Hindawi
International Journal of Intelligent Systems
Volume 2023, Article ID 9227348, 16 pages
https://doi.org/10.1155/2023/9227348

Research Article

Semi-White-Box Strategy: Enhancing Data Efficiency and Interpretability of Convolutional Neural Networks in Image Processing

Qi Wang, Jianchao Zeng, Pinle Qin, Pengcheng Zhao, Rui Chai, Zhaomin Yang, and Jianshan Zhang

School of Data Science, North University of China, 3 Xueyuan Road, Taiyuan 030051, Shanxi, China

Correspondence should be addressed to Jianchao Zeng; zjc@nuc.edu.cn

Received 7 December 2022; Revised 25 November 2023; Accepted 30 November 2023; Published 15 December 2023

Academic Editor: Alexander Hošovský

Copyright © 2023 Qi Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Data-hunger is a persistent challenge in machine learning, particularly in the field of image processing based on convolutional neural networks (CNNs). This study systematically investigates the factors contributing to data-hunger in machine-learning-based image-processing algorithms. The results revealed that the proliferation of model parameters, the lack of interpretability, and the complexity of model structure are significant factors influencing data-hunger. Based on these findings, this paper introduces a novel semi-white-box neural network model construction strategy. This approach effectively reduces the number of model parameters while enhancing the interpretability of model components. It accomplishes this by constraining uninterpretable processes within the model and leveraging prior knowledge of image processing for model. Rather than relying on a single all-in-one model, a semi-white-box model is composed of multiple smaller models, each responsible for extracting fundamental semantic features. The final output is derived from these features and prior knowledge. The proposed strategy holds the potential to substantially decrease data requirements under specific data source conditions while improving the interpretability of model components. Validation experiments are conducted on well-established datasets, including MNIST, Fashion MNIST, CIFAR, and generated data. The results demonstrate the superiority of the semi-white-box strategy over the traditional all-in-one approach in terms of accuracy when trained with equivalent data volumes. Impressively, on the tested datasets, a simplified semi-white-box model achieves performance close to that of ResNet while utilizing a small number of parameters. Furthermore, the semi-white-box strategy offers improved interpretability and parameter reusability features that are challenging to achieve with the all-in-one approach. In conclusion, this paper contributes to mitigating data-hunger challenges in machine-learning-based image processing through the introduction of a novel semi-white-box model construction strategy, backed by empirical evidence of its effectiveness.

1. Introduction

In the ever-evolving landscape of machine learning, the past decade has witnessed an exhilarating pace of development. Among the remarkable milestones achieved during this period, the widespread adoption of convolutional neural networks (CNNs) for diverse image-processing tasks stands as a testament to the transformative power of these techniques. These methods, when applied to image problems with well-defined requirements and ample data, have demonstrated their prowess in crafting efficacious models. However, beneath this facade of simplicity lies a complex dichotomy—the data that fuels machine learning is both its lifeblood and weak link.

In practice, it is highly difficult to obtain sufficient valid data due to the difficulty associated with obtaining high-quality original data and the high cost of labeling. Moreover, it is imperative to acknowledge the hunger for data exhibited by machine learning methodologies, especially neural networks. This always leads to problems of overfitting and poor model performances in common practical applications [1–3].

For example, the ImageNet image classification dataset includes 1,281,167 training images and 1000 labeled semantic categories [4]. Despite its vast scale, it remains insufficient to encompass the myriad of scenarios encountered in real-world applications. This data-hunger extends beyond image processing into other domains, such as natural language processing, where models such as GPT-3 require billions of training data, each incurring significant computational overhead [5].

Therefore, it is critical to inhibit the data-hunger tendencies of machine learning algorithms. This study was focused on the abovementioned issue within the context of image processing and CNNs.

Existing research on the problem of data-hunger is conducted mainly with respect to two aspects: the data side and the model side. The underlying principle of the data side approach is to augment a dataset. Basic data augmentation methods expand dataset size through geometric changes such as rotation, stretching, and flipping [6, 7].

The advent of generative adversarial networks (GANs) [8], introduced by Ian et al. in 2014, has opened new avenues for data augmentation [9, 10]. GANs, comprised of generative and discriminator networks, generate pseudoimages that gradually approach reality through iterative refinement. The pseudoimages generated by GAN are considered to be the exploration of existing image information instead of meaningless noise information. Therefore, these pseudoimages offer realism surpassing geometric transformations. However, while data generation potential is evident, it fails to alleviate the computational overhead associated with training. Up to now, research related to GANs remains active, but it primarily focuses on improving generative quality rather than reducing computational burden [10–12].

Among model-side solutions, the most effective application is parameter transfer, which is widely used in natural image processing and medical image-processing research [13–17]. This technique initializes untrained neural networks by transplanting parameters from related networks. This cooperates with fixed training epochs to reduce the possible state space of the model and data requirements. However, parameter transfer faces limitations, such as necessitating identical network structures and the lack of parameter interpretability, restricting its broad applicability.

This study takes a different approach by investigating the data-hunger issue from the model side. It reevaluates the commonly attributed cause, recognizing that data-hunger is not intrinsically tied to the statistical nature of machine learning. In fact, for single-parameter statistical estimation, an increase in the amount of data has a diminishing effect on the improvement of the estimation accuracy [18]. In the case of uniform sampling, the estimates of 1,000 samples and 10,000 samples generally differ only in the final few effective numbers. Therefore, in the case of certain accuracy requirements, the minimum data requirements for single-parameter estimates do not increase indefinitely.

Drawing upon possible approximate correct (PAC) learning theory, initially proposed by Valiant in 1984 [19], this study unveils the positive correlation between the minimum required sample size for training and the model's possible state space [20, 21]. Based on the PAC learning theory, as the number of model parameters escalates, the possible state space of the model expands significantly. Consequently, the minimum number of samples required to train a model is positively related to the size of the possible state space of the model. This revelation identifies the proliferation of model parameters as the core driver of data-hunger. Despite recent trends favoring the construction of complex monolithic end-to-end neural networks [22–24], this study challenges the wisdom of structuring such intricate models all-in-one frameworks, advocating for a more data-efficient paradigm.

The conventional all-in-one framework restricts trained models to specific scenarios, necessitating retraining for minor changes in tasks or scenes. Additionally, it isolates human-refined image-processing knowledge from the model's behavior, especially in high-order semantic domains.

To address these challenges, this study introduces a novel semi-white-box network construction strategy. The primary contributions of this paper are as follows:

(1) A PAC-based theoretical argument of this strategy was performed. Theoretical analyses reveal that the networks split at the semantic level demand a smaller minimum sample size compared to the all-in-one model when the data conform to hypothesized distributions.

(2) Introduction of a strategy called the "semi-white-box strategy" for constructing networks based on semantic decomposition. This strategy achieves effects that are difficult to attain with an all-in-one approach, especially in scenarios with limited available data and smaller model scales.

(3) Implementation of instances of the semi-white-box strategy and validation on both Monte Carlo datasets and real datasets, including MNIST, Fashion MNIST, and CIFAR-10. The results demonstrate the superiority of this approach, particularly in scenarios where data are scarce. Furthermore, the interface developed using the semi-white-box strategy enables modular separation and reuse of model components.

2. Related Works

In model-based approaches targeting the data-hunger issue, network parameter transfer is widely adopted. This method assumes similarity between optimal model parameters for related problems, initializing the network model for the target problem with parameters from a model pretrained on other datasets [3, 16, 17]. However, this method becomes inapplicable when suitable approximate datasets for transfer are scarce. This presents a significant limitation in practical applications, as diverse real-world scenarios might lack appropriate datasets for pretraining. Moreover, parameter transfer methods only modify network parameters without altering the network's structure, thereby failing to alleviate data-hunger problems resulting from parameter bloating.

Pruning techniques were initially employed in decision tree models and later extended to neural network models [25]. Through appropriate pruning, the computational structures of neural networks are streamlined, enabling them to eliminate the influence of redundant parameters and enhance their generalization performance [26, 27]. However, due to current limitations in neural network computation methods, pruning methods require trained models.

Therefore, while pruned models exhibit improved generalization, the initial network model before pruning still necessitates a considerable amount of data for training. In other words, when data scarcity is not severe, pruning techniques can mitigate overfitting and enhance model generalization. Yet, in scenarios with extremely limited available data, training the networks before pruning becomes challenging, resulting in less-than-ideal pruning outcomes.

In systematic research regarding the dissection and reconstruction of neural network model internal structures, split learning has emerged as a recent focus. These techniques involve disassembling all-in-one neural network models into several modules, each possessing varying degrees of independence [28–31]. These split modules can operate on different computing devices and communicate with each other via data links. This approach is meaningful for enhancing training efficiency and fostering an understanding of neural network structures. However, current research primarily focuses on distributed computing; hence, these methods are not particularly effective in addressing data-hunger issues.

Additionally, directly and manually simplifying network model structures may also reduce the network's data requirements. Such attempts are often observed in network practice. However, due to the noninterpretable nature of neural networks and the lack of a unified theoretical analysis method, the effects of directly and manually simplifying existing network structures are often unpredictable. Therefore, this paper uses several classic neural network structures to represent this type of method in Section 4.

Among these methods, transfer learning relies on approximate datasets and hence lacks comparability with other methods. The comparison of characteristics between the remaining methods and the semi-white-box strategy is shown in Table 1.

As shown in Table 1, the semi-white-box strategy can reduce data requirements while improving training and runtime efficiency. Additionally, as the semi-white-box strategy decomposes semantic labels, each of its components can operate independently from the whole, thereby possessing a certain degree of interpretability.

3. The Semi-White-Box Strategy and Theoretical Analysis

3.1. The Semi-White-Box Neural Network Construction Strategy. A digital image is the result of discretely sampling the light projected onto a plane. While the potential state space of a discretized pixel matrix is vast, only a small portion of it can be decoded by the human visual system. The remaining portion is essentially noise devoid of specific semantics [32].

In current image-processing and computational vision research, there is a general consensus that the human visual system consists of multilayer convolutional structures. Convolutional neural networks (CNNs) can achieve adequate results in the field of image processing due to their structural similarities to the human visual system.

In studies with well-defined objectives, all-in-one end-to-end CNNs are often employed as the exclusive model. However, these networks are sometimes expected to directly generate interpretable results using highly complex semantic information, which can be an overestimation of their capabilities. To draw a comparison, it is assumed that a convolutional network can process visual information as comprehensively as the human visual system. While human vision is innate, comprehending abstract concepts necessitates a prolonged learning process. The transition from specific images to intricate semantic labels demands a harmonious synergy between these two capacities. Consequently, the all-in-one end-to-end CNN is tasked with both visual processing and semantic inference in such scenarios. Nevertheless, for the latter task, CNNs may not always be the optimal choice. Furthermore, due to a lack of interpretability, CNNs cannot utilize existing formal knowledge bases or generate interpretable formal knowledge in semantic inference tasks.

Hence, the approach in this study is to decompose the image-processing task of generating semantic labels into two distinct phases, as follows. The first phase involves visual feature extraction, and the second phase centers on semantic inference to ascertain the semantic labels. To bridge these phases, we introduce a set of semantic features denoted as H, where $H = h_i$ and $i = 1...k$. Figure 1 illustrates a schematic of the semi-white box strategy model.

It is important to note that H does not represent a specific vision feature but encompasses a collection of visual features that resist further decomposition and are readily discernible. These features include attributes such as surface texture, boundary clarity, texture complexity, color bias, texture directionality, and more. Fully enumerating the composition of H within a visual scene is a formidable task; however, prior research in computational vision has identified some features. This study, which focuses on the semi-white-box strategy, refrains from explicitly listing the constituents of H in the theoretical section. Suppose that H satisfies the following requirements:

(1) Each $h_i \in H$ should represent an intuitive visual feature that does not rely on semantic inferences. For example, "fluffy" is a more suitable feature than "cat."

(2) $h_i \in H$ should be distinguishable by human vision, rejecting statistical differences that are imperceptible to the human eye, such as higher-order moments. If CNNs demonstrate potential comparable to human vision, this implies the existence of a suitable convolutional network for extracting h_i.

Table 1: Comparison of characteristics between semi-white-box and related methods.

Method	Data demand	Training efficiency	Runtime efficiency	Interpretability
Network pruning	Reduced	No effect	Improved	No
Split learning	No effect	Improved	No effect	No
Manually simplifying	Not guaranteed	Not guaranteed	Not guaranteed	Not guaranteed
Semi-white-box	Reduced	Improved	Improved	Partial

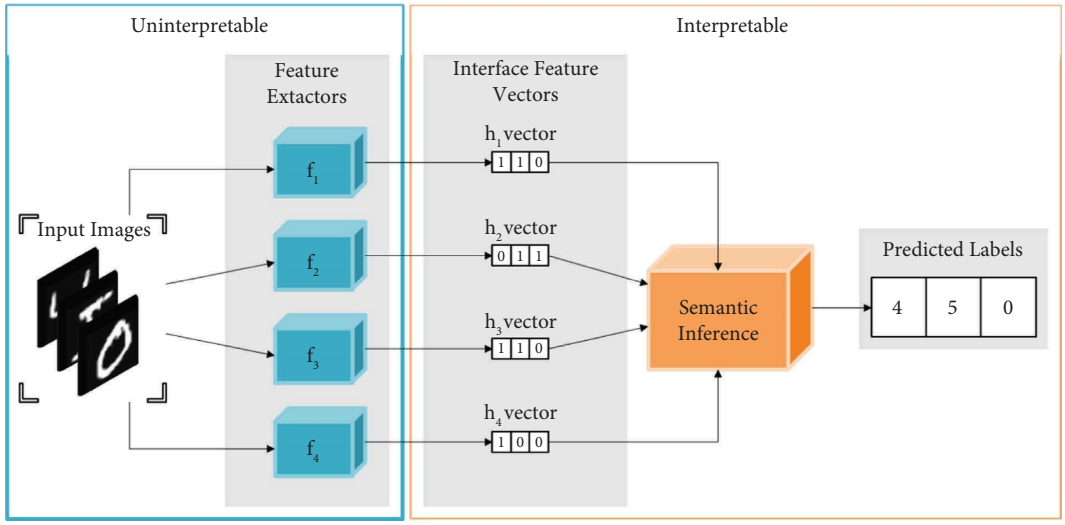

Figure 1: The schematic of the semi-white-box strategy model.

(3) $h_i \in H$ should align with the vocabulary of existing knowledge bases, ensuring the interpretability of results.

(4) The features within H should be orthogonal and maximally independent of one another, reducing redundancy within the feature set.

In the feature extraction phase, a feature extractor f_i was designed for each feature $h_i \in H$. Notably, when nonneural network methods are available, they can be utilized. However, given the strong feature extraction capabilities of neural networks and the complexity of their visual features, neural networks often serve as feature extractors. Given that each f_i uses h_i as a training target, even if f_i is not internally interpretable, its external output is interpretable. In cases where neural networks are employed as f_i, data labeling is necessary for each neural network. Nevertheless, as detailed in Section 3.2, the data labeling requirements are significantly reduced compared to those of all-in-one neural networks. Furthermore, these feature extractors, with externally interpretable outputs, can be reused for various tasks.

Semantic inference leverages the features provided by the feature extractor set F ($f_i \in F$), to derive the final semantic label. In this study, to enable interaction with a formal knowledge base, we employ the formal logic method to facilitate this function. This necessitates a knowledge base expressed in a formal language as a foundation. The inference process takes the formalized knowledge and the extracted image semantic features as premises, drawing conclusions through propositional logic inference. In cases where no suitable knowledge base is available, this stage can be substituted with fitting a statistical decision model using methods like decision trees. In the experiments conducted in this study, the final output is attained through logical operations guided by prior knowledge and informed by feature vectors extracted by the feature extractors.

The difference in the construction process of the all-in-one model and the semi-white-box model is visually depicted in Figure 2.

The semi-white-box strategy does not build an end-to-end convolutional neural network to directly complete the image classification. Instead, it initiates the process by conducting a semantic analysis of the classification labels pertinent to the images. This initial step aims to discern whether these labels can be decomposed as several fundamental and simpler concepts.

In cases where the desired classification labels indeed emerge from a combination of relatively basic concepts, the subsequent task is to delineate the constituents of this foundational concept set. This involves identifying the specific members of the set denoted as H tailored to the problem at hand. Following this, the strategy leverages semantic inferences between these conceptual elements to establish a mapping function linking the set H to the desired classification labels. This process is done by humans based on prior knowledge. At the same time, this process is also a process in which prior knowledge exerts an influence on the model and reduces the uninterpretable part.

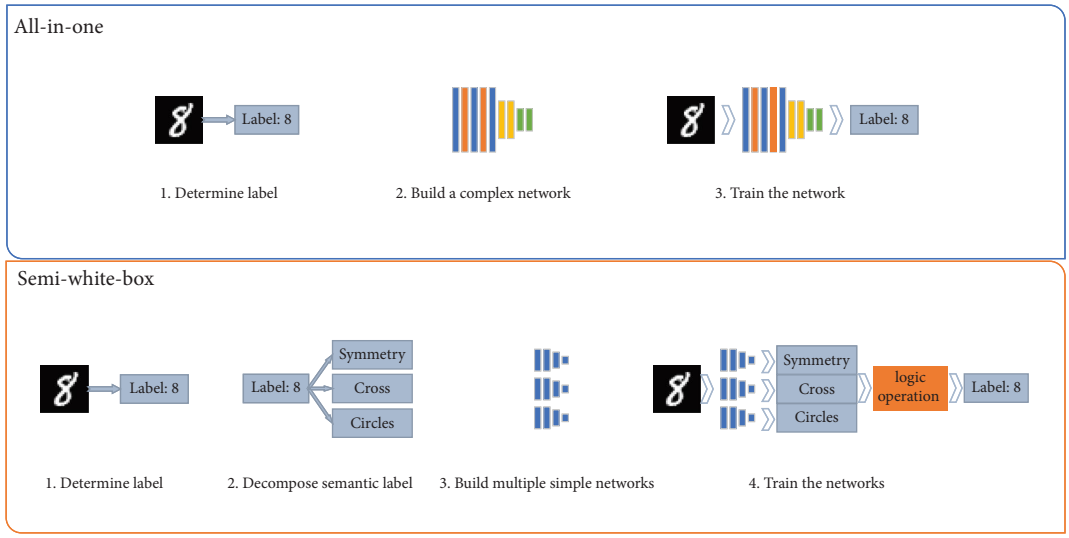

Figure 2: Comparison of all-in-one model and semi-white-box model construction process.

To illustrate, consider the common knowledge shared by human data labelers regarding the significant similarities between handwritten "9" and "6." While this knowledge readily informs human labeling, it becomes concealed within a fully black-box neural network model. In such cases, the trainers of the model must adopt an artificial stance of ignorance concerning prior knowledge related to the similarities of handwritten digits.

The semi-white-box strategy reuses these concealed prior knowledge by representing them as logical judgment conditions. As elaborated in Section 3.4, the strategy decomposes the classification of handwritten digits into stroke characteristics rooted in prior knowledge.

This transformative approach reframes the overarching task of recognizing handwritten digits as a collection of subtasks focused on identifying fundamental stroke features. For each of these converted features denoted as $h_i \in H$, dedicated statistical models are subsequently constructed for feature extraction. From a perspective that considers the number of potential states and entropy, this approach yields a composite feature extractor system for each h_i in H that is notably less complex than the all-in-one model used prior to the transformation, as expounded upon in Section 3.2.

3.2. Entropy and Sample Complexity Analysis. To analyze the process from images to semantic labels, it is first assumed that mapping can be established from semantic labels to images. Notably, graphic rendering is actually part of the inverse mapping. Let H_s represent the set of semantic features to be conveyed by an image, consisting of natural language-based semantic features. These features may comprise connotative nouns (e.g., "cat") or abstract concepts (e.g., "chasing"). Concurrently, there exists another set of features denoted as H, with a known or relatively straightforward mapping relationship to H_s. Furthermore, H and H_s may be interlinked through semantic inferences. Consequently, the mapping from H_s to an image can be viewed as a composition of the mapping from H_s to H and the mapping from H to an image.

In this system, all possible mappings from H_s to images constitute a space, with its entropy, denoted as E_t, calculated based on the number of system's possible states. The mappings from H_s to H and from H to images create their respective spaces, each with entropies represented as E_{s1} and E_{s2}.

Considering that all possible states are equally likely, according to the entropy calculation (1), entropy is solely determined by the number of basic events, as shown in (2).

$$E = -\sum_{i=1}^{i=n} p_i \log_2 p_i, \tag{1}$$

$$E = -\sum_{i=1}^{i=n} \frac{1}{n} \log_2 \frac{1}{n}. \tag{2}$$

According to the power set theorem, the number of all possible mappings from set A to set B in a finite context depends solely on the number of elements in sets A and B. This relationship can be expressed using the following equation:

$$\mathrm{Card}\left(f_{A \longrightarrow \mathrm{B}}\right) = \mathrm{Card}\,(B)^{\mathrm{Card}\,(A)}, \tag{3}$$

where Card denotes the calculation of the number of set elements, while $f_{A \longrightarrow \mathrm{B}}$ signifies the set comprising all mappings from A to B.

Assuming that the total number of images in the problem domain is n_{Image}, the overall number of potential mappings from H_s to H is determined by (4), and the total number of possible mappings from H to images is determined by (5). The combined total number of all possible mappings in the two stages is defined by (6). This represents the possible state space of the entire semi-white-box strategy model, with the model's training process involving the search for an optimal solution within this space.

$$\mathrm{Card}\left(f_{H_s \to H}\right) = \mathrm{Card}\left(H\right)^{\mathrm{Card}\left(H_s\right)}, \tag{4}$$

$$\mathrm{Card}\left(f_{H \to \mathrm{Image}}\right) = \mathrm{Card}\left(n_{\mathrm{Image}}\right)^{\mathrm{Card}\left(H\right)}, \tag{5}$$

$$\begin{aligned} n_{s1s2} &= \mathrm{Card}\left(H\right)^{\mathrm{Card}\left(H_s\right)} \\ &\quad \times \mathrm{Card}\left(n_{\mathrm{Image}}\right)^{\mathrm{Card}\left(H\right)}. \end{aligned} \tag{6}$$

For the all-in-one model, given that Card (H) is itself an adjustable parameter, the model's state space is described by the following equation:

$$n_t = \sum_{i=1}^{i=M} i^{\mathrm{Card}\left(H_s\right)} \times \mathrm{Card}\left(n_{\mathrm{Image}}\right)^i, \tag{7}$$

where M is the model hyperparameter, which is generally set to be much larger than Card (H) to ensure successful model training. Usually, M is not set explicitly but is affected by modifying the width and depth of the model. The entropy of the possible state space for the all-in-one model is expressed by (8), while for the semi-white-box strategy, it is denoted as (9).

$$E_t = -\sum_{i=1}^{i=n_t} \frac{1}{n_t} \log_2 \frac{1}{n_t}, \tag{8}$$

$$E_{s1} + E_{s2} = -\sum_{i=1}^{i=n_{s1s2}} \frac{1}{n_{s1s2}} \log_2 \frac{1}{n_{s1s2}}. \tag{9}$$

Given that $n_t \geq n_{s1s2}$, the relationship between these three entropies is expressed by (10). These signify that the semi-white-box strategy has a lower entropy than the all-in-one neural network. The inequality takes an equal sign when the semantic relations between H_s and H do not constrain the space formed by mappings between H_s and the image, and the hyperparameter settings of the all-in-one model are exceptionally reasonable. This extreme case suggests that high-level semantic features in the image-processing problem cannot be dissected into lower-level semantic features, making texture analysis and pattern recognition unfeasible. This contradicts the general consensus in the field of image-processing research, resulting in the greater-than sign in the following equation:

$$E_t \geq E_{s1} + E_{s2}. \tag{10}$$

If a knowledge base can be established detailing the semantic relations between H_s and H, the mapping from H_s to H becomes entirely certain rendering $E_{s2} = 0$. In this scenario, the semi-white-box strategy aims to train the inverse mapping from H to images, whereas the all-in-one neural network targets the inverse of the mapping from H_s to images. As a result, the entropy of the problem addressed by the semi-white-box strategy is lower than that of an all-in-one neural network, implying a smaller parameter search space and facilitating the acquisition of numerical solutions.

The above analysis represents an entropy-based examination of the problem domain. Drawing on PAC theory, the sample complexity of the two strategies can be compared. Sample complexity refers to the minimum amount of data required for model training, and when a hypothesis class is PAC-learnable, its sample complexity is depicted in the following equation [21]:

$$S = \frac{\log\left(2|H|/\delta\right)}{2\epsilon^2}, \tag{11}$$

where $|H|$ denotes the VC dimension of the hypothesis class, ϵ signifies accuracy, and δ represents confidence.

The VC dimension of a neural network is typically expressed as S (Nlog (N)) [21], with N representing the number of connections within the network, leading to the following equation:

$$S = \frac{\log\left(2O\left(N \log\left(N\right)\right)/\delta\right)}{2\epsilon^2}, \tag{12}$$

where ϵ and δ are constants when the accuracy and confidence requirements are determined. When assuming that the constructed neural network's complexity matches that of the problem, the number of connections in the all-in-one strategy model is N_t. In contrast, the semi-white-box strategy constructs k networks for extracting k features, with the number of connections in these k networks being represented as $N_{s1},...,N_{sk}$. Since the k features are orthogonal and independent, there is no need to develop separate datasets for each subnetwork. Instead, the same fully labeled data can be reused for training different subnetworks, as detailed in Sections 3.3 and 3.4. If $N_{s\max} = \max\left(N_{s1},...,N_{sk}\right)$, the sample complexity of the semi-white-box strategy is solely related to $N_{s\max}$.

Given that each feature in the semi-white-box strategy constitutes only a portion of the final output semantics, the complexity of each subnetwork is lower than that of the all-in-one neural network. This leads to the following equation:

$$N_t \geq N_{s\max}. \tag{13}$$

The equality sign holds when the final output semantic label is equivalent to a feature in H, rendering the other feature extractors redundant. As the dependency distribution of the final output semantic label on each feature extractor becomes more uniform, the gap between N_t and $N_{s\max}$ increases. Let S_t represent the sample complexity of the all-in-one strategy and S_s denote the sample complexity of the semi-white-box strategy. Substituting (13) into (12) results in the following equation:

$$S_t \geq S_s. \tag{14}$$

Hence, under specific accuracy and confidence requirements, the sample complexity of the semi-white-box strategy is no greater than that of the all-in-one strategy. When feature set H aligns with the assumption of orthogonal independence, and the final output semantic labels exhibit uniform dependence on the features in H, the sample complexity of the semi-white-box strategy is less than that of the all-in-one model.

3.3. Monte Carlo Test Design. To verify the feasibility of the semi-white-box strategy and test its specific performance, a Monte Carlo test experiment was designed in this study. In this experiment design, to ensure that the image data adhered to the premise, all the image data were randomly generated. Additionally, to mitigate any subjective bias stemming from the manual specification of image characteristics, all image generators were randomly generated. Consequently, the experimental data in this study can be characterized as a type of "second-order" random data.

The image generation process, as depicted in Figure 3, was bifurcated into two distinct processes, in accordance with the assumptions detailed in Section 3.1.

In the first process, feature maps were generated using a set of feature generators corresponding to the semantic feature interface H.

The second process involved the fusion of the feature maps from the previous stage into a single composite image. To mitigate the inclusion of a multitude of random pixel maps significantly deviating from the natural image distribution in the generated results, both the feature generator and the fusion device were designed with a CNN structure. Because the consensus in the field is that convolution aligns more closely with human visual processing. Moreover, CNN-based image generation algorithms have witnessed notable advancements in recent years.

In this study, all feature and fused images were standardized as squares with dimensions of 32 pixels in length and width. Each feature map generation network consisted of a two-layer fully connected network and a two-layer convolutional network, producing a feature map based on a random floating-point number.

Given that a completely randomly initialized feature generator cannot guarantee reversibility as expressed in (15), certain restrictions were imposed on the random selection of generator parameters. Specifically, a feature decoder was constructed with a structural complexity akin to that of the feature generator, forming an autoencoding structure paired with the feature generator for training. This ensured that all random feature generators had at least one numerical solution for the inverse function.

$$\exists h^{-1}\left(h^{-1}\left(h\left(x\right)\right)=x\right). \tag{15}$$

The input and output of the fusion network were both images; therefore, they were composed entirely of convolutional layers. After undergoing autoencoding via two downsampling layers and two upsampling layers, the fusion network amalgamated multiple feature maps into a single image. Similarly, to guarantee the reversibility of the fusion network, a decoding network was constructed corresponding to the fusion network for training.

During the generation of experimental data, a set of feature maps was initially created by the feature generation network. These feature maps were subsequently fused by the fusion network. Importantly, the semantic label corresponding to each image could be directly derived based on the random floating-point number input to the network. While these generated data may lack clear semantics from a human perspective, their distribution closely mirrors that of natural images. At the same time, the numbers of feature generators and fusion networks were saved as prior knowledge for semantic label decomposing during semi-white-box model building. This is a simulation of the situation in which a part of the prior knowledge is known in the real-world scene, but the entire image-processing process cannot be completely processed by the analytical method. In contrast to the challenge of obtaining labeled natural images, such synthetic datasets can be generated in substantial quantities and serve as excellent materials for experimental verification.

After generating the images, each image was assigned a classification label based on the random values used to generate it. These random values were floating-point numbers ranging from 0 to 1. They were divided into two categories based on whether they were greater than or equal to 0.5. Therefore, an image generated by a random feature generator could have one of two labels. The differences between images assigned to the same label, which resulted from slight variations in the randomly generated values used to create them, were considered random noise. When multiple features were combined, the number of label categories exponentially increased. If an image was generated by fusing feature maps from k feature generators, it could have 2^k possible labels. This label generation process ensured that the ultimate semantic label matching the image was derived from a set of features, denoted as H, through logical operations. However, at the image level, the final generated image data were not obtained through simple arithmetic and logical operations on a few feature maps but rather through a randomly generated function with complexity equivalent to several layers of a fully convolutional neural network. This dataset effectively simulated real-world scenarios where the semantics of images could be subjected to relatively clear logical operations, but the corresponding images themselves could not be processed through simple arithmetic and logical operations.

In fitting the dataset described above, the all-in-one model is directly trained on the generated images as inputs and the generated labels as outputs. The entire process of extracting features from image features, performing operations between semantic features, and obtaining the final label was completed within a relatively complex network. While this led to heightened network complexity, as elaborated in Section 3.2, it also hindered the effective utilization of semantic prior knowledge within the network due to the uninterpretability of neural network parameters.

In contrast, the semi-white-box model initially decomposed the classification labels based on semantic prior knowledge, breaking down a single numeric identifier into a logical combination of several feature values. Then, relatively simple and independent networks were constructed for each feature. These networks could have similar structures but did not share parameters because they had different feature extraction tasks. Throughout model training, these subnetworks were not trained directly on the final label. Instead, the final labels were transformed into feature labels for each subnetwork through logical operations based on semantic inference relationships. When using the model for inference, image features were first extracted separately using the

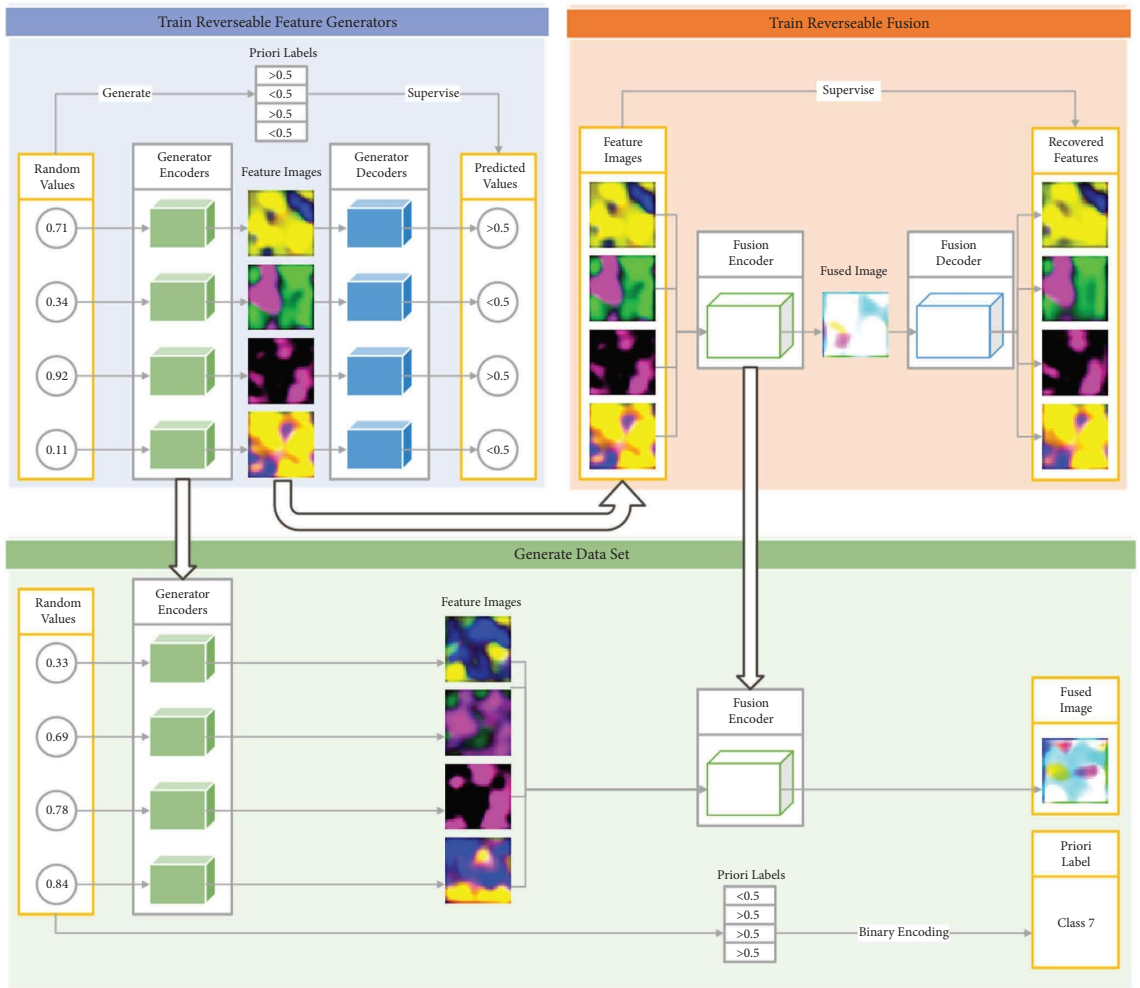

Figure 3: The schematic of generating random dataset.

subnetworks, and then the final output label was obtained through logical operations on the feature values based on semantic inference relationships.

Since the training processes of the subnetworks in the semi-white-box model were independent, the semi-white-box model had some additional advantages during tuning. In practice, to obtain well-performing models, multiple training runs were often performed, and the best results were applied. For an all-in-one model, due to the high coupling of its internal parameters, if the overall performance was poor, the entire model had to be discarded. However, the subnetworks in the semi-white-box model were not interdependent. Therefore, several semi-white-box models with poor overall performance could be disassembled and recombined to create a new model with better overall performance. The implementation was straightforward—replacing a poorly performing subnetwork in one semi-white-box model with a better-performing subnetwork from another model as long as their corresponding features were the same.

To assess the performance of these strategies, we compared the training effects of these model sets using varying amounts of data. Detailed results and analysis are presented in Section 4.

3.4. Real Image Data Test. To investigate the impact of the semi-white-box strategy on real-world image-processing datasets, validation experiments were conducted using the MNIST [33], Fashion MNIST [34], and CIFAR-10 [35]. These datasets are commonly employed in the field of image processing to assess the practical efficacy of novel strategies.

In scenarios with ample training data, traditional all-in-one models can achieve an accuracy exceeding 0.9 on these datasets. However, when the quantity of training data is limited, the performance of all-in-one models significantly deteriorates. In real-world applications, many problem domains often have access to only a small amount of high-quality labeled data, making it challenging to construct datasets on the scale of MNIST. Therefore, in the experiments conducted in this study, a small random subset of the aforementioned datasets was selected for model training to compare the performance of all-in-one models and semi-white-box models under conditions of relatively scarce training data.

The main difference between establishing semi-white--box models for real-world datasets and for generated data lies in the semantic decomposition of labels. Each specific

real-world application comes with its unique prior knowledge, necessitating research to determine the label decomposition tailored to each application scenario.

Taking the MNIST dataset as an example, it contains data with 10 distinct labels. Since $2^3 < 10 < 2^4$, a minimum of four orthogonal and independent features must be selected to cover all classification labels.

In this study, previous research on handwritten character recognition was referred to. Crosses, approximate center symmetry, sharp corners, and vertical lines were utilized as interface features. Examples of these four features are depicted in Figure 4. Crosses, sharp corners, and vertical lines can be extracted through pixel convolution and neighborhood analysis, making them readily accessible via convolutional networks (in the CIFAR-10 dataset, the original ten classifications have been decomposed into artificial, presence on land, relative size, and fur texture). Although approximate center symmetry exhibits a degree of globality, it can be obtained using a combination of convolutional and fully connected networks. Four neural networks were constructed, each dedicated to extracting one of these four features. During the training phase, these networks were trained using images and transformed labels. For example, knowing that a picture represents the handwritten number 0 implies that the picture exhibits features such as approximate center symmetry, the absence of a cross, sharp corners, and vertical lines. Through this process, the necessary label data for training the feature extractors can be inversely deduced using existing high-level semantic labels. During the inference phase, the final label is obtained by performing logical operations on the outputs of the four networks. Similar semi-white-box experiments on other datasets were conducted using analogous methods. As shown in the experimental results in Section 4, the semantic decomposition of labels for real datasets in this study is a feasible solution, but it is not the only one and may not be the optimal one. Exploring more suitable methods for semantic decomposition is a broader topic that goes beyond the scope of this paper.

It is important to note that feature extractors for features within the H do not necessarily always need to be constructed and trained from scratch. If a feature already has a clear white-box extraction method or an established model, it can be directly used to replace the corresponding subnetwork in the semi-white-box model.

4. Experiments and Results

4.1. Experiment Introduction. In this study, VGG [36], ResNet [37], EfficientNet [38], EfficientNet V2 [39], ViT [40], and Swin [41] were chosen as representatives of the all-in-one models. They have been thoroughly tested in the field of image processing and are considered highly representative. Additionally, to compare the effect of pruning methods on the performance of the all-in-one model, the latest Torch Pruning [26] was used to prune the best-performing ResNet model from the aforementioned models. The pruned ResNet model was included as a comparative item in the experimental results.

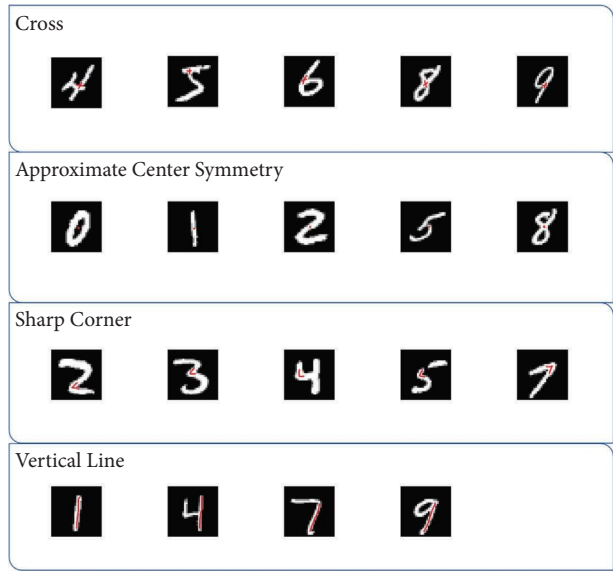

Figure 4: Features used in MNIST dataset test. Red markers represent key points of features. In the approximate center symmetry row, the marks indicate the centers of symmetry.

In contrast, the semi-white-box models consisted of four subnetworks for feature extraction, with each subnetwork being a simple neural network composed of three convolutional layers and two fully connected layers. The key difference lies in the semantic inference relationships between features and the final labels in different experimental scenarios. Additionally, to demonstrate the disassembly and reassembly capabilities of the semi-white-box model as introduced in Section 3.3, some experimental results will present two sets of semi-white-box models: one showcasing performance without recombination and the other highlighting the performance of the optimally recombined models. Since the all-in-one model lacks recombination capabilities, there are no corresponding recombined model data.

The experimental framework for this study was implemented using Python and PyTorch. Hardware acceleration was achieved using an NVIDIA GTX 1060 graphics card. Consequently, the algorithms employed in this study did not require extensive memory space.

The experimental code demo for this paper is available at https://github.com/ZhiZe-ZG/IDT-Open-Source.

The fundamental characteristics of the experimental models are presented in Figures 5 and 6. Here, model size refers to the size of the model parameter file, measured in kibibytes (KiB), while training speed indicates the number of batches trained per second in the experimental environment with a batch size of 10. As observed from the data in the table, the semi-white-box models employed in the experiments are relatively smaller in size and exhibit faster training speeds.

4.2. Data Generation. Data generation played a pivotal role in the Monte Carlo testing conducted in this study. Underpinning this investigation was the premise that if the

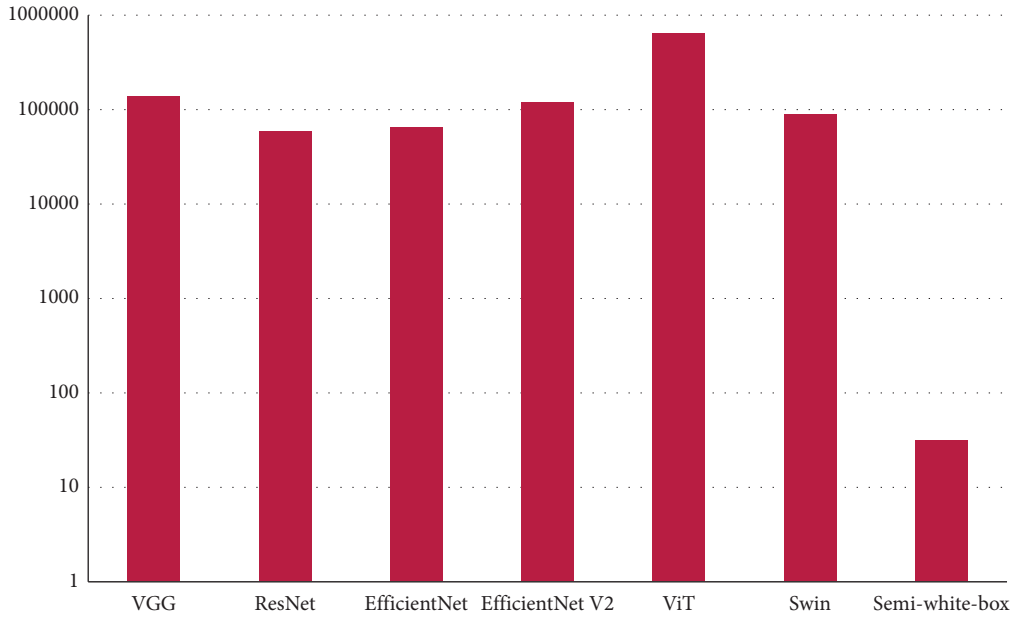

Figure 5: Comparison of model parameter sizes. Model sizes refer to the sizes of the model parameter files, measured in kibibytes (KiB). Due to the vast differences in magnitudes, logarithmic scales are used for the vertical axis here.

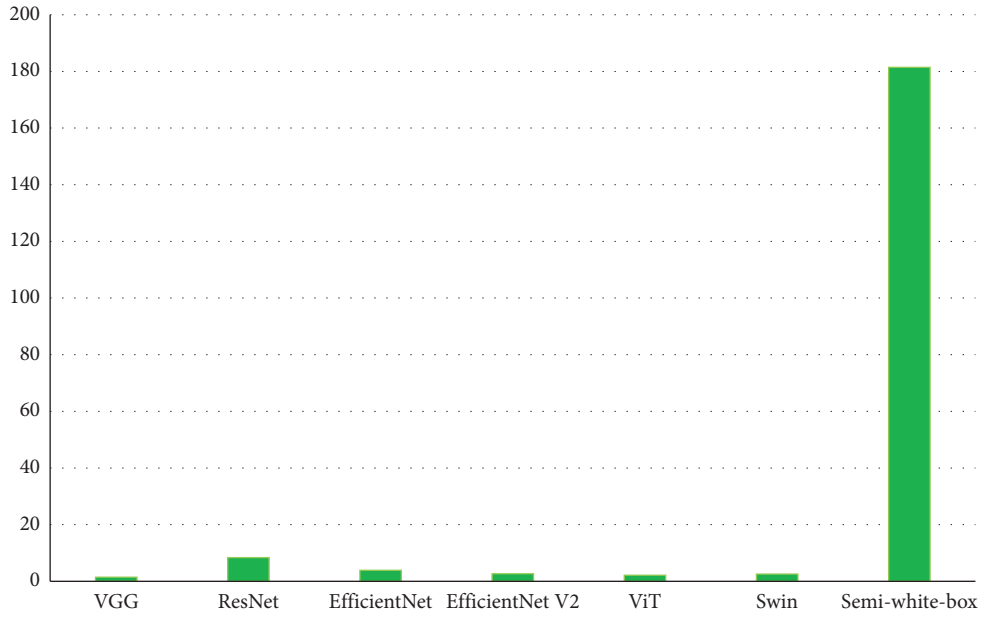

Figure 6: Comparison of model training speeds. The training speed indicates the number of batches trained per second in the experimental environment with a batch size of 10.

assumptions in the image-processing field concerning the relationship between human vision and convolutional neural networks (CNNs) hold true, then the space comprising images generated by reversible convolutional network generators encompasses the information most likely to be recognized by human vision. Within the realm of natural images, it can be inferred that the space encompassing all visually recognizable image patterns with well-defined semantics should be a subset of this larger space. In theory, this approach holds the potential to produce images akin to those captured in real-world scenarios. Nevertheless, due to the vast expanse of this space, the likelihood of randomly training the reversible feature generator to produce image patterns with explicit semantics remains considerably low.

Figure 7 showcases a collection of randomly generated feature maps derived from a reversible random feature map generator. These images exhibit various distinctions, yet they can be broadly categorized into two or three groups

corresponding to different values of the same feature. Minor divergences among the samples signify random perturbations. As illustrated in Figure 8, notable stylistic variations are evident among the feature maps generated by different feature generators.

Figure 8 presents the influence of the reversible feature fusion. This process seamlessly melds multiple feature maps into a single image, and the information contained within each feature map is preserved to an extraordinary degree in the resulting composite image. Conceptually, this simulated the process of light propagation by mixing various object information into a flat image of a natural scene. The fusion mechanism intrinsic to the actual process of generating natural images theoretically resides within the expansive space shaped by these reversible fusions. This figure illustrates a four-feature fusion scenario. The first through fourth rows depict the feature maps generated by four distinct feature generators. The fifth row showcases the composite image resulting from the fusion process. Finally, the last four rows present the four feature channels, each of which has been decomposed using the inverse function of fusion.

4.3. Monte Carlo Test Result. In this study, 25 sets of four-feature fusion and corresponding feature generators were generated. This essentially amounts to conducting experiments on 25 different distributions of datasets. Due to the stochastic nature of parameter initialization, neural networks may converge to local optima or fail to converge during training. Consequently, it is practical to repeat the training process multiple times to obtain an optimal model. Therefore, considering the extreme values of accuracy rates across multiple training cycles can provide a more accurate reflection of the model's performance in practical applications. The average results from these 25 experiments are depicted in Figure 9, while the best results are illustrated in Figure 10.

In the line charts in this paper, the horizontal axis represents the number of samples used for training a model from scratch, while the vertical axis denotes the accuracy achieved by the model. To emphasize the performance of models when data are scarce, we focus on curves involving sample sizes less than 1000. In general, as the number of samples used increases, the accuracy of all models tends to rise. However, when the dataset contains fewer than 1000 samples, most all-in-one models achieve relatively low accuracy. In contrast, ResNet outperforms other all-in-one models in these scenarios. The recombined semi-white-box models excel when data are extremely limited, surpassing other models. As the dataset size gradually increases, their accuracy approaches or even surpasses that of ResNet. However, in terms of model size and training speed, the semiwhite-box model significantly outperforms the compared all-in-one model.

4.4. Real Image Data Test Result. Testing on a real-world dataset can be considered a unique case within the expansive space created by all feature generators and fusions, resembling a Monte Carlo test scenario. The key distinction

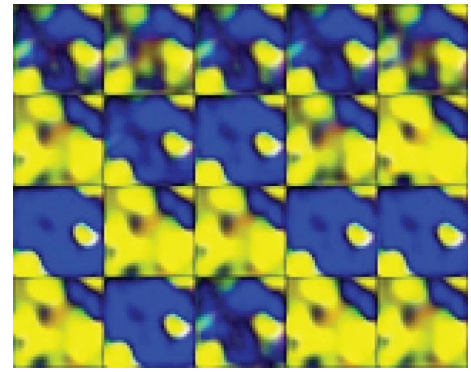

Figure 7: One of generated features. Each image in every grid is generated from a random number. Depending on whether this number is greater than 0.5, the generated images distinctly exhibit two different styles.

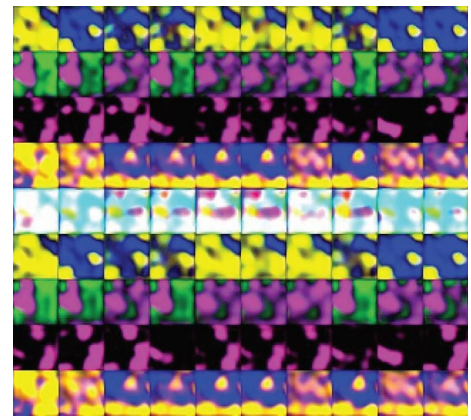

Figure 8: One of generated fusion. Rows 1 through 4 display features generated by four feature generators. Row 5 showcases the image obtained by fusing the four features. Rows 6 to 9 display different features separated from the fused image.

lies in the explicit semantics of real-world datasets. Since each dataset is used independently, it is not possible to compute averages and extremes across multiple different datasets, as is done in the Monte Carlo test. The performance of the compared models on MNIST, Fashion MNIST, and CIFAR-10 datasets is illustrated in Figures 11–13, respectively.

The overall trends in the experimental results align with the data observed in the Monte Carlo test. The recombined semi-white-box models, while maintaining a significantly smaller model size, still achieved performance that was approximately on par with or even superior to ResNet, especially when the available data were less than 200 samples.

Figure 14 presents the visualization of partial feature channels in the semi-white-box model trained on the MNIST dataset. Since the training objective was primarily focused on classification, the expressed features in the feature maps are relatively abstract. Nevertheless, it is still noticeable that the feature extractors exhibit significantly different feature maps for positive and negative samples. This evidence supports the ability of each feature extractor

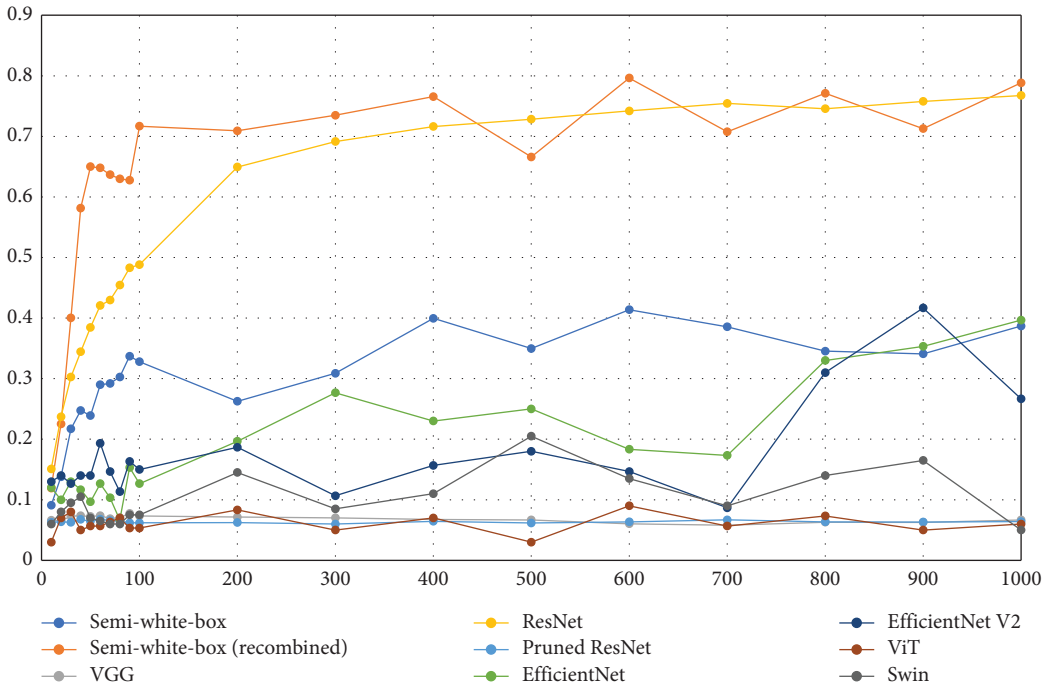

Figure 9: Monte Carlo test mean accuracy. The horizontal axis represents the maximum number of data samples used for training the model, while the vertical axis indicates the accuracy achieved by the model.

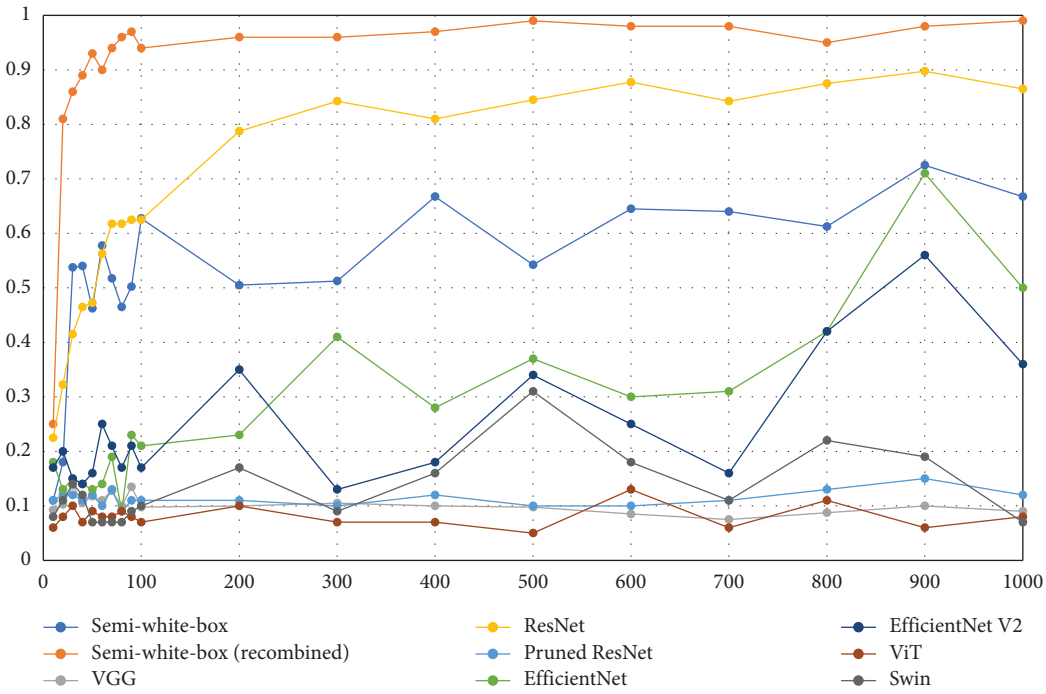

Figure 10: Monte Carlo test best accuracy. The horizontal axis represents the maximum number of data samples used for training the model, while the vertical axis indicates the accuracy achieved by the model.

within the semi-white-box framework to extract its corresponding features.

By comparing the performance of the same models on different datasets, it becomes evident that the complexity of dataset semantics can affect the upper limit of model accuracy. On the semantically more complex CIFAR-10 dataset, all models struggled to attain high accuracy. However, the relative ranking of model accuracies remained largely consistent. Thus, the improvement brought by semi-white-box models in image classification tasks has been

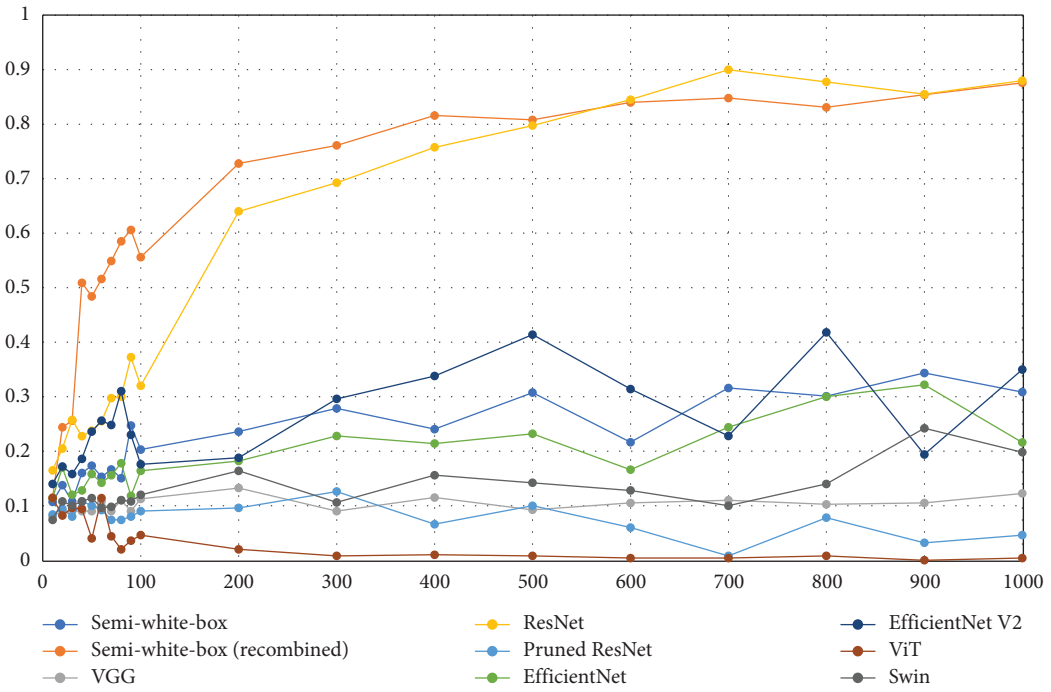

Figure 11: MNIST test accuracy. The horizontal axis represents the maximum number of data samples used for training the model, while the vertical axis indicates the accuracy achieved by the model.

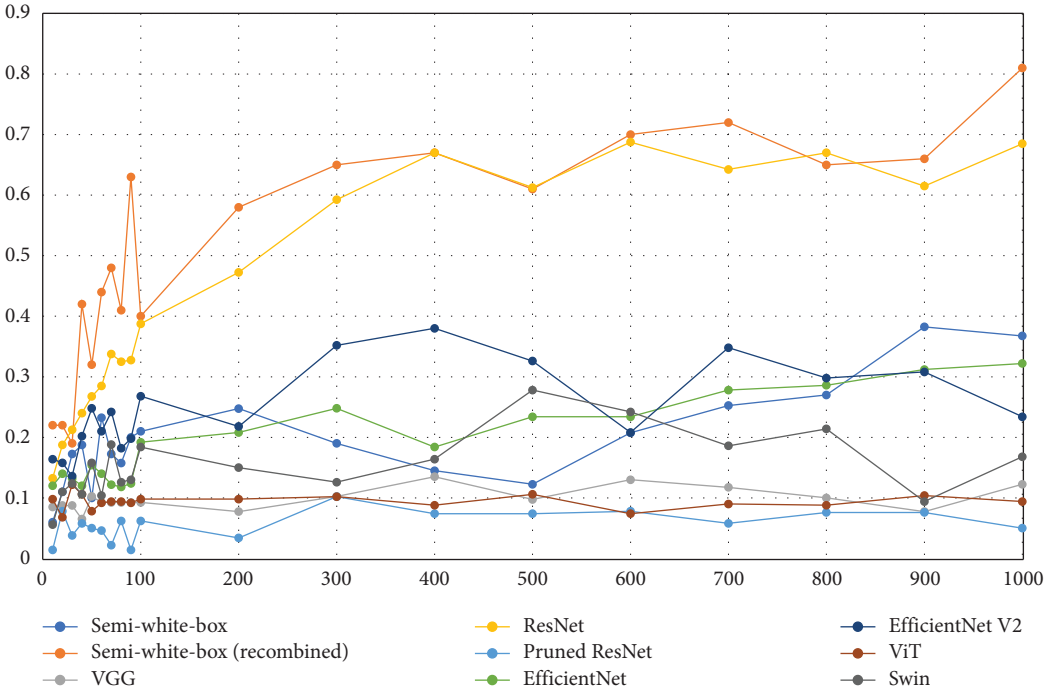

Figure 12: Fashion MNIST test accuracy. The horizontal axis represents the maximum number of data samples used for training the model, while the vertical axis indicates the accuracy achieved by the model.

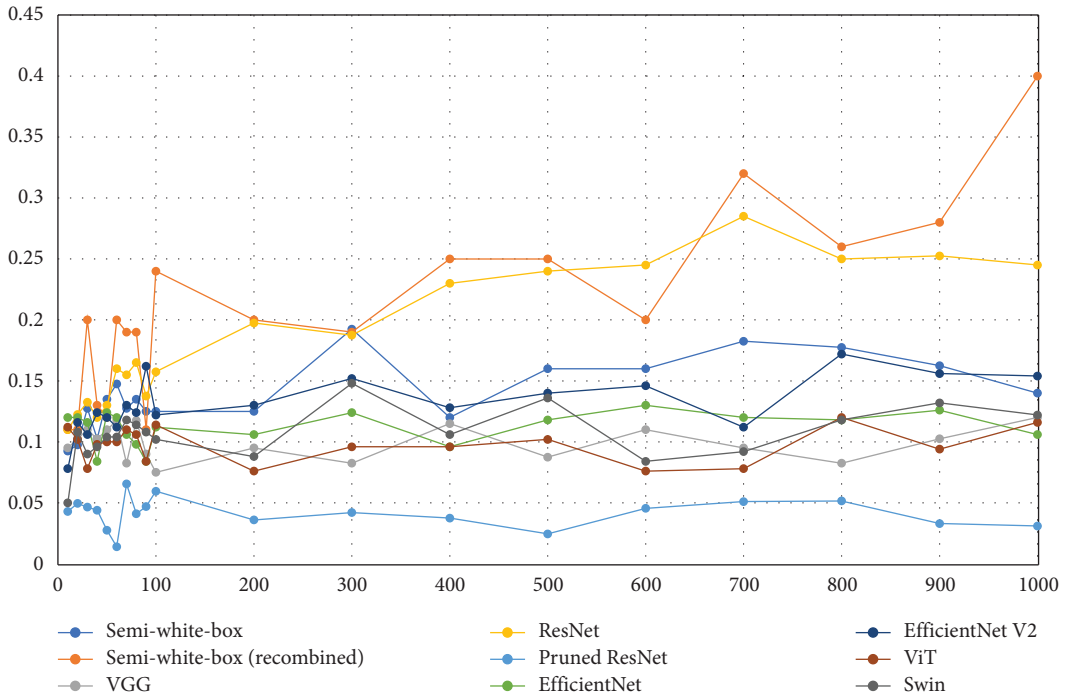

Figure 13: CIFAR-10 test accuracy. The horizontal axis represents the maximum number of data samples used for training the model, while the vertical axis indicates the accuracy achieved by the model.

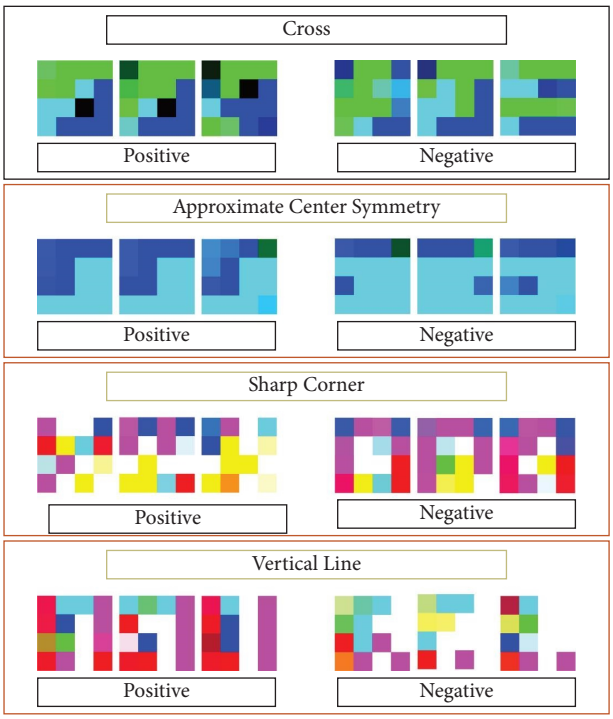

Figure 14: Visualization of partial feature channels of subnetworks corresponding to different features in the semi-white-box model on the MNIST dataset.

validated across various heterogeneous source datasets. Therefore, on the adopted real-world dataset, recombined semi-white-box demonstrates performance comparable to ResNet when available training data are limited, while maintaining a smaller network size and faster training speed.

5. Conclusion and Discussion

In this study, the data-hunger issue of machine learning algorithms was investigated, particularly in the context of image processing. The analysis revealed that the data-hunger problem primarily arises from the rapid growth in model parameters. As a solution, this paper introduces a semi-white-box CNN construction strategy. This approach leverages the semantic clarity of interface features, enabling the incorporation of prior knowledge and modular reuse. Consequently, models developed using the semi-white-box strategy achieve the same accuracy as their all-in-one counterparts while maintaining a smaller model size, particularly when trained with limited data. Using information entropy theory and PAC theory, this paper delves into the principles behind the semi-white-box strategy for reducing model parameters and the minimum number of samples required for training.

5.1. Limitations. The semi-white-box strategy brings some improvements in reducing data requirements, but it may still have limitations in certain practical scenarios. The core idea behind the semi-white-box strategy is the incorporation of prior information to reduce the data requirements for model training. Consequently, this strategy has two primary limitations.

First, when data are abundant and computational resources are ample, the semi-white-box approach may not always be the optimal choice. However, for problem domains with a scarcity of high-quality labeled data, the semi-white-box strategy is a promising avenue to explore.

Second, the effectiveness of the semi-white-box approach heavily relies on the decomposition of semantic labels based on prior knowledge. If there is no suitable prior knowledge available for splitting semantic labels in the applied problem domain, or if the desired semantic labels are inherently challenging to separate, the effectiveness of the semi-white-box approach can be diminished. Therefore, domains like medical imaging, which possess ample prior information and require the fitting of visual models, are best suited to experiment with the semi-white-box strategy.

5.2. Conclusion. This research subjected the semi-white-box approach to extensive testing, conducting experiments on various datasets, including MNIST, Fashion MNIST, CIFAR-10, and randomly generated data. The results demonstrate that the recombined semi-white-box models can achieve accuracy comparable to or even surpassing that of ResNet while maintaining a significantly smaller model size and fast training speed, especially when data availability is scarce, with fewer than 200 samples.

The effective application of the semi-white-box strategy hinges on the careful selection of interface features. This paper introduces principles for interface feature selection that are not specific to any particular application domain.

In fields dealing with natural images, an interpretable image knowledge base from prior studies serves as a valuable reference for interface feature selection. In specialized image-processing domains such as medical imaging, domain-specific knowledge, such as medical expertise and medical imaging knowledge, can guide the selection of suitable interface features.

It is important to note that this research is not a critique of large models. Instead, it offers an alternative approach, the semi-white-box strategy, for scenarios where clear prior knowledge is available. In cases where substantial prior information can be leveraged, training large models from scratch with a wealth of labeled data may not be the most economical choice. Hence, the semi-white-box strategy is presented as a complementary approach to all-in-one models rather than a direct alternative.

Data Availability

The datasets generated during and analysed during the current study are available from the corresponding author on reasonable request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by Study on the Effectiveness of RF Data and Recognition Models in Wireless Sensing (Grant no. 202203021222049), Shanxi Province Major Scientific and Technological Project "Revealing the List and Appointing the Leader" (Grant nos. 202101010101018 and 202201010101004).

References

[1] A. Adadi, "A survey on data-efficient algorithms in big data era," *Journal of Big Data*, vol. 8, no. 1, p. 24, 2021.

[2] T. van der Ploeg, P. C. Austin, and E. W. Steyerberg, "Modern modelling techniques are data hungry: a simulation study for predicting dichotomous endpoints," *BMC Medical Research Methodology*, vol. 14, no. 1, p. 137, 2014.

[3] M. Iman, H. R. Arabnia, and K. Rasheed, "A review of deep transfer learning and recent advancements," *Technologies*, vol. 11, no. 2, p. 40, 2023.

[4] O. Russakovsky, J. Deng, H. Su et al., "ImageNet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[5] T. B. Brown, B. Mann, N. Ryder et al., "Language models are few-shot learners," 2020.

[6] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, p. 60, 2019.

[7] P. Chlap, H. Min, N. Vandenberg, J. Dowling, L. Holloway, and A. Haworth, "A review of medical image data

augmentation techniques for deep learning applications," *Journal of Medical Imaging and Radiation Oncology*, vol. 65, no. 5, pp. 545–563, 2021.

[8] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza et al., "Generative adversarial networks," 2014.

[9] N.-T. Tran, V.-H. Tran, N.-B. Nguyen, T.-K. Nguyen, and N. M. Cheung, "On data augmentation for GAN training," *IEEE Transactions on Image Processing*, vol. 30, pp. 1882–1897, 2021.

[10] Y. Lu, D. Chen, E. Olaniyi, and Y. Huang, "Generative adversarial networks (GANs) for image augmentation in agriculture: a systematic review," *Computers and Electronics in Agriculture*, vol. 200, Article ID 107208, 2022.

[11] Y. Chen, X.-H. Yang, Z. Wei et al., "Generative adversarial networks in medical image augmentation: a review," *Computers in Biology and Medicine*, vol. 144, Article ID 105382, 2022.

[12] A. Kammoun, R. Slama, H. Tabia, T. Ouni, and M. Abid, "Generative adversarial networks for face generation: a survey," *ACM Computing Surveys*, vol. 55, no. 5, pp. 1–37, 2022.

[13] F. Li, J. Chen, J. Pan, and T. Pan, "Cross-domain learning in rotating machinery fault diagnosis under various operating conditions based on parameter transfer," *Measurement Science and Technology*, vol. 31, no. 8, Article ID 85104, 2020.

[14] X. Fei, S. Zhou, X. Han et al., "Doubly supervised parameter transfer classifier for diagnosis of breast cancer with imbalanced ultrasound imaging modalities," *Pattern Recognition*, vol. 120, Article ID 108139, 2021.

[15] H. Shao, W. Li, M. Xia et al., "Fault diagnosis of a rotor-bearing system under variable rotating speeds using two-stage parameter transfer and infrared thermal images," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–11, 2021.

[16] Z. Zhu, K. Lin, A. K. Jain, and J. Zhou, "Transfer learning in deep reinforcement learning: a survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 11, pp. 13344–13362, 2023.

[17] J. Y.-L. Chan, K. T. Bea, S. M. H. Leow, S. W. Phoong, and W. K. Cheng, "State of the art: a review of sentiment analysis based on sequential transfer learning," *Artificial Intelligence Review*, vol. 56, no. 1, pp. 749–780, 2023.

[18] R. V. Hogg, J. W. McKean, and A. T. Craig, *Introduction to Mathematical Statistics*, Pearson, London, UK, 8th edition, 2019.

[19] L. G. Valiant, "A theory of the learnable," in *Proceedings of the sixteenth annual ACM symposium on Theory of computing-STOC*, pp. 436–445, New York, NY, USA, December 1984.

[20] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning with Applications in R*, Springer, Singapore, 2013.

[21] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning from Theory to Algorithms*, Cambridge University Press, Cambridge, UK, 1st edition, 2014.

[22] M. Z. Alom, T. M. Taha, C. Yakopcic et al., "A state-ofthe-art survey on deep learning theory and architectures," *Electronics*, vol. 8, no. 3, p. 292, 2019.

[23] L. Jiao and J. Zhao, "A survey on the new generation of deep learning in image processing," *IEEE Access*, vol. 7, pp. 172231–172263, 2019.

[24] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: a survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 7, pp. 3523–3542, 2022.

[25] H. Cheng, M. Zhang, and J. Q. Shi, "A survey on deep neural network pruning-taxonomy, comparison, analysis, and recommendations," 2023.

[26] G. Fang, X. Ma, M. Song, M. B. Mi, and X. Wang, "Depgraph: towards any structural pruning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16091–16101, Vancouver, Canada, June 2023.

[27] J. Chang, Y. Lu, P. Xue, Y. Xu, and Z. Wei, "Iterative clustering pruning for convolutional neural networks," *Knowledge-Based Systems*, vol. 265, Article ID 110386, 2023.

[28] W. Wu, M. Li, K. Qu et al., "Split learning over wireless networks: parallel design and resource management," *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 4, pp. 1051–1066, 2023.

[29] B. Yin, Z. Chen, and M. Tao, "Predictive gan-powered multi-objective optimization for hybrid federated split learning," *IEEE Transactions on Communications*, vol. 71, no. 8, pp. 4544–4560, 2023.

[30] A. Bakhtiarnia, N. Miloševi´c, Q. Zhang, D. Bajovi´c, and A. Iosifidis, "Dynamic split computing for efficient deep edge intelligence," 2022.

[31] N. D. Pham, A. Abuadbba, Y. Gao, K. T. Phan, and N. Chilamkurti, "Binarizing split learning for data privacy enhancement and computation reduction," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 3088–3100, 2023.

[32] E. P. Simoncelli and B. A. Olshausen, "Natural image statistics and neural representation," *Annual Review of Neuroscience*, vol. 24, no. 1, pp. 1193–1216, 2001.

[33] Y. LeCun, C. Cortes, C. J. C. Burges et al., "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[34] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms," 2017.

[35] A. Krizhevsky, "Learning multiple layers of features from tiny images," 2009, https://www.cs.toronto.edu/%7Ekriz/learning-features-2009-TR.pdf.

[36] K. Simonyan and A. Zisserman, *Very Deep Convolutional Networks for Large-Scale Image Recognition*, VGG, Oxford, UK, 2014.

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, June 2015.

[38] B. Koonce and B. Koonce, "Efficientnet," *Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization*, pp. 109–123, Springer, Singapore, 2021.

[39] M. Tan and Q. Le, "Efficientnetv2: smaller models and faster training," in *Proceedings of the International Conference on Machine Learning*, pp. 10096–10106, PMLR, July 2021.

[40] A. Dosovitskiy, L. Beyer, A. Kolesnikov et al., "An image is worth 16x16 words: transformers for image recognition at scale," 2020.

[41] Z. Liu, H. Hu, Y. Lin et al., "Swin transformer v2: scaling up capacity and resolution," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12009–12019, New Orleans, LA, USA, June 2022.