MODULAR ENHANCEMENT OF ONEFORMER VIA CBAM-AUGMENTED FASTERVIT FOR SEMANTIC SEGMENTATION

MEEHIR MAHENDRA MHATRE[1], LI ZHANG[1], STUART ROTHNIE[2]

[1]Department of Computer Science, Royal Holloway, University of London, Surrey, TW20 0EX, UK
[2]Rubberatkins Ltd, Aberdeen Energy Park, Aberdeen, AB23 8GW, UK

Abstract:

Transformer-based segmentation models like OneFormer offer strong performance but remain computationally expensive for real-time and resource-limited settings. In this preliminary work, we propose a modular enhancement of OneFormer by integrating the efficient FasterViT backbone and lightweight CBAM (Convolutional Block Attention Modules). FasterViT introduces Hierarchical Attention for scalable, high-resolution processing, while CBAM enhances feature saliency through spatial and channel-wise attention strategies. We apply CBAM after the initial convolutional stage of the encoder and at the pixel decoder's input projections without altering core architecture preserving pretraining compatibility. Evaluated on the Cityscapes dataset, our approach achieves a promising 77.18 % mean IoU across 19 classes despite limited training resources. These results indicate that hybrid transformers combined with modular attention can provide an effective path toward lightweight, scalable segmentation models.

Keywords:

OneFormer; FasterViT; CBAM; Mask2Former; Vision Transformers

1. Introduction

Semantic segmentation is a fundamental task in computer vision that involves assigning a class label to each pixel in an image. It plays a vital role in applications such as autonomous driving, robotics, and urban planning, where detailed scene understanding is crucial for safe and efficient decision-making [1], [15]. Recent advancements in transformer-based architectures have significantly improved the accuracy of segmentation models. Unified architectures like OneFormer have gained attention for their ability to handle semantic, instance, and panoptic segmentation within a single framework, using shared pixel- and mask-level decoders.

Despite their effectiveness, these models come with a high computational cost. Large-scale transformers demand extensive memory and processing power, which limits their applicability in real-time or edge-device scenarios. As the demand for deployable vision systems grows, finding the right balance between accuracy and efficiency becomes increasingly critical.

1.1. Research Motivations

While OneFormer achieves state-of-the-art results, its use of heavy pixel encoders and transformer decoders can make deployment on lightweight or embedded systems challenging. At the same time, attention mechanisms like CBAM (Convolutional Block Attention Module) have shown promise in enhancing model performance with minimal overhead by refining spatial and channel-wise feature selection.

To address this gap, we explore whether modular and lightweight enhancements can improve segmentation performance without significantly increasing computational cost. In this work, we propose an architecture that integrates the FasterViT backbone a high-throughput, hybrid vision transformer with CBAM modules placed strategically within both the encoder and decoder stages. This approach maintains compatibility with existing pretraining pipelines while offering a flexible path to more efficient segmentation.

2. Related Work

2.1. Vision Transformer and Oneformer

Vision Transformers (ViTs) [6] apply the transformer architecture to images by splitting an image into patches and processing them with self-attention. Unlike convolutional networks, ViTs can model global context across the entire image [14]. They have achieved state-of-the-art performance on many vision tasks including classification, object detection, and semantic segmentation [14] because self-attention captures long-range dependencies that are important for understanding scenes. In semantic segmentation, for example, ViTs can integrate information from distant parts of the image to better

delineate objects. However, pure ViT models tend to be extremely large and data hungry. Without the built-in spatial biases of CNNs, they often require very large datasets and many parameters to train effectively [5]. In practice, ViTs can have hundreds of millions of parameters (e.g. ViT-Huge has ~632M [6]), leading to high computational and memory costs.

OneFormer, a universal architecture for semantic, instance, and panoptic segmentation. OneFormer uses a single transformer with task-specific conditioning (via text prompts) to handle all three tasks in one model [7]. It is trained once on a mix of segmentation labels, achieving state-of-the-art results on ADE20K [17], CityScapes [3], Mapillary Vistas [13] and COCO [10]. Remarkably, a single OneFormer model outperforms specialized Mask2Former [2] models on each task, despite the specialized models being trained separately with three times the resources [7]. In summary, OneFormer's unified design yields excellent accuracy across tasks. Nevertheless, like other ViT-based segmentation models, OneFormer is very heavy and overparameterized. Such models incur high compute costs and latency, which can limit their use in practice [7].

2.2. Efficient Backbones: FasterViT

To address the efficiency challenges of large ViTs, hybrid CNN-Transformer backbones have been proposed. FasterViT is one such architecture designed for high throughput and efficiency [5]. FasterViT combines standard convolutional blocks in early layers with transformer blocks later in the network. Specifically, it uses residual CNN layers in the high-resolution stages (stages 1 and 2) and switches to transformer blocks in lower-resolution, higher-level stages [5]. This hybrid design allows the model to quickly extract fine-grained features with convolutions and then capture global context with transformers.

A key innovation of FasterViT is its Hierarchical Attention (HAT) mechanism [5]. In each transformer block, FasterViT interleaves local windowed self-attention with a global "carrier token" attention. Concretely, local self-attention is computed within small windows, while a set of learned carrier tokens summarize and propagate global information across windows [5]. By decomposing full self-attention into hierarchical local and global stages, HAT captures long-range dependencies at much lower cost than naive global attention. The complexity of hierarchical attention grows roughly linearly with image resolution, making it efficient for high-res inputs [5].

Empirically, FasterViT achieves a superior accuracy-throughput trade-off on standard vision benchmarks. When tested on tasks like classification, object detection, and segmentation, FasterViT establishes a new Pareto-optimal frontier of accuracy vs. speed [5]. FasterViT yields state-of-the-art throughput (images per second) for a given accuracy on ImageNet [9], COCO [10], and ADE20K [17]. Its high efficiency and strong performance make FasterViT especially well-suited for resource-sensitive or real-time applications.

2.3. Attention Modules: CBAM

The Convolutional Block Attention Module (CBAM) is a lightweight CNN attention module that refines intermediate feature maps via both channel and spatial attention [5]. CBAM applies two sequential attention steps to any convolutional feature map:

Channel attention: The module first aggregates spatial information (e.g. using global average and max pooling) to compute a 1D attention vector that weighs each feature channel by importance [5]. This operation enables the network to emphasize 'what' feature channels are most informative for the task.

Spatial attention: Next, CBAM infers a 2D spatial attention map by pooling along the channel axis and applying a small convolution. The resulting mask highlights 'where' in the spatial map the important features lie [5].

The channel and spatial attention masks are multiplied with the original feature map to adaptively refine it. Because CBAM is very lightweight, it adds negligible computational overhead and can be easily inserted into any CNN layer [5]. In practice, CBAM consistently improves CNN performance: for example, adding CBAM to standard classification or detection networks yields higher accuracy on ImageNet [7] and COCO [10] [8].

3. The Proposed Architecture

Our architecture extends OneFormer, a universal segmentation framework that handles semantic, instance, and panoptic segmentation using a shared pipeline. We modify the pixel encoder to use FasterViT for efficient feature extraction, and we enhance salient representation learning through CBAM at both early and late stages. To preserve compatibility with OneFormer's architecture, we use lightweight adaptation/projection layers (adapter_0, projection_1, projection_2, projection_3) that ensure our outputs match the pixel decoder's expected input shapes.

3.1. Overall Architecture

We retain OneFormer's pixel decoder, transformer decoder, and task-conditioned mask decoder, but replace

the original backbone (Swin [11] and ConvNeXt [12]) with FasterViT, a hybrid CNN–transformer model designed for fast, high-resolution feature extraction [5]. FasterViT uses convolutional residual blocks in early layers (Level 0–1) and a transformer-style hierarchical attention mechanism in later stages (Level 2–3). This configuration yields high-quality multiscale features with improved speed and lower parameter counts, making it suitable for resource constrained segmentation scenarios.

We use faster_vit_4_any_res with default config but with resolution changed to [512,1024] which is the input image size. We build a wrapper module FasterVITBackbone around FasterViT that extracts four hierarchical feature maps—res2 to res5—aligned with OneFormer's expected inputs. These are passed to the pixel decoder, which upsamples and fuses them through multi-scale deformable attention [16] before handing them off to the shared transformer decoder and task-specific segmentation heads.

To ensure the FasterViT outputs are shape-compatible with OneFormer's decoder, we introduce the following adapter layers:

- adapter_0: Converts Level 0 CBAM-refined features from 392 → 192 channels, producing res2 at 80×80 resolution.
- projection_1: Projects Level 1 outputs from 784 → 384 channels, yielding res3.
- projection_2: Projects Level 2 outputs from 1568 → 768 channels, yielding res4.
- projection_3: Projects Level 3 outputs (also 1568 channels) to 1536 channels, yielding res5.

Each projection module includes a 1×1 convolution, GroupNorm, and ReLU, like OneFormer's original input_proj stages. These ensure the intermediate feature maps from FasterViT can be directly processed by OneFormer's pixel decoder without retraining it or breaking compatibility.

3.2. CBAM Integration

We augment the encoder with CBAM, a compact attention block that applies channel and spatial attention in sequence to adaptively refine feature maps. CBAM is designed to be modular and lightweight, making it suitable for plug-in use in pretrained architectures [8]. In our case, we strategically insert CBAM in two parts of the network:

We apply CBAM after FasterViT's Level 0 output, before feeding into adapter_1. At this early stage, features are high resolution and contain detailed spatial information. CBAM enhances these by emphasizing informative edges and suppressing background noise, which improves low-level saliency crucial for segmentation. The output of CBAM is projected to 192 channels by adapter_1, forming the res2 map.

We further add CBAMs just after GroupNorm inside each input_proj module (input_proj1–3). These modules project and normalize features before they are processed by the deformable attention encoder [16] in the pixel decoder. By refining feature saliency at this stage, we ensure that the transformer decoder receives cleaner and more discriminative multi-scale features.

These attention modules help the network learn what and where to focus—early CBAM improves local feature saliency, while later CBAMs help sharpen higher-level semantic activations across the fused feature hierarchy.

3.3. Design Principles

We followed three guiding principles in our architecture design:

- Modular: CBAM blocks are added as plug-in attention modules at specific stages of the encoder and decoder. They require no changes to OneFormer's pixel decoder, transformer decoder, or task-conditioned segmentation heads. Similarly, we preserve FasterViT's native architecture by wrapping it in a thin module that outputs four intermediate feature maps aligned with OneFormer expectation. FasterViT's stage-based design (Levels 0–3) aligns cleanly with OneFormer's multi-scale feature fusion requirements. This enables us to adapt FasterViT for segmentation without modifying internal attention or convolutional layers. Together, CBAM and FasterViT promote a modular architecture where components can be swapped or reused independently.
- Lightweight: Each CBAM consists of only a few small convolutional layers and pooling operations. CBAM is designed to be "lightweight" with negligible computational overhead [5]. Thus, inserting CBAM adds only a tiny fraction of extra parameters and FLOPs. This satisfies our requirement that the augmented model remains efficient and does not dramatically increase inference cost. FasterViT also prioritizes efficiency by combining early-stage convolutions (Levels 0–1) for fast spatial modeling with late-stage hierarchical transformers (Levels 2–3) for global context. This hybrid approach enables faster forward passes and lower memory usage compared to pure transformer backbones like Swin-L [11] or ConvNext-L [12], especially at high resolutions.
- Effective: The two-stage placement of CBAM (early and late) ensures the network focuses on informative features at multiple levels. Prior work demonstrates that

adding CBAM to models yields "consistent improvements in classification and detection" accuracy [5]. By analogy, we expect our segmentation performance to benefit since CBAM encourages the network to learn what and where to attend. In particular, the early CBAM steers the backbone to emphasize meaningful low-level patterns, while the decoder-stage CBAM refines the multi-scale features that the transformer decoder uses. FasterViT's hybrid structure complements this by producing rich, high-resolution features suitable for dense prediction. Its use of multi-head attention windows, convolutional residuals, and layer scaling helps stabilize training and improve spatial awareness, especially under low-data or low-resolution regimes.

4. Experiments

Our architecture extends OneFormer, a universal segmentation framework that handles semantic, instance, and panoptic segmentation using a shared pipeline. We modify the pixel encoder to use FasterViT for efficient feature extraction, and we enhance salient representation learning through CBAM at both early and late stages. To preserve compatibility with OneFormer's architecture, we use lightweight adaptation/projection layers (adapter_0, projection_1, projection_2, projection_3) that ensure our outputs match the pixel decoder's expected input shapes/

4.1. Dataset

We evaluate our model on the Cityscapes dataset [3], which includes high-resolution (1024×2048) images of urban street scenes from 50 cities. It contains 5,000 finely annotated images, split into training (2,975), validation (500), and test (1,525) sets. Although 30 classes are annotated, only 19 semantic classes are used for benchmark evaluation (e.g., road, sidewalk, building, person, car, etc.). This dataset is standard for evaluating urban scene understanding.

4.2. Training Setup and Evaluation Metrics

Our model was trained from scratch due to FasterViT's architectural sensitivity to input shape. The key training configuration is presented in Table 1.

For evaluation we use standard mean Intersection over Union (mIoU) [4] on the Cityscapes validation set. We also report per-class IoUs in Table 2.

TABLE 1. Model configuration

Configuration	*Value*
Input Image Size	512 x 1024
Hidden Dimension	256
Batch Size	8
Base Learning Rate	0.0001
Optimizer	AdamW
LR Schedule	Polynomial Decay (power=0.9)
LR Multipliers	Backbone: 0.1×, Others: 1.0×
Minimum LR	1e-6
Pretraining	None

TABLE 2. Evaluation metrics

Class	*IoU*
Road	0.9803
Sidewalk	0.8460
Building	0.9210
Wall	0.5546
Fence	0.5804
Pole	0.6339
Traffic Light	0.7039
Traffic Sign	0.7845
Vegetation	0.9238
Terrain	0.6519
Sky	0.9411
Person	0.8297
Rider	0.6180
Car	0.9481
Truck	0.7024
Bus	0.8860
Train	0.7564
Motorcycle	0.6298
Bicycle	0.7726
Mean IoU	**0.7718**

4.3. Results and Comparison

We evaluate our model on the Cityscapes dataset [3], which includes high-resolution (1024×2048) images of urban street scenes from 50 cities. It contains 5,000 finely annotated images, split into training (2,975), validation

(500), and test (1,525) sets. Although 30 classes are annotated, only 19 semantic classes are used for benchmark evaluation (e.g., road, sidewalk, building, person, car, etc.). Table 3 illustrates comparison with pre-trained models.

TABLE 3. Evaluation metrics

Model	*Params*	*Pretrained*	*Input Size*	*Hidden Dim*	*mIoU (%)*
Ours (FasterViT + CBAM)	**386.8 M**	NA (training from scratch)	512 × 1024	256	**77.18**
OneFormer (Swin-L)	219 M	ImageNet-22K	1024 × 2048	1024	83.0
OneFormer (ConvNeXt-L [12])	220 M	ImageNet-22K	1024 × 2048	1024	83.0
OneFormer (ConvNeXt-XL [12])	372 M	ImageNet-22K	1024 × 2048	1024	83.6

4. Conclusion

In this research, we proposed a modular and efficient extension of the OneFormer segmentation framework by integrating the FasterViT backbone and lightweight CBAM attention modules. Our architecture maintains compatibility with OneFormer's decoder while introducing improvements in feature extraction and saliency refinement. The use of FasterViT ensures a strong trade-off between speed and accuracy, and the strategic placement of CBAM enhances attention to relevant spatial and semantic features.

We trained the model from scratch without pretraining, used a smaller hidden dimension (256), and operated on reduced input resolution (512×1024). These preliminary settings were chosen to balance performance and cost. Even under these constrained settings, our model achieved a competitive mean IoU of 77.18% on the Cityscapes dataset. We believe the performance can be significantly improved by adopting larger configurations—such as using a hidden dimension of 1024 and full-resolution inputs (1024×2048)—and leveraging pretrained weights. These enhancements, along with evolutionary algorithm-based hyper-parameter optimisation [18-36], will be exploited in future work to fully realize the capability of our architecture.

References

[1] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 40, No. 4, pp. 834–848, Apr. 2018.

[2] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar, "Masked-attention Mask Transformer for Universal Image Segmentation," *Proceedings of the IEEE Conference on CVPR*, pp. 1280–1289, Jun. 2022.

[3] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele, "The Cityscapes Dataset for Semantic Urban Scene Understanding," *Proceedings of IEEE Conference on CVPR*, pp. 3213–3223, Jun. 2016.

[4] Mark Everingham, SM Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman, "The Pascal Visual Object Classes Challenge: A Retrospective," *International Journal of Computer Vision*, Vol. 111, pp. 98–136, Jan. 2015.

[5] Amirhossein Habibian Hatamizadeh, Greg Heinrich, Hien Yin, Alexey A. Tao, Jose M. Alvarez, Jan Kautz, and Pavlo Molchanov, "FasterViT: Fast Vision Transformers with Hierarchical Attention," *arXiv preprint*, 2023.

[6] Alexey Dosovitskiy, et al., "An Image is Worth 16×16 Words: Transformers for Image Recognition at Scale," *Proceedings of the International Conference on Learning Representations*, May 2021.

[7] Jaskirat Singh Jain, Jingyu Li, Meng Tang Chiu, Ali Hassani, Nikita Orlov, and Humphrey Shi, "OneFormer: One Transformer to Rule Universal Image Segmentation," *Proceedings of the IEEE Conference on CVPR*, pp. 10552–10561, Jun. 2022.

[8] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon, "CBAM: Convolutional Block Attention Module," *Proceedings of the European Conference on Computer Vision*, pp. 3–19, Sep. 2018.

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Communications of the ACM*, Vol. 60, No. 6, pp. 84–90, Jun. 2017.

[10] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick, "Microsoft COCO: Common Objects in Context," *Lecture Notes in Computer Science*, Vol. 8693, pp. 740–755, 2014.

[11] Ziwei Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo, "Swin

Transformer: Hierarchical Vision Transformer using Shifted Windows," *Proceedings of ICCV*, 2021.

[12] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie, "A ConvNet for the 2020s," *Proceedings of the IEEE Conference on CVPR*, pp. 11966–11976, Jun. 2022.

[13] Gernot Neuhold, Tobias Ollmann, Samuel Rota Bulò, and Peter Kontschieder, "The Mapillary Vistas Dataset for Semantic Understanding of Street Scenes," *Proceedings of ICCV*, pp. 5000–5009, Oct. 2017.

[14] Shaibal Saha and Lirong Xu, "Vision Transformers on the Edge: A Comprehensive Survey of Model Compression and Acceleration Strategies," *Neurocomputing*, Vol. 643, p. 130417, 2025.

[15] Evan Shelhamer, Jonathan Long, and Trevor Darrell, "Fully Convolutional Networks for Semantic Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 39, 2017.

[16] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai, "Deformable DETR: Deformable Transformers for End-to-End Object Detection," *Proceedings of ICLR*, May 2021.

[17] Zhou B., Zhao H., Puig X., Fidler S., Barriuso A., Torralba A., "Semantic Understanding of Scenes through the ADE20K Dataset", *International Journal of Computer Vision*, Vol. 127, pp. 302–321, 2016.

[18] Slade, S., Zhang, L., Huang, H., Asadi, H., Lim, C.P., Yu, Y., Zhao, D., Lin, H. and Gao, R., 2023. Neural inference search for multiloss segmentation models. *IEEE Transactions on Neural Networks and Learning Systems*.

[19] Zhang, L., Slade, S., Lim, C.P., Asadi, H., Nahavandi, S., Huang, H. and Ruan, H., 2023. Semantic segmentation using Firefly Algorithm-based evolving ensemble deep neural networks. *Knowledge-Based Systems*, *277*, p.110828.

[20] L. Zhang and C.P. Lim, Intelligent optic disc segmentation using improved particle swarm optimization and evolving ensemble models, *Applied Soft Computing*, vol. 92, p. 106328, 2020.

[21] L. Zhang, W. Srisukkham, S. C. Neoh, C.P. Lim, and D. Pandit, Classifier ensemble reduction using a modified firefly algorithm: An empirical evaluation, *Expert Systems with Applications*, vol. 93, 2018.

[22] L. Zhang, C.P. Lim, Y. Yu, and M. Jiang, Sound classification using evolving ensemble models and particle swarm optimization, *Applied soft computing*, vol. 116, p. 108322, 2022.

[23] L. Zhang, D. Zhao, C.P. Lim, H. Asadi, H. Huang, Y. Yu, and R. Gao, Video deepfake classification using particle swarm optimization-based evolving ensemble models, *Knowledge-Based Systems*, p.111461, 2024.

[24] H. Xie, L. Zhang, C.P. Lim, Y. Yu, C. Liu, H. Liu, and J. Walters, Improving k-means clustering with enhanced firefly algorithms, *Applied Soft Computing*, vol. 84, p. 105763, 2019.

[25] L. Cunha, L. Zhang, B. Sowan, C.P. Lim, and Y. Kong, Video deepfake detection using particle swarm optimization improved deep neural networks, *Neural Computing and Applications*, vol. 36, no. 15, 2024.

[26] L. Zhang, C.P. Lim, and C. Liu, Enhanced bare-bones particle swarm optimization based evolving deep neural networks, *Expert systems with applications*, 2023.

[27] S. Slade, L. Zhang, Y. Yu, and C.P. Lim, An evolving ensemble model of multi-stream convolutional neural networks for human action recognition in still images, *Neural computing and applications*, vol. 34, 2022.

[28] Choi, H., Zhang, L. and Watkins, C., 2025. Dual representations: A novel variant of Self-Supervised Audio Spectrogram Transformer with multi-layer feature fusion and pooling combinations for sound classification. *Neurocomputing*, *623*, p.129415.

[29] L. Zhang, C.P. Lim, and Y. Yu, Intelligent human action recognition using an ensemble model of evolving deep networks with swarm-based optimization, *Knowledge-based systems*, 2021.

[30] Slade, S., Zhang, L., Asadi, H., Lim, C.P., Yu, Y., Zhao, D., Panesar, A., Wu, P.F. and Gao, R., 2025. Cluster search optimisation of deep neural networks for audio emotion classification. *Knowledge-Based Systems*, *314*, p.113223.

[31] D. Pandit, L. Zhang, S. Chattopadhyay, C.P. Lim, and C. Liu, A scattering and repulsive swarm intelligence algorithm for solving global optimization problems, *Knowledge-Based Systems*, vol. 156, 2018.

[32] Chen, W., Zhang, L. and Jiang, M., 2022. Failure Mode Identification of Elastomer for Well Completion Systems using Mask R-CNN. In *IJCNN* (pp. 1-8).

[33] B. Fielding and L. Zhang, Evolving image classification architectures with enhanced particle swarm optimisation, *IEEE access*, vol. 6, 2018.

[34] H. Xie, L. Zhang, and C. P. Lim, Evolving cnn-lstm models for time series prediction using enhanced grey wolf optimizer, *IEEE access*, vol. 8, 2020.

[35] Arno, J., Grace, O., Larridon, I. and Zhang, L., Plant Species Classification Using Evolving Ensemble and Siamese Networks. In *IEEE SMC*. 2024.

[36] T. Lawrence, L. Zhang, K. Rogage, and C. P. Lim, Evolving deep architecture generation with residual connections for image classification using particle swarm optimization, *sensors*, p.7936, 2021.