Transfer Learning for Computer Vision: Practical Techniques for CNNs and Vision Transformers

Debabrata Pruseth
AI Architect & Applied AI Researcher
Singapore

Author Note

This article is a research-style companion version of author's blog post " Transfer learning for Vision"

This research presents an applied framework for selecting and implementing transfer learning strategies in modern computer vision systems. The study focuses on practical adaptation techniques for convolutional neural networks, Vision Transformers, multimodal vision-language models, segmentation foundation models, lightweight adapters, and self-supervised pretraining methods.

The objective of this work is to provide a structured, practitioner-oriented research framework for adapting pretrained visual models to real-world domains where labeled data, compute resources, and training time are constrained. The study is positioned within applied AI research and emphasizes implementation logic, model selection, fine-tuning strategy, troubleshooting, and responsible deployment.

This work does not present a new benchmark dataset or a new model architecture. Instead, it contributes a practical transfer learning framework that can guide applied AI teams in choosing suitable techniques for image classification, segmentation, multimodal visual understanding, and domain-specific computer vision tasks.

Abstract

Transfer learning has become a central technique in modern computer vision because it enables high-performing visual recognition systems to be developed with limited labelled data, reduced compute cost, and shorter experimentation cycles. Earlier transfer learning workflows were dominated by convolutional neural networks such as ResNet and EfficientNet, where pretrained feature extractors were adapted to downstream classification, detection, or segmentation tasks. More recently, Vision Transformers, multimodal vision-language models, and foundation segmentation models have expanded the transfer learning landscape by introducing patch-based representation learning, self-attention, prompt-based

adaptation, and parameter-efficient fine-tuning. This paper presents a practical research-oriented overview of transfer learning for computer vision, with emphasis on model selection, adaptation strategies, fine-tuning recipes, self-supervised pretraining, and troubleshooting. It compares feature extraction, partial fine-tuning, full fine-tuning, adapter-based learning, LoRA-style parameter-efficient adaptation, masked autoencoding, contrastive learning, and prompt-based transfer. The objective is to provide a structured decision framework for practitioners who need to adapt CNNs, Vision Transformers, CLIP-like models, or segmentation foundation models to real-world domains such as medical imaging, agriculture, satellite analysis, retail, and industrial inspection.

Keywords: transfer learning, computer vision, convolutional neural networks, Vision Transformer, ViT, LoRA, adapters, CLIP, SAM, MAE, self-supervised learning, fine-tuning.

---

1. Introduction

Computer vision models are rarely trained from scratch in practical settings. Modern visual recognition systems typically begin with a pretrained model that has already learned reusable low-level and high-level visual representations from large-scale image datasets. These representations may include edges, textures, shapes, object parts, spatial relationships, and semantic patterns that can be transferred to new downstream tasks.

The value of transfer learning is particularly strong in domains where labelled data is scarce, expensive, sensitive, or difficult to curate. Medical imaging, remote sensing, agriculture, manufacturing inspection, and document intelligence often involve specialised image distributions that differ from general internet-scale datasets. Training a high-capacity model from scratch in such settings may require substantial labelled data, expert annotation, and compute infrastructure. Transfer learning reduces this burden by adapting pretrained visual representations rather than learning all visual features independently.

The evolution of transfer learning in vision can be understood in two major phases. The first phase was driven by convolutional neural networks such as ResNet and EfficientNet, which provided strong reusable backbones for image classification and downstream tasks. ResNet introduced residual learning to make very deep networks easier to optimise, while EfficientNet demonstrated that careful compound scaling of depth, width, and resolution could improve accuracy-efficiency trade-offs.

The second phase has been driven by Vision Transformers and vision foundation models. Vision Transformers represent images as sequences of patches and use self-attention to model relationships across the image. When pretrained at scale, ViTs transfer effectively to downstream recognition tasks and challenge the assumption that convolution is always necessary for visual representation learning.

This paper examines how these model families can be adapted in practice and proposes a structured framework for choosing transfer learning strategies under different data, compute, and domain-shift conditions.

---

2. Background: From CNN-Based Transfer to Vision Transformers

Convolutional neural networks learn visual features through local receptive fields, hierarchical feature maps, and translation-equivariant operations. Early layers often capture generic structures such as edges and textures, while deeper layers represent object parts and semantic concepts. This layered representation makes CNNs natural candidates for transfer learning: a pretrained backbone can be frozen, partially unfrozen, or fully fine-tuned depending on the downstream task.

ResNet became one of the most widely used CNN backbones because residual connections allow deeper models to be trained more reliably. EfficientNet extended the CNN transfer learning landscape by showing that network depth, width, and input resolution should be scaled together rather than independently. These models remain practical for many production systems because they are computationally efficient, stable, and well supported in machine learning libraries.

Vision Transformers introduced a different paradigm. Instead of applying convolutional filters across an image, a ViT divides the image into fixed-size patches, projects those patches into embeddings, adds positional information, and processes the resulting sequence using Transformer blocks. This design allows the model to capture long-range dependencies using self-attention. ViTs can be data-hungry when trained from scratch, but they become highly effective when pretrained on large datasets and transferred to downstream tasks.

The rise of multimodal and foundation models has further expanded transfer learning. CLIP learns transferable visual representations through natural language supervision, enabling zero-shot and prompt-based image classification. SAM introduced a promptable segmentation model trained on a large-scale segmentation dataset, enabling broad transfer to new segmentation tasks.

---

3. Why Transfer Learning Matters in Computer Vision

Transfer learning addresses three recurring constraints in real-world computer vision projects.

First, labelled image data is often limited. In medical diagnosis, crop disease detection, defect inspection, and satellite analysis, expert labels may be expensive or slow to obtain. Transfer learning allows practitioners to start from a representation learned elsewhere and adapt it using a smaller labelled dataset.

Second, compute resources are limited. Training a high-capacity CNN, ViT, or foundation model from scratch can require significant GPU resources. Fine-tuning a pretrained model, especially with frozen layers or parameter-efficient adapters, is substantially more practical for organisations without large-scale training infrastructure.

Third, deployment timelines are often short. Business and research teams usually need working baselines quickly. Transfer learning enables rapid prototyping: a pretrained

backbone can be paired with a new classification head, evaluated, and then progressively adapted if the initial baseline is insufficient.

These benefits do not eliminate the need for careful evaluation. Transfer learning can fail when the source and target domains differ sharply, when the downstream dataset is noisy, when the learning rate is too high, or when the model overfits a small target dataset. Therefore, transfer learning should be treated as a controlled adaptation process rather than a simple reuse of pretrained weights.

---

4. Taxonomy of Transfer Learning Strategies

4.1 Feature Extraction

Feature extraction is the simplest transfer learning strategy. The pretrained backbone is frozen, and only a new task-specific head is trained. For example, a pretrained ResNet, EfficientNet, or ViT can be used as a fixed feature extractor, while a new classifier is trained for plant disease classification, skin lesion classification, or product category recognition.

This strategy is most useful when the target dataset is small and visually similar to the pretraining dataset. It is computationally efficient and reduces overfitting risk because only a small number of parameters are updated.

4.2 Partial Fine-Tuning

Partial fine-tuning unfreezes only the upper layers or final blocks of the pretrained model. In CNNs, this may involve unfreezing the last convolutional stage. In ViTs, this may involve unfreezing the final few Transformer blocks while keeping earlier blocks frozen.

This method provides a balance between stability and adaptability. Lower layers retain general visual representations, while higher layers adapt to the target task. Partial fine-tuning is often a strong default when the target domain differs moderately from the source domain.

4.3 Full Fine-Tuning

Full fine-tuning updates all model parameters. This strategy can produce strong results when the target dataset is sufficiently large and representative. However, it is more computationally expensive and increases the risk of catastrophic forgetting, especially when the target dataset is small.

Full fine-tuning is appropriate when there is a large labelled dataset, meaningful domain shift, and enough compute to train carefully with small learning rates, regularisation, and validation monitoring.

4.4 Parameter-Efficient Fine-Tuning

Parameter-efficient fine-tuning updates only a small subset of parameters while keeping most pretrained weights frozen. Adapters, LoRA-style low-rank updates, and bias-only methods are examples of this approach.

LoRA freezes pretrained weights and injects trainable low-rank matrices into selected layers, reducing the number of trainable parameters. Although originally popularised in language models, the same principle can be adapted to Transformer-based vision models where full fine-tuning is costly.

Parameter-efficient adaptation is especially useful when multiple downstream tasks must share one base model, when GPU memory is limited, or when model governance requires preserving a stable pretrained foundation.

4.5 Self-Supervised Pretraining

Self-supervised learning is useful when labelled data is scarce but unlabelled images are available. Methods such as SimCLR, DINO, and MAE learn useful visual representations without conventional human labels. SimCLR uses contrastive learning, DINO explores self-supervised learning with Vision Transformers, and MAE reconstructs masked image patches using an asymmetric encoder-decoder design.

A practical workflow is to first pretrain on unlabelled domain-specific images, then fine-tune on a smaller labelled dataset. This can be valuable in medical imaging, industrial inspection, satellite imagery, and scientific imaging.

4.6 Prompt-Based and Multimodal Transfer

Prompt-based transfer is increasingly important for multimodal models such as CLIP. Instead of training a classifier directly, a practitioner can define text prompts such as “a photo of a diseased leaf” or “an X-ray showing pneumonia” and compare text-image embedding similarity. CLIP demonstrated that natural language supervision can produce transferable visual models capable of zero-shot classification across many datasets.

This approach is attractive when rapid experimentation is needed, when labels are unavailable, or when the set of classes may change frequently. However, performance depends heavily on prompt design and domain alignment.

---

5. Practical Fine-Tuning Methodology

A robust transfer learning workflow should begin with the simplest stable baseline and increase adaptation complexity only when needed.

5.1 Stage 1: Train the Task Head

The first stage is to freeze the pretrained backbone and train only the task-specific head. This establishes a baseline and helps determine whether the pretrained representation is already

sufficient. For classification, the head may be a linear layer or small multilayer perceptron. For segmentation, the head may be a decoder or mask prediction module.

5.2 Stage 2: Unfreeze Higher Layers

If the frozen-backbone baseline underperforms, the next step is to unfreeze the final blocks. In a ViT-B/16 model, for example, the final two to four Transformer blocks may be unfrozen. In a CNN, the last convolutional stage may be unfrozen.

This stage should use a smaller learning rate for pretrained layers than for the newly initialised task head. The head must learn rapidly, while pretrained layers should be modified conservatively.

5.3 Stage 3: Progressive Unfreezing

Progressive unfreezing adapts the model gradually by unfreezing deeper layers in stages. This reduces the risk of destabilising pretrained representations. It is useful when there is moderate to high domain shift and the practitioner wants more adaptation without immediately updating the full model.

5.4 Stage 4: Discriminative Learning Rates

Discriminative learning rates assign different learning rates to different parts of the model. A practical configuration is:

Model Component	Suggested Learning Rate Behaviour
Newly added head	Highest learning rate
Final pretrained blocks	Lower learning rate
Middle pretrained blocks	Very low learning rate
Early pretrained blocks	Frozen or extremely low learning rate

This approach reflects the intuition that earlier layers capture more general visual patterns, while later layers are more task-specific.

5.5 Normalisation Considerations

CNNs often rely on Batch Normalization, which uses mini-batch statistics and may become unstable with very small batch sizes. Vision Transformers commonly use Layer Normalization, which computes statistics within each individual example and is less dependent on batch size. Batch Normalization and Layer Normalization were introduced as distinct normalisation approaches with different statistical assumptions and training behaviours.

When fine-tuning CNNs with small batches, practitioners should carefully manage BatchNorm behaviour. In some cases, freezing BatchNorm statistics improves stability.

---

6. Model-Specific Adaptation Patterns

6.1 CNN Backbones

CNNs remain strong baselines for many practical vision tasks. For small and medium-sized datasets, ResNet and EfficientNet are often easier to train and deploy than larger Transformer-based models. A recommended workflow is to train a frozen-backbone classifier first, then fine-tune the last convolutional stage if needed.

6.2 Vision Transformers

ViTs are effective when pretrained at scale and transferred carefully. Because they contain large numbers of parameters and can overfit small datasets, it is often advisable to begin with head-only training, then unfreeze the last few Transformer blocks.

Data augmentation is important for ViT fine-tuning. Techniques such as random resized crops, colour jitter, Mixup, CutMix, RandAugment, and regularisation may improve generalisation, depending on the task.

6.3 CLIP-Style Models

CLIP-style models are useful when the task can be expressed through natural language prompts. They are particularly valuable for zero-shot classification, rapid prototyping, image retrieval, and open-vocabulary recognition. However, prompt sensitivity must be evaluated. A poor prompt may underrepresent the target class, while multiple prompt templates may improve robustness.

6.4 SAM and Segmentation Foundation Models

SAM introduced a promptable segmentation framework trained on a very large segmentation dataset. It can be useful for zero-shot or interactive segmentation, but domain-specific segmentation tasks may still require adaptation. A practical strategy is to keep the image encoder frozen and fine-tune lightweight task-specific components such as mask decoders, prompt encoders, or downstream segmentation heads.

---

7. Decision Framework

The choice of transfer learning strategy should depend on dataset size, domain shift, task type, compute availability, and deployment constraints.

Scenario	Recommended Strategy
Small dataset, low domain shift	Frozen backbone + trained head
Small dataset, high domain shift	Adapters, LoRA, or progressive unfreezing
Medium dataset, moderate domain shift	Fine-tune final CNN stage or final ViT blocks

Scenario	Recommended Strategy
Large labelled dataset	Full fine-tuning with small learning rates
Large unlabelled domain dataset	Self-supervised pretraining followed by fine-tuning
No labels, flexible class definitions	CLIP-style prompt-based classification
Segmentation task	SAM, ViT-decoder, or domain-specific segmentation head
Low compute environment	Feature extraction or parameter-efficient fine-tuning
Multi-client or multi-task deployment	Shared frozen backbone with adapters per task

---

8. Common Failure Modes and Remedies

Transfer learning failures are often caused by unstable optimisation, insufficient adaptation, overfitting, or domain mismatch.

Failure Mode	Likely Cause	Practical Remedy
Validation loss rises after unfreezing	Learning rate too high	Reduce pretrained-layer learning rate
Head-only training underperforms	Representation not task-specific enough	Unfreeze final blocks or add adapters
Training accuracy high but validation poor	Overfitting	Freeze more layers, add augmentation, use dropout
Model performance degrades after full fine-tuning	Catastrophic forgetting	Use discriminative learning rates or LoRA
Poor performance on medical/satellite images	High domain shift	Use self-supervised domain pretraining
Unstable CNN fine-tuning with small batches	BatchNorm statistics unreliable	Freeze BatchNorm or increase effective batch size
CLIP zero-shot results inconsistent	Prompt sensitivity	Use prompt ensembling and domain-specific wording

---

9. Discussion

The practical value of transfer learning lies not only in higher accuracy but also in operational feasibility. In enterprise and applied research environments, model development must consider cost, governance, repeatability, deployment latency, and maintainability. A full fine-tuned model may deliver high task performance but may be harder to maintain across many clients or use cases. A frozen foundation model with lightweight adapters may be easier to govern and cheaper to update.

CNNs remain relevant because they are efficient, stable, and well understood. ViTs and foundation models provide stronger flexibility for large-scale and multimodal tasks, but they require careful fine-tuning discipline. Self-supervised methods are especially important when organisations possess large unlabelled image repositories but limited expert annotations.

The most reliable strategy is incremental adaptation. Practitioners should first establish a frozen-backbone baseline, then gradually increase model flexibility through partial fine-tuning, adapters, or self-supervised pretraining. This avoids unnecessary complexity and makes performance improvements easier to attribute.

---

10. Conclusion

Transfer learning is a foundational technique for modern computer vision. It enables practitioners to adapt pretrained CNNs, Vision Transformers, multimodal models, and segmentation foundation models to specialised downstream tasks with reduced data and compute requirements. The choice of strategy should be guided by domain similarity, dataset size, task type, compute budget, and deployment needs.

For small datasets with low domain shift, feature extraction is often sufficient. For moderate domain shift, partial fine-tuning with discriminative learning rates provides a strong balance. For high domain shift or low-label settings, self-supervised pretraining, adapters, or LoRA-style adaptation can improve performance while preserving efficiency. For open-vocabulary and segmentation tasks, CLIP-like and SAM-like models provide flexible foundations, although domain validation remains essential.

A disciplined transfer learning workflow should therefore combine pretrained representations, staged adaptation, careful learning-rate control, strong validation, and explicit troubleshooting. This approach makes computer vision development more practical, scalable, and reliable across real-world domains.

References

Pruseth, D. (2026). Transfer Learning for Vision. Debabrata Pruseth AI Blog.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., & Joulin, A. (2021). *Emerging properties in self-supervised Vision Transformers*. arXiv.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2020). *An image is worth 16x16 words: Transformers for image recognition at scale*. arXiv

He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Deep residual learning for image recognition*. CVPR.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., & Girshick, R. (2022). *Masked Autoencoders are scalable vision learners*. CVPR.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., & Chen, W. (2021). *LoRA: Low-rank adaptation of large language models*. arXiv.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollár, P., & Girshick, R. (2023). *Segment Anything*. arXiv.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021). *Learning transferable visual models from natural language supervision*. arXiv

Tan, M., & Le, Q. V. (2019). *EfficientNet:Rethinking model scaling for convolutional neural networks*. arXiv

Suggested Citation

Pruseth, D. (2026). *Transfer Learning for Computer Vision: Practical Techniques for CNNs and Vision Transformers*.