**SO SÁNH CÁC KIẾN TRÚC CNN TRÊN TẬP DỮ LIỆU CIFAR-10**

Tác giả: [Trần Nguyên Hoàng], Khoa CNTT – EAUT | Ngày: 10/09/2025

# Tóm tắt

Bài báo đánh giá thực nghiệm các kiến trúc CNN (SimpleCNN, VGG-11-BN, ResNet-18, MobileNetV2, EfficientNet-B0) trên CIFAR-10 (60.000 ảnh màu 32×32, 10 lớp). Thiết lập huấn luyện thống nhất (augmentation cơ bản, SGD/AdamW, Cosine LR), báo cáo Acc, F1-macro, số tham số, FLOPs, độ trễ suy luận trên CPU/GPU. Kết quả chỉ ra ... [viết 5–7 dòng].

# Từ khóa

CIFAR-10; CNN; ResNet-18; MobileNetV2; EfficientNet-B0; phân loại ảnh.

# 1. Giới thiệu

• Đặt vấn đề: nhu cầu so sánh mô hình CNN trên dữ liệu ảnh nhỏ (32×32) phổ biến.  
• Khoảng trống: thiếu báo cáo gọn nhẹ, thống nhất tiêu chí, dễ tái lập cho bối cảnh giáo dục/triển khai tài nguyên hạn chế.  
• Đóng góp: (i) recipe huấn luyện thống nhất; (ii) so sánh đa tiêu chí (độ chính xác & chi phí); (iii) công bố mã & log.

# 2. Công trình liên quan

Tóm lược VGG → ResNet → MobileNet → EfficientNet và các thực hành huấn luyện hiện đại (augmentation, optimizer, schedule). Nhấn mạnh điểm mạnh/yếu của từng họ kiến trúc trên ảnh 32×32.

# 3. Dữ liệu & tiền xử lý

• CIFAR-10: 50k train / 10k test, 10 lớp, ảnh màu 32×32; mean/std: (0.4914, 0.4822, 0.4465) / (0.2470, 0.2435, 0.2616).  
• Chia tập: train/val = 90/10 từ train; seed cố định.  
• Augmentation: RandomCrop(32, pad=4), RandomHorizontalFlip; Normalize theo mean/std; batch 128.  
• (Tuỳ chọn) Fine-tune từ ImageNet với mô hình lớn hơn để rút ngắn thời gian (nếu có).

# 4. Mô hình & thiết lập huấn luyện

• Mô hình: SimpleCNN; VGG-11-BN; ResNet-18; MobileNetV2; EfficientNet-B0 (num\_classes=10).  
• Siêu tham số: epochs 100 (60 nếu gấp), batch 128; optimizer: SGD(momentum=0.9, wd=5e-4, lr0=0.1) hoặc AdamW(lr0=3e-4, wd=1e-4); scheduler: Cosine/Step; label smoothing 0.1; AMP nếu có GPU.  
• Thực thi: lặp 3 seed, báo cáo mean±std; chọn best theo val-acc.

# 5. Thiết kế thí nghiệm

• Tiêu chí: Acc, F1-macro, Params (M), FLOPs (GMac), latency (ms/ảnh).  
• Ablation: (i) augmentation on/off; (ii) SGD vs AdamW; (iii) cosine vs step.  
• Báo cáo: Bảng chính + đường cong học + ma trận nhầm lẫn + scatter Acc–Params/FLOPs.

# 6. Kết quả & thảo luận

• Bảng tổng hợp kết quả chính (mean±std, 3 seeds) trên test.  
• Phân tích chi phí–hiệu quả: khi nào nên chọn mô hình nhẹ so với mô hình sâu hơn.  
• Phân tích lỗi với ví dụ minh hoạ; (tuỳ chọn) Grad-CAM cho 1–2 mẫu.

# 7. Kết luận

Tóm tắt phát hiện; khuyến nghị mô hình theo ràng buộc tài nguyên; hướng mở (tăng epochs, RandAugment, MixUp/CutMix, KD).

# 8. Tái lập & công bố

Cấu trúc mã, lệnh chạy, seed, checkpoint, log CSV; đường dẫn GitHub (sau khi ẩn thông tin nhạy cảm).

# Phụ lục — Danh mục Bảng/Hình

Bảng 1. Cấu hình huấn luyện.  
Bảng 2. Thống kê CIFAR-10.  
Bảng 3. Tham số & FLOPs từng mô hình.  
Bảng 4. Kết quả chính (mean±std, 3 seeds).  
Bảng 5. Ablation.  
Hình 1. Pipeline.  
Hình 2. Đường cong học.  
Hình 3. Ma trận nhầm lẫn.  
Hình 4. Biểu đồ Acc vs Params/FLOPs.

# Tài liệu tham khảo

[1] A. Krizhevsky, “Learning Multiple Layers of Features from Tiny Images,” Technical Report, University of Toronto, 2009. (CIFAR-10)

[2] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” arXiv:1409.1556, 2014. (VGG)

[3] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in Proc. CVPR, 2016. (ResNet)

[4] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” in Proc. CVPR, 2018.

[5] M. Tan and Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” in Proc. ICML, 2019.

[6] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2016. (Chương CNN)

[7] R. C. Gonzalez and R. E. Woods, Digital Image Processing, 4th ed., Pearson, 2018.

[8] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, “mixup: Beyond Empirical Risk Minimization,” in Proc. ICLR, 2018.

[9] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, “AutoAugment: Learning Augmentation Policies from Data,” in Proc. CVPR, 2019.

[10] I. Loshchilov and F. Hutter, “Decoupled Weight Decay Regularization,” in Proc. ICLR, 2019. (AdamW)

[11] I. Loshchilov and F. Hutter, “SGDR: Stochastic Gradient Descent with Warm Restarts,” in Proc. ICLR, 2017. (Cosine schedule)