Semantic-DARTS: Elevating Semantic Learning for Mobile Differentiable Architecture Search

Bicheng Guo, *Member, IEEE*, Shibo He, *Senior Member, IEEE*, Miaojing Shi, *Senior Member, IEEE*, Kaicheng Yu, Jiming Chen, *Fellow, IEEE*, and Xuemin Shen, *Fellow, IEEE*

***Abstract*—Differentiable architecture search (DARTS) is a prevailing direction in automatic machine learning, but it may suffer from performance collapse and generalization issues. Recent efforts mitigate them by integrating regularization into architectural parameters or rule-based operations selection. These efforts primarily emphasize learning the global class-specific features through the image classification task, while overlooking the fine-grained local information during the search process. In this article, we take the first trial to observe that three semantic challenges arise from the classification-based DARTS: 1) inaccurate class-specific features; 2) partial target attention; and 3) blurred semantic regions. To tackle them in one shot, we propose Semantic-DARTS, combining the masked image modeling (MIM) paradigm with the classification task to incorporate local semantic information into the architecture search. Specifically, we design a lightweight reconstruction head that recovers the corrupted image based on the condensed latent feature, which learns both the local semantics and their relationship patch-wisely. Simultaneously, the concurrent classification head strengthens the connection between the global category of the target and the local semantics of their parts. As evidenced by our experiments, the proposed approach achieves state-of-the-art results on CIFAR-10, CIFAR-100, and ImageNet. Furthermore, the searched model is not only able to improve global class-specific features but also to capture fine-grained local representations, improving both the classification performance and the generalization ability.**

***Index Terms*—Masked image modeling (MIM), neural architecture search (NAS), self-supervised learning, semantic learning.**

I. Introduction

DEEP neural network (DNN) due to its powerful capability in recognizing patterns, has been the de-facto tool in intelligent communication networks [1], [2], which successfully empowers numerous Internet of Things (IoT) applications, such as emotion recognition for brain-computer interface [3] and anomaly detection for smart buildings [4]. In real-world computer vision applications, particularly those within the mobile computing IoT scenarios marked by significant spatial and temporal dynamics, there is inherent heterogeneity in both devices and data distributions [5], [6]. Further, designing suitable models for IoT applications mainly poses two significant challenges, i.e., the constrained computation platform [7] and the vastly diverse data distributions [8]. For example, the computation power of a typical smartwatch is only around three GFLOPS and the memory is only 512 MB. Under this budget, it has to handle not only the IMU data for monitoring the motion activity but also the ECG data for performing disease diagnosis [9]. All of these exhibit considerable obstacles in designing effective DNN models to meet the complex requirements [10]. Aiming at saving the large efforts induced by manual design, neural architecture search (NAS) has achieved remarkable success in automatically designing neural networks given specific conditions and requirements [11], [12], [13], [14]. While reinforcement learning [12], [13] or evolutionary algorithm methods [15], [16] exhibit massive search overheads, the differentiable architecture search (DARTS) [17], [18], [19], [20], [21], [22], [23] plays an indispensable role due to its simplicity and efficiency, as well as the single-shot feature, which is IoT device-friendly.

However, there exist several downsides to the DARTS. For example, the performance collapse issue that the parameter-free operations, such as *skip* connections, hold significant influence over the characteristics of the searched architectures [22]. The poor generalization ability beyond the search data set is further identified [24]. Correspondingly, multiple methods have been suggested for enhancing DARTS, such as incorporating regularization based on empirical markers [22], imposing explicit constraints on the number of *skip* operations [20], [25], mitigating the steepness of the validation loss landscape [26], and addressing the inherent biases in *skip* operations [27], [28]. Recently, Wang et al. [29] have introduced additional fine-tuning techniques beyond the originally two-stage search-train process to select the architectures. Ye et al. [21] have proposed to regularize the probability vector instead of primitive architectural weights, leading to promising results. Unfortunately, all the previous works are limited to the classification task which merely learns the global and abstract

Received 18 May 2024; revised 14 August 2024; accepted 5 September 2024. Date of publication 25 September 2024; date of current version 9 January 2025. This work was supported in part by the National Key Research and Development Program of China under Grant 2022YFE0196000, and in part by the Key Research and Development Program of Zhejiang Program under Grant 2024C01065. *(Corresponding author: Shibo He.)*

Bicheng Guo is with the College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, Zhejiang, China, and also with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: guobc@zju.edu.cn).

Shibo He and Jiming Chen are with the College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, Zhejiang, China (e-mail: s18he@zju.edu.cn; cjm@zju.edu.cn).

Miaojing Shi is with the College of Electronic and Information Engineering, Tongji University, Shanghai 200092, China (e-mail: mshi@tongji.edu.cn).

Kaicheng Yu is with the School of Engineering, Westlake University, Hangzhou 310031, Zhejiang, China (e-mail: kyu@westlake.edu.cn).

Xuemin Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: sshen@uwaterloo.ca).

Digital Object Identifier 10.1109/JIOT.2024.3462954

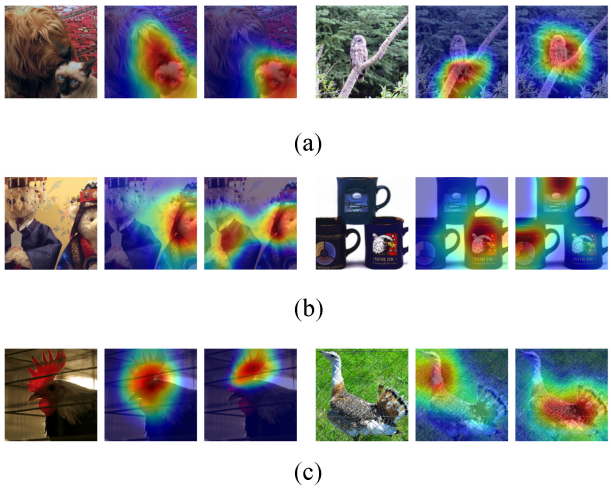


Fig. 1. Classification-based DARTS (center column) versus our Semantic-DARTS (right column). We compare the CAMs. Semantic-DARTS can (a) enhance the global class-specific feature; (b) capture multiple instances of the target class; and (c) attend central semantic regions.

class-specific features. While it is still questionable whether the varying intermediate architecture in the limited searching process of DARTS can accurately learn such comprehensive features. Moreover, the data distributions from the mobile edge computing field are various and diverse. While most of the DARTS works focus on search regulation while neglecting the data distribution adaption, which may further incur biased pattern learning.

We explore these issues by taking the first trail of observing the semantic challenges from classification-based DARTS and analyzing the possible reasons as follows.

1) It is easily interfered by the surrounding objects which are not of interest and thus attends the wrong semantic regions [Fig. 1(a)]. This may contribute to the fact that insufficient training process of DARTS [30] is inadequate for learning accurate class-specific features, which requires accurately mapping the image-level features to target classes. This becomes more severe when searching on a data set with limited samples.
2) Classification-based DARTS tends to attend only part of the targets when multiple instances of a category appear [Fig. 1(b)]. This is because the image-level classification cannot incorporate the detailed local representations distributed in the parts of an image for the changing architecture.
3) DARTS often blurs semantic regions with the backgrounds, indicating a lack of relationship modeling among local representations [Fig. 1(c)]. Though it has been verified that fine-grained semantics, including the local representations for the components of each object and their relationships, are crucial for hand-crafted model design [31], [32], it still remains completely unstudied in the field of DARTS.

In this work, we propose Semantic-DARTS to tackle the aforementioned inferior semantic challenges of classification-based DARTS. First, we inject additional semantic information into the insufficient searching [30] to address the issue of inaccurate feature learning. This is different from the previous regulation-based approaches that intentionally slow down the training. Further, we propose to learn local representation by exploiting the masked image modeling (MIM) [33], [34] task to enhance fine-grained semantic learning. The classification is retained to enable focused attention on the semantic regions of interest, and prompt for the downstream task. [Fig. 1]. By learning the local semantics and global task abstract information in one search round, our Semantic-DARTS has strong data generalization ability. In this way, we wish to offer a generalized IoT application deep learning model search paradigm to simultaneously fulfill two requirements by way of enhancing semantic learning ability.

We provide empirical evidence of the efficacy of our method across various well-known DARTS benchmarks, including DARTS and NAS-Bench-201. Overall, our Semantic-DARTS consistently surpasses classification-based DARTS and its derivatives, achieving an accuracy of 97.54% on CIFAR-10 and 83.81% on CIFAR-100, setting the SOTA top-1 accuracy of 76.5% on ImageNet. Furthermore, we have evidence that the searched models by our Semantic-DARTS generalize better to other data sets beyond the searching data set. To summarize, our primary contributions include as follows.

1) We are the first to observe and analyze the semantic challenges from the classification-based DARTS and underlie the potential reasons that may degrade performance.
2) We then propose the Semantic-DARTS that integrating the MIM auxiliary task to strengthen the fine-grained local semantic representation learning.
3) We empirically demonstrate that our approach attains SOTA performance across diverse settings and provide evidence that the models obtained excel in capturing refined local semantic information while also exhibiting enhanced generalization to other data sets.

The remainder of this article is structured as follows. Section II provides an overview of recent research and advanced methods. Section III delves into the proposed method and its implementation. Section IV presents the experimental results and ablation studies, with conclusions following in Section V.

II. Related Works

Challenges for DARTS: DARTS is a lightweight NAS method, featuring computation-friendly and single-shot. For the most recent DUCCIO [35], its motivation shares a similar idea with us, which is to utilize the lightweight feature to design models suitable for IoT devices. However, their focus is located on the optimization under multiple hardware constraints, while lack of consideration of adapting to multiple distributions for the data characteristic in IoT scenarios like ours. Auto-MCNet [36] utilizes the DARTS to enhance the performance on automatic modulation classification tasks. However, their search paradigm is to first pretrain on the auxiliary data set, then fine tune on a few-shot training data set, which is different from our style, that directly searches

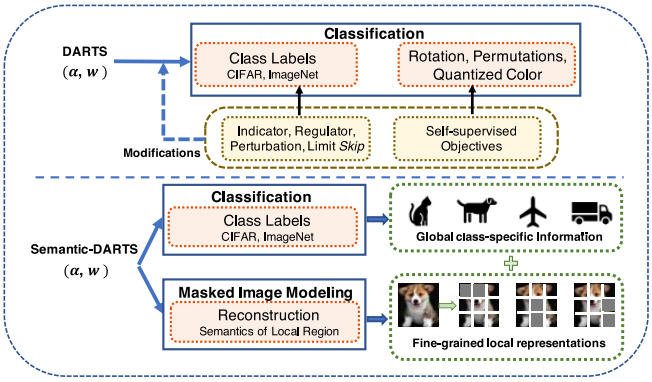


Fig. 2. Comparison of DARTS variants with our Semantic-DARTS.

for the target data set by learning both the target task and the data set's inherent semantics and distribution by MIM task.

DARTS faces several optimization challenges, with Zela et al. [22] pinpointing a performance degradation issue where the resulting architecture is heavily dominated by parameter-free operations. To address this challenge, RobustDARTS [22] and SDARTS [26] utilize the Hessian eigenvalue associated with architectural weights α to identify instances of collapse and introduce early-stop or perturbation strategies to impose regularization on the search procedure. Moreover, β-DARTS [21] suggests a direct regularization approach on the softmax output related to α. P-DARTS [20] suggests explicitly constraining the quantity of the *skip* operations. FairDARTS [28] mitigates unfair *skip* competition through the use of an independent sigmoid function. Wang et al. [29] find that α does not accurately depict the operation's effectiveness and resort to additional adjustments when selecting architectures, prolonging the search procedure. A recently uncovered observation is that DARTS encounters inadequate training, which biases it toward architectures that converge quickly [30]. The aforementioned techniques deliberately introduce search slowdown through diverse regularization approaches. In contrast, our approach suggests augmenting the search with additional information, while avoiding complicating the search pipeline.

Self-Supervised Learning in NAS: Self-supervised learning is practical for tackling obstacles in the NAS domain. For example, RLNAS [37] proposes convergent-based NAS which adopts random labels for supernet training and angle metrics for convergence measurements. BossNAS [38] employs an ensemble bootstrapping to solve the architectural bias when applying block-wise distillation from the human-designed models. There has been an initial endeavor aimed at enabling DARTS to acquire semantic information [39]. Although their initial aim was to debate the necessity of labels in architecture search, our further findings reveal that semantic information, encompassing visuospatial representations, textures, colors, and geometric transformations, proves advantageous in addressing the collapse issue. Different from optimizing with only the classification [39], we enrich the search process and learn fine-grained local representations by additionally introducing the MIM task. Moreover, we do not drop classification task which takes the responsibility of modeling relationships between contexts and target classes.

MIM: MIM [34], [40] stands as a promising avenue in self-supervised learning, focusing on reconstructing masked portions of images. The seminal approach by context encoder [40] adopts a strategy where it obscures rectangular regions, subsequently employing convolutional architectures to regenerate pixel colors. The latest Vision Transformer leverages MIM's prowess in representation learning by forecasting color clusters, [41], mean color [42], the color of raw pixels [34], [43], and patch tokens [33]. Their emphasis lies in the pretraining of model weights within fixed architectures, alongside the utilization of abundant and beneficial training samples, while our target is to optimize the parameters that encode a changing architecture, with the constrained data set to save the search expenses. LatenMIM [44] shares the same motivation as ours to learn both the task-specific high-level semantics and local semantics at the same time. Their method turns to learning the semantics in the latent representation space, in the way of reconstructing masked patches with a constrastive loss. Differently, our method adopts both the target optimization objective and the MIM task, harmonized by an auto-scaling mechanism. DMJD [45] solves the learning inefficiency by deliberately enhancing the usage of tokens (augmenting more masked images). We on the other side enhance the usage of patches by jointly learning the classification and the MIM. In this way, we do not add more training samples which is acceptable for IoT scenarios. A concurrent work [46] reinforces the masterpiece ConvNeXtV1 [47] by adapting to MIM. On the contrary, we search for architecture from the vast search spaces, without any prior.

III. Proposed Method

In this section, we first observe that classification-based DARTS lacks the special design for semantic learning and exhibits three failed semantic patterns which may result in performance degradation. To tackle this, we propose jointly learning MIM and the classification task can solve the problems. An analysis of enhanced semantics and generalization is given. The pseudocode of the proposed Semantic-DARTS is in Algorithm 1, which expatiates the detailed steps for architecture search under the original classification and semantic learning objectives.

A. Preliminary of DARTS

Formulation of DARTS: In this part, we give the basic formulation for conducting DARTS. Tailored for specific tasks like classification, where there's a training set $\mathcal{D}_{\text{train}}$ and a validation set $\mathcal{D}_{\text{val}}$, DARTS seeks a computation cell to construct the architecture. Typically, the directed acyclic graph (DAG) is used to describe the cell which comprises N nodes, with each node x^i representing a hidden feature. Each directed edge (i, j) is connected to an operation, $o^{(i,j)}$, chosen from the candidate operations set $\mathcal{O}$, which contains 3×3 DilConv, 3×3 SepConv, 3×3 MaxPool, 3×3 AvgPool, 5×5 DilConv, 5×5 SepConv, identity, and zero, as shown in Fig. 7. For each operation, it is associated with architectural parameters α to represent the importance compared to other candidate operations. DARTS performs a

Algorithm 1: Semantic-DARTS PyTorch-Style Pseudocode

```
# f(w, α): backbone
# w: trainable weights
# α: arch parameters
# x, u: the original image, (3, H, W)
# y, v: category label for image
# x_p, u_p: image in patches, (L, P x P x 3)
# P: patch size
# L: sequence length

w, α = initialize()
optimizer_w.set(w)
optimizer_a.set(α)

# load samples for w updating
for x, y in train_loader:
  # load samples for α updating
  u, v = val_loader.next()

  # split the image x into patches
  x_p, u_p= patchify(x), patchify(u)
  # randomly masked out patches
  x_m, u_m = masking(x_p, ratio), masking(u_p, ratio)

  # update arch parameters α
  ϵ = f(α, unpatchify(u_m))
  v̂ = cls_head(ϵ)
  û = rec_decoder(ϵ)
  loss_a = cls_loss(v, v̂) + λ * rec_loss(u, û)
  loss_a.backward()
  optimizer_a.step()

  # update operation weights w
  ϵ = f(w, unpatchify(x_m))
  ŷ = cls_head(ϵ)
  x̂ = rec_decoder(ϵ)
  loss_w = cls_loss(y, ŷ) + λ * rec_loss(x, x̂)
  loss_w.backward()
  optimizer_w.step()

# finish searching
arch = select_top2_edges(f)
```

continuous relaxation on learnable architectural parameters α to blend the outputs of these operations

$$\bar{o}^{(i,j)}(x) = \sum_{k\in\mathcal{O}} \frac{\exp\left(\alpha_k^{(i,j)}\right)}{\sum_{k'\in\mathcal{O}} \exp\left(\alpha_{k'}^{(i,j)}\right)} o(x). \tag{1}$$

In this setup, the mixed output of a cell, denoted as $\bar{o}$, is generated. In the search phase, computation cells are stacked to construct the data encoder $E(\alpha, w)$ for processing input data.

TABLE I
LIST OF NOTATIONS AND ABBREVIATIONS

Notation/Abbr.	Definition
α	trainable architectural parameters
w	trainable weights for operations
$\mathcal{D}_{train}$	training set
$\mathcal{D}_{val}$	validation set
x^i	latent feature associated with node i in DAG
(i,j)	directed edge in DAG, connecting node i and node j
$o^{(i,j)}$	operation associated with edge (i,j)
$\mathcal{O}$	candidate operations set
$\bar{o}$	mixed output for computation cell
$E(\alpha, w)$	data encoder stacked by several computation cells
H_{cls}	classification task head
H_{MIM}	masked image modeling task head, reconstruct masked image
w^*	approximated weights using architectural parameters
H, W, C	size of image, Height, Width, and Channel
H', W', C'	size of intermediate feature
P	patch resolution
N	sequence length for patches partitioned from an image, $N = HW/P^2$
$\boldsymbol{m}$	binary mask, shape of $N \times (P^2 \cdot C)$
$\boldsymbol{x}$	original input image, shape $H \times W \times C$
$\boldsymbol{x}_p$	partitioned image, patches format, shape of $N \times (P^2 \cdot C)$
$\boldsymbol{x}_{mask}$	sequence of patches after applying mask $\boldsymbol{m}$
$\boldsymbol{x}_{input}$	image format for $\boldsymbol{x}_{mask}$, shape $H \times W \times C$
$\boldsymbol{x}_{inter}$	intermediate feature, shape $H' \times W' \times C'$
$\boldsymbol{x}_{rec}$	reconstructed image, output by H_{MIM}
$\mathcal{L}_{cls}$	cross-entropy loss for classification task
$\mathcal{L}_{msk}$	MSE loss for reconstruction task
λ	loss harmonize factor
DARTS	Differentiable ARchitecture Search
MIM	Masked Image Modeling
DAG	Directed Acyclic Graph, representing computational cell

This encoder is then combined with the task-specific head H_{cls}. Here, we adopt image classification as an example. Through an alternating optimization process, both the architectural parameters α and the weights w of the operations are optimized

$$\begin{aligned} &\min_{\alpha} \ \mathcal{L}_{\text{val}}\left(E(w^*(\alpha), \alpha), H_{\text{cls}}, \mathcal{D}_{\text{val}}\right) \\ &\text{s.t.} \quad w^*(\alpha) = \arg\min_{w} \mathcal{L}_{\text{train}}(E(w, \alpha), H_{\text{cls}}, \mathcal{D}_{\text{train}}) \end{aligned} \tag{2}$$

here w^* is approximated using either the one-step forward approach or the current w, refereed to second and first methods, respectively. Table I summarizes the notations and abbreviations used in this article.

Issues of DARTS: The original classification-based DARTS, as described by the above formulation, demonstrates a significant issue of performance collapse. Zela et al. [22] discovered that the networks searched from four simplified DARTS search spaces with the CIFAR-10 data set, are dominated by the *skip* connections. We further empirically identify this issue on the standard DARTS search space with the CIFAR-100 data set, shown in the left four columns of Fig. 4: the average top-1 accuracy from four runs indicates a significant decrease and fluctuation during the one-step forward (2nd) approximation. Even with the more stable first-order approximation, over 75% of the operations collapse into *skip* connections. This highlights a notable generalization issue: architectures searched on CIFAR-100 significantly lag behind those optimized for another data set, such as CIFAR-10.

B. Semantic Challenges for DARTS

The optimization objective of previous attempts is limited to the classification task, which learns the abstract class-specific feature for the model. However, it is questionable whether the classification is appropriate for searching an architecture since

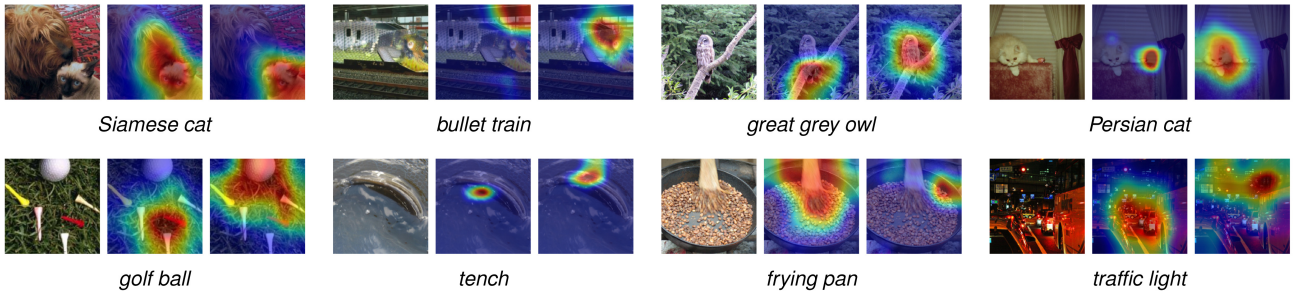


Fig. 3. *Semantic-DARTS achieves precise localization of the target class.* Semantic-DARTS distinguishes from interfering objects, and precisely attends to the corresponding region. The original images on the first column are from the ImageNet validation set. The second and third columns are CAMs overlaid on the original images, obtained from classification-based DARTS and our Semantic-DARTS, respectively.

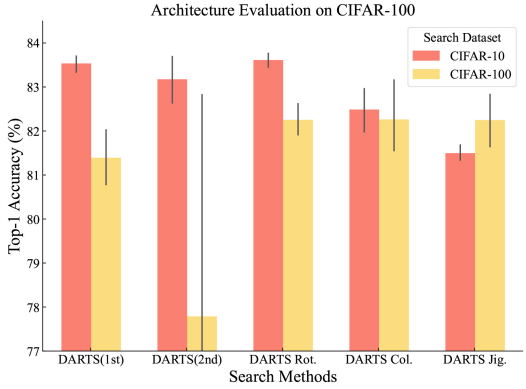


Fig. 4. *Comparison of the original DARTS and the self-supervised DARTS.* Our experiments involved utilizing both human-annotated classification labels and self-supervised pretext tasks on both the CIFAR-10 and CIFAR-100 data sets. Subsequently, we assessed the searched architectures on CIFAR-100. Surprisingly, the models searched and evaluated on the same data set exhibited inferior performance compared to those searched on CIFAR-10, with a margin of 3% and 7% for two DARTS settings. Conversely, leveraging self-supervised labels significantly boosted the performance of the CIFAR-100 searched architecture.

local representation is learned well for a fixed architecture in classification [32], while this cannot be guaranteed for nonfixed architecture in NAS. Consequently, DARTS lacks a special design for learning the local contexts during the search.

Additional support for the significance of semantics can be gleaned from an initial exploration [39]. In this endeavor, they substituted human-annotated labels with visual data generated through three pretext tasks as follows.

1) *Rotation Prediction (Rot.) [48]:* The pretext task is formulated as a four-way classification problem which is to predict the preset rotation angle of the input image (0, 90, 180, and 270 degrees).
2) *Colorization (Col.) [49]:* The pretext task is formulated as a pixel-wise classification problem, which takes as input a grayscale image and outputs the probability of the quantized color value for each pixel. In [49], the image color space is quantized into 313 values.
3) *Solving Jigsaw Puzzles (Jig.) [50]:* The pretext task is formulated as an image-wise classification problem, that is the input image is divided into fixed-size patches which are then randomly shuffled. The task chooses the correct one from a set of preset permutations.

We further find that the visual information, i.e., visuospatial representation, textures and colors, and geometric transformation, provided by these pretext tasks is beneficial for enhancing the performance of DARTS. Nevertheless, their performance fails to outperform the original DARTS. Further, we observe that the objective of [39] is still within the classification scope, similar to all other DARTS variants. Hence, our hypothesis posits that the sole reliance on a single classification paradigm might prove inadequate for the architecture search process to capture valuable semantic insights, potentially leading to suboptimal performance.

Apart from the above analysis, we further investigate the learned semantics of the classification-based DARTS to find out whether their pattern is suitable for architecture search. To do this, we visualize the class activation maps (CAMs) [51] by the LayerCAM method [52] on the last computation cell of the classification-based DARTS, indicating how the trained supernet attends to the regions associated with the target class. LayerCAM is an advanced CAM visualization version. Compared to the previous counterparts [53], [54], LayerCAM utilizes the gradients to highlight the different importance of various feature maps in different locations. Thus, the resulting CAMs not only contain coarse and small class information from the final layer but also high-resolution rich semantics from the shallow layer. As a result, the fine-grained details of the target objects can be effectively kept, which guarantees its fitness for our paper to investigate the ability to learn fine-grained semantic local information. Consequently, we discover three failed and misleading semantic patterns from the classification-supervised supernet.

First, in the middle column of Fig. 3, classification-based DARTS is interfered with by the surrounding objects that are not of interest and thus it attends the wrong semantic regions. For example, for the class *Siamese cat*, it attends additionally to a nearby dog; for the class *great gray owl*, due to the camouflage of the owl being similar to the gray branch, the supernet mistakenly attends to the latter; for the class *traffic light*, the supernet attends wrongly to the alike tail light instead of the upper region containing the target. This pattern may suggest that the insufficient training [30] with limited samples, e.g., usually a portion of the training data set, is unable to learn the useful class-specific feature. Besides, the architecture is changing, making global feature learning much harder.

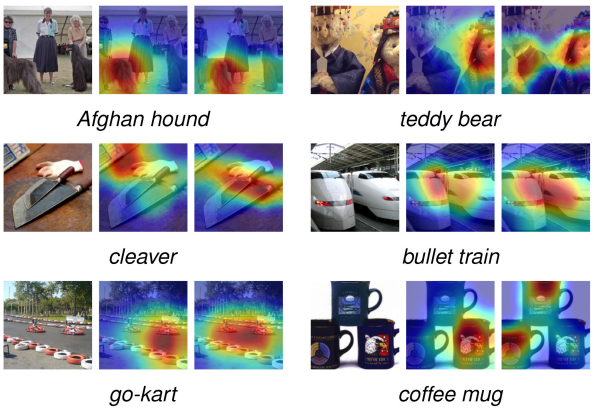


Fig. 5. *Semantic-DARTS captures multiple instances of the target class.* Patch-wisely learning the local representations enables Semantic-DARTS to focus on fine-grained details, rather than a single representative object.

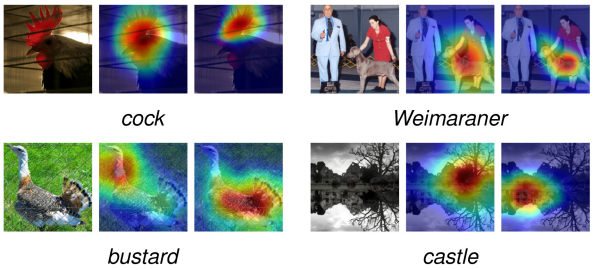


Fig. 6. *Semantic-DARTS obtains refined and central features* avoiding the blurred range of mixed background.

Second, the classification-based DARTS fails to detect most of the objects when multiple instances appear in the image. For example, the right *Afghan hound* and the left two *coffee mugs* are overlooked shown in the examples of Fig. 5. This failure may come from the fact that the classification-based DARTS learns a mapping between the image-level input and a specific class. The rough learning preserves the class-specific features, but it also discards many fine-grained details. In other words, classification-based DARTS absents local representing ability.

Finally, even though the DARTS can attend the right semantic region, it features an enlarged attention area, which is blurred with the background. As shown in Fig. 6, DARTS attends more on the blurred and dark *chicken* body other than the most salient comb; for the *bustard*, the semantic region is on an insignificant slender neck and blurred with the grass background. We attribute it to the lack of relationship modeling among different local contexts, that learned supernet is unable to distinguish objects from noises.

C. Semantic-DARTS

The classification-based DARTS exposes obvious deficiencies of lacking a semantic understanding. To this end, we suggest tackling all three semantic challenges simultaneously without extending the search duration. More precisely, we integrate semantic information into the short supernet search process by introducing an additional learning task known as MIM. This task not only learns the local representations by

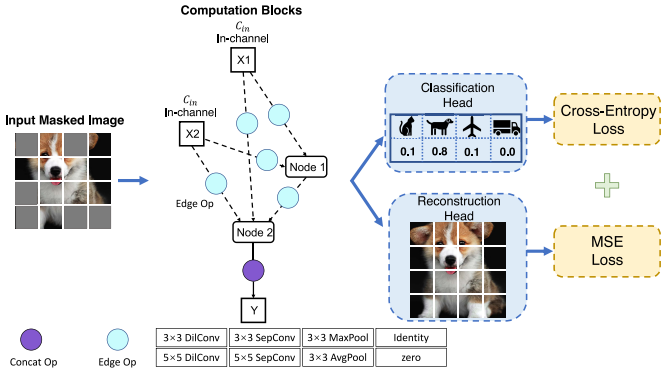


Fig. 7. *Systematic presentation of the proposed method.* We learn the local representations and model their relationship by jointly performing classification and MIM tasks.

recovering the masked contexts but also a holistic understanding beyond low-level image statistics [34]. Moreover, the classification is retained to tighten the local information with the target class and also give a prompt to the architecture of the downstream task. A comprehensive depiction of the proposed method can be observed in Fig. 7. A detailed discussion of its key components is provided below.

Masking: We convert the primitive image into one with masking applied. The architecture infers and recovers the masked patches given the around contexts, hence it learns to model the local representation distributed inside the image. Specifically, given an input image $\boldsymbol{x} \in \mathbb{R}^{H\times W\times C}$, where H and W is the height and width, C is the channel of the image, we partition it into regular, nonoverlapping patches $\boldsymbol{x}_p \in \mathbb{R}^{N\times(P^2\cdot C)}$, where P represents the patch resolution and N is calculated as $N = HW/P^2$. Subsequently, we uniformly select a predetermined percentage of patches without replacement and apply the "dropout" operation to them by setting their values to zero. The masking implementation follows:

$$\boldsymbol{x}_{\text{mask}} = (1 - \boldsymbol{m}) \odot \boldsymbol{x}_p. \tag{3}$$

Here, we have $\boldsymbol{x}_{\text{mask}} \in \mathbb{R}^{N\times(P^2\cdot C)}$ representing the sequence of the visible patches after making operation; $\boldsymbol{m}$ is binary mask of the same dimensions as $\boldsymbol{x}_p$, with a value of 1 indicating a masked pixel; and $\odot$ denotes the element-wise product operation. We then transform $\boldsymbol{x}_{\text{mask}}$ into the shape of a normal image, i.e., $\boldsymbol{x}_{\text{input}} \in \mathbb{R}^{H\times W\times C}$.

Encoder: We assemble the computational cells [17] into the data encoder E, which functions as both the processor for the masked image and the entity responsible for architecture optimization during the process. The encoder includes all the operations of the architecture parameterized by the α. The architecture obtains abundant semantic and class-specific information supervised by two tasks. Besides the standard cell, we have included reduction cells intended to reduce the spatial resolution of the latent feature. These reduction cells incorporate candidate operations with a stride of two

$$\boldsymbol{x}_{\text{inter}} = E\big(\boldsymbol{x}_{\text{input}}\big) \tag{4}$$

where $\boldsymbol{x}_{\text{inter}}$ represents the intermediate feature with dimensions $\mathbb{R}^{H'\times W'\times C'}$.

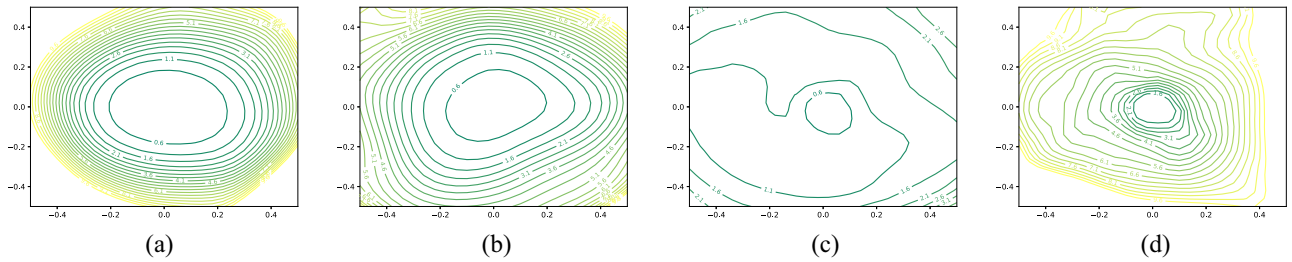


Fig. 8. *Loss landscapes of the searched supernet.* The center corresponds to the obtained supernet weights, two axis parameterize two random directions of the architecture's weights. Loss values are obtained on the training set of CIFAR-10. (a) DARTS, $\mathcal{L}_{\text{cls}}$. (b) Semantic-DARTS, $\mathcal{L}_{\text{cls}}$. (c) Semantic-DARTS, $\mathcal{L}_{\text{msk}}$. (d) Semantic-DARTS, $\mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{msk}}$.

Reconstruction Decoder: The decoder H_{MIM} takes the $\boldsymbol{x}_{\text{inter}}$ as input and reconstructs the masked patches by utilizing information from adjacent visible patches. In contrast to the encoder, which employs variable architecture and stage-specific down-sampling, the decoder maintains a fixed structure aimed at restoring the spatial resolution to match that of the input image. Different from segmentation tasks that can adopt parameter-free decoder to cope with resolution restoration and classification on a limited number of categories, our method, especially the masked patches reconstruction branch, focuses on regressing the difficult color space of the masked patches. Moreover, since we focus more on training the weights w and architectural parameter α of the encoder to learn both the general class information and semantic features, we design a parameter-efficient learnable decoder without introducing significant overheads.

We initiate the process by diffusing information spatially through a 3×3 convolutional operation and across channels using an 1×1 convolutional layer. Subsequently, we employ a sequence of two transposed convolutional layers to systematically restore the spatial resolution, aligning it with that of the original input image. Before each transposed convolution operation, we incorporate a learned filter, implement batch normalization, and activate it with ReLU. In the final step, a 3×3 convolution generates a feature representation for the three color channels, and we ensure value constraints within acceptable ranges using the element-wise HardTanh function. The PyTorch clamp function is also feasible for clipping the color values

$$\boldsymbol{x}_{\text{rec}} = H_{\text{MIM}}(\boldsymbol{x}_{\text{inter}}). \tag{5}$$

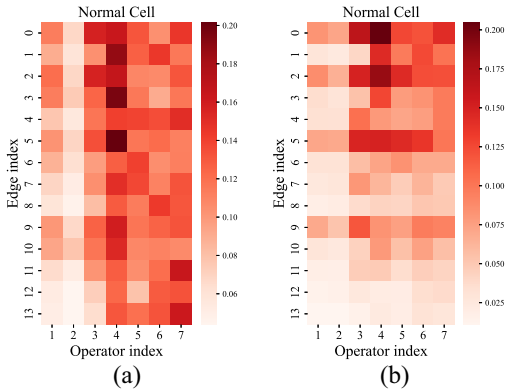
$\boldsymbol{x}_{\text{rec}}$ denotes the reconstructed image with dims $\mathbb{R}^{H \times W \times C}$.

Reconstruction Target: We compute the mean squared error (MSE) by comparing the reconstructed images to the originals, focusing solely on the masked patches during the calculation

$$\mathcal{L}_{\text{msk}} = \|\boldsymbol{m} \odot (\boldsymbol{x}_{\text{rec}} - \boldsymbol{x})\|_2. \tag{6}$$

Simultaneous Learning of Classification and MIM: In addition to the MIM task, we maintain the classification branch to capture the relationship between the local representations and the target class: $\boldsymbol{p} = H_{\text{cls}}(\boldsymbol{x}_{\text{inter}})$ and employ the cross-entropy loss, denoted as $\mathcal{L}_{\text{cls}}$, to guide the training, where $\boldsymbol{p}$ is the categories possibilities. This allows us to simultaneously learn both tasks while maintaining simplicity and flexibility

$$\mathcal{L} = \mathcal{L}_{\text{cls}} + \lambda \mathcal{L}_{\text{msk}} \tag{7}$$



(a) (b)

Fig. 9. *α distribution of normal cell learned by our method and DARTS on CIFAR-10.* Operator indexes 1 through 6 correspond to different operations: 3×3 max pooling, 3×3 average pooling, skip connect, 3×3 and 5×5 separable convolution, and 3×3 and 5×5 dilated convolution, respectively. The total standard deviation is obtained by summing the standard deviation of all the edges. Each edge's α values are normalized using softmax. (a) Ours, total std = 0.70. (b) DARTS, total std = 2.19.

where λ is the loss harmonize factor. Moreover, the classification task can also provide guidance on the downstream task for the trained supernet. Here, we present a straightforward implementation to simplify the weighting scheme. We assign equal significance to the two types of tasks and harmonize the two losses by setting $\lambda = (\mathcal{L}_{\text{cls}}/\mathcal{L}_{\text{msk}})$, with the computation of λ not influencing the gradient update process.

D. Analysis

Enhanced Semantic: To investigate the learned semantic level of the Semantic-DARTS, we visualize the CAMs following Section III-B. It demonstrates that the proposed method learns well the local representations in: our method well addresses the above three failed patterns. In Fig. 3, Semantic-DARTS attends accurately to the region that is associated with the target class. For example, it attends only to the *Siamese cat* and the *gray owl*, excluding the interference of the nearby furry dog and gray branch. It should be noticed that this fine-grained semantic level does not involve any prolongation of the search process, nor does it result in any significant overheads. This indicates that enriching the insufficient search process by MIM task is a promising way to tackle the challenge of learning inaccurate information. Moreover, in Fig. 5, we find the Semantic-DARTS is capable of covering multiple instances of the target in the image:

it attends all of two *Afghan hounds* and *teddy bears*. This is thanks to the fine-grained local representing capability of patch-wisely inferring and recovering the masked patches. Further, Semantic-DARTS obtains more refined features that closely concentrate on the discriminate feature of the target in Fig. 6. Interestingly, Semantic-DARTS even captures the plausible feature: the reflection of the castle.

Stronger Generalization: The classification-based DARTS endures a fast convergence during the insufficient search process [30], resulting in poor generalization ability. We further investigate it by visualizing the loss landscape [55]. It shows a smooth and benign loss landscape of the classification-based DARTS in Fig. 8(a), yielding a wide and shallow architecture with poor generalization [30]. On the contrary, Semantic-DARTS in Fig. 8(d) demonstrates a more agitated landscape, which can search for an architecture that is deeper enough to generalize well. Moreover, by visualizing the branch of classification in Fig. 8(b), we find the loss landscape is much flatter around the center minima when compared to the classification-based DARTS. This indicates that the MIM task can help to ease the fast-changing loss during the training and strengthen the generalization ability [55].

Comparing the α distributions of architectures derived from Semantic-DARTS and the original classification-based DARTS [17] (see Fig. 9), we observe a reduction in the accumulation of standard deviation across all the edges from 2.19 (original DARTS) to 0.70 (Semantic-DARTS). This decrease signifies minimal variations in α values, indicating that the learned architectural parameters converge to a stable zone, resilient to the final operation selection process. Additionally, in contrast to the original DARTS, our approach exhibits a consistent preference for parametric operators (indexed 4, 5, 6, 7) across all the edges, while the original DARTS demonstrates this tendency only in the initial edges.

IV. Performance Evaluation

This section comprises comprehensive experiments aimed at confirming the effectiveness and generalization capabilities of the proposed Semantic-DARTS. We offer an extensive dissection of the essential components and design principles. Following common routine [17], each experiment encompasses the following stages.

1) The pursuit of an optimal cell through validation loss minimization.
2) The composition of cells into a complete architecture, followed by training from scratch for evaluation.

These experiments are carried out across CIFAR-10, CIFAR-100, and ImageNet, utilizing two distinct search spaces: 1) DARTS and 2) NAS-Bench-201.

A. Comparing With State-of-the-Art Methods

Settings: We demonstrate the effectiveness of our proposed method by comparing it to previous SOTA approaches. We assess obtained architectures and evaluate them in two scenarios: 1) within the DARTS search space using CIFAR-10 and CIFAR-100 and 2) within the NAS-Bench-201 search space using CIFAR-10. DARTS stands as a prominent and demanding benchmark for assessing NAS algorithms. It features cells with four intermediate nodes, 14 edges, and eight candidate operations per edge. NAS-Bench-201 represents an alternative and widely adopted cell-based search environment, featuring cells consisting of four intermediate nodes and each edge presenting a selection of five available operations. We adhere to common implementations for consistency [17], [21].

TABLE II
Comparison Results on CIFAR

Method	Search Cost	CIFAR-10		CIFAR-100	
	(GPU-Days)	Params(M)	Acc(%)	Params(M)	Acc(%)
NASNet-A	2000	3.3	97.35	3.3	83.18
DARTS(1st)	0.2*	3.4	97.00±0.14	3.4	82.46
DARTS(2nd)	0.9*	3.3	97.24±0.09	-	-
SNAS	1.5	2.8	97.15±0.02	2.8	82.45
GDAS	0.2	3.4	97.07	3.4	81.62
P-DARTS(C100)	0.3	-	97.38	3.6	84.08
P-DARTS(C10)	0.3	3.4	97.50		82.80
PC-DARTS	0.1	3.6	97.43±0.07	3.6	83.10
CyDAS	0.3	3.6	97.60	-	-
P-DARTS	0.3	3.3±0.21	97.19±0.14	-	-
R-DARTS(L2)	1.6	-	97.05±0.21	-	81.99±0.26
SDARTS-ADV	1.3	3.3	97.39±0.02	-	-
DARTS+PT	0.8	3.0	97.39±0.08	-	-
DARTS-	0.4	3.5±0.13	97.41±0.08	3.4	82.49±0.25
β-DARTS(C100)	0.4	3.78±0.08	97.49±0.07	3.83±0.08	83.48±0.03
β-DARTS(C10)	0.4	3.75±0.15	97.47±0.08	3.80±0.15	83.76±0.22
DOTS	0.3	3.5	97.51±0.06	4.1	83.52±0.13
CyDAS	0.3	3.9±0.08	97.52±0.04	-	84.31
ADARTS	0.2	2.9	97.54	-	82.94
ours(C100)	0.2*	3.61±0.24	97.43±0.11	3.66±0.24	**83.71±0.66**
ours(C100 best)	0.2*	3.73	97.59	3.78	**84.45**
ours(C10)	0.2*	4.05±0.23	**97.54±0.15**	4.10±0.23	**83.81±0.44**
ours(C10 best)	0.2*	4.10	**97.71**	4.05	84.23

The top block exhibits outcomes obtained through the training of the top-performing architecture. Meanwhile, the middle block presents the average accuracy derived from multiple search trails. C10 and C100 correspond to the architectures discovered within CIFAR-10 and CIFAR-100. We have averaged the accuracy across the most optimal architectures identified in four independent runs. $*$ All results was recorded under the same environment on a single RTX 3090.

Results on DARTS: The comparative performance of the architectures we obtained is displayed in Table II. In all four runs conducted on CIFAR-10 and CIFAR-100, we effectively mitigated the performance collapse issue. The average performance of architectures found via CIFAR-100 (83.71%) experiences only a slight decline in comparison to those discovered through CIFAR-10 (83.81%). The obtained architectures are not predominantly influenced by the *skip*. Additionally, the mean accuracy across four runs (83.71%) and the highest attained accuracy (84.45%) outperform CIFAR-100 counterparts equipped with a relatively modest parameter count, averaging just 3.66 million parameters. We further achieve SOTA average accuracy on two data sets, namely 97.54%, and 83.81%. This further underscores the advantages of acquiring fine-grained semantic information through additional learning, as opposed to focusing solely on a single classification task. The top-performing architecture from the four runs attains SOTA results among DARTS variants, reaching 97.71% on CIFAR-10 and 84.45% on CIFAR-100. These results underscore the superior architectural search capabilities of our method.

Results on NAS-Bench-201: We extend the evaluation of our approach to NAS-Bench-201, performing four separate searches and assessing the resulting architectures using CIFAR-10. The comparative outcomes are presented in

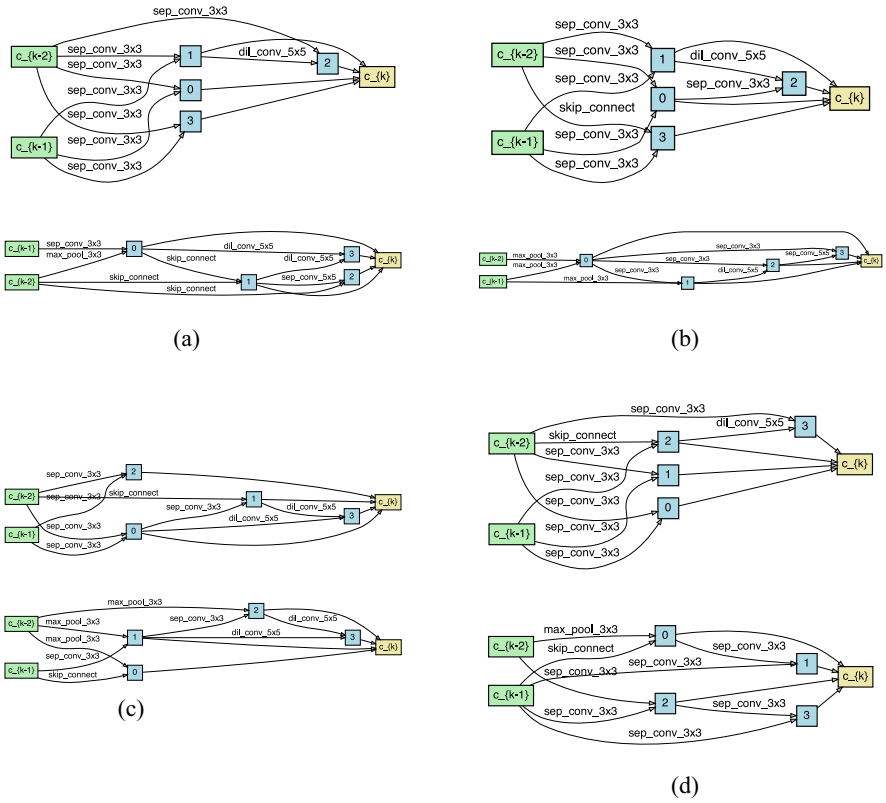


Fig. 10. *Normal and Reduction cells discovered by Semantic-DARTS on CIFAR-10.* (a) ImageNet top-1 accuracy 76.22%. (b) ImageNet top-1 accuracy 76.02%. (c) ImageNet top-1 accuracy 75.71%. (d) ImageNet top-1 accuracy 76.52%.

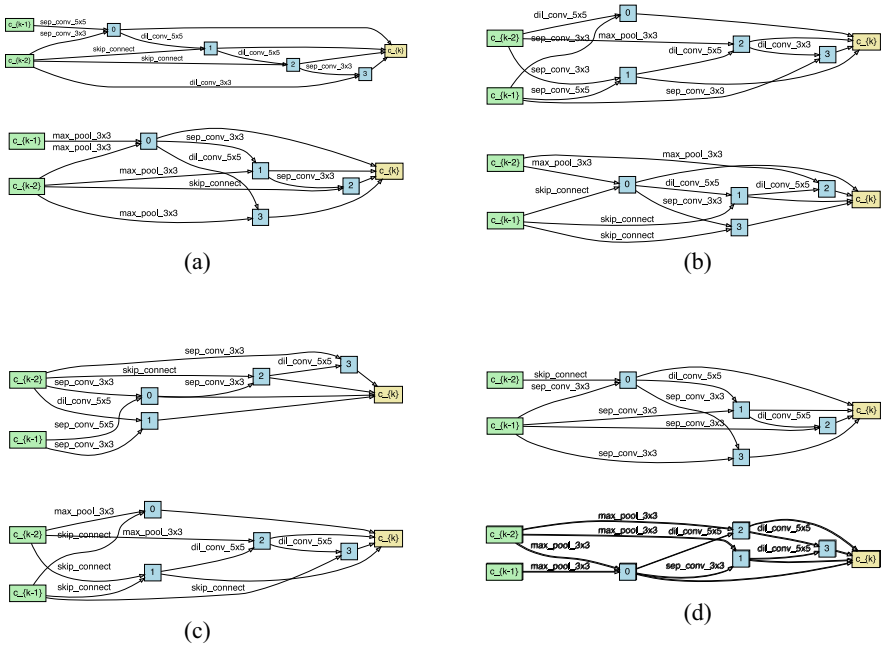


Fig. 11. *Normal and Reduction cells discovered by Semantic-DARTS on CIFAR-100.* (a) ImageNet top-1 accuracy 75.06%. (b) ImageNet top-1 accuracy 75.24%. (c) ImageNet top-1 accuracy 75.44%. (d) ImageNet top-1 accuracy 75.71%.

Table III, where we attain state-of-the-art average validation and test accuracies, specifically 91.49% and 94.30%. The average accuracy remains consistently near the optimum with minimal fluctuations, highlighting the ability of enhanced semantic learning to uncover superior architectures within a compact search space.

TABLE III
COMPARISON RESULTS ON NAS-BENCH-201

Methods	Search Cost	CIFAR-10	
	GPU-Hours	valid Acc.(%)	test Acc.(%)
DARTS(1st)	2.6*	39.77±0.00	54.30±0.00
DARTS(2nd)	8.8*	39.77±0.00	54.30±0.00
GDAS	8.7	89.89±0.08	93.61±0.09
SNAS	-	90.10±1.04	92.77±0.83
DSNAS	-	89.66±0.29	93.08±0.13
PC-DARTS	-	89.96±0.15	93.41±0.30
iDARTS	-	89.86±0.60	93.58±0.32
DARTS-	3.2	91.03±0.44	93.80±0.40
CyDAS	-	91.12±0.44	94.02±0.31
β-DARTS†	2.8*	91.47±0.17	94.23±0.27
ours	2.9*	**91.49±0.12**	**94.30±0.13**
optimal	-	91.61	94.37

Optimal signifies the top-performing network. † represents the replicated results. We compare the validation and test accuracy of the searched architecture. All results were averaged using best-searched architectures from four independent runs. $*$ denotes results were recorded on one RTX 3090 under the same environment.

TABLE IV
COMPARISON OF GENERALIZATION ABILITY ON IMAGENET

Method	Search Cost	Params	FLOPs	Test Acc	
	(GPU-Days)	(M)	(M)	Top1(%)	Top5 (%)
AmoebaNet-C(C10)	3150	6.4	570	75.7	92.4
SNAS(C10)	1.5	4.3	522	72.7	90.8
P-DARTS(C100)	0.3	5.1	577	75.3	92.5
SDARTS-ADV(C10)	1.3	5.4	594	74.8	92.2
DOTS(C10)	0.3	5.2	581	75.7	92.6
DARTS+PT(C10)	0.8	4.6	-	74.5	92.0
β-DARTS(C100)	0.4	5.4	597	75.8	92.9
β-DARTS(C10)	0.4	5.5	609	76.1	93.0
AdaptNAS-S	1.8	5.0	552	74.7	92.2
AdaptNAS-C	2.0	5.3	583	75.8	92.6
MnasNet-92	1667	4.4	388	74.8	92.0
FairDARTS	3	4.3	440	75.6	92.6
PC-DARTS	3.8	5.3	597	75.8	92.7
DOTS	1.3	5.3	596	76.0	92.8
DARTS-	4.5	4.9	467	76.2	93.0
CyDAS	1.7	6.1±0.2	701±32	76.3±0.3	92.9±0.2
ours(C100)	0.2	5.2	592	75.7	92.7
ours(C10)	0.2	5.8	642	**76.5**	93.0

The top three blocks represent candidates searched: 1) on CIFAR-10 (C10) or CIFAR-100 (C100); 2) via samples from CIFAR-10 and portions of ImageNet; 3) directly on ImageNet. As per the ImageNet evaluation routine [56], 14 searched cells are integrated into the final architecture.

B. Generalization Ability of Semantic-DARTS

Settings: We undertake thorough evaluations to assess the generalization potential. Our methodology entails conducting the search on one data set and subsequently evaluating the discovered architectures on others. For example, we search for a model CIFAR-10 and then evaluate it on CIFAR-100 and ImageNet. Conversely, in the case of NAS-Bench-201, we customize the architectures derived from CIFAR-10 specifically for evaluation on CIFAR-100 and ImageNet-16-120. Our implementations on ImageNet adhere to [37], while those on the other data sets are in accordance with the guidelines outlined in Section IV-A.

Results on ImageNet: We adopt the architectures discovered on CIFAR-10 and CIFAR-100 for evaluation on the ImageNet. In Table IV, our method demonstrates highly competitive generalization performance with reduced search costs. The model obtained through CIFAR-10 achieves a new SOTA top-1 accuracy of 76.5%. It outperforms competing models, which either require additional training samples from different domains [24] or necessitate direct search from the ImageNet, incurring significantly higher search costs, ranging from 1.7 to 1667 GPU-Days, as opposed to our 0.2 GPU-Days. The visualizations of the searched architectures are shown in Figs. 10 and 11, respectively. Our method successfully mitigates the performance collapse issue: the obtained architectures are deep and full of nonlinear operations, instead of shallow ones or are full of nonparameter operations. They also achieve good generalization performance on the ImageNet as discussed above, which demonstrates the superiority of the proposed method.

TABLE V
COMPARISON OF GENERALIZATION ABILITY ON CIFAR DATA SETS

Methods	C10 to C100		C100 to C10	
	average	best	average	best
P-DARTS [20]	-	82.80	-	97.38
DARTS+ [25]	-	83.72	-	97.54
β-DARTS [21]	83.76±0.22	-	**97.49±0.07**	-
ours	**83.81±0.44**	**84.23**	97.43±0.11	**97.59**

The notation "C10 to C100" signifies that architectures, initially searched on CIFAR-10, undergo evaluation on CIFAR-100. "Average" and "best" refer respectively to the average and best top-1 accuracy derived from four independent runs.

Results on CIFAR: Table V displays the comparative outcomes across both CIFAR data sets. When assessing architectures originally searched from CIFAR-100 on CIFAR-10, our method demonstrates competitiveness with prior state-of-the-art approaches [21]: Semantic-DARTS achieves an average top-1 accuracy of 97.43% and a best performance of 97.59%. Furthermore, architectures discovered from CIFAR-10 exhibit strong generalization capabilities to CIFAR-100, showcasing state-of-the-art performance: an average top-1 accuracy of 83.81% and a best result of 84.23%. Therefore, by fortifying local representations, Semantic-DARTS significantly enhances generalization capabilities to unprecedented levels on the CIFAR data sets.

Results on NAS-Bench-201: In Table VI, our approach demonstrates strong generalization capabilities within a limited search space. Similar to what we observed in the DARTS search space, the architectures discovered on CIFAR-10 exhibit strong generalization to CIFAR-100, achieving the highest average validation (73.21%) and test accuracy (73.16%). While on ImageNet-16-120, it trails marginally (by 0.05%) compared to the latest state-of-the-art [21] performance on the validation set, nonetheless, it attains the highest test accuracy, surpassing it by 0.42%.

C. Comparing With Self-Supervised Tasks

Settings: The semantic knowledge acquired by Rot., Col., and Jig. fails to address the collapse problem while improving accuracy. We attribute this challenge to the limitation of their optimization within a single classification scope. We verify this by comparing it with their modification that performs classification and self-supervised tasks jointly.

Results: The comparison results are in Table VII. Though the architectures are learned by two tasks, the performance basically maintains the same with their single objective prototype. Besides, Rot. and its classification variants obtain fair performance with relatively few parameters on CIFAR-10,

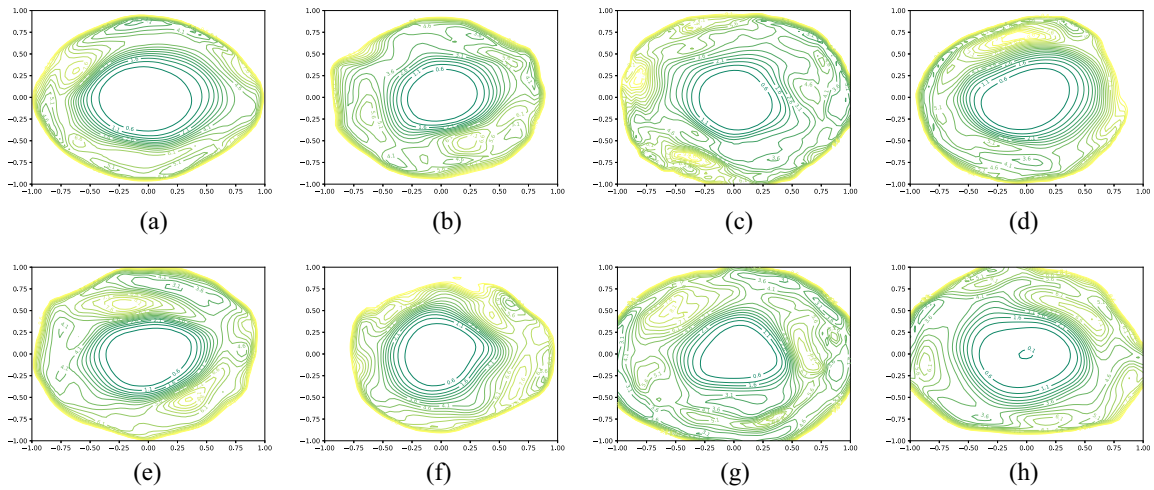


Fig. 12. *Loss landscape of the searched models on the test data set of CIFAR-10.* The top-1 accuracy is reported (%). (a) Rot., 97.47%, $\mathcal{L}_{\text{cls}}$. (b) Col., 97.1%, $\mathcal{L}_{\text{cls}}$. (c) Jig., 96.95%, $\mathcal{L}_{\text{cls}}$. (d) DARTS, 97.00%, $\mathcal{L}_{\text{cls}}$. (e) Cls.+Rot., 97.53%, $\mathcal{L}_{\text{cls}}+\mathcal{L}_{\text{cls}}$. (f) Cls.+Col., 97.47%, $\mathcal{L}_{\text{cls}}+\mathcal{L}_{\text{cls}}$. (g) Cls.+Jig., 97.37%, $\mathcal{L}_{\text{cls}}+\mathcal{L}_{\text{cls}}$. (h) Ours, 97.71%, $\mathcal{L}_{\text{cls}}+\mathcal{L}_{msk}$.

TABLE VI
COMPARISON OF GENERALIZATION ABILITY ON NAS-BENCH-201

Methods	CIFAR-100		ImageNet-16-120	
	valid Acc.(%)	test Acc.(%)	valid Acc.(%)	test Acc.(%)
DARTS(1st)	15.03±0.00	15.61±0.00	16.43±0.00	16.32±0.00
DARTS(2nd)	15.03±0.00	15.61±0.00	16.43±0.00	16.32±0.00
GDAS	71.34±0.04	70.70±0.30	41.59±1.33	41.71±0.98
SNAS	69.69±2.39	69.34±1.98	42.84±1.79	43.16±2.64
DSNAS	30.87±16.40	31.01±16.38	40.61±0.09	41.07±0.09
PC-DARTS	67.12±0.39	67.48±0.89	40.83±0.08	41.31±0.22
iDARTS	70.57±0.24	70.83±0.48	40.38±0.59	40.89±0.68
DARTS-	71.36±1.51	71.53±1.51	44.87±1.46	45.12±0.82
CyDAS	72.12±1.23	71.92±1.30	45.09±0.61	45.51±0.72
β-DARTS†	73.02±0.95	73.10±0.82	**46.22±0.31**	45.92±0.85
ours	**73.21±0.56**	**73.16±0.70**	46.17±0.39	**46.34±0.00**
optimal	73.49	73.51	46.77	47.31

Candidates are obtained on CIFAR-10 and assessed on CIFAR-100 and ImageNet-16-120. Optimal signifies the top-performing network in the search space. † indicates our replicated accuracy. We compare the validation and test accuracy of the searched architecture. All architectures are searched through 4 independent runs.

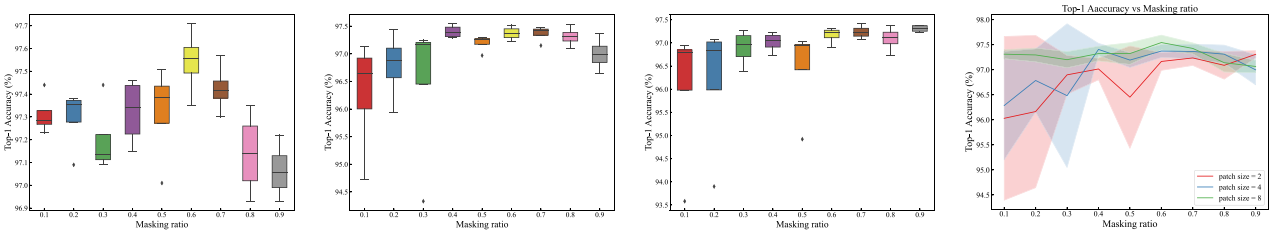
TABLE VII
COMPARISON WITH CLASSIFICATION-BASED SELF-SUPERVISED TASKS

Methods	Search Cost		CIFAR-10		CIFAR-100	
	GPU-Days	Mem.(GB)	Params(M)	Acc(%)	Params(M)	Acc(%)
Rot.(C10)	0.5	16.3*	3.35±0.25	97.39±0.10	3.40±0.25	83.60±0.19
Cls.+ Rot.(C10)	0.5	16.3*	3.44±0.54	97.35±0.12	3.49±0.54	83.57±0.64
Col.(C10)	0.2	8.6	3.65±0.33	97.04±0.07	3.70±0.33	82.48±0.54
Cls.+ Col.(C10)	0.2	8.6	2.37±0.33	97.19±0.25	2.42±0.33	82.12±0.81
Jig.(C10)	0.2	4.8	4.57±0.11	96.93±0.03	4.62±0.11	81.72±0.48
Cls.+ Jig.(C10)	0.2	4.8	4.48±0.25	96.91±0.43	4.53±0.25	81.72±1.11
ours(C10)	0.2	9.6	4.05±0.23	**97.54±0.15**	4.10±0.23	**83.81±0.44**
Rot.(C100)	0.5	16.3*	2.73±0.27	97.21±0.09	2.78±0.27	82.24±0.44
Cls.+ Rot.(C100)	0.5	16.3*	2.19±0.18	97.03±0.07	2.24±0.18	82.10±0.39
Col.(C100)	0.2	8.6	4.11±0.36	97.14±0.10	4.17±0.36	82.25±0.94
Cls.+ Col.(C100)	0.2	8.6	2.09±0.29	97.05±0.22	2.14±0.29	81.73±0.74
Jig.(C100)	0.2	4.8	3.94±0.23	97.00±0.40	3.99±0.23	82.20±0.94
Cls.+ Jig.(C100)	0.2	4.8	4.31±0.29	97.02±0.04	4.36±0.29	82.40±0.39
ours(C100)	0.2	9.6	3.61±0.24	**97.43±0.11**	3.66±0.24	**83.71±0.66**

Cls. signifies CIFAR dataset classification. Rot., Col., and Jig. represent classification with labels from Rotation, Colorization, and Jigsaw puzzle tasks. All experiments were independently executed 4 times on CIFAR-10 (C10) and CIFAR-100 (C100). Search costs are measured under same environment of a single RTX 3090 GPU. $*$ indicates GPU memory measured using batch size 32, default being 64.

but its generalization performance on CIFAR-100 falls largely behind ours and recent DARTS variants due to fewer parameters, Moreover, Rot. costs 2.5 times more search duration and 3.4 times more memory overheads, which is unaffordable and way too high for the CIFAR data sets. This suggests that the combination of classification and self-supervised objectives does not lead to any performance improvement. It further testifies that the single classification paradigm is not enough to learn rich semantic information for architecture search. Yet, through the concurrent execution of classification and MIM tasks, we surpass the performance of classification-based self-supervised objectives and attain a new SOTA. This illustrates that: 1) MIM proves to be a more fitting approach for acquiring valuable semantic insights compared to prior classification-based self-supervised techniques and 2) learning well the local representations as well as their relationship is a promising way to make up the semantic deficiencies of classification-based DARTS, which learns only the abstract class-specific features. We wish to shed light on designing specialized semantic learning methods for NAS to further boost performance.

We further look up the generalization ability of the searched models by an analysis of the loss landscapes [55] in Fig. 12. The proposed method shows a visually flatter landscape in that the loss changes slowly as we move in some directions [Fig. 12(h)]. This contributes to a good generalization ability which corresponds to a higher test accuracy of 97.71%. On the contrary, the minimizer of the classification-based DARTS [Fig. 12(d)] is enclosed by a steep landscape, i.e., fast-changing and dense contour lines, which yields a rather poor generalization performance. The self-supervised counterparts show a different phenomenon. In the landscape of Jig. [Fig. 12(c)], the center minimizer is surrounded by chaos and server nonconvexity, which leads to the lowest generalization performance (96.95%). For Col. [Fig. 12(b)], though the chaotic phenomenon has been alleviated to some extent, the entire landscape still remains tight around the center minimizer, resulting in dramatic changes in the loss and less improvement in performance. The Rot. [Fig. 12(a)] shows a large and flatter center, resulting in better performance than other classification-based self-supervised methods. However, compared to our method, it still shows a sharper loss change in most of the directions. Moreover, even with the introduction of the classification task, the steep loss landscape does not



(a) (b) (c) (d)

Fig. 13. *Ablations investigating various patch sizes and masking ratios.* We find that architectural performance stabilizes at a masking ratio of 0.6 for a patch size of 8, and at 0.4 for a patch size of 4. Conversely, when using a patch size of 2, performance consistently improves as the masking ratio increases. The most effective architectures are identified with a patch size of 8 and a masking ratio of 0.6. Each experiment was conducted independently four times on CIFAR-10. (a) Patch size = 8. (b) Patch size = 4. (c) Patch size = 2. (d) Acc. versus masking ratio.

TABLE VIII
ABLATIONS ON KEY COMPONENTS

Cls.	Rec.	mask	Accuracy(%)	Params(M)
✓			97.00±0.14	3.4
✓		✓	97.05±0.26	4.24±0.23
	✓		84.09±0.00	1.59±0.00
	✓	✓	97.18±0.12	2.46±0.35
✓	✓	✓	**97.54±0.15**	4.05±0.23

Cls.: CIFAR-10 classification. Rec.: clean image reconstruction. Mask: random masking.

TABLE IX
SEARCH EXPENSES ON DARTS SEARCH SPACE

Search	Semantic-DARTS						DARTS	
Dataset	0.4	0.5	0.6	0.7	0.8	0.9	1st	2nd
CIFAR-10	17842	18861	17944	17794	18668	17421	17166	80995
CIFAR-100	18933	18248	18155	18525	17866	17825	16853	78281

We measure the GPU-Seconds of the search process for 1) Semantic-DARTS with patch size 8 and masking ratio ranging from 0.4 to 0.9; 2) 1st and 2nd approximation variants for the original DARTS. Recorded under the same environment on a single RTX 3090 GPU.

significantly alleviate [Fig. 12(e)–(g)]. This may indicate that the Rot., Col., and Jig. can not work well with classification. Besides, our method is the only one that obtains a lower minimum [0.1 in Fig. 12(h)], demonstrating the superior performance of the obtained architecture.

D. Ablation Studies

Exploring Patch Size and Masking Ratio: We commence by exploring how varying patch sizes and masking ratios affect the performance on the CIFAR-10 and CIFAR-100 data sets. For our experiments, we consider patch sizes of 2, 4, and 8, corresponding to 256, 64, and 16 patches for CIFAR images with dimensions of 32. In Fig. 13, observing varying patch sizes (8, 4, and 2), we note that the optimal architectures arise at masking ratios of 0.6, 0.4, and 0.9, respectively. Their average top-1 accuracy rates are recorded at 97.54%, 97.40%, and 97.31%, respectively. When using a smaller patch size of 2, accuracy sees enhancements with an increasing masking ratio, while for larger sizes of 4 and 8, performance plateaus at intermediate masking ratios and starts to decline at higher values. This suggests that small patch sizes may not capture sufficient information. Furthermore, analyzing the comparison of different patch sizes and masking ratios for CIFAR-10 depicted in Fig. 13(d), we note that performance stability is more pronounced at a patch size of 8 compared to sizes 4 and 2, across various masking ratios. This indicates a broader range of effective masking ratios for larger patch sizes. The optimal ones are found at: 1) patch size 8 with a masking ratio of 0.6 for CIFAR-10 and 2) patch size 8 with a masking ratio of 0.7 for CIFAR-100.

In What Scenarios Is the MIM Effective for NAS? We are the pioneers in introducing the MIM task to the domain of NAS. It has become evident that the concurrent learning of class-specific and local representations by injecting additional information yields favorable results for DARTS. When solely engaging the MIM task (as indicated in the fourth row of Table VIII), the average accuracy attains a level of 97.18%, surpassing the original DARTS, which focuses solely on classification. This underscores the potential of the MIM task as a promising approach to extracting local semantics for vision-related tasks. Nevertheless, it continues to fall short of recent DARTS variants, including ours (97.54%). This supports the notion that, in the absence of an extensive search process, the incorporation of additional information is necessary to discover well-performing architectures. Consequently, we opt to simultaneously learn both classification and MIM.

Essential Elements: We further validate the efficacy of pivotal elements by progressively incorporating them into the original DARTS framework, as demonstrated in Table VIII. It is important to highlight that in the second row, performing pure classification on masked images resembles the application of cutout regularization. Yet, the primary distinction arises from our strategy of concealing arbitrary segments as opposed to a square area. This approach yields only modest enhancements. (97.05% compared to the original DARTS's 97.00%). Also, in the fourth line, we take the MIM task as the search objective which is different from the classification objective, the accuracy growth is somewhat significant, but it still cannot compare to modern methods. When concentrating solely on reconstruction for clean images, the resultant networks are mainly influenced by *skip* connections, leading to inferior outcomes (84.09%). This is unsurprising because searching for an architecture to reconstruct a figure is essentially equivalent to learning an identity function. Ultimately, by simultaneously engaging in classification and MIM, we infuse class-specific and local contexts into the search process and achieve SOTA.

Search Expenses: We measure the search expenses on DARTS search space with the CIFAR-10 data set. As shown in Table IX, Semantic-DARTS does not introduce significant search expenses compared to the original DARTS,

due to the efficient decoder design (Section III-C). On the NAS-Bench-201 search space, It attains nearly identical affordability to [21], i.e., 2.9 GPU-Hours in Table III, which is slightly longer than the DARTS (1st) of 2.6 GPU-Hours. Moreover, compared with the self-supervised variants (Table VII), our framework shows its superiority by achieving SOTA results while maintaining low search duration and acceptable memory expenses.

V. Conclusion

In this work, we have tackled the semantic deficiencies of the traditional classification-based DARTS, and validated such semantic deficiencies through a large number of empirical and visualization experiments. By formulating an MIM task, we have enriched the architectures from the inadequate search process with more fine-grained local representations. Extensive experiment results have shown exceptional performance in identifying cutting-edge architectures in both the DARTS and NAS-Bench-201 search domains. These architectural designs have demonstrated remarkable generalization capabilities across CIFAR-10, CIFAR-100, and ImageNet, outperforming earlier DARTS iterations. Our insights and novel search objective design enhance the generalization and robustness of DARTS, making it more applicable in the real-world vision scenarios. In our future work, we will extend the semantic learning paradigm to architectures designed for multimodal inputs, considering the limitations of mobile or edge devices.

References

[1] X. Shen, J. Gao, W. Wu, M. Li, C. Zhou, and W. Zhuang, "Holistic network virtualization and pervasive network intelligence for 6G," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 1, pp. 1–30, 1st Quart., 2022.

[2] Z. M. Fadlullah et al., "State-of-the-art deep learning: Evolving machine intelligence toward tomorrow's intelligent network traffic control systems," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2432–2455, 4th Quart., 2017.

[3] Y. Du, J. Liu, X. Wang, and P. Wang, "SSVEP-based emotion recognition for IoT via multiobjective neural architecture search," *IEEE Internet Things J.*, vol. 9, no. 21, pp. 21432–21443, Nov. 2022.

[4] M. Dissem, M. Amayri, and N. Bouguila, "Neural architecture search for anomaly detection in time-series data of smart buildings: A reinforcement learning approach for optimal autoencoder design," *IEEE Internet Things J.*, vol. 11, no. 10, pp. 18059–18073, May 2024.

[5] P. Li et al., "Filling the missing: Exploring generative AI for enhanced federated learning over heterogeneous mobile edge devices," *IEEE Trans. Mobile Comput.*, vol. 23, no. 10, pp. 10001–10015, Oct. 2024.

[6] T. K. Rodrigues, S. Verma, Y. Kawamoto, N. Kato, M. M. Fouda, and M. Ismail, "Smart handover with predicted user behavior using convolutional neural networks for WiGig systems," *IEEE Netw.*, vol. 38, no. 4, pp. 190–196, Jul. 2024.

[7] H. Lee, H. Ko, C. Bae, and S. Pack, "Accelerating convolutional neural network inference in split computing: An in-network computing approach," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, 2024, pp. 773–776.

[8] S. Mo, R. Salakhutdinov, L. P. Morency, and P. P. Liang, "IoT-LM: Large multisensory language models for the Internet of Things," 2024, *arXiv:2407.09801*.

[9] Z. Khan et al., "Diabetic retinopathy detection using VGG-NIN a deep learning architecture," *IEEE Access*, vol. 9, pp. 61408–61416, 2021.

[10] J. Xue, K. Yu, T. Zhang, H. Zhou, L. Zhao, and X. Shen, "Cooperative deep reinforcement learning enabled power allocation for packet duplication URLLC in multi-connectivity vehicular networks," *IEEE Trans. Mobile Comput.*, vol. 23, no. 8, pp. 8143–8157, Aug. 2024.

[11] T. Elsken, J. H. Metzen, and F. Hutter, "Neural architecture search: A survey," *J. Mach. Learn. Res.*, vol. 20, no. 1, pp. 1997–2017, 2021.

[12] B. Zoph and Q. Le, "Neural architecture search with reinforcement learning," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2017, pp. 1–16.

[13] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 1–14.

[14] H. Liu, K. Simonyan, O. Vinyals, C. Fernando, and K. Kavukcuoglu, "Hierarchical representations for efficient architecture search," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–12.

[15] E. Real, A. Aggarwal, Y. Huang, and Q. V. Le, "Regularized evolution for image classifier architecture search," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, vol. 33, 2019, pp. 4780–4789.

[16] E. Real et al., "Large-scale evolution of image classifiers," in *Proc. 34th Int. Conf. Mach. Learn. (ICML)*, vol. 70, 2017, pp. 2902–2911.

[17] H. Liu, K. Simonyan, and Y. Yang, "DARTS: Differentiable architecture search," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2019, pp. 1–13.

[18] C. Zhang, X. Yuan, Q. Zhang, G. Zhu, L. Cheng, and N. Zhang, "Toward tailored models on private AIoT devices: Federated direct neural architecture search," *IEEE Internet Things J.*, vol. 9, no. 18, pp. 17309–17322, Sep. 2022.

[19] H. Cai, L. Zhu, and S. Han, "ProxylessNAS: Direct neural architecture search on target task and hardware," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2019, pp. 1–13.

[20] X. Chen, L. Xie, J. Wu, and Q. Tian, "Progressive differentiable architecture search: Bridging the depth gap between search and evaluation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2019, pp. 1294–1303.

[21] P. Ye, B. Li, Y. Li, T. Chen, J. Fan, and W. Ouyang, "b-DARTS: Beta-decay regularization for differentiable architecture search," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 10874–10883.

[22] A. Zela, T. Elsken, T. Saikia, Y. Marrakchi, T. Brox, and F. Hutter, "Understanding and robustifying differentiable architecture search," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020, pp. 1–28.

[23] Y. Xue and J. Qin, "Partial connection based on channel attention for differentiable neural architecture search," *IEEE Trans. Ind. Informat.*, vol. 19, no. 5, pp. 6804–6813, May 2023.

[24] Y. Li, Z. Yang, Y. Wang, and C. Xu, "Adapting neural architectures between domains," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 33, 2020, pp. 789–798.

[25] H. Liang et al., "Darts+: Improved differentiable architecture search with early stopping," 2019, *arXiv:1909.06035*.

[26] X. Chen and C.-J. Hsieh, "Stabilizing differentiable architecture search via perturbation-based Regularization," in *Proc. 37th Int. Conf. Mach. Learn. (ICML)*, vol. 119, 2020, pp. 1554–1565.

[27] X. Chu, X. Wang, B. Zhang, S. Lu, X. Wei, and J. Yan, "DARTS-: Robustly stepping out of performance collapse without indicators," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2021, pp. 1–22.

[28] X. Chu, T. Zhou, B. Zhang, and J. Li, "Fair DARTS: Eliminating unfair advantages in differentiable architecture search," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 465–480.

[29] R. Wang, M. Cheng, X. Chen, X. Tang, and C.-J. Hsieh, "Rethinking architecture selection in differentiable NAS," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2021, pp. 1–18.

[30] Y. Shu, W. Wang, and S. Cai, "Understanding architectures learnt by cell-based neural architecture search," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020, pp. 1–21.

[31] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2015, pp. 1422–1430.

[32] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative Localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 2921–2929.

[33] H. Bao, L. Dong, S. Piao, and F. Wei, "BEiT: BERT pre-training of image transformers," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2022, pp. 1–18.

[34] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 16000–16009.

[35] A. Burrello, M. Risso, B. A. Motetti, E. Macii, L. Benini, and D. J. Pagliari, "Enhancing neural architecture search with multiple hardware constraints for deep learning model deployment on tiny IoT devices," *IEEE Trans. Emerg. Topics Comput.*, vol. 12, no. 3, pp. 780–794, Jul.–Sep. 2024.

[36] X. Zhang, Y. Wang, H. Huang, Y. Lin, H. Zhao, and G. Gui, "Few-shot automatic modulation classification using architecture search and knowledge transfer in radar-communication coexistence scenarios," *IEEE Internet Things J.*, early access, Jul. 4, 2024, doi: 10.1109/JIOT.2024.3423018.

[37] X. Zhang, P. Hou, X. Zhang, and J. Sun, "Neural architecture search with random labels," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021, pp. 10907–10916.
[38] C. Li et al., "BossNAS: Exploring hybrid CNN-transformers with block-wisely self-supervised neural architecture search," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 12281–12291.
[39] C. Liu, P. Dollár, K. He, R. Girshick, A. Yuille, and S. Xie, "Are labels necessary for neural architecture search?" in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 798–813.
[40] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 1–12.
[41] M. Chen et al., "Generative pretraining from pixels," in *Proc. 37th Int. Conf. Mach. Learn. (ICML)*, vol. 119, 2020, pp. 1691–1703.
[42] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2021, pp. 1–22.
[43] Z. Xie et al., "SimMIM: A simple framework for masked image modeling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 9653–9663.
[44] Y. Wei, A. Gupta, and P. Morgado, "Towards latent masked image modeling for self-supervised visual representation learning," 2024, *arXiv:2407.15837*.
[45] X. Ma, C. Liu, C. Xie, L. Ye, Y. Deng, and X. Ji, "Disjoint masking with joint distillation for efficient masked image modeling," *IEEE Trans. Multimedia*, vol. 26, pp. 3077–3087, 2024.
[46] S. Woo et al., "ConvNeXt V2: Co-designing and scaling ConvNets with masked autoencoders," 2023, *arXiv:2301.00808*.
[47] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 11976–11986.
[48] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–16 .
[49] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 649–666.
[50] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving Jigsaw puzzles," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 69–84.
[51] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 1–10.
[52] P. Jiang, C. Zhang, Q. Hou, M. Cheng, and Y. Wei, "LayerCAM: Exploring hierarchical class activation maps for localization," *IEEE Trans. Image Process.*, vol. 30, pp. 5875–5888, 2021.
[53] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 618–626.
[54] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, 2018, pp. 839–847.
[55] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, "Visualizing the loss landscape of neural nets," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 31, 2018, pp. 1–11.
[56] Y. Xu et al., "PC-DARTS: Partial channel connections for memory-efficient architecture search," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020, pp. 1–13.

Bicheng Guo (Member, IEEE) received the B.S. degree from Chongqing University, Chongqing, China, in 2017, and the M.S. degree from Wuhan University, Wuhan, China, in 2020. He is currently pursuing the Ph.D. degree with the College of Control Science and Engineering, Zhejiang University, Hangzhou, China.

From June 2023 to June 2024, he was a Visiting Student with the University of Waterloo, Waterloo, ON, Canada. His research interests include pattern recognition, neural architecture search, and autonomous driving.

Shibo He (Senior Member, IEEE) received the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 2012.

He is currently a Professor with Zhejiang University. He was an Associate Research Scientist from March 2014 to May 2014 and a Postdoctoral Scholar from May 2012 to February 2014 with Arizona State University, Tempe, AZ, USA. From November 2010 to November 2011, he was a Visiting Scholar with the University of Waterloo, Waterloo, ON, Canada. His research interests include the IoT, crowdsensing, and big data analysis.

Dr. He served as a Symposium Co-Chair for IEEE ICC 2017, the Finance Registration Chair for ACM MobiHoc 2015, and the Technical Program Committee Co-Chair for IEEE ScalCom 2014. He serves on the Editorial Board for several journals, including IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING.

Miaojing Shi (Senior Member, IEEE) received the Ph.D. degree from Peking University, Beijing, China, in 2015.

He was engaged with a joint Ph.D. program with the University of Oxford, Oxford, U.K, and INRIA, Rennes, France for a year. He held a post-doctoral position with the University of Edinburgh, Edinburgh, Scotland, and was a Research Scientist with INRIA. From 2020 to 2022, he has been a Lecturer/Senior Lecturer with the Department of Informatics, King's College London, London, U.K. Since 2023, he has been a Full Professor with Tongji University, Shanghai, China, and a Visiting Senior Lecturer with King's College London. He has authored or co-authored over 70 papers in prestigious journals, such as IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE and PROCEEDINGS OF THE IEEE, as well as the top AI conferences, including CVPR, ICCV, and NeurIPS. His current research focus is on visual learning with few data, vision-language learning, and medical imaging analysis.

Kaicheng Yu received the B.S. degree (First-Class Hons.) from the University of Hong Kong, Hong Kong, in 2016, and the Ph.D. degree from EPFL, Lausanne, Switzerland.

He is an Assistant Professor with Westlake University, Hangzhou, China, and founded the Autonomous Intelligence Lab in 2023. Prior to that, he worked with the research institutes, such as Intel Lab, Beijing, China, and Alibaba Damo Academy, Hangzhou. He received Qualcomm Innovation Fellowship Europe in 2019. He published on the top conference venues like CVPR, ICCV, ECCV, ICML, ICLR, and NeurIPS, and journals like TPAMI. His research interests cover broader areas of computer vision and machine learning methods for the real-world.

Jiming Chen (Fellow, IEEE) received the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 2005.

He is currently a Professor with the Department of Control Science and Engineering, the Deputy Director of the State Key Laboratory of Industrial Control Technology, and the Director of the Institute of Industrial Process Control, Zhejiang University. His research interests include Internet of Things, networked control, and wireless networks.

Dr. Chen was a recipient of the Seventh IEEE ComSoc Asia/Pacific Outstanding Paper Award, the JSPS Invitation Fellowship, and the IEEE ComSoc AP Outstanding Young Researcher Award. He serves as the General Co-Chair for IEEE RTCSA 2019, IEEE Datacom 2019, and IEEE PST 2020. He serves on the editorial boards for multiple IEEE Transactions. He is an IEEE VTS Distinguished Lecturer and a Fellow of CAA.

Xuemin (Sherman) Shen (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990.

He is a University Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. His research focuses on network resource management, wireless network security, Internet of Things, 5G and beyond, and vehicular networks.

Dr. Shen received the West Lake Friendship Award from Zhejiang Province in 2023, the President's Excellence in Research from the University of Waterloo in 2022, the Canadian Award for Telecommunications Research from the Canadian Society of Information Theory in 2021, the R.A. Fessenden Award in 2019 from IEEE, Canada, the Award of Merit from the Federation of Chinese Canadian Professionals, Ontario, in 2019, the James Evans Avant Garde Award in 2018 from the IEEE Vehicular Technology Society, the Joseph LoCicero Award in 2015, the Education Award in 2017 from the IEEE Communications Society (ComSoc), and the Technical Recognition Award from Wireless Communications Technical Committee in 2019 and AHSN Technical Committee in 2013. He has also received the Excellent Graduate Supervision Award in 2006 from the University of Waterloo and the Premier's Research Excellence Award in 2003 from the Province of Ontario, Canada. He serves/served as the General Chair for the 6G Global Conference 2023 and ACM Mobihoc 2015; the Technical Program Committee Chair/Co-Chair for IEEE Globecom 2024, 2016, and 2007, IEEE Infocom 2014, and IEEE VTC 2010 Fall; and the Chair for the IEEE ComSoc Technical Committee on Wireless Communications. He is the President of the IEEE ComSoc. He was the Vice President for Technical and Educational Activities, the Vice President for Publications, Member-at-Large on the Board of Governors, the Chair of the Distinguished Lecturer Selection Committee, and a Member of the IEEE Fellow Selection Committee of the ComSoc. He served as the Editor-in-Chief for IEEE INTERNET OF THINGS JOURNAL, IEEE NETWORK, and IET COMMUNICATIONS.