Small Object Detection: Milestone 1

Arnab Dey
Computer Science & Artificial Intelligence
Plaksha University

Chandan Yadav
Computer Science & Artificial Intelligence
Plaksha University

Nishant Mahajan
Computer Science & Artificial Intelligence
Plaksha University

Rishi Sharma
Computer Science & Artificial Intelligence
Plaksha University

Abstract

This paper proposes a novel technique in existing object detection tasks, especially in Small Object Detection (SOD). The adaptation of transformers in image detection has advanced the discipline to unprecedented heights. However, SOD remains a challenging task. This is because small objects appear as noisy, blurry, and less informative images. This is impeded further by the scarcity and lack of diversity of SOD datasets. To address the issue of SOD and its effects, we introduce a scalable, and customizable, feature-fusion technique for SOD which may be extended to normal object detection tasks. This paradigm is greatly influenced by ensemble techniques although it clearly distinguishes itself in terms of ease of implementation, and training time required. We also introduce a practical implementation of using Swin Transformer v1[7] with some nuances borrowed from Swinv2[6] as the neck; the sub-network between the backbone and the prediction head. We employ another parallel network in the neck loosely based on a Feature Pyramid Network (FPN) to support the Swin-based neck network and perform feature fusion. We employ a traditional CenterNet as the head. Further, we decrease the window size of the transformer to benefit small object detection.

1 Introduction

One of the core technologies in the field of machine vision is object detection. After Convolutional Neural Networks (CNN) took center stage in the industry a few years ago, Vision Transformers (ViT) [2] for image have advanced the discipline to unprecedented heights. A new era of capabilities and possible applications is heralded by the introduction and adaptation of transformer models [9] to visual data, which were originally meant for natural language processing (NLP).

Nevertheless, because small objects appear as noisy, blurry, and less informative images, Small Object Detection (SOD) remains a difficult task [5]. This is impeded even further by the scarcity and lack of diversity of SOD datasets.

In recent years, various methods have been proposed to solve limitations in the detection of particularly small objects. Super Resolution is used to recover the information of low-resolution objects. Employing pre-trained detectors, they obtain object regions and then use a generator to generate corresponding super-resolution objects of an image. These are generally GAN-based models that have stability difficulties when training and are not efficient in detection tasks[4]. These methods only alleviate issues of specific datasets but do not address the difficulties in learning efficient features for small objects.

To sufficiently learn object features, we focus on the architecture of the object detection model for fitting the small object scale. In general, the common components of a detection model are a backbone

Code Files: Github

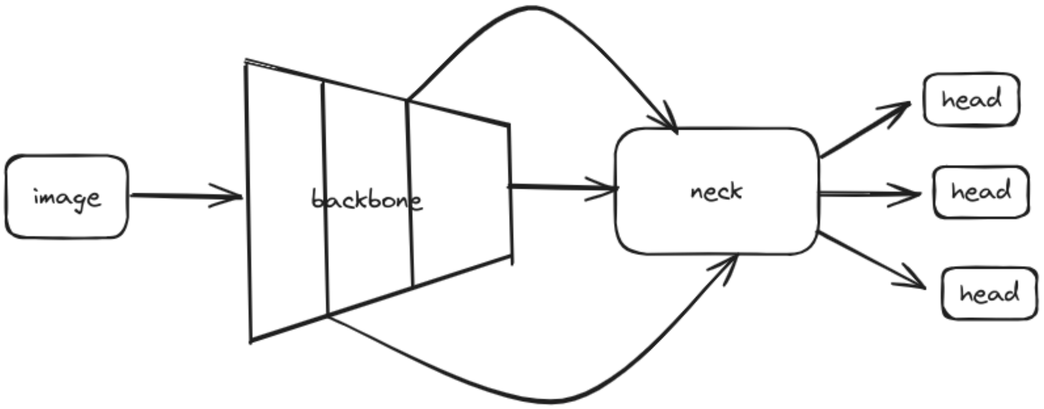

Figure 1: General Object Detection Network

for image feature extraction, a next with pyramid-like architecture combining different sizes of image feature maps for feature merging, and a prediction head for object classes and bounding box final prediction.

We theorize that our swin-based neck network, further strengthened by a parallel feature extractor network, potentially solves the previous limitation of the inconsistency of features, particularly while merging respective ones. This is backed by [4] who adapted to the small object scale by changing the window size of the swin transformer.

2 Related Work

SOD4SB Dataset We utilize the SOD4SB dataset, gathered through onboard drone cameras, comprising images with a resolution of 3,840 × 2,160 pixels. This dataset encompasses various avian species such as hawks, crows, and sparrows, among others. Many images depict instances of mutual occlusion and crowding, while others suffer from blurring caused by drone motion and bird flight. Despite the diversity of wild birds, all instances have been uniformly annotated as "bird".

The SOD4SB dataset comprises 39,070 annotated images and 137,121 bird instances. These annotated images are divided into three subsets: the **Training Subset** (9,759 images), the **Public Test Subset** (9,699 images), and the **Private Test Subset** (20,512 images).

This was introduced by [5] for the MVA2023 Challenge for Spotting Birds. We only utilized a part of this dataset due to hardware constraints.

Transformer based vision backbones, also known as Vision Transformers (ViT), use a transformer architecture on medium-sized, non-overlapping picture patches to classify images. Compared to convolutional networks, ViT offers an amazing speed-accuracy trade-off for image classification; yet, it needs big training datasets to function well. On the other hand, architectures such as DINO from FAIR [8] offer a novel and reliable technique for transformer self-supervised learning.

However, because of its low-resolution feature maps and quadratic complexity increase with image size, its architecture is not suited for usage as a general-purpose backbone network when the input image quality is large. As a result, we resort to Swin Transformer [7], which operates locally and generates multi-resolution feature maps while maintaining a linear level of complexity. This is useful in simulating the strong correlation found in visual signals.

State of the Art (SOTA): Weighted Box Fusion (WBF) Weighted box fusion achieves the best results for tiny object detection on the SOD4SB dataset, with an AP_{50} of 77.6. The model makes use of an ensemble fusion technique, which capitalizes on the advantages of current methodologies to improve overall performance. The rationale for this is that ensemble approaches frequently enable increased generalizability by taking advantage of the diversity of these models, resulting in more reliable and accurate predictions [3].

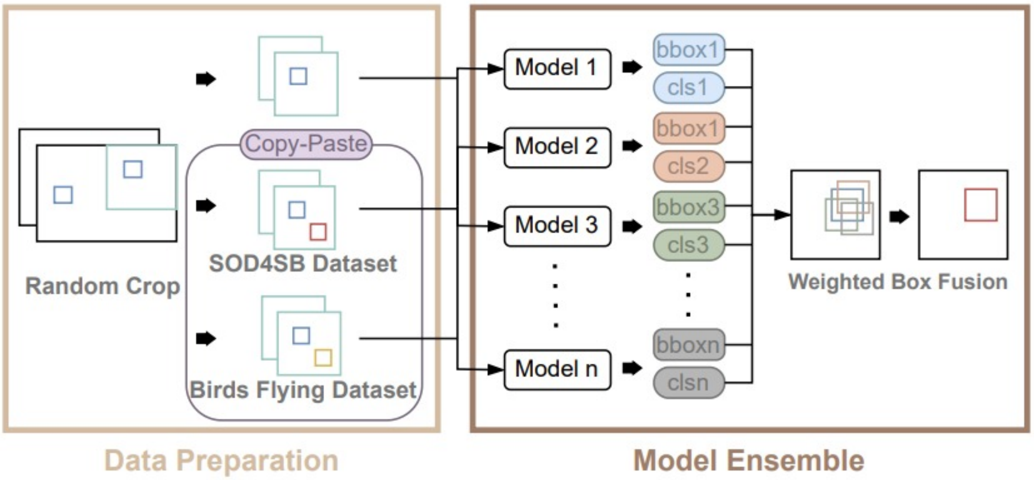

Figure 2: Overview of the framework in WBF [3]

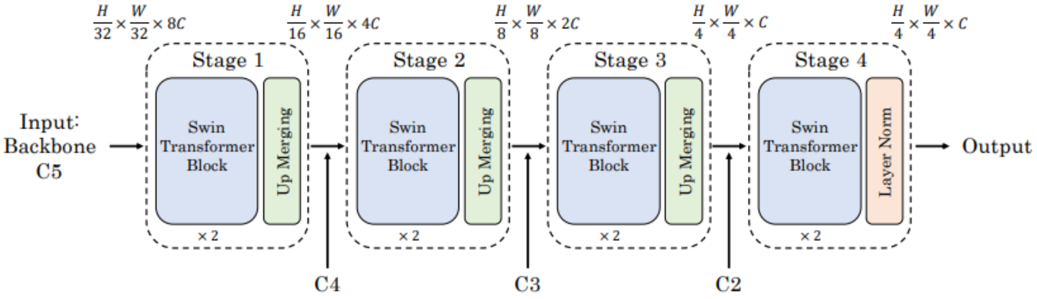

Figure 3: Architecture of the proposed neck network in [4]

SOD for Birds with Swin Transformer uses a swin transformer that advantages mAPs for small object identification and has a small window size (by default, of 2). Similar to a convolution procedure, this enables a greater emphasis on local information. Furthermore, the delicate shifting window design takes into account the attentions of overlapping windows with the shifting windows in each stage.

We provide a slightly tweaked implementation of their proposed network to adapt to our available hardware. This works as our baseline. Further, it is important to note that their code implementation was not available. However, in their paper, they mention using mmlab's mmdetection[1] to conduct their experiments but we provide an implementation that suits our needs. Our implementation does not depend on MMDetection.

3 Methodology

3.1 Proposed Claim

We propose an innovative approach to enhance small object detection leveraging a novel architecture employing **dual necks** within the detection pipeline. Our methods combine the robustness of a modified Swin Transformer backbone with a customized secondary Swin-based neck module followed by our implementation of a Centernet head for bounding box predictions.

We theorize that using multiple necks (2 in this case) may lead to much richer feature map extraction. This is different from ensemble weighted box fusion. This approach allows for scaling the neck to multiple feature extraction networks. Compared to the ensemble approach, this method does not

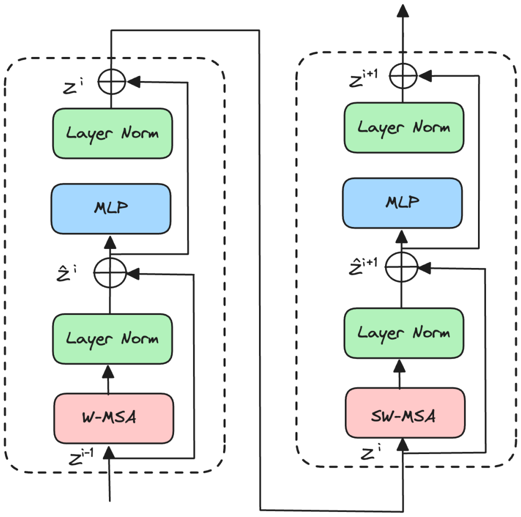

Figure 4: Post Norm in SwinV1

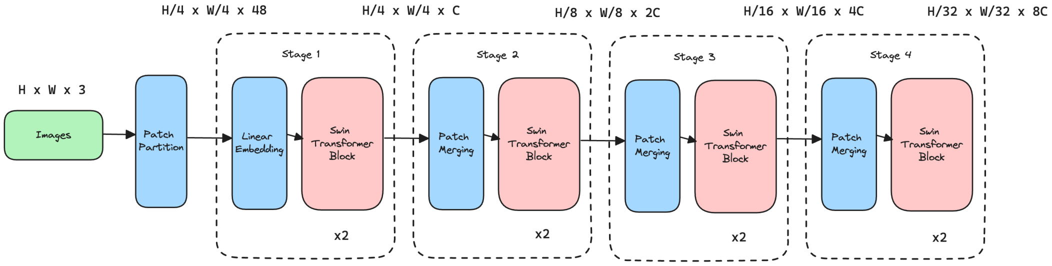

Figure 5: Proposed Backbone

require training all the object detection networks end-to-end. This saves us training and inference time. However, adding multiple necks will lead to an increase in the number of parameters. Thus, there exists a sweet ratio comparing the number of necks and the number of parameters.

3.2 Overall Architecture

3.3 Modified Swin Transformer

We employ a swinv1 architecture with some nuances from swinv2, especially the post-normalization technique. We also change the layers to [2, 2, 2, 2]. See figure 5. This becomes our teeny-tiny swin transformer as the actual swin-T transformer has layers [2, 2, 6, 2].

We started with employing a swin-S transformer pre-trained on imagenet22k as a backbone. This posed us with two difficulties: (a) It had a fixed size of 224. When compared to our 4K images with small objects less than 32 pixels, using this backbone would result in these being projected as a point. Further, it had a window size of 7 which easily failed to capture the small objects. The attention was simply neglected as mentioned in [5]. (b) We had compute constraints and could not train the swin-S or even the swin-T models on the drone2021 data which is a significant part of the SOD4SB dataset.

3.4 Feature Upsampling

It also needs to meet various feature map sizes between neck network stages to support multiple small-scale detections. We provide an upsampling module [5] that replaces transposed convolution with simple operations in light of the neck's efficiency. The upsampling mechanism is the opposite of the patch merging for downsampling; it is also known as the Up Merging module. PixelShuffle is a way of super-resolution for images. We have chosen a stride of two for this procedure. Following

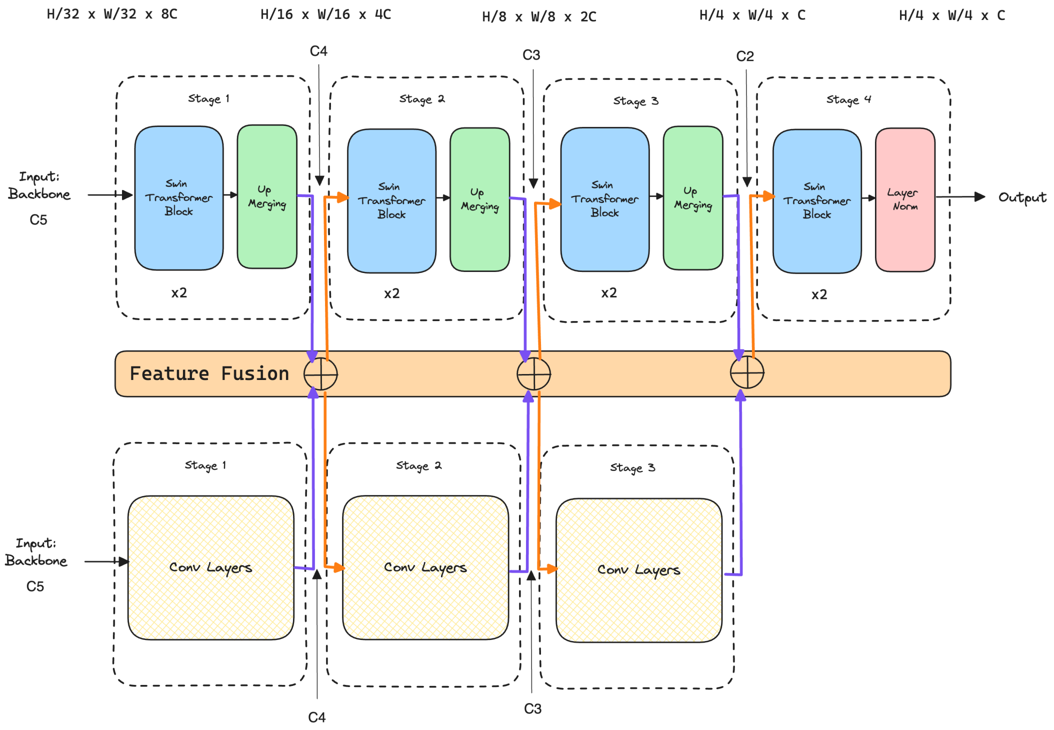

Figure 6: Proposed Parallel Feature Extractor Network

this, the input channel (designated as 4C) becomes channel C in the feature maps. We utilize a concatenation (had to change from a linear layer because of compute constraints) to convert the channel from C to 2C to maintain consistency with the feature merging backbone.

3.5 Feature Fusion

We use a network loosely based on the feature pyramid network (FPN). You may experiment with different necks to substitute for this parallel network. See figure 6. Convolutional networks have stood the test of time, and for the smaller dataset that we are working with, we think that using a convnet-based feature extractor will further augment the features extracted by our proposed swin neck network. We do a simple addition after each stage of the individual networks. This is sent as input to the second stage of both the neck networks. There are $n-1$ stages for the secondary neck network. Do note that we also take into consideration the features extracted by the backbone denoted as C_i, where $i = 5, 4, 3, 2$. Only the output from the last stage of the swin network is used in the head.

4 Experiments and Results

4.1 Small Object Dataset

We make use of the SOD4SB dataset, which was collected using onboard drone cameras and consists of 3,840 $\times$ 2,160 pixel-resolution images. This dataset includes a variety of bird species, including sparrows, crows, and hawks. Numerous photos show instances of crowding and mutual occlusion, while other photos have blurred from flying birds and drone motion. Despite the variety of wild birds, "bird" has been consistently annotated in every instance.

There are 137,121 bird occurrences and 39,070 annotated photos in the SOD4SB collection. The **Training Subset** (9,759 photos), the **Public Test Subset** (9,699 images), and the **Private Test Subset** (20,512 images) are the three subsets into which these annotated images are separated.

We only train on the Training subset, creating a separate validation set for evaluation.

4.2 Settings and Results

We train the model with a SGD optimizer as it performs well with image training tasks. We fully train the network end-to-end. However, one could pretrain and fine tune the backbone on the drone2021 data. We use a center net head for the prediction of the bounding box.

Here are the four proposed experiments: 1) Baseline on train dataset. 2) Feature Fusion on train dataset. 3) Baseline on augmented train dataset. 4) Feature Fusion on augmented train dataset.

5 Conclusion

We implemented a novel architecture to detect small objects. Our contributions are as follows: 1) We implemented a custom, novel, neck based on the swin transformer. 2) We augmented this model with a network loosely based on the feature pyramid network.

Further study could include experimenting with hyperparameters for the backbone architecture, and neck architecture, especially the window size. One could also experiment with the different neck architectures for feature fusion.

References

[1] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. Mmdetection: Open mmlab detection toolbox and benchmark, 2019.

[2] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[3] Hao-Yu Hou, Mu-Yi Shen, Chia-Chi Hsu, En-Ming Huang, Yu-Chen Huang, Yu-Cheng Xia, Chien-Yao Wang, and Chun-Yi Lee. Ensemble fusion for small object detection. In *2023 18th International Conference on Machine Vision and Applications (MVA)*, pages 1–6, 2023.

[4] Da Huo, Marc A. Kastner, Tingwei Liu, Yasutomo Kawanishi, Takatsugu Hirayama, Takahiro Komamizu, and Ichiro Ide. Small object detection for birds with swin transformer. In *2023 18th International Conference on Machine Vision and Applications (MVA)*, pages 1–5, 2023.

[5] Yuki Kondo, Norimichi Ukita, Takayuki Yamaguchi, Hao-Yu Hou, Mu-Yi Shen, Chia-Chi Hsu, En-Ming Huang, Yu-Chen Huang, Yu-Cheng Xia, Chien-Yao Wang, Chun-Yi Lee, Da Huo, Marc A. Kastner, Tingwei Liu, Yasutomo Kawanishi, Takatsugu Hirayama, Takahiro Komamizu, Ichiro Ide, Yosuke Shinya, Xinyao Liu, Guang Liang, and Syusuke Yasui. Mva2023 small object detection challenge for spotting birds: Dataset, methods, and results. In *2023 18th International Conference on Machine Vision and Applications (MVA)*, pages 1–11, 2023.

[6] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12009–12019, 2022.

[7] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

[8] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.