances in neural information processing systems*, 30, 2017.

[4] Kevin Alex Zhang, Alfredo Cuesta-Infante, Lei Xu, and Kalyan Veeramachaneni. Steganogan: High capacity image steganography with gans. *arXiv preprint arXiv:1901.03892*, 2019.

[5] Ru Zhang, Shiqi Dong, and Jianyi Liu. Invisible steganography via generative adversarial networks. *Multimedia tools and applications*, 78(7):8559–8575, 2019.

[6] Xintao Duan, Liu Nao, Gou Mengxiao, Dongli Yue, Zimei Xie, Yuanyuan Ma, and Chuan Qin. High-capacity image steganography based on improved fc-densenet. *IEEE Access*, 8:170174–170182, 2020.

[7] Junpeng Jing, Xin Deng, Mai Xu, Jianyi Wang, and Zhenyu Guan. Hinet: Deep image hiding by invertible network. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4733–4742, 2021.

[8] Zihan Wang, Neng Gao, Xin Wang, Ji Xiang, Daren Zha, and Linghui Li. Hidinggan: High capacity information hiding with generative adversarial network. In *Computer Graphics Forum*, volume 38, pages 393–401. Wiley Online Library, 2019.

[9] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[10] Pietro Morerio, Jacopo Cavazza, Riccardo Volpi, René Vidal, and Vittorio Murino. Curriculum dropout. In *Proceedings of the IEEE international conference on computer vision*, pages 3544–3552, 2017.

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[12] Lu Zeng, Ning Yang, Xiang Li, Aidong Chen, Hongyuan Jing, and Jiancheng Zhang. Advanced image steganography using a u-net-based architecture with multi-scale fusion and perceptual loss. *Electronics*, 12(18):3808, 2023.

[13] Le Zhang, Yao Lu, Tong Li, and Guangming Lu. Joint adjustment image steganography networks. *Signal Processing: Image Communication*, 118:117022, 2023.

[14] Yinyin Peng, Donghui Hu, Gang Pei, and Yaofei Wang. Image steganography with deep orthogonal fusion of multi-scale channel attention. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4850–4854. IEEE, 2024.

[15] Ye Yao, Junyu Wang, Qi Chang, Yizhi Ren, and Weizhi Meng. High invisibility image steganography with wavelet transform and generative adversarial network. *Expert Systems with Applications*, 249:123540, 2024.

[16] Mohammmad Kashif, Iman Shakeel, Shahnawaz Ahmad, and Shabana Mehfuz. Enhanced pixel privacy: Leveraging deep learning for advanced image steganography. In *2023 7th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, pages 535–541. IEEE, 2023.

[17] Zhengze Li, Xiaoyuan Yang, Kangqing Shen, Fazhen Jiang, Jin Jiang, Huwei Ren, and Yixiao Li. Adversarial feature hybrid framework for steganography with shifted window local loss. *Neural Networks*, 165:358–369, 2023.

[18] Xiao Ke, Huanqi Wu, and Wenzhong Guo. Stegformer: rebuilding the glory of autoencoder-based steganography. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 2723–2731, 2024.

[19] Youmin Xu, Chong Mou, Yujie Hu, Jingfen Xie, and Jian Zhang. Robust invertible image steganography. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7875–7884, 2022.

[20] Yuhuan Feng, Yunjie Liu, Hongjuan Wang, Jin Dong, Rujia Wang, and Chunpeng Tian. Image hide with invertible network and swin transformer. In *International Conference on Data Mining and Big Data*, pages 385–394. Springer, 2022.

[21] Fengyong Li, Yang Sheng, Xinpeng Zhang, and Chuan Qin. iscmis: Spatial-channel attention based deep invertible network for multi-image steganography. *IEEE Transactions on Multimedia*, 26:3137–3152, 2023.

[22] Zhenyu Guan, Junpeng Jing, Xin Deng, Mai Xu, Lai Jiang, Zhou Zhang, and Yipeng Li. Deepmih: Deep invertible network for multiple image hiding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):372–390, 2022.

[23] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48, 2009.

[24] Fengchun Liu, Tong Zhang, Chunying Zhang, Lu Liu, Liya Wang, and Bin Liu. A review of the evaluation system for curriculum learning. *Electronics*, 12(7):1676, 2023.

[25] Yuxing Tang, Xiaosong Wang, Adam P Harrison, Le Lu, Jing Xiao, and Ronald M Summers. Attention-guided curriculum learning for weakly supervised classification and localization of thoracic diseases on chest radiographs. In *Machine Learning in Medical Imaging: 9th International Workshop, MLMI 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Proceedings 9*, pages 249–258. Springer, 2018.

[26] Andrew Jesson, Nicolas Guizard, Sina Hamidi Ghalehjegh, Damien Goblot, Florian Soudan, and Nicolas Chapados. Cased: curriculum adaptive sampling for extreme data imbalance. In *International conference on medical image computing and computer-assisted intervention*, pages 639–646. Springer, 2017.

[27] Fengbei Liu, Yu Tian, Yuanhong Chen, Yuyuan Liu, Vasileios Belagiannis, and Gustavo Carneiro. Acpl: Anti-curriculum pseudo-labelling for semi-supervised medical image classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20697–20706, 2022.

[28] Yang Zhang, Philip David, Hassan Foroosh, and Boqing Gong. A curriculum domain adaptation approach to the semantic segmentation of urban scenes. *IEEE transactions on pattern analysis and machine intelligence*, 42(8):1823–1841, 2019.

[29] Anastasia Pentina, Viktoriia Sharmanska, and Christoph H Lampert. Curriculum learning of multiple tasks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5492–5500, 2015.

[30] Samarth Sinha, Animesh Garg, and Hugo Larochelle. Curriculum by smoothing. *Advances in Neural Information Processing Systems*, 33:21653–21664, 2020.

[31] Vinod Kumar Kurmi, Vipul Bajaj, Venkatesh K Subramanian, and Vinay P Namboodiri. Curriculum based dropout discriminator for domain adaptation. *arXiv preprint arXiv:1907.10628*, 2019.

[32] Rishi Sharma, Shane Barratt, Stefano Ermon, and Vijay Pande. Improved training with curriculum gans. *arXiv preprint arXiv:1807.09295*, 2018.

[33] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[34] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[35] Maoke Yang, Kun Yu, Chi Zhang, Zhiwei Li, and Kuiyuan Yang. Denseaspp for semantic segmentation in street scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3684–3692, 2018.

[36] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[37] Qilong Wang, Banggu Wu, Pengfei Zhu, Peihua Li, Wangmeng Zuo, and Qinghua Hu. Eca-net: Efficient channel attention for deep convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11534–11542, 2020.

[38] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.