Assignment 0 Report: Exploring Deep Learning Architectures and Dynamics

Ahmed Hassan
26100308@lums.edu.pk

Course: CS6304 – Advanced Topics in Machine Learning

Instructor: Dr. Muhammad Tahir

Repository:
https://github.com/tk1475/ATML-PA0

Abstract

This report presents experiments conducted as part of Assignment 0 for CS6304: Advanced Topics in Machine Learning. We investigate the inner workings of ResNet-152, analyze Vision Transformer attention maps, study GAN training dynamics, explore posterior collapse in VAEs, and examine the modality gap in CLIP. The report emphasizes reproducibility, systematic experimentation, and research-style documentation.

1. Introduction

Deep learning has enabled transformative advances across computer vision, natural language processing, and multimodal tasks. However, training stability, representation learning, and cross-modal alignment remain central challenges. This assignment investigates these themes through a series of targeted experiments.

The overarching problem is to understand the behavior of state-of-the-art architectures under different conditions, and to critically evaluate their strengths, limitations, and tradeoffs.

- **ResNet:** Transfer learning, residual connections, and feature hierarchies.
- **Vision Transformers (ViTs):** Interpretability, robustness, and pooling strategies.
- **GANs:** Adversarial dynamics, vanishing gradients, mode collapse, and overfitting.
- **Variational Autoencoders (VAEs):** Reconstruction–regularization trade-offs, posterior collapse, and KL annealing.
- **CLIP:** Zero-shot classification, modality gap, and embedding alignment.

2. Task 1: ResNet-152

2.1. Methodology

A pretrained ResNet-152 was adapted to CIFAR-10 by replacing its final layer with a 10-class head and training only this layer while freezing the backbone. We also conducted controlled experiments to study the role of residual connections, feature hierarchies, transfer learning strategies, and model depth. Visualization methods (UMAP, t-SNE) and confusion matrix analysis were employed to better understand the learned feature spaces.

2.2. Results

Baseline Setup. Validation accuracy reached ∼85% within 5 epochs, demonstrating the effectiveness of transfer learning.

Epoch	Train Acc (%)	Val Acc (%)
1	83.10	82.72
2	83.29	84.23
3	83.66	84.97
4	83.75	85.23
5	84.14	83.81

Table 1. Training and validation accuracy for ResNet-152 baseline (frozen backbone).

Residual Connections. After removing skip connections, training collapsed to near-random accuracy:

Feature Hierarchies. UMAP visualizations show early layers encode low-level features, while penultimate layers form linearly separable clusters.

Epoch	Train Acc (%)	Val Acc (%)
1	12.18	13.45
2	13.34	13.01
3	13.39	13.38
4	13.35	14.01
5	13.22	14.05

Table 2. Training and validation accuracy after removing skip connections in ResNet-152.

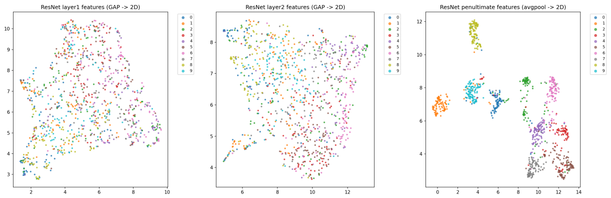


Figure 1. UMAP projections of CIFAR-10 features at different depths.

Transfer Learning Strategies. We compared different fine-tuning strategies:

Table 3. Comparison of transfer learning strategies on CIFAR-10 (5 epochs).

Setting	Final Train Acc (%)	Final Val Acc (%)
Pretrained + FC only	85.94	84.66
Random Initialization	78.68	77.30
Pretrained + Layer4 + FC	**98.65**	**92.02**

Optional Analyses. t-SNE vs. UMAP: UMAP gave tighter clusters. Confusion matrix: revealed confusion between semantically similar classes (e.g., cat vs. dog).

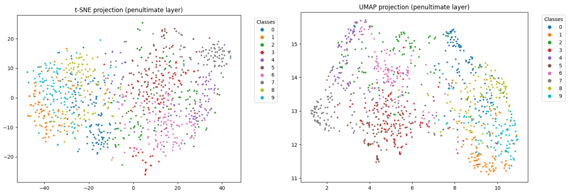


Figure 2. t-SNE (left) vs UMAP (right) projections of ResNet-152 penultimate features on CIFAR-10.

ResNet-152 vs. ResNet-18. The deeper ResNet-152 achieved higher accuracy but required more computation:

2.3. Discussion

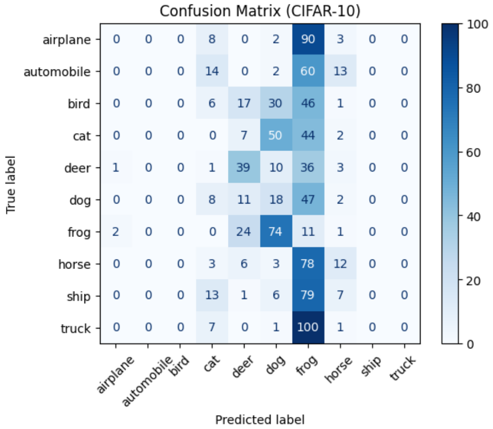


Figure 3. Confusion Matrix for ResNet-152 on CIFAR-10 (subset).

Table 4. Comparison of ResNet-18 vs ResNet-152 (Pretrained + FC fine-tuned, 3 epochs, CIFAR-10 subset).

Model	Final Train Acc	Final Val Acc
ResNet-18	76.64%	74.10%
ResNet-152	80.76%	76.50%

The experiments show that transfer learning with ResNet-152 is highly effective on CIFAR-10. - **Baseline**: Training only the classifier head quickly achieved 85% accuracy. - **Residual connections**: Essential for optimization—without (He et al., 2016). them, accuracy collapsed to ∼13%. - **Feature hierarchies**: Early features capture edges/textures, deeper layers form discriminative embeddings. - **Transfer learning strategies**: Fine-tuning the last residual block with the classifier yielded the best balance (92% accuracy). - **Optional analyses**: UMAP was superior for visualization; confusion matrix highlighted semantic misclassifications. - **Depth comparison**: ResNet-152 outperformed ResNet-18 but at higher compute cost, suggesting ResNet-18 is more practical for constrained settings.

Overall, Task 1 illustrates the power of residual learning, transfer learning, and hierarchical feature representations in deep CNNs.

3. Task 2: Understanding Vision Transformers

3.1. Methodology

We used a pretrained ViT-B/16 model (ImageNet-21k → ImageNet-1k pretrained) to evaluate classification on custom images (cat, dog, car). Experiments included:

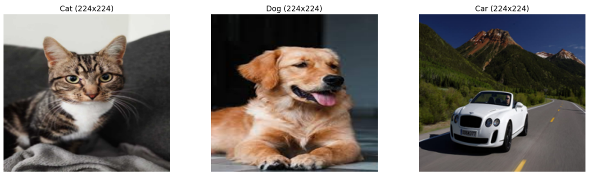
- Evaluating top-1 predictions on resized inputs (224×224).
- Extracting and visualizing patch-level attention maps from the final transformer layer.
- Analyzing attention map interpretability compared to

CNN-based CAM methods.

- Testing robustness under random and structured patch masking.
- Comparing linear probes using `[CLS]` token vs. mean pooling of patch tokens on CIFAR-10 subset.

3.2. Results

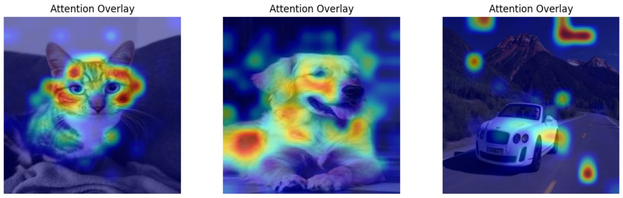
Classification. ViT produced fine-grained predictions such as “Tabby Cat” and “Golden Retriever”, demonstrating transferable semantic knowledge.



Tabby Cat **Golden Retriever** **Sports Car**

Figure 4. Top-1 predictions from ViT-B/16 on custom images (224×224).

Attention Maps. ViT attention highlights object regions directly, unlike CNNs that require CAM-based posthoc methods.



Cat **Dog** **Car**

Figure 5. Attention heatmaps from ViT-B/16. Cat and dog heads emphasize face/body; car heads are more diffuse.

Masked Patches. We tested robustness to random (30%) vs. structured masking (central block).

Table 5. Effect of random vs. structured patch masking on ViT-B/16 predictions.

Image	Original	Random Mask	Center Mask
Dog	Golden Retriever (97.40%)	Golden Retriever (96.81%)	Golden Retriever (96.39%)
Cat	Tabby Cat (73.67%)	Tabby Cat (79.65%)	Tabby Cat (69.72%)
Car	Sports Car (93.18%)	Convertible (78.39%)	Sports Car (53.86%)

Linear Probes. We compared `[CLS]` token vs. mean pooling strategies on CIFAR-10:

Table 6. Linear probe validation accuracy (ViT-B/16 features on CIFAR-10 subset).

Pooling method	Validation Accuracy (%)
CLS token	91.90
Mean of patch tokens	**93.50**

3.3. Discussion

Our experiments highlight several key insights about Vision Transformers (Dosovitskiy et al., 2021):

- **Classification:** ViT produced fine-grained, transferable predictions, confirming its pretrained semantic richness.
- **Attention Interpretability:** Certain heads aligned with salient objects (cat/dog faces), while others captured context (car image). Unlike CNNs, ViTs provide native attention maps without gradient-based post-processing.
- **Robustness:** Random masking caused minimal degradation, confirming ViTs’ global information distribution. Structured masking (central block) significantly reduced confidence, especially for the car, showing sensitivity to systematic occlusion.
- **Pooling Strategies:** Mean pooling outperformed CLS pooling (93.5% vs 91.9%), suggesting that distributed patch embeddings carry richer transferable information than the single CLS token.

In summary, Task 2 demonstrates that ViTs offer built-in interpretability through attention maps, robustness to random patch corruption, and flexible pooling strategies for downstream tasks. However, attention interpretability can vary across heads, and structured occlusions remain a limitation.

4. Task 3: GAN Dynamics on MNIST

4.1. Methodology

We implemented and evaluated a simple MLP-based GAN on MNIST.

- **Generator (G):** Noise $z \in \mathbb{R}^{100} \rightarrow$ FC layers ($100 \rightarrow 256 \rightarrow 512 \rightarrow 784$) with ReLU activations, $\tanh$ output reshaped to 28×28.
- **Discriminator (D):** FC layers ($784 \rightarrow 256 \rightarrow 256 \rightarrow 1$) with LeakyReLU activations, final sigmoid output.
- **Training setup:** Normal initialization ($\sigma = 0.02$), Adam ($lr = 2 \times 10^{-4}, \beta_1 = 0.5, \beta_2 = 0.999$), binary cross-entropy loss, batch size 64, trained for 20 epochs.

- **Experiments:**
  - Baseline GAN training dynamics.
  - Comparison with Erik Linder-Norén's `PyTorch-GAN` reference implementation.
  - Training instabilities: vanishing gradients, mode collapse, and discriminator overfitting.
  - Mitigation strategies: label smoothing, balanced updates, adjusted learning rates, dropout regularization.

4.2. Results

Baseline GAN Training. Losses oscillated as expected in adversarial training (Fig. 6), with D loss around 1.2–1.4 and G loss 0.7–1.0. Generated samples (Fig. 7) showed digit-like structures, albeit blurry.

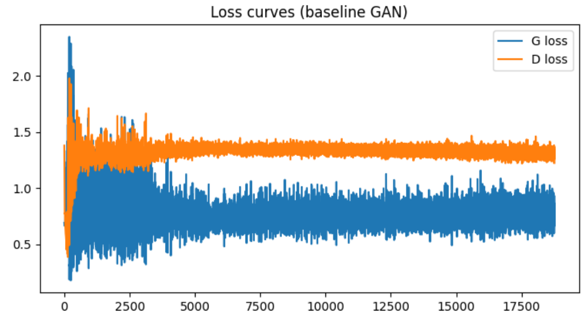


Figure 6. Loss curves for Generator (G) and Discriminator (D). Oscillations reflect adversarial dynamics.

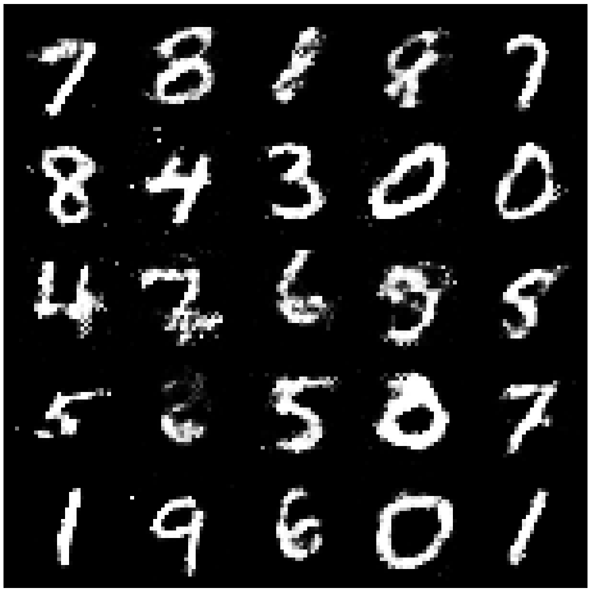


Figure 7. Generated samples after 20 epochs. Digits are recognizable but blurry.

Comparison with Reference Implementation. Our model produced blurry digits after 20 epochs. The reference `PyTorch-GAN` (200+ epochs, stronger batch normalization) produced sharper, more diverse digits. Differences stemmed from:

- Longer training improving convergence.
- Batch normalization stabilizing training.
- Random initialization/seed variability.

Vanishing Gradients. With an overpowering discriminator, G gradients vanished (Table 1). After applying label smoothing and balanced updates, training stabilized (Table 2).

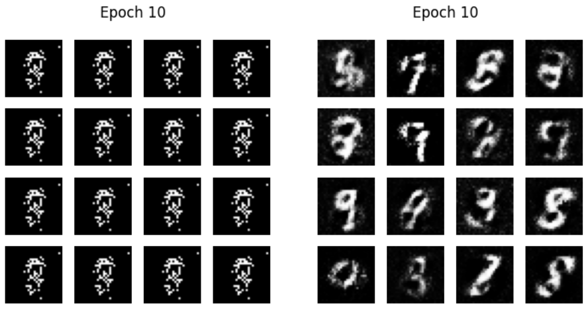
Table 7. Discriminator dominates, leading to vanishing gradients.

Epoch	D_{loss}	G_{loss}	$D_{\text{real avg}}$	$D_{\text{fake avg}}$
1	0.0003	0.0002	1.000	0.000
2	0.0001	0.0000	1.000	0.000
8	0.0000	0.0000	1.000	0.000

Table 8. After applying label smoothing + balanced updates.

Epoch	D_{loss}	G_{loss}	$D_{\text{real avg}}$	$D_{\text{fake avg}}$
1	0.9549	0.7781	0.875	0.465
2	0.7913	1.0109	0.879	0.370
3	0.7567	1.1723	0.827	0.318
5	0.7742	1.3127	0.752	0.308

Mode Collapse. With high generator LR and low discriminator LR, G collapsed to producing only digit **8** (Fig. 8). Balanced updates, learning rate adjustment, and minibatch discrimination restored diversity.



(a) Mode Collapse **(b) After Mitigation**

Figure 8. Generator outputs. Left: collapsed to mostly digit 8. Right: diverse outputs after mitigation.

Discriminator Overfitting. On a reduced dataset (1,000 images), a high-capacity D without dropout overfit and provided poor gradients to G. Adding Dropout (0.4) improved generalization and G diversity (Fig. 9).

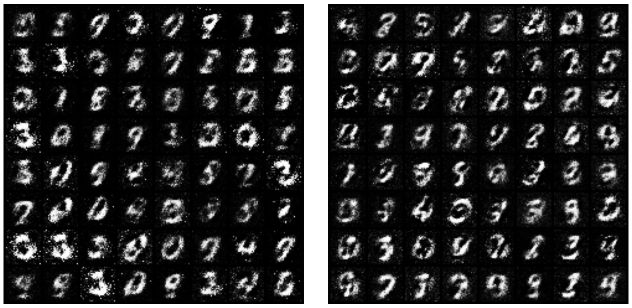


Figure 8. Overfitting D (no Dropout).

Figure 8. Regularized D (Dropout 0.4).

Figure 9. GAN outputs under overfitting vs. regularized discriminator.

4.3. Discussion

Task 3 highlights critical aspects of GAN dynamics:

- **Baseline:** Even simple MLP GANs generate digit-like outputs, though blurry without long training or normalization.
- **Vanishing gradients:** Overpowering D eliminated G gradients; mitigated by label smoothing and balanced training.
- **Mode collapse:** High G LR led to collapse (digit 8 only); mitigations restored diversity.
- **Discriminator overfitting:** Without regularization, D memorized training data and failed to guide G; dropout improved stability and diversity.

Overall, GAN stability depends on maintaining adversarial balance. Careful control of learning rates, training ratios, and regularization ensures meaningful gradients for the generator and prevents degenerate equilibria such as vanishing gradients or mode collapse.

5. Task 4: Variational Autoencoders (VAEs)

5.1. Methodology

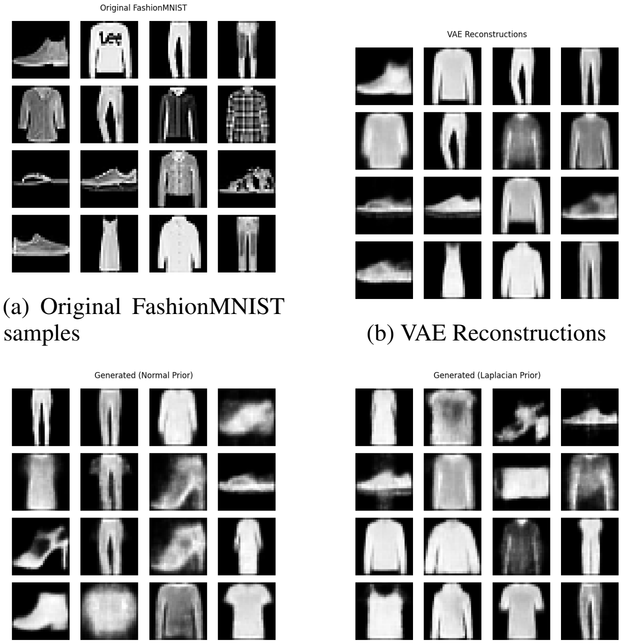
A convolutional VAE was trained on the FashionMNIST dataset with latent dimension 20.

- **Architecture:** Convolutional encoder–decoder with latent $z \in \mathbb{R}^{20}$.
- **Loss:** Combination of reconstruction MSE and KL divergence:
$$\mathcal{L} = \text{MSE}(x, \hat{x}) + D_{KL}(q_\phi(z|x)||p(z)).$$
- **Training:** Adam optimizer ($lr = 1e{-}3$), 20 epochs.
- **Experiments:**
  - Baseline training convergence on FashionMNIST.
  - Reconstructions and generations from different priors ($\mathcal{N}(0, 1)$, Laplace).
  - Posterior collapse analysis via ELBO components.
  - Mitigation through KL annealing schedule:
$$\beta = \min\left(1.0, \frac{\text{epoch}}{\text{ramp\_fraction} \times \text{total\_epochs}}\right), \quad \text{ramp\_fraction} =$$

5.2. Results

Training Convergence. The VAE converged smoothly without posterior collapse:

- Final Reconstruction Loss: 27.04
- Final KL Loss: 2.967



(a) Original FashionMNIST samples

(b) VAE Reconstructions

(c) Generated samples (Normal prior)

(d) Generated samples (Laplacian prior)

Figure 10. Reconstructions and generations from different priors. Reconstructions are slightly fuzzy; generated samples show diversity.

Reconstructions and Generations.

Posterior Collapse Analysis. ELBO decomposition shows reconstruction loss dominates KL, preventing collapse.

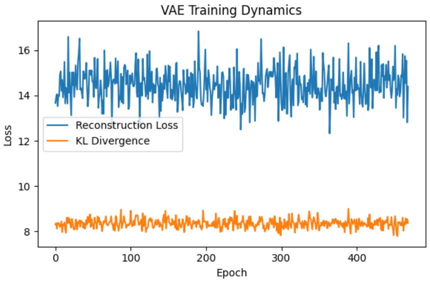


Figure 11. ELBO components over epochs. KL remains small but non-zero, avoiding posterior collapse.

KL Annealing Results. With KL annealing (β ramping to 1.0 over the first 30% of epochs), the KL term stayed active while reconstruction remained stable.

Table 9. Training metrics with KL annealing. Table resized to fit column width.

Epoch	Train Loss	Val Loss	Recon Loss	KL Loss	Beta
1	0.9402	0.9163	0.0001	0.0115	0.17
2	1.6937	1.5897	0.0001	0.0106	0.33
3	2.1404	1.9207	0.0001	0.0089	0.50
4	2.1969	1.8765	0.0001	0.0069	0.67
5	1.9278	1.7386	0.0001	0.0048	0.83
6	2.0700	2.0690	0.0001	0.0043	1.00
⋮	⋮	⋮	⋮	⋮	⋮
20	2.0629	2.0628	0.0001	0.0043	1.00

5.3. Discussion

In the case of the VAE on FashionMNIST, several insights emerge:

- **Training:** Losses converged smoothly with a balance between reconstruction and KL terms. No evidence of posterior collapse was observed.
- **Reconstructions:** Slightly fuzzy but structurally faithful to inputs, consistent with stochastic latent sampling.
- **Generations:** Diverse outputs from both Normal and Laplacian priors confirm robust latent representations.
- **Posterior collapse:** ELBO analysis showed KL remained small but non-zero, meaning the encoder retained dependency on input data.
- **KL annealing:** Allowed gradual incorporation of regularization, preventing premature dominance of the KL term and stabilizing latent space learning.

Overall, the VAE experiment demonstrates that convolutional architectures, combined with KL annealing, can learn stable and meaningful latent spaces on FashionMNIST while avoiding posterior collapse.

6. CLIP and the Modality Gap

6.1. Methodology

Contrastive Language–Image Pretraining (CLIP) jointly trains a vision encoder and a text encoder to map images and natural language into a shared embedding space. This enables zero-shot classification by comparing image embeddings against text prompt embeddings.

We conducted experiments on STL-10 to investigate:

- The effect of prompt engineering on zero-shot classification.
- The presence of a modality gap between CLIP's image and text embeddings.
- The effectiveness of orthogonal Procrustes alignment for bridging this gap.

Embeddings were normalized and projected with t-SNE for qualitative visualization, while cosine similarity was used as a quantitative measure of alignment.

6.2. Results

Zero-Shot Classification. Different prompting strategies produced notable accuracy variations:

Table 10. Zero-shot classification accuracy on STL-10 with different prompting strategies.

Prompting Strategy	**Accuracy (%)**
Plain Labels	96.26
Prompted Text ("a photo of a cat")	97.36
Descriptive Prompts	97.19

Exploring the Modality Gap. For 100 STL-10 samples, embeddings revealed:

- Average cosine similarity between image and text embeddings: **0.2539**.
- Image embeddings clustered near the center, while text embeddings were more dispersed.
- This separation indicated a modality gap between modalities.

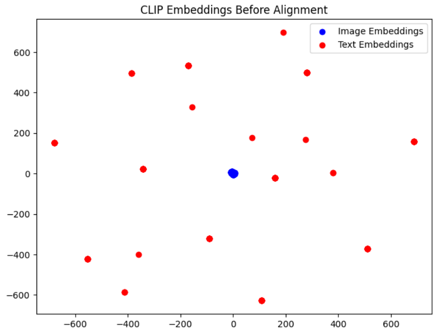


Figure 12. t-SNE projection of image (blue) and text (red) embeddings before alignment. Distinct clustering reflects the modality gap.

Bridging the Modality Gap. An orthogonal Procrustes transform was applied to align embeddings:

$$\min_R \|XR - Y\|_F, \quad \text{s.t. } R^\top R = I$$

where X and Y are image and text embeddings. The optimal R was obtained via SVD.

After alignment:

- Average cosine similarity improved from **0.25** to **0.87**.
- Embeddings appeared more aligned in t-SNE projections, though distortions limited visual clarity.

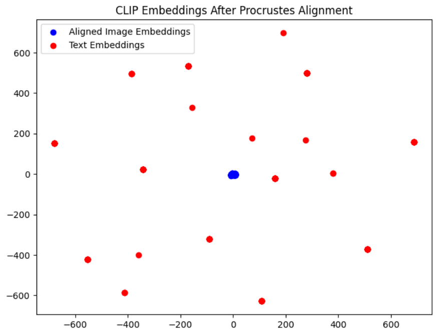


Figure 13. t-SNE projection after Procrustes alignment. Cosine similarity increased substantially (0.25 → 0.87), confirming improved alignment.

6.3. Discussion

The CLIP experiments reveal three key insights:

- **Prompt engineering:** Natural language phrasing significantly affects performance. Even small changes ("cat" vs. "a photo of a cat") improved accuracy.
- **Modality gap:** Raw image and text embeddings occupy misaligned regions of the embedding space, with low cosine similarity (0.25).
- **Alignment:** A simple linear Procrustes transformation improved similarity to 0.87, reducing the modality gap. However, visualization with t-SNE remained less definitive due to dimensionality reduction distortions.

Overall, this study demonstrates that CLIP embeddings, while powerful, exhibit a modality gap that can be mitigated with alignment techniques. Prompt engineering and embedding alignment are both effective strategies to enhance zero-shot multimodal performance.

7. Conclusion

This report has explored five major paradigms in deep learning. Across the experiments, several themes emerged:

- **Transferability:** Pretrained convolutional and transformer models demonstrated strong adaptability with minimal fine-tuning.
- **Stability:** GAN training requires careful balance between generator and discriminator; regularization and adaptive strategies prevent collapse.
- **Representation learning:** Both CNNs and VAEs highlight the importance of hierarchical and structured latent features, with KL annealing providing a practical safeguard against posterior collapse.
- **Interpretability:** Vision Transformers offer inherent attention maps, enabling insights into decision-making compared to CNN posthoc methods.
- **Multimodality:** CLIP reveals both the power and challenges of joint vision–language models; prompt design and embedding alignment significantly affect performance.

Collectively, these findings underscore the trade-offs between performance, interpretability, and stability across architectures. While pretrained models accelerate progress, careful design choices in optimization, regularization, and representation pooling remain critical. Future work could extend these experiments to larger datasets, multimodal retrieval tasks, and more advanced alignment techniques, bridging the gap between theoretical understanding and real-world deployment.

References

Dosovitskiy, A., Beyer, L., Kolesnikov, A., and et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.