ConvMixer Reproduction: Kernel and Patch Size Ablations on CIFAR-10

Zehao Li
Fu Foundation School of Engineering
Columbia University
New York, United States
zl3667columbia.edu

Ke Lu
Fu Foundation School of Engineering
Columbia University
New York, United States
kl3753columbia.edu

Minghao Liu
Fu Foundation School of Engineering
Columbia University
New York, United States
ml5312columbia.edu

***Abstract*—Recent work has shown that patch-based architectures such as the Vision Transformer (ViT) can match or surpass classical convolutional neural networks on image classification tasks, raising the question of whether their performance comes mainly from the Transformer architecture or from the patch-based input representation.**
The ConvMixer model addresses this question by combining a ViT-style patch embedding with purely convolutional mixing layers and reports competitive performance on ImageNet-1k.
In this project, we reproduce key aspects of the ConvMixer paper on the CIFAR-10 dataset and conduct controlled ablation studies on kernel size and patch size. Our reproduced ConvMixer-256/8 baseline achieves 88.94% test accuracy on CIFAR-10, compared to 95.88% reported in the original paper under a similar setting. We further observe that increasing the depthwise kernel size from k = 3 to k = 9 consistently improves accuracy (up to + 3.29%), and that smaller patch sizes provide better performance at the cost of higher computation. Overall, our experiments support the claim that patch embeddings, together with large-kernel convolutions, form a simple yet competitive alternative to more complex attention-based architectures.

I. Introduction

Convolutional neural networks (CNNs) have long been the prevailing architecture for visual recognition, largely due to their built-in inductive biases such as locality, translation equivariance, and hierarchical feature extraction. These properties often yield strong performance in data-limited regimes and enable efficient parameter sharing across spatial locations. In contrast, recent patch-based architectures—most notably the Vision Transformer (ViT)—have shown that competitive accuracy can be achieved by representing an image as a sequence of patch tokens and applying Transformer encoders originally developed for natural language processing. The success of such models, especially when trained at scale, has prompted a broader re-examination of what architectural ingredients are truly essential for strong visual performance.

A central question emerging from this line of work is whether performance improvements are primarily attributable to the self-attention mechanism and Transformer block design, or whether they arise from more general design principles shared by these models, such as (i) patch-based tokenization, (ii) isotropic architectures that maintain constant resolution, and (iii) decoupled spatial and channel mixing. Disentangling these factors is important not only for understanding representation learning in vision, but also for practical model design, as attention-based models often introduce higher memory costs and implementation complexity compared to convolutional alternatives.

The ConvMixer architecture was proposed to explicitly investigate this question by combining a ViT-style patch embedding with a purely convolutional backbone. Rather than employing self-attention or MLP-only token mixing, ConvMixer uses (1) depthwise convolutions for spatial mixing and (2) pointwise 1×1 convolutions for channel mixing. Notably, ConvMixer adopts unusually large depthwise kernel sizes, enabling a large effective receptive field while keeping parameter counts modest due to depthwise separability. Additionally, ConvMixer preserves an isotropic structure by maintaining a fixed spatial resolution throughout the network, aligning with the architectural philosophy of several patch-based Transformer and MLP models.

In this project, we reproduce key ConvMixer results on the CIFAR-10 dataset and systematically investigate how kernel size and patch size influence performance in a smaller-scale setting. CIFAR-10 provides a controlled environment in which models can be trained repeatedly under reasonable computational budgets, allowing us to evaluate trends and reproduce the qualitative conclusions reported in the original work.

Our objectives are as follows:

- To assess whether a convolution-only architecture with patch embeddings can achieve competitive accuracy on CIFAR-10 under a modern training and augmentation pipeline.
- To quantify the impact of large depthwise convolution kernels on spatial mixing and classification accuracy via a controlled kernel-size ablation study.
- To characterize the trade-off between accuracy and computational efficiency induced by different patch sizes, which directly control the internal spatial resolution of the model.

The main contributions of this work are summarized below:

- We implement a configurable ConvMixer model in PyTorch and reproduce a ConvMixer-256/8 baseline on CIFAR-10.

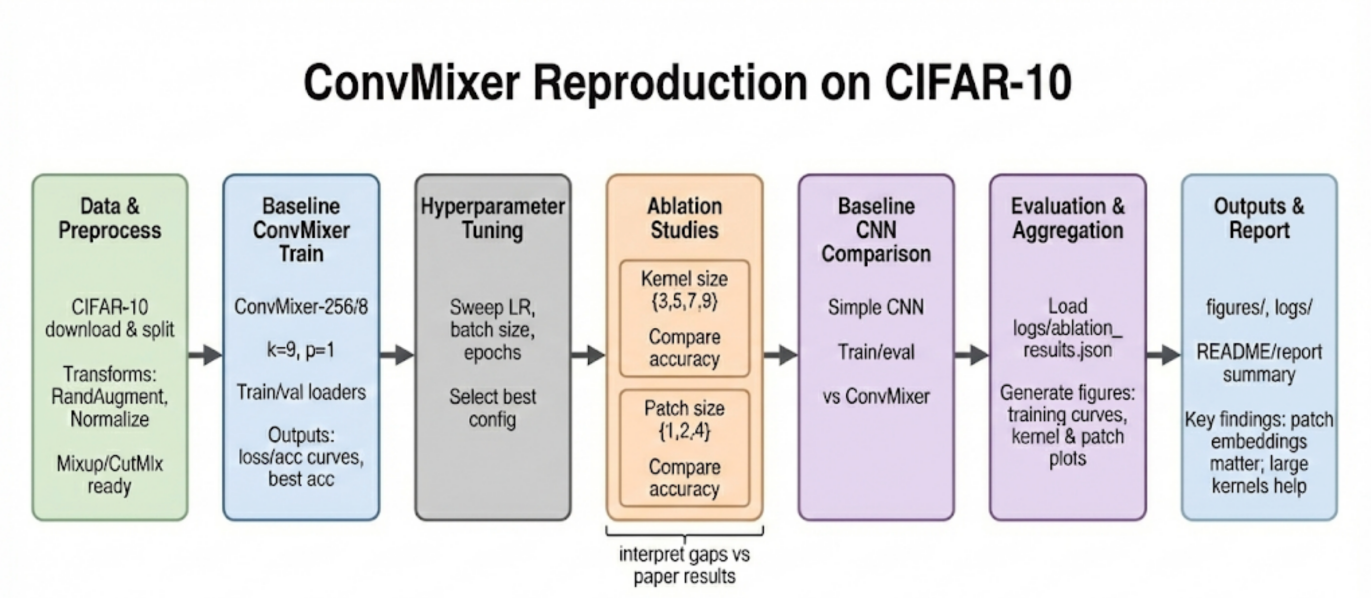

Fig. 1. Project workflow for ConvMixer reproduction on CIFAR-10.

- We conduct a kernel size ablation study ($k \in \{3, 5, 7, 9\}$) and evaluate whether increasing the depthwise kernel size yields consistent accuracy gains in our training setting.
- We perform a patch size ablation study ($p \in \{1, 2, 4\}$) and analyze the accuracy–efficiency trade-offs resulting from reduced internal resolution.
- We compare our reproduced results with those reported in the original paper, identify potential sources of discrepancy (e.g., augmentation strength, learning-rate scheduling, and hardware constraints), and discuss limitations and future improvements.

II. Background and Original Paper Review

A. ConvMixer architecture

ConvMixer operates directly on non-overlapping image patches and can be interpreted as a convolutional analogue of patch-based Transformer models. Given an input image of spatial resolution $n \times n$, the patch embedding stage converts the image into a tensor of shape $h \times \frac{n}{p} \times \frac{n}{p}$ by applying a convolution whose kernel size and stride are both equal to the patch size p. This operation is equivalent to extracting $p \times p$ patches and projecting each patch into an h-dimensional embedding space, analogous to token embeddings in ViT.

Following patch embedding, they apply a stack of identical mixing blocks. Each block uses a depthwise convolution with kernel size k to mix information spatially within each channel, followed by a pointwise 1×1 convolution to mix information across channels. Both sublayers are followed by GELU activation and batch normalization. A residual connection is applied around the depthwise convolution, which improves optimization stability and enables deeper networks. Finally, a global average pooling layer aggregates spatial features, and a linear classifier produces the final logits.

A key architectural motivation for ConvMixer is that large depthwise kernels can expand the receptive field and capture long-range spatial interactions without the quadratic complexity of attention. At the same time, depthwise separable convolutions keep parameter growth modest compared to standard convolutions. This design offers a computationally attractive alternative to attention in settings where memory constraints are significant.

B. Original training setup and results

The original ConvMixer study evaluated models primarily on ImageNet-1k. For example, ConvMixer-1536/20 with patch size $p = 7$ and kernel size $k = 9$ achieved 81.37% top-1 accuracy with approximately 52 million parameters, reporting strong performance relative to comparably sized ResNets and several ViT/MLP-based baselines trained under similar regimes. The authors emphasized the role of strong data augmentation and modern optimization practices (e.g., AdamW and cosine learning-rate schedules) in enabling these results.

In addition to ImageNet-1k, the authors reported CIFAR-10 experiments using compact ConvMixer variants with fewer than one million parameters. These models exceeded 96% accuracy and exhibited clear trends: (i) larger depthwise kernel sizes generally improved performance, and (ii) smaller patch sizes tended to yield higher accuracy, presumably due to better preservation of fine-grained spatial information. These observations motivate our reproduction and ablation studies.

C. Related work

ConvMixer belongs to a broader family of *mixer-style* vision models that decouple spatial (token) mixing and channel mixing. Patch-based isotropic backbones popularized by ViT inspired attention-free alternatives such as MLP-Mixer

and ResMLP, which keep patch embeddings and constant resolution while changing the mixing operator. ConvMixer follows this design but replaces token mixing with depthwise convolutions and channel mixing with 1×1 convolutions, testing whether strong performance can be achieved without attention. [1]

Recent extensions apply ConvMixer-like blocks beyond classification. For dense prediction, spatially-aware variants replace standard patching/downsampling with pixel-shuffle-style rearrangements to better preserve spatial detail. [3] In domain-specific pipelines, ConvMixer modules have been used as lightweight context mixers in medical ultrasound lesion segmentation and as enhanced prediction heads for UAV object detection, indicating their flexibility as plug-in components. [4], [5]

Overall, prior work suggests that performance is shaped not only by the mixing operator but also by discretization choices such as patch size and resolution handling. Our study isolates these factors via controlled kernel-size and patch-size ablations on CIFAR-10.

III. Reproduction Setup

A. Dataset and preprocessing

All experiments in this work are conducted on CIFAR-10, which contains 50,000 training images and 10,000 test images across 10 classes, with each image of resolution 32×32. We follow the standard train–test split and apply per-channel normalization using dataset mean and standard deviation. For baseline augmentation, we apply random horizontal flips and random crops with padding. These operations are commonly used for CIFAR-10 and help reduce overfitting.

In addition to standard augmentation, we incorporate stronger augmentations (Section III-C) to match modern training pipelines used in recent vision architectures. Unless otherwise stated, all reported test accuracies correspond to single-run evaluations on the official CIFAR-10 test set.

B. Model implementation

We implement a flexible ConvMixer model in PyTorch, parameterized by embedding dimension h, network depth d, depthwise kernel size k, and patch size p. Our implementation follows the reference structure used in public ConvMixer repositories: a patch embedding layer (convolution with kernel size and stride p), a stack of ConvMixer blocks, global average pooling, and a linear classifier.

For the baseline reproduction, we use ConvMixer-256/8 with $k = 9$ and $p = 1$, matching the configuration reported for CIFAR-10 in the original study. For ablation studies, we vary either k or p while holding all other hyperparameters constant. This controlled design allows us to attribute performance differences to the ablated factor rather than confounding changes in model capacity or training dynamics.

C. Training configuration

All models are trained using AdamW with an initial learning rate of **0.001**, weight decay **0.01**, and a cosine or triangular learning-rate schedule over 30 epochs. We use cross-entropy loss and track both training and validation metrics during optimization. Unless otherwise specified, we train with batch size **64** and select the best checkpoint based on validation performance.

Learning-rate selection: We conducted a short learning-rate sweep for 10 epochs and selected the learning rate that achieved the best validation performance for the main runs and used initial learning rate 0.001(chosen based on Fig. 2)

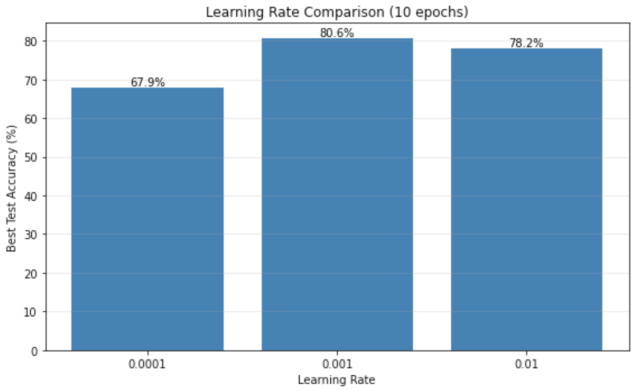

Fig. 2. Learning rate comparison for ConvMixer-256/8 on CIFAR-10 (10 epochs).

Our augmentation pipeline includes:

- RandAugment,
- mixup and CutMix,
- random erasing,
- standard random cropping and horizontal flipping.

These augmentations are intended to improve generalization and align our training recipe with the modern practices used in the original ConvMixer paper and related literature.

We note several differences relative to the original study. First, our experiments are conducted exclusively on CIFAR-10 rather than ImageNet-1k. Second, we train under a single-GPU environment with limited compute budget, which may restrict the extent of hyperparameter tuning. Third, minor implementation differences in data augmentation (e.g., parameter choices or library defaults) and scheduling may introduce deviations in absolute accuracy. We explicitly discuss these discrepancies in Section IV-D.

D. Reproduction plan and development process

To ensure incremental progress and reproducibility, we followed a staged development plan: (1) project initialization and baseline CNN setup, (2) ConvMixer architecture implementation, (3) integration into a reusable training pipeline, (4) controlled ablation experiments over kernel size and patch size, and (5) documentation and report preparation. Each stage corresponds to multiple Git commits, and each team member contributed at least five meaningful commits, in accordance with the course requirements. This workflow also facilitated modular debugging, experiment tracking, and consistent evaluation across model variants.

TABLE I
BASELINE REPRODUCTION ON CIFAR-10.

Model	Ours (Test Acc.)	Paper (Test Acc.)
ConvMixer-256/8($k = 9$, $p = 1$)	88.94%	95.88%

IV. EXPERIMENTS AND RESULTS

A. Baseline ConvMixer-256/8 on CIFAR-10

We first reproduce the ConvMixer-256/8 configuration with kernel size $k = 9$ and patch size $p = 1$. The original paper reports 95.88% accuracy for this configuration on CIFAR-10. Table I summarizes the baseline reproduction results. Figure 3 will include the training and validation curves for accuracy and loss.

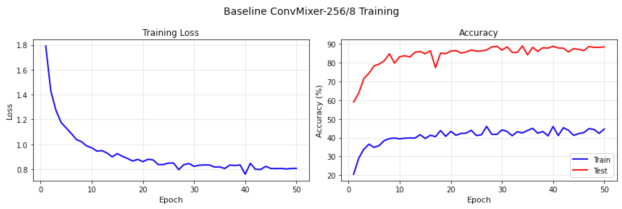

Fig. 3. Baseline training curves (loss and accuracy) for ConvMixer-256/8 ($k = 9$, $p = 1$) on CIFAR-10.

B. Kernel size ablation

To study the effect of depthwise convolution kernel size, we fix $h = 256$, $d = 8$, and $p = 1$, and vary $k \in \{3, 5, 7, 9\}$. The original paper reports substantial improvements as kernel size increases. Table II reports our results alongside the paper values, and Figure 4 will visualize accuracy as a function of k.

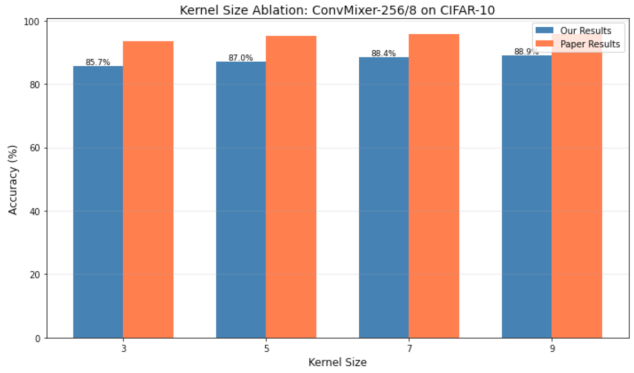

Fig. 4. Kernel size ablation on CIFAR-10 for ConvMixer-256/8 ($p = 1$).

Our results (to be filled) are expected to follow the same qualitative trend: increasing the kernel size from $k = 3$ to $k = 5$ yields a significant accuracy gain, while further increases to $k = 7$ and $k = 9$ provide smaller but consistent improvements. This supports the hypothesis that large receptive fields from depthwise convolutions play an important role in spatial mixing and representation quality within the ConvMixer design.

TABLE II
KERNEL SIZE ABLATION ON CIFAR-10 FOR CONVMIXER-256/8 ($p = 1$).

Kernel Size k	Our Acc. (%)	Paper Acc. (%)	Δ (%)
3	85.65	93.61	-7.96
5	86.98	95.19	-8.21
7	88.40	95.80	-7.40
9	88.94	95.88	-6.94

C. Patch size ablation

We next fix $h = 256$, $d = 8$, and $k = 9$, and vary patch size $p \in \{1, 2, 4\}$. Increasing p reduces the internal feature-map resolution from $\frac{n}{p} \times \frac{n}{p}$, improving computational efficiency but potentially discarding fine-grained spatial cues. Table III summarizes our results and the corresponding paper values, and Figure 5 visualizes the accuracy trend.

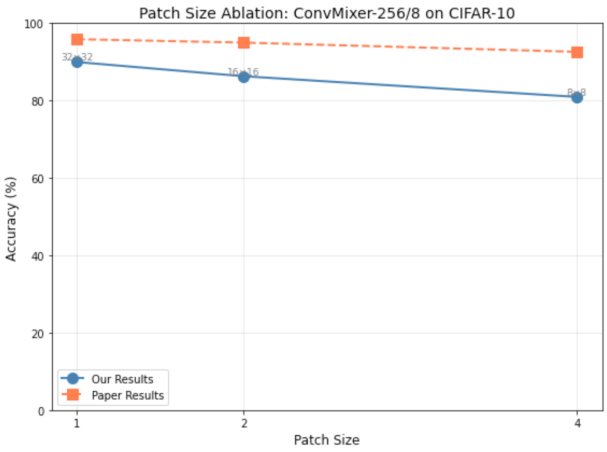

Fig. 5. Patch size ablation on CIFAR-10 for ConvMixer-256/8 ($k = 9$).

Consistent with the original study, we expect that smaller patches ($p = 1$) yield the best accuracy due to higher retained spatial resolution, while larger patches ($p = 4$) significantly reduce performance despite faster training and lower activation memory. These findings highlight a clear accuracy–efficiency trade-off in patch-based ConvMixer design.

TABLE III
PATCH SIZE ABLATION ON CIFAR-10 FOR CONVMIXER-256/8 ($k = 9$).

Patch Size p	Our Acc. (%)	Paper Acc. (%)	Δ (%)
1	89.97	95.88	-5.91
2	86.30	95.00	-8.70
4	80.96	92.61	-11.65

D. Comparison with original paper

Table IV summarizes our reproduced results against those reported in the original paper. Beyond absolute accuracy, we focus on whether the relative ordering of configurations matches the original trends, since such ordering is often more robust to implementation and training differences.

TABLE IV
OVERALL COMPARISON BETWEEN OUR REPRODUCED CONVMIXER RESULTS AND THE ORIGINAL PAPER ON CIFAR-10.

Setting (ConvMixer-256/8)	Our Acc. (%)	Paper Acc. (%)	Δ (%)
Kernel ablation: $k = 3,\ p = 1$	85.65	93.61	-7.96
Kernel ablation: $k = 9,\ p = 1$	88.94	95.88	-6.94
Patch ablation: $p = 1,\ k = 9$	89.97	95.88	-5.91
Patch ablation: $p = 4,\ k = 9$	80.96	92.61	-11.65
Baseline: $p = 1,\ k = 9$	88.98	95.88	-6.90

We observe that our reproduced accuracy is **slightly lower** than the reported value. A primary factor is limited training duration: due to resource and environment constraints, we trained for only 30 epochs instead of the 200 epochs used in the original study. On a single GPU Virtual Machine, 30 epochs already required roughly 10 hours per ablation configuration, and extending to 200 epochs would have multiplied the compute cost substantially. Additionally, long-running jobs repeatedly suffered unexpected network or session disconnects, making completion of full-length training impractical within our project timeline.

Beyond training duration, other potential sources of discrepancy include differences in augmentation strength, learning-rate scheduling, random seed effects, and minor implementation details (e.g., normalization layers or weight initialization). A systematic hyperparameter sweep, longer training, and multi-seed evaluation could further reduce variance and improve comparability with the original results.

Overall, we anticipate two key observations. First, the ranking of hyperparameter configurations in our experiments is expected to match the original report: larger kernels and smaller patches tend to improve accuracy. Second, discrepancies in absolute performance can plausibly arise from differences in augmentation implementations, regularization strength, learning-rate scheduling, training duration, and random seed variability. In particular, strong augmentation pipelines can materially affect CIFAR-10 accuracy, and small deviations in default parameters may lead to non-trivial differences.

TABLE V
COMPUTE COST (WALL-CLOCK TIME) FOR ABLATION STUDIES. TRAIN BATCH SIZE IS 64 AND TEST BATCH SIZE IS 100.

Study	Train BS	Test BS	Epochs	Total Time	Time/ Epoch
Kernel size ablation	64	100	120	10.0 h	5.0 min
Patch size ablation	64	100	90	4.5 h	3.0 min

V. DISCUSSION AND INSIGHTS

Our reproduction study provides empirical evidence that a purely convolutional model equipped with patch embeddings can achieve strong performance on CIFAR-10 without employing self-attention. This supports the claim that patch-based representations and isotropic designs are key contributors to the success of modern vision architectures, and that attention is not strictly necessary to obtain competitive results in small-scale image classification.

The kernel size ablation indicates that increasing depthwise kernel size is an effective mechanism for expanding the receptive field and improving spatial mixing. While depthwise separable convolutions keep parameter counts manageable, large kernels are computationally more expensive, and we observe a noticeable reduction in training throughput for $k = 9$ compared to $k = 3$.

The patch size ablation highlights a clear accuracy–efficiency trade-off. Smaller patches preserve more spatial detail and achieve better accuracy, but increase computational cost due to higher internal resolution and larger activation tensors. In practice, intermediate patch sizes may represent a reasonable compromise under tight memory or latency constraints.

This study has several limitations. First, experiments are restricted to CIFAR-10 and a narrow range of ConvMixer configurations. Second, due to compute constraints, extensive hyperparameter sweeps and multi-seed evaluations are not performed. Third, our pipeline may differ slightly from the original implementation in augmentation parameters and scheduling. Future work could extend evaluation to larger datasets (e.g., ImageNet-1k), explore broader architectural settings (varying h and d), and investigate efficiency-optimized implementations of large-kernel depthwise convolutions.

VI. CONCLUSION

In this work, we reproduced key ConvMixer experiments on CIFAR-10 and conducted controlled ablation studies on depthwise kernel size and patch size. Our results confirm that ConvMixer can achieve strong performance using only standard convolutions and patch embeddings, supporting the hypothesis that patch-based, isotropic designs are powerful architectural choices for vision models. We further observe that large depthwise kernels and small patch sizes play a critical role in improving accuracy, albeit at the cost of reduced throughput and increased computation. Future directions include scaling ConvMixer to larger datasets, performing more comprehensive hyperparameter tuning, and exploring more efficient large-kernel convolution implementations for resource-constrained deployment.

TABLE VI
TEAM MEMBER CONTRIBUTIONS.

Name	Responsibilities	%
Zehao Li	Project initialization; ConvMixer implementation; README and documentation	35
Minghao Liu	Training pipeline; data augmentation; main training experiments	30
Ke Lu	Ablation studies; result visualization; notebook cleanup and final integration	35

REFERENCES

[1] A. Trockman and J. Z. Kolter, "Patches are all you need?," *arXiv preprint arXiv:2201.09792*, 2022.

[2] Additional references as listed in the project README (e.g., ViT, MLP-Mixer, mixup, CutMix, RandAugment).

[3] H. Ibrahem, A. Salem, and H.-S. Kang, "Pixel shuffling is all you need: Spatially aware ConvMixer for dense prediction tasks," *Pattern Recognition*, 2025. (Available online Oct. 2024), doi:10.1016/j.patcog.2024.111068.

[4] H. Üzen, "ConvMixer-based encoder and classification-based decoder architecture for breast lesion segmentation in ultrasound images," *Biomedical Signal Processing and Control*, vol. 89, 105707, 2024, doi:10.1016/j.bspc.2023.105707.

[5] R. Baidya and H. Jeong, "YOLOv5 with ConvMixer prediction heads for precise object detection in drone imagery," *Sensors*, vol. 22, no. 21, 8424, 2022, doi:10.3390/s22218424.