CNN-Free Lightweight Vision Model Using Weight-Shared MLP on Micro-Patches

Taehyeon Kim
Department of Computer Engineering, Kyonggi University
Suwon, Republic of Korea
dannykim05@kyonggi.ac.kr

Abstract

Convolutional neural networks (CNNs) achieve strong performance in vision, but their convolutional operators and feature hierarchies can impose non-trivial compute and parameter overhead in lightweight settings. Meanwhile, fully connected (MLP-based) vision models often sacrifice key CNN inductive biases such as locality and weight sharing. In this paper, we present MP-MLP (Micro-Patch Multi-Layer Perceptron), a convolution-free lightweight architecture that recovers CNN-like behavior using only fully connected layers. MP-MLP partitions an input image into non-overlapping micro-patches and applies a single weight-shared MLP block to every patch, acting as a pseudo-convolutional filter without any convolution operations. Patch-wise features are concatenated and fed into a shallow classifier MLP for end-to-end recognition. We evaluate MP-MLP on MNIST, Fashion-MNIST, and SVHN, covering increasing task complexity from clean grayscale digits to RGB street-view digits. With substantially fewer parameters, MP-MLP achieves competitive accuracy on MNIST and Fashion-MNIST, and slightly outperforms a lightweight CNN baseline on SVHN, demonstrating that carefully designed weight-shared MLPs can be a compelling convolution-free alternative for structured lightweight vision tasks.

Keywords: Multi-Layer Perceptron; Lightweight Vision; Micro-Patches; Weight Sharing; Edge Computing

1 Introduction

CNNs have dominated visual recognition for over a decade due to two key inductive biases: *local receptive fields* and *weight sharing*. These properties enable efficient local pattern extraction and translation robustness. However, convolutional stacks can still be computationally heavy and may be undesirable in extremely lightweight or operator-restricted environments.

Recent interest in non-convolutional vision models (e.g., MLP-based designs) has shown that fully connected architectures can achieve competitive performance under specific conditions, but they often lack explicit locality and weight sharing. To compensate, they may require large width/depth, reducing their efficiency.

We propose MP-MLP (Micro-Patch Multi-Layer Perceptron), a convolution-free architecture that reconstructs locality and weight sharing using only MLP layers. MP-MLP divides an image into micro-patches and applies a single shared patch-MLP to all patches. This shared mapping behaves similarly to scanning a filter across spatial locations, but avoids convolution operators. (For compactness, we omit an architecture figure; the complete model is fully specified by the micro-patch formulation and shared MLP equations in Section 3.) We validate MP-MLP on MNIST, Fashion-MNIST, and SVHN. Our results show that MP-MLP provides a strong accuracy–efficiency trade-off in structured lightweight settings.

1.1 Contributions

- We introduce MP-MLP, a convolution-free lightweight vision architecture that preserves CNN-like locality and weight sharing using a shared patch-wise MLP.
- We provide a simple micro-patch framework where a single shared MLP functions as a pseudo-convolutional filter.
- We evaluate on MNIST, Fashion-MNIST, and SVHN, demonstrating competitive accuracy with significantly fewer parameters, including a slight gain over a lightweight CNN baseline on SVHN.

2 Related Work

2.1 Lightweight CNNs

Mobile-oriented CNN families reduce compute using depthwise separable convolutions and optimized blocks, improving efficiency while retaining convolutional operators. Despite success, such models still depend on convolution and intermediate feature maps that may be costly in constrained settings.

2.2 MLP-Based Vision Architectures

MLP-Mixer, gMLP, and ResMLP revisit MLP designs for vision and can be competitive with sufficient scale and training. However, many MLP-based vision models lack explicit locality and weight sharing, potentially requiring larger capacity to recover similar behaviors.

2.3 Patch-Based Representations

Patch tokenization is widely used in modern vision models. Transformers process patches with attention mechanisms; MLP-based models often mix information globally across patches. In contrast, MP-MLP focuses on *local* patch processing with *shared* weights across spatial locations, explicitly targeting CNN-like inductive bias without convolution.

3 Proposed Method

3.1 Overview

MP-MLP consists of three components:

1. Micro-patch partitioning
2. Weight-shared patch MLP block
3. Patch feature aggregation and classification MLP

3.2 Micro-Patch Partitioning

Given an image $X \in \mathbb{R}^{H\times W\times C}$, we partition it into non-overlapping patches of size $p \times p$. The number of patches is

$$N = \frac{H}{p} \cdot \frac{W}{p}. \tag{1}$$

Each patch is flattened into $x_i \in \mathbb{R}^{p^2C}$ for $i = 1, \ldots, N$.

3.3 Weight-Shared Patch MLP

We define a single patch MLP

$$f_\theta : \mathbb{R}^{p^2C} \to \mathbb{R}^d, \tag{2}$$

and *share* its parameters θ across all patches:

$$h_i = f_\theta(x_i), \quad i = 1, \ldots, N. \tag{3}$$

This yields CNN-like weight sharing across spatial locations while preserving locality via patch-limited receptive fields.

In our implementation, f_θ is a 2-layer MLP:

$$f_\theta(x) = W_2\, \sigma(W_1 x + b_1) + b_2, \tag{4}$$

where σ is ReLU (or GELU), $W_1 \in \mathbb{R}^{d\times p^2C}$, and $W_2 \in \mathbb{R}^{d\times d}$.

3.4 Aggregation and Classification

Patch features are concatenated:

$$H = [h_1; h_2; \ldots; h_N] \in \mathbb{R}^{Nd}. \tag{5}$$

A shallow classifier MLP produces logits:

$$\hat{y} = g_\phi(H), \quad g_\phi : \mathbb{R}^{Nd} \to \mathbb{R}^K, \tag{6}$$

where K is the number of classes.

3.5 Parameter/Compute Notes

The shared patch MLP parameters scale with $p^2C \cdot d + d^2$, and the classifier scales with $Nd \cdot K$ (plus optional hidden layer). Smaller p increases N (more patches) while enhancing locality; larger d increases representational capacity.

4 Experiments

4.1 Datasets

We evaluate on:

- **MNIST**: 28×28 grayscale digit images (10 classes) [1].
- **Fashion-MNIST**: 28×28 grayscale fashion items (10 classes) [2].
- **SVHN**: 32×32 RGB street-view digit images (10 classes) [3].

4.2 Baselines and Settings

We compare MP-MLP against a lightweight CNN baseline:

- **CNN**: Conv($C \to 16$, 3×3) + ReLU + MaxPool(2×2) + Linear $\to$ 10 classes.

MP-MLP uses non-overlapping patches and a shared 2-layer MLP per patch. Unless specified, we use batch size 64, Adam optimizer (lr $= 10^{-3}$), and train for 8 epochs. For MNIST/Fashion-MNIST we use $p = 4$, $d = 16$; for SVHN we use the same $p = 4$, $d = 16$ with $C = 3$ and $H = W = 32$. (All models converge within 8 epochs under these settings.)

5 Results

5.1 Main Comparison

Table 1 summarizes the best test accuracy and parameter counts. Tables 2–4 report epoch-wise results for MNIST, Fashion-MNIST, and SVHN, respectively.

Table 1: Summary of best test accuracy and parameter counts.

Dataset	Model	# Params	Best Test Acc (%)
MNIST	CNN	31,530	98.37
MNIST	MP-MLP (p=4, d=16)	8,394	96.63
Fashion-MNIST	CNN	31,530	89.79
Fashion-MNIST	MP-MLP (p=4, d=16)	8,394	86.25
SVHN	CNN	41,418	79.72
SVHN	MP-MLP (p=4, d=16)	11,306	**80.57**

5.2 Epoch-wise Results (MNIST)

Table 2: Epoch-wise accuracy on MNIST (8 epochs).

Ep	CNN Train	CNN Test	MP Train	MP Test
1	0.9200	0.9684	0.8615	0.9114
2	0.9726	0.9784	0.9123	0.9220
3	0.9797	0.9801	0.9301	0.9429
4	0.9829	0.9808	0.9446	0.9542
5	0.9848	0.9827	0.9527	0.9586
6	0.9864	0.9837	0.9574	0.9623
7	0.9877	0.9808	0.9608	0.9638
8	0.9889	0.9822	0.9641	0.9663

5.3 Epoch-wise Results (Fashion-MNIST)

Table 3: Epoch-wise accuracy on Fashion-MNIST (8 epochs).

Ep	CNN Train	CNN Test	MP Train	MP Test
1	0.8274	0.8546	0.7764	0.8027
2	0.8789	0.8804	0.8372	0.8311
3	0.8902	0.8864	0.8520	0.8391
4	0.8972	0.8901	0.8586	0.8485
5	0.9029	0.8877	0.8619	0.8445
6	0.9087	0.8957	0.8665	0.8460
7	0.9116	0.8934	0.8693	0.8516
8	0.9157	0.8979	0.8729	0.8625

5.4 Epoch-wise Results (SVHN)

Table 4: Epoch-wise accuracy on SVHN (8 epochs).

Ep	CNN Train	CNN Test	MP Train	MP Test
1	0.6739	0.7719	0.6041	0.7463
2	0.8050	0.7796	0.7846	0.7732
3	0.8205	0.7969	0.8039	0.7819
4	0.8271	0.7966	0.8148	0.7812
5	0.8313	0.7917	0.8217	0.8002
6	0.8346	0.7972	0.8256	0.7995
7	0.8381	0.7952	0.8296	0.8006
8	0.8412	0.7963	0.8322	0.8057

6 Discussion

6.1 Accuracy–Efficiency Trade-off

Across MNIST and Fashion-MNIST, MP-MLP achieves competitive accuracy while reducing parameter count by approximately 73% relative to the CNN baseline (8,394 vs. 31,530). This supports the hypothesis that locality and weight sharing, when implemented via a shared patch MLP, can capture much of the structure needed for lightweight recognition tasks.

6.2 SVHN: RGB Digits with Background

SVHN introduces RGB inputs and real-world background variation. Notably, MP-MLP slightly outperforms the CNN baseline (80.57% vs. 79.72%) while using far fewer parameters (11,306 vs. 41,418). This suggests that for structured tasks such as digit recognition, a shared patch-wise MLP can be an effective convolution-free alternative even in RGB settings.

6.3 Limitations

MP-MLP uses only local patch processing and simple feature concatenation; it does not explicitly model cross-patch interactions or build deep hierarchical representations. Extending MP-MLP with lightweight global mixing or multi-stage patch processing is a promising direction.

7 Conclusion

We introduced MP-MLP, a convolution-free lightweight vision architecture that recovers CNN-like inductive biases—locality and weight sharing—using only fully connected layers. By applying a single shared MLP to non-overlapping micro-patches and aggregating patch features with a shallow classifier, MP-MLP achieves competitive performance on MNIST and Fashion-MNIST with substantially fewer parameters, and slightly surpasses a lightweight CNN baseline on SVHN. These results highlight that carefully designed weight-shared MLPs can serve as practical convolution-free alternatives for structured lightweight vision tasks.

References

[1] Y. LeCun, C. Cortes, and C. J. C. Burges, "The MNIST database of handwritten digits," 1998.

[2] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms," *arXiv:1708.07747*, 2017.

[3] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Ng, "Reading Digits in Natural Images with Unsupervised Feature Learning," in *NIPS Workshop*, 2011.

[4] A. G. Howard *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," *arXiv:1704.04861*, 2017.

[5] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices," in *Proc. CVPR*, 2018.

[6] I. Tolstikhin *et al.*, "MLP-Mixer: An all-MLP Architecture for Vision," in *Proc. NeurIPS*, 2021.

[7] H. Liu *et al.*, "Pay Attention to MLPs," in *Proc. NeurIPS*, 2021.

[8] H. Touvron *et al.*, "ResMLP: Feedforward networks for image classification with data-efficient training," *arXiv:2105.03404*, 2021.

[9] A. Dosovitskiy *et al.*, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," in *Proc. ICLR*, 2021.