homework6

April 18, 2025

[1]:
```python
#ViT fromscratch - 4 heads, 4 layers - 256 hidden, 512 mlp

import torch
import torch.nn as nn
import torchvision
import torchvision.transforms as transforms
from torch.utils.data import DataLoader
from tqdm import tqdm
import matplotlib.pyplot as plt
import numpy as np
from torchinfo import summary
import gc
import time

# Device configuration
device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')

# Hyperparameters
image_size = 32
patch_size = 4
num_classes = 100
num_epochs = 50
batch_size = 64
learning_rate = 0.001
num_heads = 4
num_layers = 4
hidden_dim = 256
mlp_dim = 512

# Data preparation
transform = transforms.Compose([
    transforms.RandomCrop(32, padding=4),
    transforms.RandomHorizontalFlip(),
    transforms.ToTensor(),
    transforms.Normalize((0.5071, 0.4867, 0.4408), (0.2675, 0.2565, 0.2761))
])
```

```
# CIFAR-100 dataset
train_dataset = torchvision.datasets.CIFAR100(root='./data', train=True,
                                              download=True, transform=transform)
test_dataset = torchvision.datasets.CIFAR100(root='./data', train=False,
                                             download=True, transform=transform)

train_loader = DataLoader(dataset=train_dataset, batch_size=batch_size,␣
 ↪shuffle=True)
test_loader = DataLoader(dataset=test_dataset, batch_size=batch_size,␣
 ↪shuffle=False)

# Patch embedding layer
class PatchEmbedding(nn.Module):
    def __init__(self, image_size, patch_size, in_channels=3, embed_dim=256):
        super().__init__()
        self.num_patches = (image_size // patch_size) ** 2
        self.proj = nn.Conv2d(in_channels, embed_dim,
                              kernel_size=patch_size, stride=patch_size)

    def forward(self, x):
        x = self.proj(x)  # [B, embed_dim, H', W']
        x = x.flatten(2)  # [B, embed_dim, num_patches]
        x = x.transpose(1, 2)  # [B, num_patches, embed_dim]
        return x

# Custom Multi-Head Self-Attention module
class MultiHeadSelfAttention(nn.Module):
    def __init__(self, embed_dim, num_heads):
        super().__init__()
        self.embed_dim = embed_dim
        self.num_heads = num_heads
        self.head_dim = embed_dim // num_heads
        assert self.head_dim * num_heads == embed_dim, "embed_dim must be␣
 ↪divisible by num_heads"

        self.qkv = nn.Linear(embed_dim, embed_dim * 3)
        self.proj = nn.Linear(embed_dim, embed_dim)

    def forward(self, x):
        # x: [B, n_patches + 1, embed_dim]
        B, N, C = x.shape

        qkv = self.qkv(x).reshape(B, N, 3, self.num_heads, self.head_dim).
 ↪permute(2, 0, 3, 1, 4)
        q, k, v = qkv.unbind(0)  # Each has shape [B, num_heads, N, head_dim]

        # Scaled dot-product attention
```

```
        attn = (q @ k.transpose(-2, -1)) * (self.head_dim ** -0.5)  # [B,␣
 ↪num_heads, N, N]
        attn = attn.softmax(dim=-1)

        # Apply attention weights to values
        x = (attn @ v).transpose(1, 2).reshape(B, N, C)  # [B, N, C]
        x = self.proj(x)
        return x

# MLP block
class MLP(nn.Module):
    def __init__(self, in_features, hidden_features, out_features):
        super().__init__()
        self.fc1 = nn.Linear(in_features, hidden_features)
        self.act = nn.GELU()
        self.fc2 = nn.Linear(hidden_features, out_features)

    def forward(self, x):
        x = self.fc1(x)
        x = self.act(x)
        x = self.fc2(x)
        return x

# Transformer Encoder with pre-normalization
class TransformerEncoder(nn.Module):
    def __init__(self, embed_dim, num_heads, mlp_dim, dropout=0.1):
        super().__init__()
        self.norm1 = nn.LayerNorm(embed_dim)
        self.attention = MultiHeadSelfAttention(embed_dim, num_heads)
        self.norm2 = nn.LayerNorm(embed_dim)
        self.mlp = MLP(
            in_features=embed_dim,
            hidden_features=mlp_dim,
            out_features=embed_dim
        )
        self.dropout = nn.Dropout(dropout)

    def forward(self, x):
        # Pre-norm architecture
        x = x + self.attention(self.norm1(x))
        x = x + self.mlp(self.norm2(x))
        return x

# Vision Transformer
class VisionTransformer(nn.Module):
    def __init__(self, image_size, patch_size, num_classes, embed_dim,
                 num_heads, num_layers, mlp_dim, dropout=0.1):
```

```
        super().__init__()
        self.patch_embed = PatchEmbedding(image_size, patch_size, 3, embed_dim)
        self.cls_token = nn.Parameter(torch.zeros(1, 1, embed_dim))
        num_patches = (image_size // patch_size) ** 2
        self.pos_embed = nn.Parameter(torch.zeros(1, num_patches + 1,␣
↪embed_dim))
        self.dropout = nn.Dropout(dropout)

        self.transformer = nn.ModuleList(
            [TransformerEncoder(embed_dim, num_heads, mlp_dim, dropout)
             for _ in range(num_layers)]
        )

        self.layer_norm = nn.LayerNorm(embed_dim)
        self.head = nn.Linear(embed_dim, num_classes)

        # Initialize parameters
        self._init_weights()

    def _init_weights(self):
        # Initialize patch embedding, class token, and position embedding
        nn.init.normal_(self.cls_token, std=0.02)
        nn.init.normal_(self.pos_embed, std=0.02)

        # Initialize transformer blocks
        for m in self.modules():
            if isinstance(m, nn.Linear):
                nn.init.xavier_uniform_(m.weight)
                if m.bias is not None:
                    nn.init.zeros_(m.bias)
            elif isinstance(m, nn.LayerNorm):
                nn.init.ones_(m.weight)
                nn.init.zeros_(m.bias)

    def forward(self, x):
        B = x.shape[0]
        x = self.patch_embed(x)

        cls_tokens = self.cls_token.expand(B, -1, -1)
        x = torch.cat((cls_tokens, x), dim=1)
        x = x + self.pos_embed
        x = self.dropout(x)

        for transformer in self.transformer:
            x = transformer(x)

        x = self.layer_norm(x)
```

```
        cls_token_final = x[:, 0]
        x = self.head(cls_token_final)
        return x

# Initialize model
model = VisionTransformer(
    image_size=image_size,
    patch_size=patch_size,
    num_classes=num_classes,
    embed_dim=hidden_dim,
    num_heads=num_heads,
    num_layers=num_layers,
    mlp_dim=mlp_dim
).to(device)


# Display model summary
summary(model,
        input_size=(batch_size, 3, image_size, image_size),
        col_names=["input_size", "output_size", "kernel_size", "num_params",␣
 ↪"mult_adds", "trainable"],
        col_width=20,
        depth=5,
        verbose=True,
        device=device)

# Loss and optimizer
criterion = nn.CrossEntropyLoss()
optimizer = torch.optim.AdamW(model.parameters(), lr=learning_rate,␣
 ↪weight_decay=0.01)

# Training loop
def train():
    model.train()
    train_losses = []
    train_accuracies = []
    epoch_times = []
    for epoch in range(num_epochs):
        start_time = time.time()
        progress_bar = tqdm(train_loader, desc=f'Epoch {epoch+1}/{num_epochs}')
        running_loss = 0.0
        correct = 0
        total = 0

        for i, (images, labels) in enumerate(progress_bar):
            images = images.to(device)
            labels = labels.to(device)
```

```
            # Debug information
            if i == 0 and epoch == 0:
                print(f"Input images shape: {images.shape}")
                print(f"Labels shape: {labels.shape}")
                print(f"Labels values: {labels[:10]}")  # Print first 10 labels

            # Forward pass
            outputs = model(images)

            # Debug information
            if i == 0 and epoch == 0:
                print(f"Model outputs shape: {outputs.shape}")
                print(f"Expected outputs shape: {torch.Size([batch_size,␣
↪num_classes])}")

            loss = criterion(outputs, labels)

            # Backward and optimize
            optimizer.zero_grad()
            loss.backward()
            optimizer.step()

            # Calculate accuracy
            _, predicted = torch.max(outputs.data, 1)
            total += labels.size(0)
            correct += (predicted == labels).sum().item()
            running_loss += loss.item()

            # Update progress bar
            progress_bar.set_postfix({
                'loss': f'{loss.item():.4f}',
                'acc': f'{100 * correct / total:.2f}%'
            })

        # Calculate epoch metrics
        epoch_loss = running_loss / len(train_loader)
        epoch_acc = 100 * correct / total

        # Calculate epoch time
        epoch_time = time.time() - start_time
        epoch_times.append(epoch_time)

        # Store metrics
        train_losses.append(epoch_loss)
        train_accuracies.append(epoch_acc)
```

```
        # Print epoch summary
        print(f'Epoch {epoch+1}/{num_epochs} - Loss: {epoch_loss:.4f}, Accuracy:
 ↪ {epoch_acc:.2f}% Time: {epoch_time:.2f}s')
    # calculate average epoch time
    avg_epoch_time = sum(epoch_times) / len(epoch_times)
    print(f'Average epoch training time: {avg_epoch_time:.2f} seconds')

    return train_losses, train_accuracies, epoch_times

# Test the model
def test():
    model.eval()
    test_losses = []
    test_accuracies = []

    with torch.no_grad():
        correct = 0
        total = 0
        running_loss = 0.0
        progress_bar = tqdm(test_loader, desc='Testing')

        for images, labels in progress_bar:
            images = images.to(device)
            labels = labels.to(device)
            outputs = model(images)
            loss = criterion(outputs, labels)

            _, predicted = torch.max(outputs.data, 1)
            total += labels.size(0)
            correct += (predicted == labels).sum().item()
            running_loss += loss.item()

            # Update progress bar with current accuracy
            accuracy = 100 * correct / total
            progress_bar.set_postfix({'accuracy': f'{accuracy:.2f}%'})

        # Calculate final metrics
        final_loss = running_loss / len(test_loader)
        final_acc = 100 * correct / total

        # Store metrics
        test_losses.append(final_loss)
        test_accuracies.append(final_acc)

        print(f'Final Test Loss: {final_loss:.4f}, Final Test Accuracy:␣
 ↪{final_acc:.2f}%')
```

```
    return test_losses, test_accuracies

# Visualize training and testing results
def visualize_results(train_losses, train_accuracies, test_losses,␣
 ↪test_accuracies):
    plt.figure(figsize=(12, 5))

    # Plot losses
    plt.subplot(1, 2, 1)
    plt.plot(train_losses, label='Training Loss')
    plt.plot([len(train_losses)-1], test_losses, 'ro', label='Test Loss')
    plt.xlabel('Epoch')
    plt.ylabel('Loss')
    plt.title('Training and Test Loss')
    plt.legend()
    plt.grid(True)

    # Plot accuracies
    plt.subplot(1, 2, 2)
    plt.plot(train_accuracies, label='Training Accuracy')
    plt.plot([len(train_accuracies)-1], test_accuracies, 'ro', label='Test␣
 ↪Accuracy')
    plt.xlabel('Epoch')
    plt.ylabel('Accuracy (%)')
    plt.title('Training and Test Accuracy')
    plt.legend()
    plt.grid(True)

    plt.tight_layout()
    plt.savefig('vit_training_results.png')
    plt.show()

# Run training and testing
if __name__ == '__main__':
    print("Training started...")
    train_losses, train_accuracies, epoch_times = train()
    print("\nTesting started...")
    test_losses, test_accuracies = test()

    # Visualize results
    print("\nVisualizing results...")
    visualize_results(train_losses, train_accuracies, test_losses,␣
 ↪test_accuracies)
```

```
Layer (type:depth-idx)                        Input Shape          Output Shape
Kernel Shape         Param #              Mult-Adds            Trainable
==========================================================================================
==========================================================================================
=====
VisionTransformer                             [64, 3, 32, 32]      [64, 100]
--                   16,896               --                   True
 PatchEmbedding: 1-1                            [64, 3, 32, 32]      [64, 64, 256]
--                   --                   --                   True
     Conv2d: 2-1                              [64, 3, 32, 32]      [64, 256, 8,
8]      [4, 4]               12,544               51,380,224           True
 Dropout: 1-2                                 [64, 65, 256]        [64, 65, 256]
--                   --                   --                   --
 ModuleList: 1-3                                --                   --
--                   --                   --                   True
     TransformerEncoder: 2-2                  [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
          LayerNorm: 3-1                      [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
          MultiHeadSelfAttention: 3-2         [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
               Linear: 4-1                    [64, 65, 256]        [64, 65, 768]
--                   197,376              12,632,064           True
               Linear: 4-2                    [64, 65, 256]        [64, 65, 256]
--                   65,792               4,210,688            True
          LayerNorm: 3-3                      [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
          MLP: 3-4                            [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
               Linear: 4-3                    [64, 65, 256]        [64, 65, 512]
--                   131,584              8,421,376            True
               GELU: 4-4                      [64, 65, 512]        [64, 65, 512]
--                   --                   --                   --
               Linear: 4-5                    [64, 65, 512]        [64, 65, 256]
--                   131,328              8,404,992            True
     TransformerEncoder: 2-3                  [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
          LayerNorm: 3-5                      [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
          MultiHeadSelfAttention: 3-6         [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
               Linear: 4-6                    [64, 65, 256]        [64, 65, 768]
--                   197,376              12,632,064           True
               Linear: 4-7                    [64, 65, 256]        [64, 65, 256]
--                   65,792               4,210,688            True
          LayerNorm: 3-7                      [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
          MLP: 3-8                            [64, 65, 256]        [64, 65, 256]
```

```
--                  --                  --                  True
            Linear: 4-8                  [64, 65, 256]       [64, 65, 512]
--                  131,584             8,421,376           True
            GELU: 4-9                    [64, 65, 512]       [64, 65, 512]
--                  --                  --                  --
            Linear: 4-10                 [64, 65, 512]       [64, 65, 256]
--                  131,328             8,404,992           True
    TransformerEncoder: 2-4               [64, 65, 256]       [64, 65, 256]
--                  --                  --                  True
        LayerNorm: 3-9                    [64, 65, 256]       [64, 65, 256]
--                  512                 32,768              True
        MultiHeadSelfAttention: 3-10      [64, 65, 256]       [64, 65, 256]
--                  --                  --                  True
            Linear: 4-11                 [64, 65, 256]       [64, 65, 768]
--                  197,376             12,632,064          True
            Linear: 4-12                 [64, 65, 256]       [64, 65, 256]
--                  65,792              4,210,688           True
        LayerNorm: 3-11                   [64, 65, 256]       [64, 65, 256]
--                  512                 32,768              True
        MLP: 3-12                         [64, 65, 256]       [64, 65, 256]
--                  --                  --                  True
            Linear: 4-13                 [64, 65, 256]       [64, 65, 512]
--                  131,584             8,421,376           True
            GELU: 4-14                   [64, 65, 512]       [64, 65, 512]
--                  --                  --                  --
            Linear: 4-15                 [64, 65, 512]       [64, 65, 256]
--                  131,328             8,404,992           True
    TransformerEncoder: 2-5               [64, 65, 256]       [64, 65, 256]
--                  --                  --                  True
        LayerNorm: 3-13                   [64, 65, 256]       [64, 65, 256]
--                  512                 32,768              True
        MultiHeadSelfAttention: 3-14      [64, 65, 256]       [64, 65, 256]
--                  --                  --                  True
            Linear: 4-16                 [64, 65, 256]       [64, 65, 768]
--                  197,376             12,632,064          True
            Linear: 4-17                 [64, 65, 256]       [64, 65, 256]
--                  65,792              4,210,688           True
        LayerNorm: 3-15                   [64, 65, 256]       [64, 65, 256]
--                  512                 32,768              True
        MLP: 3-16                         [64, 65, 256]       [64, 65, 256]
--                  --                  --                  True
            Linear: 4-18                 [64, 65, 256]       [64, 65, 512]
--                  131,584             8,421,376           True
            GELU: 4-19                   [64, 65, 512]       [64, 65, 512]
--                  --                  --                  --
            Linear: 4-20                 [64, 65, 512]       [64, 65, 256]
--                  131,328             8,404,992           True
 LayerNorm: 1-4                            [64, 65, 256]       [64, 65, 256]
```

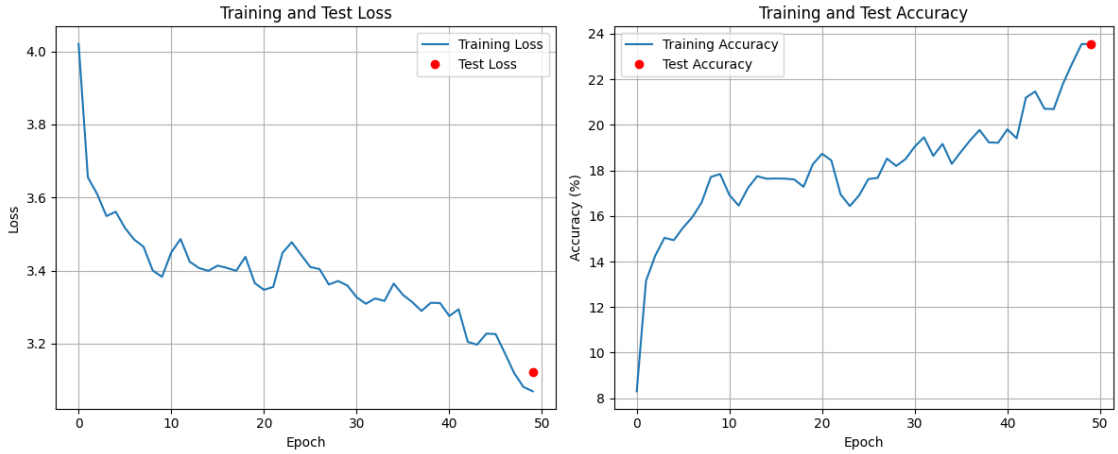
```
--                    512                   32,768                True
 Linear: 1-5                                   [64, 256]             [64, 100]
--                    25,700                1,644,800             True
==========================================================================
==========================================================================
=====
Total params: 2,164,068
Trainable params: 2,164,068
Non-trainable params: 0
Total mult-adds (Units.MEGABYTES): 188.00
==========================================================================
==========================================================================
=====
Input size (MB): 0.79
Forward/backward pass size (MB): 323.67
Params size (MB): 8.59
Estimated Total Size (MB): 333.04
==========================================================================
==========================================================================
=====
Training started…

Epoch 1/50:   0%|          | 1/782 [00:00<01:26,  9.04it/s, loss=5.0707,
acc=1.17%]

Input images shape: torch.Size([64, 3, 32, 32])
Labels shape: torch.Size([64])
Labels values: tensor([59, 44, 49, 79, 99, 65, 99,  5, 29, 58], device='cuda:0')
Model outputs shape: torch.Size([64, 100])
Expected outputs shape: torch.Size([64, 100])

Epoch 1/50: 100%|     | 782/782 [00:15<00:00, 48.91it/s, loss=3.2348,
acc=8.30%]

Epoch 1/50 - Loss: 4.0205, Accuracy: 8.30% Time: 15.99s

Epoch 2/50: 100%|     | 782/782 [00:15<00:00, 50.25it/s, loss=3.2329,
acc=13.15%]

Epoch 2/50 - Loss: 3.6549, Accuracy: 13.15% Time: 15.56s

Epoch 3/50: 100%|     | 782/782 [00:15<00:00, 50.23it/s, loss=3.5342,
acc=14.26%]

Epoch 3/50 - Loss: 3.6094, Accuracy: 14.26% Time: 15.57s

Epoch 4/50: 100%|     | 782/782 [00:15<00:00, 50.06it/s, loss=3.8848,
acc=15.04%]

Epoch 4/50 - Loss: 3.5491, Accuracy: 15.04% Time: 15.62s

Epoch 5/50: 100%|     | 782/782 [00:15<00:00, 50.07it/s, loss=3.7650,
acc=14.94%]
```

```
Epoch 5/50 - Loss: 3.5609, Accuracy: 14.94% Time: 15.62s

Epoch 6/50: 100%|      | 782/782 [00:15<00:00, 50.09it/s, loss=3.6654,
acc=15.49%]

Epoch 6/50 - Loss: 3.5166, Accuracy: 15.49% Time: 15.61s

Epoch 7/50: 100%|      | 782/782 [00:15<00:00, 50.20it/s, loss=3.6982,
acc=15.95%]

Epoch 7/50 - Loss: 3.4847, Accuracy: 15.95% Time: 15.58s

Epoch 8/50: 100%|      | 782/782 [00:15<00:00, 49.96it/s, loss=2.9210,
acc=16.59%]

Epoch 8/50 - Loss: 3.4653, Accuracy: 16.59% Time: 15.65s

Epoch 9/50: 100%|      | 782/782 [00:15<00:00, 49.99it/s, loss=2.9793,
acc=17.71%]

Epoch 9/50 - Loss: 3.3999, Accuracy: 17.71% Time: 15.65s

Epoch 10/50: 100%|      | 782/782 [00:15<00:00, 49.83it/s, loss=3.4362,
acc=17.84%]

Epoch 10/50 - Loss: 3.3826, Accuracy: 17.84% Time: 15.69s

Epoch 11/50: 100%|      | 782/782 [00:15<00:00, 49.91it/s, loss=2.7499,
acc=16.92%]

Epoch 11/50 - Loss: 3.4497, Accuracy: 16.92% Time: 15.67s

Epoch 12/50: 100%|      | 782/782 [00:15<00:00, 50.03it/s, loss=3.5379,
acc=16.45%]

Epoch 12/50 - Loss: 3.4862, Accuracy: 16.45% Time: 15.63s

Epoch 13/50: 100%|      | 782/782 [00:15<00:00, 49.96it/s, loss=3.7011,
acc=17.23%]

Epoch 13/50 - Loss: 3.4238, Accuracy: 17.23% Time: 15.65s

Epoch 14/50: 100%|      | 782/782 [00:15<00:00, 49.93it/s, loss=3.1529,
acc=17.75%]

Epoch 14/50 - Loss: 3.4067, Accuracy: 17.75% Time: 15.66s

Epoch 15/50: 100%|      | 782/782 [00:15<00:00, 49.96it/s, loss=3.6582,
acc=17.63%]

Epoch 15/50 - Loss: 3.3991, Accuracy: 17.63% Time: 15.65s

Epoch 16/50: 100%|      | 782/782 [00:15<00:00, 50.02it/s, loss=3.6178,
acc=17.64%]

Epoch 16/50 - Loss: 3.4136, Accuracy: 17.64% Time: 15.63s

Epoch 17/50: 100%|      | 782/782 [00:15<00:00, 49.90it/s, loss=3.5897,
acc=17.64%]
```

```
Epoch 17/50 - Loss: 3.4071, Accuracy: 17.64% Time: 15.67s

Epoch 18/50: 100%|     | 782/782 [00:15<00:00, 49.97it/s, loss=2.9046,
acc=17.60%]

Epoch 18/50 - Loss: 3.3992, Accuracy: 17.60% Time: 15.65s

Epoch 19/50: 100%|     | 782/782 [00:15<00:00, 50.06it/s, loss=3.7685,
acc=17.28%]

Epoch 19/50 - Loss: 3.4376, Accuracy: 17.28% Time: 15.62s

Epoch 20/50: 100%|     | 782/782 [00:15<00:00, 49.84it/s, loss=2.8521,
acc=18.27%]

Epoch 20/50 - Loss: 3.3657, Accuracy: 18.27% Time: 15.69s

Epoch 21/50: 100%|     | 782/782 [00:15<00:00, 50.04it/s, loss=3.4598,
acc=18.73%]

Epoch 21/50 - Loss: 3.3471, Accuracy: 18.73% Time: 15.63s

Epoch 22/50: 100%|     | 782/782 [00:15<00:00, 49.97it/s, loss=3.6753,
acc=18.44%]

Epoch 22/50 - Loss: 3.3551, Accuracy: 18.44% Time: 15.65s

Epoch 23/50: 100%|     | 782/782 [00:15<00:00, 50.02it/s, loss=3.4018,
acc=16.95%]

Epoch 23/50 - Loss: 3.4485, Accuracy: 16.95% Time: 15.64s

Epoch 24/50: 100%|     | 782/782 [00:15<00:00, 49.95it/s, loss=3.5803,
acc=16.44%]

Epoch 24/50 - Loss: 3.4775, Accuracy: 16.44% Time: 15.66s

Epoch 25/50: 100%|     | 782/782 [00:15<00:00, 50.08it/s, loss=3.1998,
acc=16.90%]

Epoch 25/50 - Loss: 3.4432, Accuracy: 16.90% Time: 15.62s

Epoch 26/50: 100%|     | 782/782 [00:15<00:00, 49.98it/s, loss=3.2658,
acc=17.62%]

Epoch 26/50 - Loss: 3.4096, Accuracy: 17.62% Time: 15.65s

Epoch 27/50: 100%|     | 782/782 [00:15<00:00, 50.03it/s, loss=2.8942,
acc=17.67%]

Epoch 27/50 - Loss: 3.4038, Accuracy: 17.67% Time: 15.63s

Epoch 28/50: 100%|     | 782/782 [00:15<00:00, 50.01it/s, loss=3.5724,
acc=18.52%]

Epoch 28/50 - Loss: 3.3616, Accuracy: 18.52% Time: 15.64s

Epoch 29/50: 100%|     | 782/782 [00:15<00:00, 49.74it/s, loss=3.5862,
acc=18.19%]
```

```
Epoch 29/50 - Loss: 3.3713, Accuracy: 18.19% Time: 15.72s

Epoch 30/50: 100%|      | 782/782 [00:15<00:00, 49.45it/s, loss=3.3986,
acc=18.49%]

Epoch 30/50 - Loss: 3.3589, Accuracy: 18.49% Time: 15.82s

Epoch 31/50: 100%|      | 782/782 [00:15<00:00, 50.10it/s, loss=2.5215,
acc=19.04%]

Epoch 31/50 - Loss: 3.3268, Accuracy: 19.04% Time: 15.61s

Epoch 32/50: 100%|      | 782/782 [00:15<00:00, 49.92it/s, loss=3.2487,
acc=19.45%]

Epoch 32/50 - Loss: 3.3090, Accuracy: 19.45% Time: 15.67s

Epoch 33/50: 100%|      | 782/782 [00:15<00:00, 49.92it/s, loss=3.3336,
acc=18.64%]

Epoch 33/50 - Loss: 3.3236, Accuracy: 18.64% Time: 15.67s

Epoch 34/50: 100%|      | 782/782 [00:15<00:00, 50.59it/s, loss=3.2965,
acc=19.16%]

Epoch 34/50 - Loss: 3.3167, Accuracy: 19.16% Time: 15.46s

Epoch 35/50: 100%|      | 782/782 [00:15<00:00, 50.53it/s, loss=3.6106,
acc=18.29%]

Epoch 35/50 - Loss: 3.3642, Accuracy: 18.29% Time: 15.48s

Epoch 36/50: 100%|      | 782/782 [00:15<00:00, 50.54it/s, loss=3.4407,
acc=18.82%]

Epoch 36/50 - Loss: 3.3333, Accuracy: 18.82% Time: 15.47s

Epoch 37/50: 100%|      | 782/782 [00:15<00:00, 50.64it/s, loss=3.2387,
acc=19.33%]

Epoch 37/50 - Loss: 3.3136, Accuracy: 19.33% Time: 15.44s

Epoch 38/50: 100%|      | 782/782 [00:15<00:00, 50.72it/s, loss=3.2528,
acc=19.77%]

Epoch 38/50 - Loss: 3.2893, Accuracy: 19.77% Time: 15.42s

Epoch 39/50: 100%|      | 782/782 [00:15<00:00, 50.65it/s, loss=3.2583,
acc=19.23%]

Epoch 39/50 - Loss: 3.3115, Accuracy: 19.23% Time: 15.44s

Epoch 40/50: 100%|      | 782/782 [00:15<00:00, 50.61it/s, loss=3.1123,
acc=19.21%]

Epoch 40/50 - Loss: 3.3111, Accuracy: 19.21% Time: 15.45s

Epoch 41/50: 100%|      | 782/782 [00:15<00:00, 50.56it/s, loss=3.6482,
acc=19.80%]
```

```
Epoch 41/50 - Loss: 3.2754, Accuracy: 19.80% Time: 15.47s

Epoch 42/50: 100%|     | 782/782 [00:15<00:00, 50.58it/s, loss=2.6546,
acc=19.41%]

Epoch 42/50 - Loss: 3.2938, Accuracy: 19.41% Time: 15.46s

Epoch 43/50: 100%|     | 782/782 [00:15<00:00, 50.57it/s, loss=2.9812,
acc=21.20%]

Epoch 43/50 - Loss: 3.2045, Accuracy: 21.20% Time: 15.46s

Epoch 44/50: 100%|     | 782/782 [00:15<00:00, 50.47it/s, loss=2.9685,
acc=21.47%]

Epoch 44/50 - Loss: 3.1968, Accuracy: 21.47% Time: 15.49s

Epoch 45/50: 100%|     | 782/782 [00:15<00:00, 50.51it/s, loss=3.1832,
acc=20.71%]

Epoch 45/50 - Loss: 3.2271, Accuracy: 20.71% Time: 15.48s

Epoch 46/50: 100%|     | 782/782 [00:15<00:00, 50.52it/s, loss=3.0940,
acc=20.69%]

Epoch 46/50 - Loss: 3.2261, Accuracy: 20.69% Time: 15.48s

Epoch 47/50: 100%|     | 782/782 [00:15<00:00, 50.45it/s, loss=2.7300,
acc=21.80%]

Epoch 47/50 - Loss: 3.1736, Accuracy: 21.80% Time: 15.50s

Epoch 48/50: 100%|     | 782/782 [00:15<00:00, 50.66it/s, loss=3.0633,
acc=22.70%]

Epoch 48/50 - Loss: 3.1192, Accuracy: 22.70% Time: 15.44s

Epoch 49/50: 100%|     | 782/782 [00:15<00:00, 50.70it/s, loss=3.3388,
acc=23.54%]

Epoch 49/50 - Loss: 3.0813, Accuracy: 23.54% Time: 15.42s

Epoch 50/50: 100%|     | 782/782 [00:15<00:00, 50.75it/s, loss=3.4897,
acc=23.55%]

Epoch 50/50 - Loss: 3.0693, Accuracy: 23.55% Time: 15.41s
Average epoch training time: 15.59 seconds

Testing started…

Testing: 100%|     | 157/157 [00:02<00:00, 67.42it/s, accuracy=23.52%]

Final Test Loss: 3.1229, Final Test Accuracy: 23.52%

Visualizing results…
```



[2]:
```
#ViT from scratch - 8x8 patch size

# Delete model and optimizer variables
del model, optimizer, test_losses, test_accuracies, train_losses,␣
 ↪train_accuracies, epoch_times

# Clear CUDA cache if using GPU
gc.collect()
# Clear CUDA cache
if torch.cuda.is_available():
    torch.cuda.empty_cache()


# Initialize model
model = VisionTransformer(
    image_size=image_size,
    patch_size=patch_size,
    num_classes=num_classes,
    embed_dim=hidden_dim,
    num_heads=num_heads,
    num_layers=num_layers,
    mlp_dim=mlp_dim
).to(device)

# Loss and optimizer
criterion = nn.CrossEntropyLoss()
optimizer = torch.optim.AdamW(model.parameters(), lr=learning_rate,␣
 ↪weight_decay=0.01)

# Display model summary
summary(model,
```

```
        input_size=(batch_size, 3, image_size, image_size),
        col_names=["input_size", "output_size", "kernel_size", "num_params",␣
 ↪"mult_adds", "trainable"],
        col_width=20,
        depth=5,
        verbose=True,
        device=device)

# Hyperparameters
image_size = 32
patch_size = 8
num_classes = 100
num_epochs = 50
batch_size = 64
learning_rate = 0.001
num_heads = 4
num_layers = 4
hidden_dim = 256
mlp_dim = 512

# Run training and testing
if __name__ == '__main__':
    print("Training started...")
    train_losses, train_accuracies, epoch_times = train()
    print("\nTesting started...")
    test_losses, test_accuracies = test()

    # Visualize results
    print("\nVisualizing results...")
    visualize_results(train_losses, train_accuracies, test_losses,␣
 ↪test_accuracies)
```

```
==========================================================================
==========================================================================
=====
Layer (type:depth-idx)                        Input Shape          Output Shape
Kernel Shape         Param #              Mult-Adds            Trainable
==========================================================================
==========================================================================
=====
VisionTransformer                             [64, 3, 32, 32]      [64, 100]
--                   16,896               --                   True
 PatchEmbedding: 1-1                            [64, 3, 32, 32]      [64, 64, 256]
--                   --                   --                   True
      Conv2d: 2-1                               [64, 3, 32, 32]      [64, 256, 8,
8]       [4, 4]               12,544               51,380,224           True
 Dropout: 1-2                                   [64, 65, 256]        [64, 65, 256]
--                   --                   --                   --
```

```
 ModuleList: 1-3                                --                --
--                    --                       --                True
     TransformerEncoder: 2-2                     [64, 65, 256]     [64, 65, 256]
--                    --                       --                True
          LayerNorm: 3-1                         [64, 65, 256]     [64, 65, 256]
--                    512                      32,768            True
          MultiHeadSelfAttention: 3-2            [64, 65, 256]     [64, 65, 256]
--                    --                       --                True
               Linear: 4-1                       [64, 65, 256]     [64, 65, 768]
--                    197,376                  12,632,064        True
               Linear: 4-2                       [64, 65, 256]     [64, 65, 256]
--                    65,792                   4,210,688         True
          LayerNorm: 3-3                         [64, 65, 256]     [64, 65, 256]
--                    512                      32,768            True
          MLP: 3-4                               [64, 65, 256]     [64, 65, 256]
--                    --                       --                True
               Linear: 4-3                       [64, 65, 256]     [64, 65, 512]
--                    131,584                  8,421,376         True
               GELU: 4-4                         [64, 65, 512]     [64, 65, 512]
--                    --                       --                --
               Linear: 4-5                       [64, 65, 512]     [64, 65, 256]
--                    131,328                  8,404,992         True
     TransformerEncoder: 2-3                     [64, 65, 256]     [64, 65, 256]
--                    --                       --                True
          LayerNorm: 3-5                         [64, 65, 256]     [64, 65, 256]
--                    512                      32,768            True
          MultiHeadSelfAttention: 3-6            [64, 65, 256]     [64, 65, 256]
--                    --                       --                True
               Linear: 4-6                       [64, 65, 256]     [64, 65, 768]
--                    197,376                  12,632,064        True
               Linear: 4-7                       [64, 65, 256]     [64, 65, 256]
--                    65,792                   4,210,688         True
          LayerNorm: 3-7                         [64, 65, 256]     [64, 65, 256]
--                    512                      32,768            True
          MLP: 3-8                               [64, 65, 256]     [64, 65, 256]
--                    --                       --                True
               Linear: 4-8                       [64, 65, 256]     [64, 65, 512]
--                    131,584                  8,421,376         True
               GELU: 4-9                         [64, 65, 512]     [64, 65, 512]
--                    --                       --                --
               Linear: 4-10                      [64, 65, 512]     [64, 65, 256]
--                    131,328                  8,404,992         True
     TransformerEncoder: 2-4                     [64, 65, 256]     [64, 65, 256]
--                    --                       --                True
          LayerNorm: 3-9                         [64, 65, 256]     [64, 65, 256]
--                    512                      32,768            True
          MultiHeadSelfAttention: 3-10           [64, 65, 256]     [64, 65, 256]
--                    --                       --                True
```

```
                Linear: 4-11                  [64, 65, 256]        [64, 65, 768]
--                    197,376                 12,632,064           True
                Linear: 4-12                  [64, 65, 256]        [64, 65, 256]
--                    65,792                  4,210,688            True
          LayerNorm: 3-11                       [64, 65, 256]        [64, 65, 256]
--                    512                     32,768               True
          MLP: 3-12                             [64, 65, 256]        [64, 65, 256]
--                    --                      --                   True
                Linear: 4-13                  [64, 65, 256]        [64, 65, 512]
--                    131,584                 8,421,376            True
                GELU: 4-14                    [64, 65, 512]        [64, 65, 512]
--                    --                      --                   --
                Linear: 4-15                  [64, 65, 512]        [64, 65, 256]
--                    131,328                 8,404,992            True
      TransformerEncoder: 2-5                   [64, 65, 256]        [64, 65, 256]
--                    --                      --                   True
          LayerNorm: 3-13                       [64, 65, 256]        [64, 65, 256]
--                    512                     32,768               True
          MultiHeadSelfAttention: 3-14          [64, 65, 256]        [64, 65, 256]
--                    --                      --                   True
                Linear: 4-16                  [64, 65, 256]        [64, 65, 768]
--                    197,376                 12,632,064           True
                Linear: 4-17                  [64, 65, 256]        [64, 65, 256]
--                    65,792                  4,210,688            True
          LayerNorm: 3-15                       [64, 65, 256]        [64, 65, 256]
--                    512                     32,768               True
          MLP: 3-16                             [64, 65, 256]        [64, 65, 256]
--                    --                      --                   True
                Linear: 4-18                  [64, 65, 256]        [64, 65, 512]
--                    131,584                 8,421,376            True
                GELU: 4-19                    [64, 65, 512]        [64, 65, 512]
--                    --                      --                   --
                Linear: 4-20                  [64, 65, 512]        [64, 65, 256]
--                    131,328                 8,404,992            True
 LayerNorm: 1-4                                 [64, 65, 256]        [64, 65, 256]
--                    512                     32,768               True
 Linear: 1-5                                    [64, 256]            [64, 100]
--                    25,700                  1,644,800            True
==========================================================================
==========================================================================
=====
Total params: 2,164,068
Trainable params: 2,164,068
Non-trainable params: 0
Total mult-adds (Units.MEGABYTES): 188.00
==========================================================================
==========================================================================
=====
```

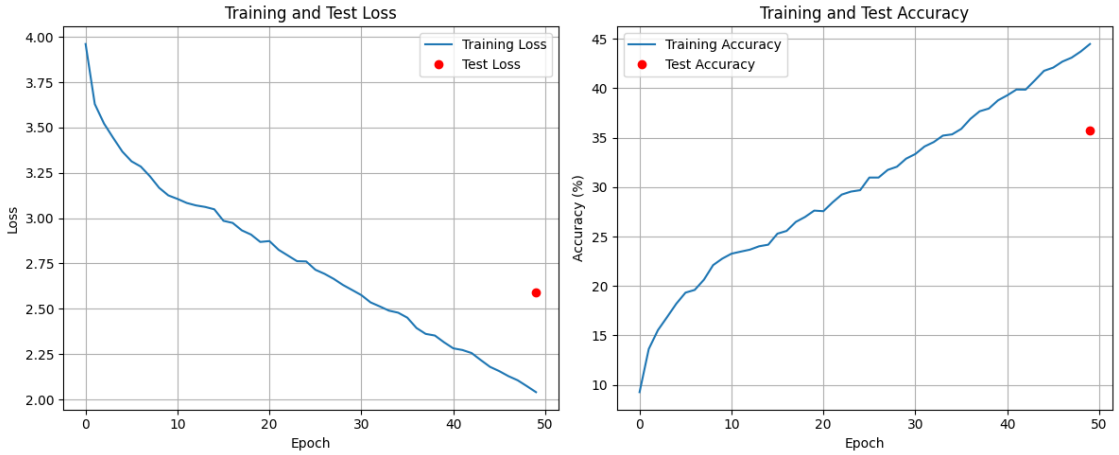
```
Input size (MB): 0.79
Forward/backward pass size (MB): 323.67
Params size (MB): 8.59
Estimated Total Size (MB): 333.04
================================================================================
================================================================================
=====
Training started…

Epoch 1/50:   1%|          | 5/782 [00:00<00:16, 47.57it/s, loss=4.8190,
acc=2.60%]

Input images shape: torch.Size([64, 3, 32, 32])
Labels shape: torch.Size([64])
Labels values: tensor([78, 36, 78, 72, 97, 99, 77, 34, 39, 13], device='cuda:0')
Model outputs shape: torch.Size([64, 100])
Expected outputs shape: torch.Size([64, 100])

Epoch 1/50: 100%|     | 782/782 [00:15<00:00, 51.24it/s, loss=3.5393,
acc=9.25%]

Epoch 1/50 - Loss: 3.9606, Accuracy: 9.25% Time: 15.26s

Epoch 2/50: 100%|     | 782/782 [00:15<00:00, 52.11it/s, loss=3.4655,
acc=13.62%]

Epoch 2/50 - Loss: 3.6295, Accuracy: 13.62% Time: 15.01s

Epoch 3/50: 100%|     | 782/782 [00:14<00:00, 52.26it/s, loss=3.0965,
acc=15.53%]

Epoch 3/50 - Loss: 3.5222, Accuracy: 15.53% Time: 14.97s

Epoch 4/50: 100%|     | 782/782 [00:15<00:00, 52.08it/s, loss=3.4326,
acc=16.85%]

Epoch 4/50 - Loss: 3.4431, Accuracy: 16.85% Time: 15.02s

Epoch 5/50: 100%|     | 782/782 [00:15<00:00, 52.06it/s, loss=3.3121,
acc=18.21%]

Epoch 5/50 - Loss: 3.3671, Accuracy: 18.21% Time: 15.02s

Epoch 6/50: 100%|     | 782/782 [00:15<00:00, 51.79it/s, loss=3.5285,
acc=19.32%]

Epoch 6/50 - Loss: 3.3134, Accuracy: 19.32% Time: 15.10s

Epoch 7/50: 100%|     | 782/782 [00:15<00:00, 52.12it/s, loss=3.3615,
acc=19.60%]

Epoch 7/50 - Loss: 3.2848, Accuracy: 19.60% Time: 15.01s

Epoch 8/50: 100%|     | 782/782 [00:15<00:00, 51.82it/s, loss=3.2827,
acc=20.63%]

Epoch 8/50 - Loss: 3.2310, Accuracy: 20.63% Time: 15.09s
```

```
Epoch 9/50: 100%|      | 782/782 [00:15<00:00, 52.05it/s, loss=3.0183,
acc=22.11%]

Epoch 9/50 - Loss: 3.1676, Accuracy: 22.11% Time: 15.02s

Epoch 10/50: 100%|      | 782/782 [00:15<00:00, 52.03it/s, loss=3.2921,
acc=22.76%]

Epoch 10/50 - Loss: 3.1257, Accuracy: 22.76% Time: 15.03s

Epoch 11/50: 100%|      | 782/782 [00:15<00:00, 52.00it/s, loss=3.2266,
acc=23.26%]

Epoch 11/50 - Loss: 3.1062, Accuracy: 23.26% Time: 15.04s

Epoch 12/50: 100%|      | 782/782 [00:15<00:00, 51.94it/s, loss=3.4694,
acc=23.47%]

Epoch 12/50 - Loss: 3.0840, Accuracy: 23.47% Time: 15.06s

Epoch 13/50: 100%|      | 782/782 [00:15<00:00, 51.94it/s, loss=2.6883,
acc=23.67%]

Epoch 13/50 - Loss: 3.0705, Accuracy: 23.67% Time: 15.06s

Epoch 14/50: 100%|      | 782/782 [00:15<00:00, 52.02it/s, loss=3.1246,
acc=24.01%]

Epoch 14/50 - Loss: 3.0620, Accuracy: 24.01% Time: 15.03s

Epoch 15/50: 100%|      | 782/782 [00:15<00:00, 52.06it/s, loss=3.0042,
acc=24.17%]

Epoch 15/50 - Loss: 3.0491, Accuracy: 24.17% Time: 15.02s

Epoch 16/50: 100%|      | 782/782 [00:15<00:00, 52.07it/s, loss=3.0261,
acc=25.29%]

Epoch 16/50 - Loss: 2.9854, Accuracy: 25.29% Time: 15.02s

Epoch 17/50: 100%|      | 782/782 [00:14<00:00, 52.24it/s, loss=3.2354,
acc=25.57%]

Epoch 17/50 - Loss: 2.9738, Accuracy: 25.57% Time: 14.97s

Epoch 18/50: 100%|      | 782/782 [00:15<00:00, 52.00it/s, loss=3.4922,
acc=26.48%]

Epoch 18/50 - Loss: 2.9327, Accuracy: 26.48% Time: 15.04s

Epoch 19/50: 100%|      | 782/782 [00:15<00:00, 51.99it/s, loss=2.6071,
acc=26.98%]

Epoch 19/50 - Loss: 2.9096, Accuracy: 26.98% Time: 15.04s

Epoch 20/50: 100%|      | 782/782 [00:15<00:00, 51.96it/s, loss=3.3008,
acc=27.62%]

Epoch 20/50 - Loss: 2.8690, Accuracy: 27.62% Time: 15.05s
```

```
Epoch 21/50: 100%|      | 782/782 [00:14<00:00, 52.17it/s, loss=2.7702,
acc=27.56%]

Epoch 21/50 - Loss: 2.8737, Accuracy: 27.56% Time: 14.99s

Epoch 22/50: 100%|      | 782/782 [00:15<00:00, 52.07it/s, loss=3.1567,
acc=28.46%]

Epoch 22/50 - Loss: 2.8255, Accuracy: 28.46% Time: 15.02s

Epoch 23/50: 100%|      | 782/782 [00:15<00:00, 52.00it/s, loss=3.4064,
acc=29.25%]

Epoch 23/50 - Loss: 2.7943, Accuracy: 29.25% Time: 15.04s

Epoch 24/50: 100%|      | 782/782 [00:15<00:00, 51.91it/s, loss=3.1387,
acc=29.55%]

Epoch 24/50 - Loss: 2.7628, Accuracy: 29.55% Time: 15.07s

Epoch 25/50: 100%|      | 782/782 [00:15<00:00, 52.01it/s, loss=3.7193,
acc=29.69%]

Epoch 25/50 - Loss: 2.7614, Accuracy: 29.69% Time: 15.04s

Epoch 26/50: 100%|      | 782/782 [00:15<00:00, 52.04it/s, loss=2.6281,
acc=30.97%]

Epoch 26/50 - Loss: 2.7155, Accuracy: 30.97% Time: 15.03s

Epoch 27/50: 100%|      | 782/782 [00:15<00:00, 51.99it/s, loss=2.3743,
acc=30.97%]

Epoch 27/50 - Loss: 2.6929, Accuracy: 30.97% Time: 15.04s

Epoch 28/50: 100%|      | 782/782 [00:15<00:00, 52.10it/s, loss=3.3216,
acc=31.74%]

Epoch 28/50 - Loss: 2.6649, Accuracy: 31.74% Time: 15.01s

Epoch 29/50: 100%|      | 782/782 [00:15<00:00, 52.06it/s, loss=2.4153,
acc=32.06%]

Epoch 29/50 - Loss: 2.6317, Accuracy: 32.06% Time: 15.02s

Epoch 30/50: 100%|      | 782/782 [00:15<00:00, 52.00it/s, loss=2.9936,
acc=32.88%]

Epoch 30/50 - Loss: 2.6037, Accuracy: 32.88% Time: 15.04s

Epoch 31/50: 100%|      | 782/782 [00:15<00:00, 52.07it/s, loss=2.8196,
acc=33.35%]

Epoch 31/50 - Loss: 2.5755, Accuracy: 33.35% Time: 15.02s

Epoch 32/50: 100%|      | 782/782 [00:15<00:00, 51.93it/s, loss=2.4644,
acc=34.11%]

Epoch 32/50 - Loss: 2.5356, Accuracy: 34.11% Time: 15.06s
```

```
Epoch 33/50: 100%|     | 782/782 [00:15<00:00, 51.95it/s, loss=2.2392,
acc=34.56%]

Epoch 33/50 - Loss: 2.5134, Accuracy: 34.56% Time: 15.05s

Epoch 34/50: 100%|     | 782/782 [00:15<00:00, 51.98it/s, loss=3.0979,
acc=35.21%]

Epoch 34/50 - Loss: 2.4900, Accuracy: 35.21% Time: 15.05s

Epoch 35/50: 100%|     | 782/782 [00:15<00:00, 51.94it/s, loss=2.7125,
acc=35.34%]

Epoch 35/50 - Loss: 2.4788, Accuracy: 35.34% Time: 15.06s

Epoch 36/50: 100%|     | 782/782 [00:15<00:00, 51.89it/s, loss=2.7180,
acc=35.89%]

Epoch 36/50 - Loss: 2.4513, Accuracy: 35.89% Time: 15.07s

Epoch 37/50: 100%|     | 782/782 [00:15<00:00, 51.83it/s, loss=2.3343,
acc=36.91%]

Epoch 37/50 - Loss: 2.3941, Accuracy: 36.91% Time: 15.09s

Epoch 38/50: 100%|     | 782/782 [00:15<00:00, 52.09it/s, loss=1.9673,
acc=37.67%]

Epoch 38/50 - Loss: 2.3617, Accuracy: 37.67% Time: 15.01s

Epoch 39/50: 100%|     | 782/782 [00:15<00:00, 51.91it/s, loss=2.7149,
acc=37.95%]

Epoch 39/50 - Loss: 2.3527, Accuracy: 37.95% Time: 15.07s

Epoch 40/50: 100%|     | 782/782 [00:15<00:00, 52.08it/s, loss=2.5244,
acc=38.78%]

Epoch 40/50 - Loss: 2.3153, Accuracy: 38.78% Time: 15.02s

Epoch 41/50: 100%|     | 782/782 [00:15<00:00, 51.94it/s, loss=2.3808,
acc=39.29%]

Epoch 41/50 - Loss: 2.2821, Accuracy: 39.29% Time: 15.06s

Epoch 42/50: 100%|     | 782/782 [00:15<00:00, 52.00it/s, loss=1.8405,
acc=39.87%]

Epoch 42/50 - Loss: 2.2727, Accuracy: 39.87% Time: 15.04s

Epoch 43/50: 100%|     | 782/782 [00:15<00:00, 52.03it/s, loss=2.1176,
acc=39.86%]

Epoch 43/50 - Loss: 2.2560, Accuracy: 39.86% Time: 15.03s

Epoch 44/50: 100%|     | 782/782 [00:15<00:00, 51.97it/s, loss=2.4451,
acc=40.80%]

Epoch 44/50 - Loss: 2.2171, Accuracy: 40.80% Time: 15.05s
```

```
Epoch 45/50: 100%|      | 782/782 [00:15<00:00, 52.01it/s, loss=2.2449,
acc=41.77%]

Epoch 45/50 - Loss: 2.1799, Accuracy: 41.77% Time: 15.04s

Epoch 46/50: 100%|      | 782/782 [00:15<00:00, 51.96it/s, loss=2.4183,
acc=42.09%]

Epoch 46/50 - Loss: 2.1564, Accuracy: 42.09% Time: 15.05s

Epoch 47/50: 100%|      | 782/782 [00:15<00:00, 51.97it/s, loss=2.2990,
acc=42.70%]

Epoch 47/50 - Loss: 2.1288, Accuracy: 42.70% Time: 15.05s

Epoch 48/50: 100%|      | 782/782 [00:15<00:00, 52.11it/s, loss=1.7552,
acc=43.10%]

Epoch 48/50 - Loss: 2.1060, Accuracy: 43.10% Time: 15.01s

Epoch 49/50: 100%|      | 782/782 [00:15<00:00, 51.90it/s, loss=1.8592,
acc=43.72%]

Epoch 49/50 - Loss: 2.0737, Accuracy: 43.72% Time: 15.07s

Epoch 50/50: 100%|      | 782/782 [00:15<00:00, 51.90it/s, loss=1.8686,
acc=44.48%]

Epoch 50/50 - Loss: 2.0405, Accuracy: 44.48% Time: 15.07s
Average epoch training time: 15.04 seconds

Testing started…

Testing: 100%|      | 157/157 [00:02<00:00, 70.48it/s, accuracy=35.76%]

Final Test Loss: 2.5896, Final Test Accuracy: 35.76%

Visualizing results…
```



[3]:
```
#ViT from scratch - 4x4  patch size - 2 heads 4 layers 256 dim 512 mlp

# Delete model and optimizer variables
del model, optimizer, test_losses, test_accuracies, train_losses,␣
 ↪train_accuracies, epoch_times

# Clear CUDA cache if using GPU
gc.collect()
# Clear CUDA cache
if torch.cuda.is_available():
    torch.cuda.empty_cache()


# Initialize model
model = VisionTransformer(
    image_size=image_size,
    patch_size=patch_size,
    num_classes=num_classes,
    embed_dim=hidden_dim,
    num_heads=num_heads,
    num_layers=num_layers,
    mlp_dim=mlp_dim
).to(device)
# Loss and optimizer
criterion = nn.CrossEntropyLoss()
optimizer = torch.optim.AdamW(model.parameters(), lr=learning_rate,␣
 ↪weight_decay=0.01)


# Display model summary
summary(model,
        input_size=(batch_size, 3, image_size, image_size),
        col_names=["input_size", "output_size", "kernel_size", "num_params",␣
 ↪"mult_adds", "trainable"],
        col_width=20,
        depth=5,
        verbose=True,
        device=device)

# Hyperparameters
image_size = 32
patch_size = 4
num_classes = 100
num_epochs = 50
batch_size = 64
learning_rate = 0.001
num_heads = 2
```

```python
num_layers = 4
hidden_dim = 256
mlp_dim = 512

# Run training and testing
if __name__ == '__main__':
    print("Training started...")
    train_losses, train_accuracies, epoch_times = train()
    print("\nTesting started...")
    test_losses, test_accuracies = test()

    # Visualize results
    print("\nVisualizing results...")
    visualize_results(train_losses, train_accuracies, test_losses,␣
 ↪test_accuracies)
```

```
==========================================================================================
==========================================================================================
=====
Layer (type:depth-idx)                        Input Shape          Output Shape
Kernel Shape         Param #              Mult-Adds            Trainable
==========================================================================================
==========================================================================================
=====
VisionTransformer                             [64, 3, 32, 32]      [64, 100]
--                   4,608                --                   True
 PatchEmbedding: 1-1                          [64, 3, 32, 32]      [64, 16, 256]
--                   --                   --                   True
     Conv2d: 2-1                             [64, 3, 32, 32]      [64, 256, 4,
4]       [8, 8]               49,408               50,593,792           True
 Dropout: 1-2                                [64, 17, 256]        [64, 17, 256]
--                   --                   --                   --
 ModuleList: 1-3                             --                   --
--                   --                   --                   True
     TransformerEncoder: 2-2                 [64, 17, 256]        [64, 17, 256]
--                   --                   --                   True
          LayerNorm: 3-1                     [64, 17, 256]        [64, 17, 256]
--                   512                  32,768               True
          MultiHeadSelfAttention: 3-2        [64, 17, 256]        [64, 17, 256]
--                   --                   --                   True
               Linear: 4-1                   [64, 17, 256]        [64, 17, 768]
--                   197,376              12,632,064           True
               Linear: 4-2                   [64, 17, 256]        [64, 17, 256]
--                   65,792               4,210,688            True
          LayerNorm: 3-3                     [64, 17, 256]        [64, 17, 256]
--                   512                  32,768               True
          MLP: 3-4                           [64, 17, 256]        [64, 17, 256]
--                   --                   --                   True
```

```
            Linear: 4-3                    [64, 17, 256]        [64, 17, 512]
--                  131,584                8,421,376            True
            GELU: 4-4                      [64, 17, 512]        [64, 17, 512]
--                  --                     --                   --
            Linear: 4-5                    [64, 17, 512]        [64, 17, 256]
--                  131,328                8,404,992            True
    TransformerEncoder: 2-3                [64, 17, 256]        [64, 17, 256]
--                  --                     --                   True
        LayerNorm: 3-5                     [64, 17, 256]        [64, 17, 256]
--                  512                    32,768               True
        MultiHeadSelfAttention: 3-6        [64, 17, 256]        [64, 17, 256]
--                  --                     --                   True
            Linear: 4-6                    [64, 17, 256]        [64, 17, 768]
--                  197,376                12,632,064           True
            Linear: 4-7                    [64, 17, 256]        [64, 17, 256]
--                  65,792                 4,210,688            True
        LayerNorm: 3-7                     [64, 17, 256]        [64, 17, 256]
--                  512                    32,768               True
        MLP: 3-8                           [64, 17, 256]        [64, 17, 256]
--                  --                     --                   True
            Linear: 4-8                    [64, 17, 256]        [64, 17, 512]
--                  131,584                8,421,376            True
            GELU: 4-9                      [64, 17, 512]        [64, 17, 512]
--                  --                     --                   --
            Linear: 4-10                   [64, 17, 512]        [64, 17, 256]
--                  131,328                8,404,992            True
    TransformerEncoder: 2-4                [64, 17, 256]        [64, 17, 256]
--                  --                     --                   True
        LayerNorm: 3-9                     [64, 17, 256]        [64, 17, 256]
--                  512                    32,768               True
        MultiHeadSelfAttention: 3-10       [64, 17, 256]        [64, 17, 256]
--                  --                     --                   True
            Linear: 4-11                   [64, 17, 256]        [64, 17, 768]
--                  197,376                12,632,064           True
            Linear: 4-12                   [64, 17, 256]        [64, 17, 256]
--                  65,792                 4,210,688            True
        LayerNorm: 3-11                    [64, 17, 256]        [64, 17, 256]
--                  512                    32,768               True
        MLP: 3-12                          [64, 17, 256]        [64, 17, 256]
--                  --                     --                   True
            Linear: 4-13                   [64, 17, 256]        [64, 17, 512]
--                  131,584                8,421,376            True
            GELU: 4-14                     [64, 17, 512]        [64, 17, 512]
--                  --                     --                   --
            Linear: 4-15                   [64, 17, 512]        [64, 17, 256]
--                  131,328                8,404,992            True
    TransformerEncoder: 2-5                [64, 17, 256]        [64, 17, 256]
--                  --                     --                   True
```

```
          LayerNorm: 3-13                   [64, 17, 256]        [64, 17, 256]
--                    512                   32,768               True
          MultiHeadSelfAttention: 3-14      [64, 17, 256]        [64, 17, 256]
--                    --                    --                   True
               Linear: 4-16                 [64, 17, 256]        [64, 17, 768]
--                    197,376               12,632,064           True
               Linear: 4-17                 [64, 17, 256]        [64, 17, 256]
--                    65,792                4,210,688            True
          LayerNorm: 3-15                   [64, 17, 256]        [64, 17, 256]
--                    512                   32,768               True
          MLP: 3-16                         [64, 17, 256]        [64, 17, 256]
--                    --                    --                   True
               Linear: 4-18                 [64, 17, 256]        [64, 17, 512]
--                    131,584               8,421,376            True
               GELU: 4-19                   [64, 17, 512]        [64, 17, 512]
--                    --                    --                   --
               Linear: 4-20                 [64, 17, 512]        [64, 17, 256]
--                    131,328               8,404,992            True
 LayerNorm: 1-4                               [64, 17, 256]        [64, 17, 256]
--                    512                   32,768               True
 Linear: 1-5                                  [64, 256]            [64, 100]
--                    25,700                1,644,800            True
==========================================================================
==========================================================================
=====
Total params: 2,188,644
Trainable params: 2,188,644
Non-trainable params: 0
Total mult-adds (Units.MEGABYTES): 187.21
==========================================================================
==========================================================================
=====
Input size (MB): 0.79
Forward/backward pass size (MB): 84.59
Params size (MB): 8.74
Estimated Total Size (MB): 94.12
==========================================================================
==========================================================================
=====
Training started…

Epoch 1/50:   1%|          | 6/782 [00:00<00:13, 56.85it/s, loss=4.6195,
acc=1.85%]

Input images shape: torch.Size([64, 3, 32, 32])
Labels shape: torch.Size([64])
Labels values: tensor([26, 49, 72, 97, 71, 15,  2, 72, 81, 37], device='cuda:0')
Model outputs shape: torch.Size([64, 100])
Expected outputs shape: torch.Size([64, 100])
```

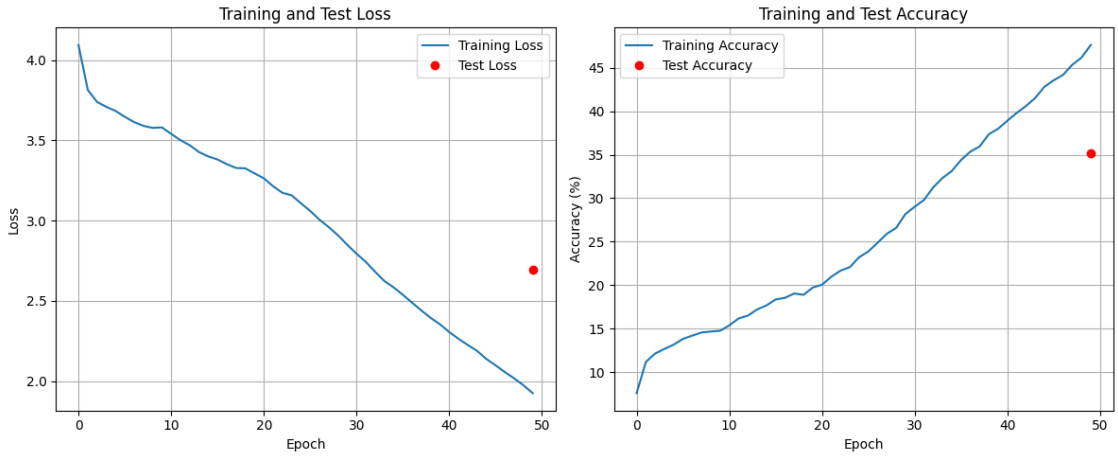
```
Epoch 1/50: 100%|      | 782/782 [00:13<00:00, 59.76it/s, loss=3.7640,
acc=7.58%]

Epoch 1/50 - Loss: 4.0936, Accuracy: 7.58% Time: 13.09s

Epoch 2/50: 100%|      | 782/782 [00:12<00:00, 60.47it/s, loss=3.4438,
acc=11.17%]

Epoch 2/50 - Loss: 3.8135, Accuracy: 11.17% Time: 12.93s

Epoch 3/50: 100%|      | 782/782 [00:12<00:00, 60.56it/s, loss=4.1943,
acc=12.14%]

Epoch 3/50 - Loss: 3.7390, Accuracy: 12.14% Time: 12.91s

Epoch 4/50: 100%|      | 782/782 [00:13<00:00, 60.06it/s, loss=3.6431,
acc=12.67%]

Epoch 4/50 - Loss: 3.7081, Accuracy: 12.67% Time: 13.02s

Epoch 5/50: 100%|      | 782/782 [00:12<00:00, 60.62it/s, loss=3.4996,
acc=13.16%]

Epoch 5/50 - Loss: 3.6831, Accuracy: 13.16% Time: 12.90s

Epoch 6/50: 100%|      | 782/782 [00:13<00:00, 58.85it/s, loss=3.5769,
acc=13.82%]

Epoch 6/50 - Loss: 3.6463, Accuracy: 13.82% Time: 13.29s

Epoch 7/50: 100%|      | 782/782 [00:13<00:00, 57.66it/s, loss=3.3669,
acc=14.20%]

Epoch 7/50 - Loss: 3.6133, Accuracy: 14.20% Time: 13.56s

Epoch 8/50: 100%|      | 782/782 [00:13<00:00, 57.09it/s, loss=3.3686,
acc=14.55%]

Epoch 8/50 - Loss: 3.5901, Accuracy: 14.55% Time: 13.70s

Epoch 9/50: 100%|      | 782/782 [00:13<00:00, 57.44it/s, loss=3.6785,
acc=14.67%]

Epoch 9/50 - Loss: 3.5775, Accuracy: 14.67% Time: 13.62s

Epoch 10/50: 100%|      | 782/782 [00:13<00:00, 56.90it/s, loss=3.2495,
acc=14.75%]

Epoch 10/50 - Loss: 3.5802, Accuracy: 14.75% Time: 13.75s

Epoch 11/50: 100%|      | 782/782 [00:13<00:00, 57.66it/s, loss=3.5738,
acc=15.38%]

Epoch 11/50 - Loss: 3.5399, Accuracy: 15.38% Time: 13.56s

Epoch 12/50: 100%|      | 782/782 [00:13<00:00, 58.57it/s, loss=3.7209,
acc=16.17%]

Epoch 12/50 - Loss: 3.5007, Accuracy: 16.17% Time: 13.35s
```

```
Epoch 13/50: 100%|      | 782/782 [00:13<00:00, 57.74it/s, loss=3.4859,
acc=16.50%]

Epoch 13/50 - Loss: 3.4694, Accuracy: 16.50% Time: 13.54s

Epoch 14/50: 100%|      | 782/782 [00:13<00:00, 58.83it/s, loss=3.3528,
acc=17.21%]

Epoch 14/50 - Loss: 3.4266, Accuracy: 17.21% Time: 13.29s

Epoch 15/50: 100%|      | 782/782 [00:13<00:00, 58.33it/s, loss=3.6926,
acc=17.66%]

Epoch 15/50 - Loss: 3.3993, Accuracy: 17.66% Time: 13.41s

Epoch 16/50: 100%|      | 782/782 [00:13<00:00, 57.39it/s, loss=3.4201,
acc=18.35%]

Epoch 16/50 - Loss: 3.3803, Accuracy: 18.35% Time: 13.63s

Epoch 17/50: 100%|      | 782/782 [00:13<00:00, 58.20it/s, loss=3.8694,
acc=18.54%]

Epoch 17/50 - Loss: 3.3514, Accuracy: 18.54% Time: 13.44s

Epoch 18/50: 100%|      | 782/782 [00:13<00:00, 58.10it/s, loss=3.2690,
acc=19.04%]

Epoch 18/50 - Loss: 3.3276, Accuracy: 19.04% Time: 13.46s

Epoch 19/50: 100%|      | 782/782 [00:13<00:00, 57.80it/s, loss=2.8873,
acc=18.89%]

Epoch 19/50 - Loss: 3.3252, Accuracy: 18.89% Time: 13.53s

Epoch 20/50: 100%|      | 782/782 [00:13<00:00, 57.73it/s, loss=3.6775,
acc=19.72%]

Epoch 20/50 - Loss: 3.2934, Accuracy: 19.72% Time: 13.55s

Epoch 21/50: 100%|      | 782/782 [00:13<00:00, 57.87it/s, loss=3.3006,
acc=20.05%]

Epoch 21/50 - Loss: 3.2631, Accuracy: 20.05% Time: 13.51s

Epoch 22/50: 100%|      | 782/782 [00:13<00:00, 57.62it/s, loss=3.5173,
acc=20.97%]

Epoch 22/50 - Loss: 3.2135, Accuracy: 20.97% Time: 13.57s

Epoch 23/50: 100%|      | 782/782 [00:13<00:00, 58.06it/s, loss=3.7378,
acc=21.67%]

Epoch 23/50 - Loss: 3.1730, Accuracy: 21.67% Time: 13.47s

Epoch 24/50: 100%|      | 782/782 [00:13<00:00, 57.01it/s, loss=3.2954,
acc=22.07%]

Epoch 24/50 - Loss: 3.1568, Accuracy: 22.07% Time: 13.72s
```

```
Epoch 25/50: 100%|      | 782/782 [00:13<00:00, 57.11it/s, loss=2.5425,
acc=23.21%]

Epoch 25/50 - Loss: 3.1073, Accuracy: 23.21% Time: 13.69s

Epoch 26/50: 100%|      | 782/782 [00:13<00:00, 57.44it/s, loss=3.1361,
acc=23.87%]

Epoch 26/50 - Loss: 3.0594, Accuracy: 23.87% Time: 13.62s

Epoch 27/50: 100%|      | 782/782 [00:13<00:00, 57.15it/s, loss=3.3904,
acc=24.88%]

Epoch 27/50 - Loss: 3.0051, Accuracy: 24.88% Time: 13.68s

Epoch 28/50: 100%|      | 782/782 [00:13<00:00, 58.32it/s, loss=2.7648,
acc=25.91%]

Epoch 28/50 - Loss: 2.9590, Accuracy: 25.91% Time: 13.41s

Epoch 29/50: 100%|      | 782/782 [00:13<00:00, 58.48it/s, loss=2.3182,
acc=26.59%]

Epoch 29/50 - Loss: 2.9078, Accuracy: 26.59% Time: 13.37s

Epoch 30/50: 100%|      | 782/782 [00:13<00:00, 58.49it/s, loss=3.3639,
acc=28.18%]

Epoch 30/50 - Loss: 2.8485, Accuracy: 28.18% Time: 13.37s

Epoch 31/50: 100%|      | 782/782 [00:13<00:00, 58.52it/s, loss=1.5699,
acc=29.02%]

Epoch 31/50 - Loss: 2.7931, Accuracy: 29.02% Time: 13.36s

Epoch 32/50: 100%|      | 782/782 [00:13<00:00, 58.43it/s, loss=2.4822,
acc=29.79%]

Epoch 32/50 - Loss: 2.7422, Accuracy: 29.79% Time: 13.38s

Epoch 33/50: 100%|      | 782/782 [00:13<00:00, 58.37it/s, loss=2.2379,
acc=31.25%]

Epoch 33/50 - Loss: 2.6809, Accuracy: 31.25% Time: 13.40s

Epoch 34/50: 100%|      | 782/782 [00:13<00:00, 58.42it/s, loss=2.1067,
acc=32.32%]

Epoch 34/50 - Loss: 2.6230, Accuracy: 32.32% Time: 13.39s

Epoch 35/50: 100%|      | 782/782 [00:13<00:00, 58.21it/s, loss=2.2259,
acc=33.13%]

Epoch 35/50 - Loss: 2.5833, Accuracy: 33.13% Time: 13.44s

Epoch 36/50: 100%|      | 782/782 [00:13<00:00, 58.25it/s, loss=3.0151,
acc=34.39%]

Epoch 36/50 - Loss: 2.5373, Accuracy: 34.39% Time: 13.43s
```

```
Epoch 37/50: 100%|      | 782/782 [00:13<00:00, 58.03it/s, loss=2.6638,
acc=35.36%]

Epoch 37/50 - Loss: 2.4878, Accuracy: 35.36% Time: 13.48s

Epoch 38/50: 100%|      | 782/782 [00:13<00:00, 58.36it/s, loss=2.0926,
acc=35.95%]

Epoch 38/50 - Loss: 2.4394, Accuracy: 35.95% Time: 13.40s

Epoch 39/50: 100%|      | 782/782 [00:13<00:00, 58.16it/s, loss=2.3077,
acc=37.36%]

Epoch 39/50 - Loss: 2.3933, Accuracy: 37.36% Time: 13.45s

Epoch 40/50: 100%|      | 782/782 [00:13<00:00, 58.25it/s, loss=2.4948,
acc=37.99%]

Epoch 40/50 - Loss: 2.3534, Accuracy: 37.99% Time: 13.43s

Epoch 41/50: 100%|      | 782/782 [00:13<00:00, 58.47it/s, loss=2.0013,
acc=38.91%]

Epoch 41/50 - Loss: 2.3054, Accuracy: 38.91% Time: 13.38s

Epoch 42/50: 100%|      | 782/782 [00:13<00:00, 58.41it/s, loss=2.1808,
acc=39.81%]

Epoch 42/50 - Loss: 2.2630, Accuracy: 39.81% Time: 13.39s

Epoch 43/50: 100%|      | 782/782 [00:13<00:00, 57.41it/s, loss=2.4386,
acc=40.60%]

Epoch 43/50 - Loss: 2.2248, Accuracy: 40.60% Time: 13.62s

Epoch 44/50: 100%|      | 782/782 [00:13<00:00, 57.98it/s, loss=3.0988,
acc=41.53%]

Epoch 44/50 - Loss: 2.1882, Accuracy: 41.53% Time: 13.49s

Epoch 45/50: 100%|      | 782/782 [00:13<00:00, 58.07it/s, loss=2.2534,
acc=42.82%]

Epoch 45/50 - Loss: 2.1377, Accuracy: 42.82% Time: 13.47s

Epoch 46/50: 100%|      | 782/782 [00:13<00:00, 58.36it/s, loss=3.1742,
acc=43.56%]

Epoch 46/50 - Loss: 2.0979, Accuracy: 43.56% Time: 13.40s

Epoch 47/50: 100%|      | 782/782 [00:13<00:00, 58.78it/s, loss=2.1448,
acc=44.18%]

Epoch 47/50 - Loss: 2.0560, Accuracy: 44.18% Time: 13.31s

Epoch 48/50: 100%|      | 782/782 [00:13<00:00, 58.78it/s, loss=2.3608,
acc=45.34%]

Epoch 48/50 - Loss: 2.0168, Accuracy: 45.34% Time: 13.31s
```

```
Epoch 49/50: 100%|      | 782/782 [00:13<00:00, 58.39it/s, loss=2.1622,
acc=46.18%]

Epoch 49/50 - Loss: 1.9740, Accuracy: 46.18% Time: 13.39s

Epoch 50/50: 100%|      | 782/782 [00:13<00:00, 58.59it/s, loss=2.7135,
acc=47.62%]

Epoch 50/50 - Loss: 1.9247, Accuracy: 47.62% Time: 13.35s
Average epoch training time: 13.42 seconds


Testing started…

Testing: 100%|      | 157/157 [00:02<00:00, 75.20it/s, accuracy=35.12%]

Final Test Loss: 2.6956, Final Test Accuracy: 35.12%


Visualizing results…
```



[4]:
```python
#ViT from scratch - 8x8  patch size - 8 heads 8 layers 128 dim 256 mlp

# Delete model and optimizer variables
del model, optimizer, test_losses, test_accuracies, train_losses,␣
 ↪train_accuracies, epoch_times

# Clear CUDA cache if using GPU
gc.collect()
# Clear CUDA cache
if torch.cuda.is_available():
    torch.cuda.empty_cache()


# Initialize model
model = VisionTransformer(
```

```
    image_size=image_size,
    patch_size=patch_size,
    num_classes=num_classes,
    embed_dim=hidden_dim,
    num_heads=num_heads,
    num_layers=num_layers,
    mlp_dim=mlp_dim
).to(device)
# Loss and optimizer
criterion = nn.CrossEntropyLoss()
optimizer = torch.optim.AdamW(model.parameters(), lr=learning_rate,␣
 ↪weight_decay=0.01)


# Display model summary
summary(model,
        input_size=(batch_size, 3, image_size, image_size),
        col_names=["input_size", "output_size", "kernel_size", "num_params",␣
 ↪"mult_adds", "trainable"],
        col_width=20,
        depth=5,
        verbose=True,
        device=device)

# Hyperparameters
image_size = 32
patch_size = 8
num_classes = 100
num_epochs = 50
batch_size = 64
learning_rate = 0.001
num_heads = 8
num_layers = 8
hidden_dim = 128
mlp_dim = 256

# Run training and testing
if __name__ == '__main__':
    print("Training started...")
    train_losses, train_accuracies, epoch_times = train()
    print("\nTesting started...")
    test_losses, test_accuracies = test()

    # Visualize results
    print("\nVisualizing results...")
    visualize_results(train_losses, train_accuracies, test_losses,␣
 ↪test_accuracies)
```

```
==========================================================================================
==========================================================================================
=====
Layer (type:depth-idx)                        Input Shape          Output Shape
Kernel Shape         Param #              Mult-Adds            Trainable
==========================================================================================
==========================================================================================
=====
VisionTransformer                             [64, 3, 32, 32]      [64, 100]
--                   16,896               --                   True
 PatchEmbedding: 1-1                            [64, 3, 32, 32]      [64, 64, 256]
--                   --                   --                   True
     Conv2d: 2-1                              [64, 3, 32, 32]      [64, 256, 8,
8]      [4, 4]               12,544               51,380,224           True
 Dropout: 1-2                                 [64, 65, 256]        [64, 65, 256]
--                   --                   --                   --
 ModuleList: 1-3                              --                   --
--                   --                   --                   True
     TransformerEncoder: 2-2                  [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
          LayerNorm: 3-1                      [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
          MultiHeadSelfAttention: 3-2         [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
               Linear: 4-1                    [64, 65, 256]        [64, 65, 768]
--                   197,376              12,632,064           True
               Linear: 4-2                    [64, 65, 256]        [64, 65, 256]
--                   65,792               4,210,688            True
          LayerNorm: 3-3                      [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
          MLP: 3-4                            [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
               Linear: 4-3                    [64, 65, 256]        [64, 65, 512]
--                   131,584              8,421,376            True
               GELU: 4-4                      [64, 65, 512]        [64, 65, 512]
--                   --                   --                   --
               Linear: 4-5                    [64, 65, 512]        [64, 65, 256]
--                   131,328              8,404,992            True
     TransformerEncoder: 2-3                  [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
          LayerNorm: 3-5                      [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
          MultiHeadSelfAttention: 3-6         [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
               Linear: 4-6                    [64, 65, 256]        [64, 65, 768]
--                   197,376              12,632,064           True
               Linear: 4-7                    [64, 65, 256]        [64, 65, 256]
--                   65,792               4,210,688            True
```

```
          LayerNorm: 3-7                  [64, 65, 256]       [64, 65, 256]
--                    512                 32,768              True
          MLP: 3-8                        [64, 65, 256]       [64, 65, 256]
--                    --                  --                  True
               Linear: 4-8                [64, 65, 256]       [64, 65, 512]
--                    131,584             8,421,376           True
               GELU: 4-9                  [64, 65, 512]       [64, 65, 512]
--                    --                  --                  --
               Linear: 4-10               [64, 65, 512]       [64, 65, 256]
--                    131,328             8,404,992           True
     TransformerEncoder: 2-4              [64, 65, 256]       [64, 65, 256]
--                    --                  --                  True
          LayerNorm: 3-9                  [64, 65, 256]       [64, 65, 256]
--                    512                 32,768              True
          MultiHeadSelfAttention: 3-10    [64, 65, 256]       [64, 65, 256]
--                    --                  --                  True
               Linear: 4-11               [64, 65, 256]       [64, 65, 768]
--                    197,376             12,632,064          True
               Linear: 4-12               [64, 65, 256]       [64, 65, 256]
--                    65,792              4,210,688           True
          LayerNorm: 3-11                 [64, 65, 256]       [64, 65, 256]
--                    512                 32,768              True
          MLP: 3-12                       [64, 65, 256]       [64, 65, 256]
--                    --                  --                  True
               Linear: 4-13               [64, 65, 256]       [64, 65, 512]
--                    131,584             8,421,376           True
               GELU: 4-14                 [64, 65, 512]       [64, 65, 512]
--                    --                  --                  --
               Linear: 4-15               [64, 65, 512]       [64, 65, 256]
--                    131,328             8,404,992           True
     TransformerEncoder: 2-5              [64, 65, 256]       [64, 65, 256]
--                    --                  --                  True
          LayerNorm: 3-13                 [64, 65, 256]       [64, 65, 256]
--                    512                 32,768              True
          MultiHeadSelfAttention: 3-14    [64, 65, 256]       [64, 65, 256]
--                    --                  --                  True
               Linear: 4-16               [64, 65, 256]       [64, 65, 768]
--                    197,376             12,632,064          True
               Linear: 4-17               [64, 65, 256]       [64, 65, 256]
--                    65,792              4,210,688           True
          LayerNorm: 3-15                 [64, 65, 256]       [64, 65, 256]
--                    512                 32,768              True
          MLP: 3-16                       [64, 65, 256]       [64, 65, 256]
--                    --                  --                  True
               Linear: 4-18               [64, 65, 256]       [64, 65, 512]
--                    131,584             8,421,376           True
               GELU: 4-19                 [64, 65, 512]       [64, 65, 512]
--                    --                  --                  --
```

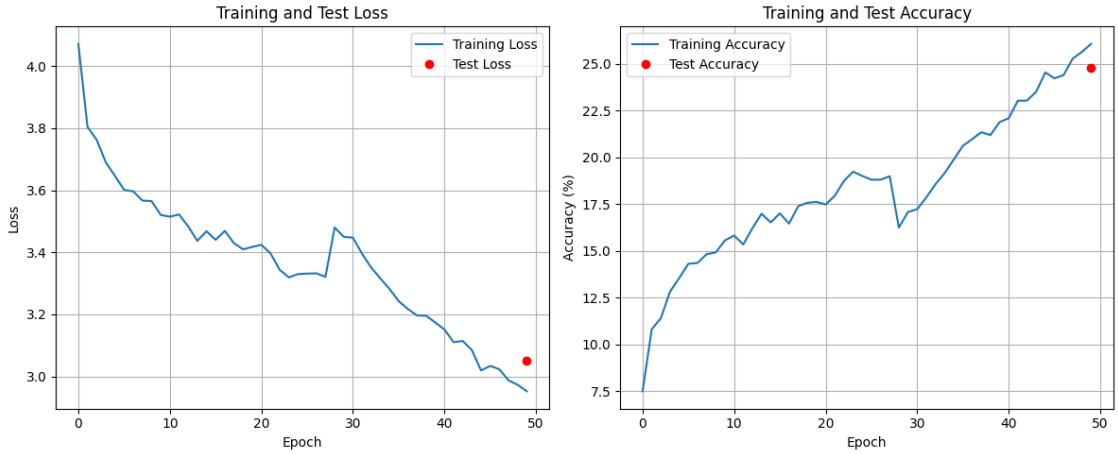
```
              Linear: 4-20                   [64, 65, 512]         [64, 65, 256]
--                    131,328              8,404,992             True
 LayerNorm: 1-4                              [64, 65, 256]         [64, 65, 256]
--                    512                  32,768                True
 Linear: 1-5                                 [64, 256]             [64, 100]
--                    25,700               1,644,800             True
==========================================================================
==========================================================================
=====
Total params: 2,164,068
Trainable params: 2,164,068
Non-trainable params: 0
Total mult-adds (Units.MEGABYTES): 188.00
==========================================================================
==========================================================================
=====
Input size (MB): 0.79
Forward/backward pass size (MB): 323.67
Params size (MB): 8.59
Estimated Total Size (MB): 333.04
==========================================================================
==========================================================================
=====
Training started…

Epoch 1/50:   1%|          | 5/782 [00:00<00:16, 48.37it/s, loss=4.6743,
acc=1.74%]

Input images shape: torch.Size([64, 3, 32, 32])
Labels shape: torch.Size([64])
Labels values: tensor([49, 60, 79,  4, 46, 20, 91,  8, 27, 72], device='cuda:0')
Model outputs shape: torch.Size([64, 100])
Expected outputs shape: torch.Size([64, 100])

Epoch 1/50: 100%|     | 782/782 [00:14<00:00, 53.48it/s, loss=4.5227,
acc=7.49%]

Epoch 1/50 - Loss: 4.0720, Accuracy: 7.49% Time: 14.62s

Epoch 2/50: 100%|     | 782/782 [00:14<00:00, 54.09it/s, loss=3.6253,
acc=10.81%]

Epoch 2/50 - Loss: 3.8044, Accuracy: 10.81% Time: 14.46s

Epoch 3/50: 100%|     | 782/782 [00:14<00:00, 53.93it/s, loss=4.2667,
acc=11.40%]

Epoch 3/50 - Loss: 3.7627, Accuracy: 11.40% Time: 14.50s

Epoch 4/50: 100%|     | 782/782 [00:14<00:00, 53.81it/s, loss=3.5379,
acc=12.82%]

Epoch 4/50 - Loss: 3.6899, Accuracy: 12.82% Time: 14.53s
```

```
Epoch 5/50: 100%|      | 782/782 [00:14<00:00, 53.87it/s, loss=3.6015,
acc=13.55%]

Epoch 5/50 - Loss: 3.6464, Accuracy: 13.55% Time: 14.52s

Epoch 6/50: 100%|      | 782/782 [00:14<00:00, 53.96it/s, loss=3.8604,
acc=14.31%]

Epoch 6/50 - Loss: 3.6014, Accuracy: 14.31% Time: 14.49s

Epoch 7/50: 100%|      | 782/782 [00:14<00:00, 54.21it/s, loss=3.8767,
acc=14.35%]

Epoch 7/50 - Loss: 3.5964, Accuracy: 14.35% Time: 14.43s

Epoch 8/50: 100%|      | 782/782 [00:14<00:00, 53.74it/s, loss=3.4773,
acc=14.83%]

Epoch 8/50 - Loss: 3.5667, Accuracy: 14.83% Time: 14.55s

Epoch 9/50: 100%|      | 782/782 [00:14<00:00, 53.81it/s, loss=3.5850,
acc=14.92%]

Epoch 9/50 - Loss: 3.5652, Accuracy: 14.92% Time: 14.53s

Epoch 10/50: 100%|      | 782/782 [00:14<00:00, 53.85it/s, loss=3.1912,
acc=15.56%]

Epoch 10/50 - Loss: 3.5205, Accuracy: 15.56% Time: 14.52s

Epoch 11/50: 100%|      | 782/782 [00:14<00:00, 53.81it/s, loss=3.4857,
acc=15.82%]

Epoch 11/50 - Loss: 3.5147, Accuracy: 15.82% Time: 14.53s

Epoch 12/50: 100%|      | 782/782 [00:14<00:00, 52.68it/s, loss=3.3031,
acc=15.34%]

Epoch 12/50 - Loss: 3.5220, Accuracy: 15.34% Time: 14.84s

Epoch 13/50: 100%|      | 782/782 [00:15<00:00, 51.33it/s, loss=3.6058,
acc=16.21%]

Epoch 13/50 - Loss: 3.4835, Accuracy: 16.21% Time: 15.24s

Epoch 14/50: 100%|      | 782/782 [00:15<00:00, 51.64it/s, loss=3.8761,
acc=16.98%]

Epoch 14/50 - Loss: 3.4369, Accuracy: 16.98% Time: 15.14s

Epoch 15/50: 100%|      | 782/782 [00:15<00:00, 51.70it/s, loss=3.0232,
acc=16.52%]

Epoch 15/50 - Loss: 3.4680, Accuracy: 16.52% Time: 15.13s

Epoch 16/50: 100%|      | 782/782 [00:15<00:00, 51.91it/s, loss=2.9667,
acc=17.00%]

Epoch 16/50 - Loss: 3.4401, Accuracy: 17.00% Time: 15.07s
```

```
Epoch 17/50: 100%|      | 782/782 [00:15<00:00, 51.80it/s, loss=3.6697,
acc=16.46%]

Epoch 17/50 - Loss: 3.4691, Accuracy: 16.46% Time: 15.10s

Epoch 18/50: 100%|      | 782/782 [00:15<00:00, 51.88it/s, loss=3.3488,
acc=17.39%]

Epoch 18/50 - Loss: 3.4292, Accuracy: 17.39% Time: 15.08s

Epoch 19/50: 100%|      | 782/782 [00:15<00:00, 51.73it/s, loss=3.7846,
acc=17.57%]

Epoch 19/50 - Loss: 3.4094, Accuracy: 17.57% Time: 15.12s

Epoch 20/50: 100%|      | 782/782 [00:15<00:00, 51.62it/s, loss=3.4846,
acc=17.62%]

Epoch 20/50 - Loss: 3.4173, Accuracy: 17.62% Time: 15.15s

Epoch 21/50: 100%|      | 782/782 [00:15<00:00, 51.82it/s, loss=2.7287,
acc=17.48%]

Epoch 21/50 - Loss: 3.4240, Accuracy: 17.48% Time: 15.09s

Epoch 22/50: 100%|      | 782/782 [00:15<00:00, 51.69it/s, loss=3.3085,
acc=17.95%]

Epoch 22/50 - Loss: 3.3964, Accuracy: 17.95% Time: 15.13s

Epoch 23/50: 100%|      | 782/782 [00:15<00:00, 51.77it/s, loss=3.5449,
acc=18.74%]

Epoch 23/50 - Loss: 3.3435, Accuracy: 18.74% Time: 15.11s

Epoch 24/50: 100%|      | 782/782 [00:15<00:00, 51.86it/s, loss=3.3244,
acc=19.23%]

Epoch 24/50 - Loss: 3.3191, Accuracy: 19.23% Time: 15.08s

Epoch 25/50: 100%|      | 782/782 [00:15<00:00, 51.83it/s, loss=3.4498,
acc=19.01%]

Epoch 25/50 - Loss: 3.3298, Accuracy: 19.01% Time: 15.09s

Epoch 26/50: 100%|      | 782/782 [00:15<00:00, 51.81it/s, loss=3.5642,
acc=18.80%]

Epoch 26/50 - Loss: 3.3313, Accuracy: 18.80% Time: 15.09s

Epoch 27/50: 100%|      | 782/782 [00:15<00:00, 51.81it/s, loss=3.3372,
acc=18.80%]

Epoch 27/50 - Loss: 3.3321, Accuracy: 18.80% Time: 15.09s

Epoch 28/50: 100%|      | 782/782 [00:15<00:00, 51.55it/s, loss=3.4855,
acc=18.99%]

Epoch 28/50 - Loss: 3.3208, Accuracy: 18.99% Time: 15.17s
```

```
Epoch 29/50: 100%|      | 782/782 [00:15<00:00, 51.27it/s, loss=3.7396,
acc=16.25%]

Epoch 29/50 - Loss: 3.4801, Accuracy: 16.25% Time: 15.25s

Epoch 30/50: 100%|      | 782/782 [00:15<00:00, 51.76it/s, loss=3.6960,
acc=17.08%]

Epoch 30/50 - Loss: 3.4497, Accuracy: 17.08% Time: 15.11s

Epoch 31/50: 100%|      | 782/782 [00:15<00:00, 52.00it/s, loss=3.8995,
acc=17.23%]

Epoch 31/50 - Loss: 3.4472, Accuracy: 17.23% Time: 15.04s

Epoch 32/50: 100%|      | 782/782 [00:15<00:00, 52.01it/s, loss=2.8157,
acc=17.86%]

Epoch 32/50 - Loss: 3.3954, Accuracy: 17.86% Time: 15.04s

Epoch 33/50: 100%|      | 782/782 [00:14<00:00, 52.21it/s, loss=3.2336,
acc=18.57%]

Epoch 33/50 - Loss: 3.3507, Accuracy: 18.57% Time: 14.98s

Epoch 34/50: 100%|      | 782/782 [00:15<00:00, 52.04it/s, loss=2.9949,
acc=19.16%]

Epoch 34/50 - Loss: 3.3157, Accuracy: 19.16% Time: 15.03s

Epoch 35/50: 100%|      | 782/782 [00:15<00:00, 51.92it/s, loss=3.1473,
acc=19.89%]

Epoch 35/50 - Loss: 3.2817, Accuracy: 19.89% Time: 15.06s

Epoch 36/50: 100%|      | 782/782 [00:15<00:00, 52.07it/s, loss=2.4144,
acc=20.62%]

Epoch 36/50 - Loss: 3.2426, Accuracy: 20.62% Time: 15.02s

Epoch 37/50: 100%|      | 782/782 [00:14<00:00, 52.19it/s, loss=3.7233,
acc=20.97%]

Epoch 37/50 - Loss: 3.2171, Accuracy: 20.97% Time: 14.99s

Epoch 38/50: 100%|      | 782/782 [00:15<00:00, 52.12it/s, loss=3.4301,
acc=21.34%]

Epoch 38/50 - Loss: 3.1964, Accuracy: 21.34% Time: 15.01s

Epoch 39/50: 100%|      | 782/782 [00:14<00:00, 52.78it/s, loss=2.7143,
acc=21.19%]

Epoch 39/50 - Loss: 3.1953, Accuracy: 21.19% Time: 14.82s

Epoch 40/50: 100%|      | 782/782 [00:14<00:00, 52.85it/s, loss=3.7725,
acc=21.88%]

Epoch 40/50 - Loss: 3.1735, Accuracy: 21.88% Time: 14.80s
```

```
Epoch 41/50: 100%|      | 782/782 [00:14<00:00, 52.98it/s, loss=3.5079,
acc=22.09%]

Epoch 41/50 - Loss: 3.1509, Accuracy: 22.09% Time: 14.76s

Epoch 42/50: 100%|      | 782/782 [00:14<00:00, 52.78it/s, loss=3.5203,
acc=23.03%]

Epoch 42/50 - Loss: 3.1102, Accuracy: 23.03% Time: 14.82s

Epoch 43/50: 100%|      | 782/782 [00:14<00:00, 52.81it/s, loss=3.2405,
acc=23.04%]

Epoch 43/50 - Loss: 3.1139, Accuracy: 23.04% Time: 14.81s

Epoch 44/50: 100%|      | 782/782 [00:14<00:00, 53.02it/s, loss=3.2019,
acc=23.51%]

Epoch 44/50 - Loss: 3.0845, Accuracy: 23.51% Time: 14.75s

Epoch 45/50: 100%|      | 782/782 [00:14<00:00, 52.85it/s, loss=2.9134,
acc=24.54%]

Epoch 45/50 - Loss: 3.0190, Accuracy: 24.54% Time: 14.80s

Epoch 46/50: 100%|      | 782/782 [00:14<00:00, 52.75it/s, loss=3.0885,
acc=24.22%]

Epoch 46/50 - Loss: 3.0336, Accuracy: 24.22% Time: 14.83s

Epoch 47/50: 100%|      | 782/782 [00:14<00:00, 52.85it/s, loss=2.8746,
acc=24.40%]

Epoch 47/50 - Loss: 3.0227, Accuracy: 24.40% Time: 14.80s

Epoch 48/50: 100%|      | 782/782 [00:15<00:00, 51.73it/s, loss=2.9947,
acc=25.28%]

Epoch 48/50 - Loss: 2.9870, Accuracy: 25.28% Time: 15.12s

Epoch 49/50: 100%|      | 782/782 [00:15<00:00, 52.01it/s, loss=2.9308,
acc=25.64%]

Epoch 49/50 - Loss: 2.9726, Accuracy: 25.64% Time: 15.04s

Epoch 50/50: 100%|      | 782/782 [00:15<00:00, 51.97it/s, loss=3.1578,
acc=26.06%]

Epoch 50/50 - Loss: 2.9518, Accuracy: 26.06% Time: 15.05s
Average epoch training time: 14.91 seconds

Testing started…

Testing: 100%|      | 157/157 [00:02<00:00, 68.24it/s, accuracy=24.80%]

Final Test Loss: 3.0491, Final Test Accuracy: 24.80%

Visualizing results…
```



[5]:
```
#ViT from scratch - 8x8  patch size - 4 heads 4 layers 256  dim 512 mlp

# Delete model and optimizer variables
del model, optimizer, test_losses, test_accuracies, train_losses,␣
 ↪train_accuracies, epoch_times

# Clear CUDA cache if using GPU
gc.collect()
# Clear CUDA cache
if torch.cuda.is_available():
    torch.cuda.empty_cache()


# Initialize model
model = VisionTransformer(
    image_size=image_size,
    patch_size=patch_size,
    num_classes=num_classes,
    embed_dim=hidden_dim,
    num_heads=num_heads,
    num_layers=num_layers,
    mlp_dim=mlp_dim
).to(device)
# Loss and optimizer
criterion = nn.CrossEntropyLoss()
optimizer = torch.optim.AdamW(model.parameters(), lr=learning_rate,␣
 ↪weight_decay=0.01)


# Display model summary
summary(model,
```

```
        input_size=(batch_size, 3, image_size, image_size),
        col_names=["input_size", "output_size", "kernel_size", "num_params",␣
 ↪"mult_adds", "trainable"],
        col_width=20,
        depth=5,
        verbose=True,
        device=device)

# Hyperparameters
image_size = 32
patch_size = 8
num_classes = 100
num_epochs = 50
batch_size = 64
learning_rate = 0.001
num_heads = 4
num_layers = 4
hidden_dim = 256
mlp_dim = 512

# Run training and testing
if __name__ == '__main__':
    print("Training started...")
    train_losses, train_accuracies, epoch_times = train()
    print("\nTesting started...")
    test_losses, test_accuracies = test()

    # Visualize results
    print("\nVisualizing results...")
    visualize_results(train_losses, train_accuracies, test_losses,␣
 ↪test_accuracies)
```

```
==========================================================================
==========================================================================
=====
Layer (type:depth-idx)                        Input Shape          Output Shape
Kernel Shape         Param #              Mult-Adds            Trainable
==========================================================================
==========================================================================
=====
VisionTransformer                             [64, 3, 32, 32]      [64, 100]
--                   2,304                --                   True
 PatchEmbedding: 1-1                          [64, 3, 32, 32]      [64, 16, 128]
--                   --                   --                   True
      Conv2d: 2-1                             [64, 3, 32, 32]      [64, 128, 4,
4]       [8, 8]               24,704               25,296,896           True
 Dropout: 1-2                                 [64, 17, 128]        [64, 17, 128]
--                   --                   --                   --
```

```
 ModuleList: 1-3                          --                   --
--                   --                   --                   True
     TransformerEncoder: 2-2                [64, 17, 128]        [64, 17, 128]
--                   --                   --                   True
          LayerNorm: 3-1                    [64, 17, 128]        [64, 17, 128]
--                   256                  16,384               True
          MultiHeadSelfAttention: 3-2       [64, 17, 128]        [64, 17, 128]
--                   --                   --                   True
               Linear: 4-1                  [64, 17, 128]        [64, 17, 384]
--                   49,536               3,170,304            True
               Linear: 4-2                  [64, 17, 128]        [64, 17, 128]
--                   16,512               1,056,768            True
          LayerNorm: 3-3                    [64, 17, 128]        [64, 17, 128]
--                   256                  16,384               True
          MLP: 3-4                          [64, 17, 128]        [64, 17, 128]
--                   --                   --                   True
               Linear: 4-3                  [64, 17, 128]        [64, 17, 256]
--                   33,024               2,113,536            True
               GELU: 4-4                    [64, 17, 256]        [64, 17, 256]
--                   --                   --                   --
               Linear: 4-5                  [64, 17, 256]        [64, 17, 128]
--                   32,896               2,105,344            True
     TransformerEncoder: 2-3                [64, 17, 128]        [64, 17, 128]
--                   --                   --                   True
          LayerNorm: 3-5                    [64, 17, 128]        [64, 17, 128]
--                   256                  16,384               True
          MultiHeadSelfAttention: 3-6       [64, 17, 128]        [64, 17, 128]
--                   --                   --                   True
               Linear: 4-6                  [64, 17, 128]        [64, 17, 384]
--                   49,536               3,170,304            True
               Linear: 4-7                  [64, 17, 128]        [64, 17, 128]
--                   16,512               1,056,768            True
          LayerNorm: 3-7                    [64, 17, 128]        [64, 17, 128]
--                   256                  16,384               True
          MLP: 3-8                          [64, 17, 128]        [64, 17, 128]
--                   --                   --                   True
               Linear: 4-8                  [64, 17, 128]        [64, 17, 256]
--                   33,024               2,113,536            True
               GELU: 4-9                    [64, 17, 256]        [64, 17, 256]
--                   --                   --                   --
               Linear: 4-10                 [64, 17, 256]        [64, 17, 128]
--                   32,896               2,105,344            True
     TransformerEncoder: 2-4                [64, 17, 128]        [64, 17, 128]
--                   --                   --                   True
          LayerNorm: 3-9                    [64, 17, 128]        [64, 17, 128]
--                   256                  16,384               True
          MultiHeadSelfAttention: 3-10      [64, 17, 128]        [64, 17, 128]
--                   --                   --                   True
```

```
            Linear: 4-11                [64, 17, 128]       [64, 17, 384]
--                  49,536              3,170,304           True
            Linear: 4-12                [64, 17, 128]       [64, 17, 128]
--                  16,512              1,056,768           True
        LayerNorm: 3-11                 [64, 17, 128]       [64, 17, 128]
--                  256                 16,384              True
        MLP: 3-12                       [64, 17, 128]       [64, 17, 128]
--                  --                  --                  True
            Linear: 4-13                [64, 17, 128]       [64, 17, 256]
--                  33,024              2,113,536           True
            GELU: 4-14                  [64, 17, 256]       [64, 17, 256]
--                  --                  --                  --
            Linear: 4-15                [64, 17, 256]       [64, 17, 128]
--                  32,896              2,105,344           True
    TransformerEncoder: 2-5             [64, 17, 128]       [64, 17, 128]
--                  --                  --                  True
        LayerNorm: 3-13                 [64, 17, 128]       [64, 17, 128]
--                  256                 16,384              True
        MultiHeadSelfAttention: 3-14    [64, 17, 128]       [64, 17, 128]
--                  --                  --                  True
            Linear: 4-16                [64, 17, 128]       [64, 17, 384]
--                  49,536              3,170,304           True
            Linear: 4-17                [64, 17, 128]       [64, 17, 128]
--                  16,512              1,056,768           True
        LayerNorm: 3-15                 [64, 17, 128]       [64, 17, 128]
--                  256                 16,384              True
        MLP: 3-16                       [64, 17, 128]       [64, 17, 128]
--                  --                  --                  True
            Linear: 4-18                [64, 17, 128]       [64, 17, 256]
--                  33,024              2,113,536           True
            GELU: 4-19                  [64, 17, 256]       [64, 17, 256]
--                  --                  --                  --
            Linear: 4-20                [64, 17, 256]       [64, 17, 128]
--                  32,896              2,105,344           True
    TransformerEncoder: 2-6             [64, 17, 128]       [64, 17, 128]
--                  --                  --                  True
        LayerNorm: 3-17                 [64, 17, 128]       [64, 17, 128]
--                  256                 16,384              True
        MultiHeadSelfAttention: 3-18    [64, 17, 128]       [64, 17, 128]
--                  --                  --                  True
            Linear: 4-21                [64, 17, 128]       [64, 17, 384]
--                  49,536              3,170,304           True
            Linear: 4-22                [64, 17, 128]       [64, 17, 128]
--                  16,512              1,056,768           True
        LayerNorm: 3-19                 [64, 17, 128]       [64, 17, 128]
--                  256                 16,384              True
        MLP: 3-20                       [64, 17, 128]       [64, 17, 128]
--                  --                  --                  True
```

```
            Linear: 4-23                [64, 17, 128]       [64, 17, 256]
--                  33,024              2,113,536           True
            GELU: 4-24                  [64, 17, 256]       [64, 17, 256]
--                  --                  --                  --
            Linear: 4-25                [64, 17, 256]       [64, 17, 128]
--                  32,896              2,105,344           True
    TransformerEncoder: 2-7             [64, 17, 128]       [64, 17, 128]
--                  --                  --                  True
        LayerNorm: 3-21                 [64, 17, 128]       [64, 17, 128]
--                  256                 16,384              True
        MultiHeadSelfAttention: 3-22    [64, 17, 128]       [64, 17, 128]
--                  --                  --                  True
            Linear: 4-26                [64, 17, 128]       [64, 17, 384]
--                  49,536              3,170,304           True
            Linear: 4-27                [64, 17, 128]       [64, 17, 128]
--                  16,512              1,056,768           True
        LayerNorm: 3-23                 [64, 17, 128]       [64, 17, 128]
--                  256                 16,384              True
        MLP: 3-24                       [64, 17, 128]       [64, 17, 128]
--                  --                  --                  True
            Linear: 4-28                [64, 17, 128]       [64, 17, 256]
--                  33,024              2,113,536           True
            GELU: 4-29                  [64, 17, 256]       [64, 17, 256]
--                  --                  --                  --
            Linear: 4-30                [64, 17, 256]       [64, 17, 128]
--                  32,896              2,105,344           True
    TransformerEncoder: 2-8             [64, 17, 128]       [64, 17, 128]
--                  --                  --                  True
        LayerNorm: 3-25                 [64, 17, 128]       [64, 17, 128]
--                  256                 16,384              True
        MultiHeadSelfAttention: 3-26    [64, 17, 128]       [64, 17, 128]
--                  --                  --                  True
            Linear: 4-31                [64, 17, 128]       [64, 17, 384]
--                  49,536              3,170,304           True
            Linear: 4-32                [64, 17, 128]       [64, 17, 128]
--                  16,512              1,056,768           True
        LayerNorm: 3-27                 [64, 17, 128]       [64, 17, 128]
--                  256                 16,384              True
        MLP: 3-28                       [64, 17, 128]       [64, 17, 128]
--                  --                  --                  True
            Linear: 4-33                [64, 17, 128]       [64, 17, 256]
--                  33,024              2,113,536           True
            GELU: 4-34                  [64, 17, 256]       [64, 17, 256]
--                  --                  --                  --
            Linear: 4-35                [64, 17, 256]       [64, 17, 128]
--                  32,896              2,105,344           True
    TransformerEncoder: 2-9             [64, 17, 128]       [64, 17, 128]
--                  --                  --                  True
```

```
          LayerNorm: 3-29                   [64, 17, 128]        [64, 17, 128]
--                    256                   16,384               True
          MultiHeadSelfAttention: 3-30      [64, 17, 128]        [64, 17, 128]
--                    --                    --                   True
               Linear: 4-36                 [64, 17, 128]        [64, 17, 384]
--                    49,536                3,170,304            True
               Linear: 4-37                 [64, 17, 128]        [64, 17, 128]
--                    16,512                1,056,768            True
          LayerNorm: 3-31                   [64, 17, 128]        [64, 17, 128]
--                    256                   16,384               True
          MLP: 3-32                         [64, 17, 128]        [64, 17, 128]
--                    --                    --                   True
               Linear: 4-38                 [64, 17, 128]        [64, 17, 256]
--                    33,024                2,113,536            True
               GELU: 4-39                   [64, 17, 256]        [64, 17, 256]
--                    --                    --                   --
               Linear: 4-40                 [64, 17, 256]        [64, 17, 128]
--                    32,896                2,105,344            True
 LayerNorm: 1-4                               [64, 17, 128]        [64, 17, 128]
--                    256                   16,384               True
 Linear: 1-5                                  [64, 128]            [64, 100]
--                    12,900                825,600              True
==========================================================================
==========================================================================
=====
Total params: 1,100,004
Trainable params: 1,100,004
Non-trainable params: 0
Total mult-adds (Units.MEGABYTES): 93.97
==========================================================================
==========================================================================
=====
Input size (MB): 0.79
Forward/backward pass size (MB): 82.43
Params size (MB): 4.39
Estimated Total Size (MB): 87.61
==========================================================================
==========================================================================
=====
Training started…

Epoch 1/50:   1%|          | 5/782 [00:00<00:17, 43.56it/s, loss=4.8081,
acc=2.26%]

Input images shape: torch.Size([64, 3, 32, 32])
Labels shape: torch.Size([64])
Labels values: tensor([18, 69, 93, 44, 24, 60, 17, 87, 98, 84], device='cuda:0')
Model outputs shape: torch.Size([64, 100])
Expected outputs shape: torch.Size([64, 100])
```

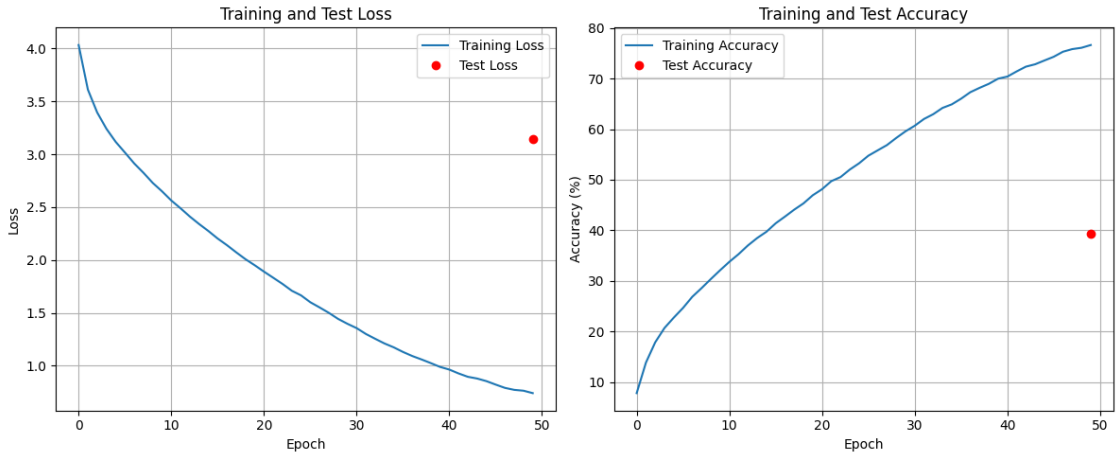
```
Epoch 1/50: 100%|      | 782/782 [00:15<00:00, 48.89it/s, loss=4.1248,
acc=7.80%]

Epoch 1/50 - Loss: 4.0322, Accuracy: 7.80% Time: 16.00s

Epoch 2/50: 100%|      | 782/782 [00:15<00:00, 49.14it/s, loss=3.8749,
acc=13.84%]

Epoch 2/50 - Loss: 3.6101, Accuracy: 13.84% Time: 15.92s

Epoch 3/50: 100%|      | 782/782 [00:15<00:00, 49.15it/s, loss=3.1416,
acc=17.85%]

Epoch 3/50 - Loss: 3.3948, Accuracy: 17.85% Time: 15.91s

Epoch 4/50: 100%|      | 782/782 [00:16<00:00, 48.52it/s, loss=3.2164,
acc=20.68%]

Epoch 4/50 - Loss: 3.2403, Accuracy: 20.68% Time: 16.12s

Epoch 5/50: 100%|      | 782/782 [00:16<00:00, 47.20it/s, loss=2.9037,
acc=22.73%]

Epoch 5/50 - Loss: 3.1165, Accuracy: 22.73% Time: 16.57s

Epoch 6/50: 100%|      | 782/782 [00:16<00:00, 47.13it/s, loss=3.4892,
acc=24.65%]

Epoch 6/50 - Loss: 3.0154, Accuracy: 24.65% Time: 16.59s

Epoch 7/50: 100%|      | 782/782 [00:16<00:00, 47.13it/s, loss=2.3596,
acc=26.85%]

Epoch 7/50 - Loss: 2.9124, Accuracy: 26.85% Time: 16.59s

Epoch 8/50: 100%|      | 782/782 [00:16<00:00, 47.18it/s, loss=3.4162,
acc=28.56%]

Epoch 8/50 - Loss: 2.8239, Accuracy: 28.56% Time: 16.58s

Epoch 9/50: 100%|      | 782/782 [00:16<00:00, 47.57it/s, loss=2.2830,
acc=30.36%]

Epoch 9/50 - Loss: 2.7282, Accuracy: 30.36% Time: 16.44s

Epoch 10/50: 100%|     | 782/782 [00:16<00:00, 47.34it/s, loss=2.5729,
acc=32.10%]

Epoch 10/50 - Loss: 2.6490, Accuracy: 32.10% Time: 16.52s

Epoch 11/50: 100%|     | 782/782 [00:16<00:00, 47.20it/s, loss=2.5805,
acc=33.77%]

Epoch 11/50 - Loss: 2.5622, Accuracy: 33.77% Time: 16.57s

Epoch 12/50: 100%|     | 782/782 [00:16<00:00, 47.30it/s, loss=2.7615,
acc=35.29%]

Epoch 12/50 - Loss: 2.4884, Accuracy: 35.29% Time: 16.53s
```

```
Epoch 13/50: 100%|     | 782/782 [00:16<00:00, 47.10it/s, loss=2.2964,
acc=37.00%]

Epoch 13/50 - Loss: 2.4105, Accuracy: 37.00% Time: 16.61s

Epoch 14/50: 100%|     | 782/782 [00:16<00:00, 47.25it/s, loss=2.3299,
acc=38.47%]

Epoch 14/50 - Loss: 2.3399, Accuracy: 38.47% Time: 16.55s

Epoch 15/50: 100%|     | 782/782 [00:16<00:00, 47.32it/s, loss=2.7738,
acc=39.71%]

Epoch 15/50 - Loss: 2.2739, Accuracy: 39.71% Time: 16.53s

Epoch 16/50: 100%|     | 782/782 [00:16<00:00, 47.16it/s, loss=2.3764,
acc=41.39%]

Epoch 16/50 - Loss: 2.2028, Accuracy: 41.39% Time: 16.58s

Epoch 17/50: 100%|     | 782/782 [00:16<00:00, 47.33it/s, loss=3.3342,
acc=42.71%]

Epoch 17/50 - Loss: 2.1404, Accuracy: 42.71% Time: 16.52s

Epoch 18/50: 100%|     | 782/782 [00:16<00:00, 47.19it/s, loss=2.5933,
acc=44.08%]

Epoch 18/50 - Loss: 2.0725, Accuracy: 44.08% Time: 16.57s

Epoch 19/50: 100%|     | 782/782 [00:16<00:00, 46.84it/s, loss=1.7346,
acc=45.32%]

Epoch 19/50 - Loss: 2.0076, Accuracy: 45.32% Time: 16.70s

Epoch 20/50: 100%|     | 782/782 [00:16<00:00, 47.24it/s, loss=2.3707,
acc=46.93%]

Epoch 20/50 - Loss: 1.9501, Accuracy: 46.93% Time: 16.55s

Epoch 21/50: 100%|     | 782/782 [00:16<00:00, 47.42it/s, loss=2.5458,
acc=48.12%]

Epoch 21/50 - Loss: 1.8898, Accuracy: 48.12% Time: 16.49s

Epoch 22/50: 100%|     | 782/782 [00:16<00:00, 47.43it/s, loss=1.2346,
acc=49.71%]

Epoch 22/50 - Loss: 1.8318, Accuracy: 49.71% Time: 16.49s

Epoch 23/50: 100%|     | 782/782 [00:16<00:00, 47.29it/s, loss=2.2335,
acc=50.53%]

Epoch 23/50 - Loss: 1.7736, Accuracy: 50.53% Time: 16.54s

Epoch 24/50: 100%|     | 782/782 [00:16<00:00, 47.29it/s, loss=1.7626,
acc=52.02%]

Epoch 24/50 - Loss: 1.7093, Accuracy: 52.02% Time: 16.54s
```

```
Epoch 25/50: 100%|      | 782/782 [00:16<00:00, 47.36it/s, loss=2.1752,
acc=53.25%]

Epoch 25/50 - Loss: 1.6653, Accuracy: 53.25% Time: 16.51s

Epoch 26/50: 100%|      | 782/782 [00:16<00:00, 47.18it/s, loss=1.9644,
acc=54.75%]

Epoch 26/50 - Loss: 1.6004, Accuracy: 54.75% Time: 16.58s

Epoch 27/50: 100%|      | 782/782 [00:16<00:00, 47.23it/s, loss=1.0469,
acc=55.80%]

Epoch 27/50 - Loss: 1.5513, Accuracy: 55.80% Time: 16.56s

Epoch 28/50: 100%|      | 782/782 [00:16<00:00, 47.22it/s, loss=1.7464,
acc=56.84%]

Epoch 28/50 - Loss: 1.5002, Accuracy: 56.84% Time: 16.56s

Epoch 29/50: 100%|      | 782/782 [00:16<00:00, 47.38it/s, loss=2.2837,
acc=58.25%]

Epoch 29/50 - Loss: 1.4428, Accuracy: 58.25% Time: 16.51s

Epoch 30/50: 100%|      | 782/782 [00:16<00:00, 47.32it/s, loss=1.1303,
acc=59.56%]

Epoch 30/50 - Loss: 1.3969, Accuracy: 59.56% Time: 16.53s

Epoch 31/50: 100%|      | 782/782 [00:16<00:00, 47.24it/s, loss=2.2466,
acc=60.67%]

Epoch 31/50 - Loss: 1.3555, Accuracy: 60.67% Time: 16.56s

Epoch 32/50: 100%|      | 782/782 [00:16<00:00, 47.79it/s, loss=1.1696,
acc=62.01%]

Epoch 32/50 - Loss: 1.3004, Accuracy: 62.01% Time: 16.37s

Epoch 33/50: 100%|      | 782/782 [00:16<00:00, 47.52it/s, loss=0.9782,
acc=62.96%]

Epoch 33/50 - Loss: 1.2548, Accuracy: 62.96% Time: 16.46s

Epoch 34/50: 100%|      | 782/782 [00:16<00:00, 47.59it/s, loss=1.6134,
acc=64.20%]

Epoch 34/50 - Loss: 1.2108, Accuracy: 64.20% Time: 16.43s

Epoch 35/50: 100%|      | 782/782 [00:16<00:00, 47.62it/s, loss=0.9294,
acc=64.90%]

Epoch 35/50 - Loss: 1.1740, Accuracy: 64.90% Time: 16.42s

Epoch 36/50: 100%|      | 782/782 [00:16<00:00, 47.49it/s, loss=1.3610,
acc=66.02%]

Epoch 36/50 - Loss: 1.1305, Accuracy: 66.02% Time: 16.47s
```

```
Epoch 37/50: 100%|      | 782/782 [00:16<00:00, 47.70it/s, loss=1.3148,
acc=67.31%]

Epoch 37/50 - Loss: 1.0921, Accuracy: 67.31% Time: 16.39s

Epoch 38/50: 100%|      | 782/782 [00:16<00:00, 47.55it/s, loss=1.5116,
acc=68.17%]

Epoch 38/50 - Loss: 1.0596, Accuracy: 68.17% Time: 16.45s

Epoch 39/50: 100%|      | 782/782 [00:16<00:00, 47.52it/s, loss=1.2805,
acc=68.96%]

Epoch 39/50 - Loss: 1.0250, Accuracy: 68.96% Time: 16.46s

Epoch 40/50: 100%|      | 782/782 [00:16<00:00, 47.53it/s, loss=1.5749,
acc=69.99%]

Epoch 40/50 - Loss: 0.9890, Accuracy: 69.99% Time: 16.45s

Epoch 41/50: 100%|      | 782/782 [00:16<00:00, 47.71it/s, loss=1.2840,
acc=70.40%]

Epoch 41/50 - Loss: 0.9639, Accuracy: 70.40% Time: 16.39s

Epoch 42/50: 100%|      | 782/782 [00:16<00:00, 47.80it/s, loss=0.9919,
acc=71.42%]

Epoch 42/50 - Loss: 0.9271, Accuracy: 71.42% Time: 16.36s

Epoch 43/50: 100%|      | 782/782 [00:16<00:00, 47.07it/s, loss=0.9794,
acc=72.36%]

Epoch 43/50 - Loss: 0.8956, Accuracy: 72.36% Time: 16.61s

Epoch 44/50: 100%|      | 782/782 [00:16<00:00, 47.35it/s, loss=1.1928,
acc=72.83%]

Epoch 44/50 - Loss: 0.8792, Accuracy: 72.83% Time: 16.51s

Epoch 45/50: 100%|      | 782/782 [00:16<00:00, 47.18it/s, loss=1.1855,
acc=73.58%]

Epoch 45/50 - Loss: 0.8546, Accuracy: 73.58% Time: 16.58s

Epoch 46/50: 100%|      | 782/782 [00:16<00:00, 47.38it/s, loss=0.7225,
acc=74.31%]

Epoch 46/50 - Loss: 0.8219, Accuracy: 74.31% Time: 16.51s

Epoch 47/50: 100%|      | 782/782 [00:16<00:00, 47.26it/s, loss=1.1033,
acc=75.31%]

Epoch 47/50 - Loss: 0.7908, Accuracy: 75.31% Time: 16.55s

Epoch 48/50: 100%|      | 782/782 [00:16<00:00, 47.27it/s, loss=1.5732,
acc=75.83%]

Epoch 48/50 - Loss: 0.7721, Accuracy: 75.83% Time: 16.54s
```

```
Epoch 49/50: 100%|      | 782/782 [00:16<00:00, 47.18it/s, loss=0.5318,
acc=76.09%]

Epoch 49/50 - Loss: 0.7641, Accuracy: 76.09% Time: 16.58s

Epoch 50/50: 100%|      | 782/782 [00:16<00:00, 47.69it/s, loss=0.7666,
acc=76.64%]

Epoch 50/50 - Loss: 0.7411, Accuracy: 76.64% Time: 16.40s
Average epoch training time: 16.47 seconds


Testing started…

Testing: 100%|      | 157/157 [00:02<00:00, 70.27it/s, accuracy=39.34%]

Final Test Loss: 3.1431, Final Test Accuracy: 39.34%


Visualizing results…
```



[6]:
```
#ViT from scratch - 4x4  patch size - 2 heads 8 layers 256 dim 512 mlp

# Delete model and optimizer variables
del model, optimizer, test_losses, test_accuracies, train_losses,␣
 ↪train_accuracies, epoch_times

# Clear CUDA cache if using GPU
gc.collect()
# Clear CUDA cache
if torch.cuda.is_available():
    torch.cuda.empty_cache()


# Initialize model
model = VisionTransformer(
```

```
    image_size=image_size,
    patch_size=patch_size,
    num_classes=num_classes,
    embed_dim=hidden_dim,
    num_heads=num_heads,
    num_layers=num_layers,
    mlp_dim=mlp_dim
).to(device)
# Loss and optimizer
criterion = nn.CrossEntropyLoss()
optimizer = torch.optim.AdamW(model.parameters(), lr=learning_rate,␣
 ↪weight_decay=0.01)

# Hyperparameters
image_size = 32
patch_size = 4
num_classes = 100
num_epochs = 50
batch_size = 64
learning_rate = 0.001
num_heads = 2
num_layers = 8
hidden_dim = 256
mlp_dim = 512

# Run training and testing
if __name__ == '__main__':
    print("Training started...")
    train_losses, train_accuracies, epoch_times = train()
    print("\nTesting started...")
    test_losses, test_accuracies = test()

    # Visualize results
    print("\nVisualizing results...")
    visualize_results(train_losses, train_accuracies, test_losses,␣
 ↪test_accuracies)
```

```
Training started…

Epoch 1/50:   1%|          | 6/782 [00:00<00:14, 53.85it/s, loss=4.6928,
acc=2.34%]

Input images shape: torch.Size([64, 3, 32, 32])
Labels shape: torch.Size([64])
Labels values: tensor([61, 61, 90, 25, 67, 77, 46, 32, 29, 60], device='cuda:0')
Model outputs shape: torch.Size([64, 100])
Expected outputs shape: torch.Size([64, 100])

Epoch 1/50: 100%|     | 782/782 [00:13<00:00, 57.04it/s, loss=3.8178,
```

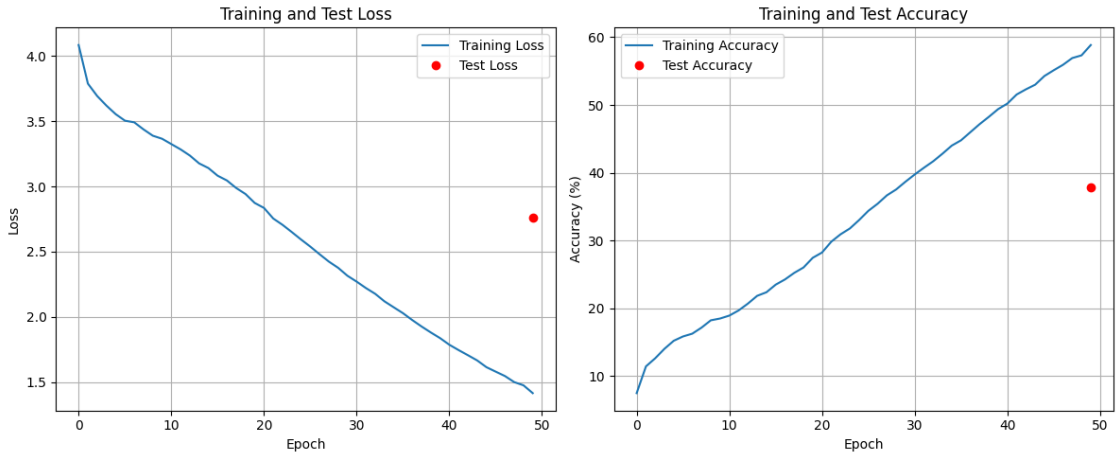
```
acc=7.49%]

Epoch 1/50 - Loss: 4.0851, Accuracy: 7.49% Time: 13.71s

Epoch 2/50: 100%|      | 782/782 [00:13<00:00, 57.55it/s, loss=4.4849,
acc=11.43%]

Epoch 2/50 - Loss: 3.7890, Accuracy: 11.43% Time: 13.59s

Epoch 3/50: 100%|      | 782/782 [00:13<00:00, 57.33it/s, loss=3.5032,
acc=12.63%]

Epoch 3/50 - Loss: 3.6941, Accuracy: 12.63% Time: 13.64s

Epoch 4/50: 100%|      | 782/782 [00:13<00:00, 57.30it/s, loss=2.8305,
acc=14.03%]

Epoch 4/50 - Loss: 3.6202, Accuracy: 14.03% Time: 13.65s

Epoch 5/50: 100%|      | 782/782 [00:13<00:00, 57.21it/s, loss=3.2250,
acc=15.22%]

Epoch 5/50 - Loss: 3.5547, Accuracy: 15.22% Time: 13.67s

Epoch 6/50: 100%|      | 782/782 [00:13<00:00, 57.07it/s, loss=3.8995,
acc=15.85%]

Epoch 6/50 - Loss: 3.5047, Accuracy: 15.85% Time: 13.70s

Epoch 7/50: 100%|      | 782/782 [00:13<00:00, 57.14it/s, loss=3.4422,
acc=16.26%]

Epoch 7/50 - Loss: 3.4922, Accuracy: 16.26% Time: 13.69s

Epoch 8/50: 100%|      | 782/782 [00:13<00:00, 57.23it/s, loss=3.3214,
acc=17.16%]

Epoch 8/50 - Loss: 3.4380, Accuracy: 17.16% Time: 13.67s

Epoch 9/50: 100%|      | 782/782 [00:13<00:00, 57.33it/s, loss=2.5587,
acc=18.23%]

Epoch 9/50 - Loss: 3.3896, Accuracy: 18.23% Time: 13.64s

Epoch 10/50: 100%|      | 782/782 [00:13<00:00, 57.07it/s, loss=2.8600,
acc=18.50%]

Epoch 10/50 - Loss: 3.3663, Accuracy: 18.50% Time: 13.70s

Epoch 11/50: 100%|      | 782/782 [00:13<00:00, 56.97it/s, loss=3.1966,
acc=18.93%]

Epoch 11/50 - Loss: 3.3254, Accuracy: 18.93% Time: 13.73s

Epoch 12/50: 100%|      | 782/782 [00:13<00:00, 57.03it/s, loss=3.7618,
acc=19.69%]

Epoch 12/50 - Loss: 3.2846, Accuracy: 19.69% Time: 13.71s
```

```
Epoch 13/50: 100%|      | 782/782 [00:13<00:00, 56.90it/s, loss=3.6476,
acc=20.70%]

Epoch 13/50 - Loss: 3.2374, Accuracy: 20.70% Time: 13.74s

Epoch 14/50: 100%|      | 782/782 [00:13<00:00, 57.18it/s, loss=3.4225,
acc=21.84%]

Epoch 14/50 - Loss: 3.1771, Accuracy: 21.84% Time: 13.68s

Epoch 15/50: 100%|      | 782/782 [00:13<00:00, 57.09it/s, loss=3.4059,
acc=22.37%]

Epoch 15/50 - Loss: 3.1416, Accuracy: 22.37% Time: 13.70s

Epoch 16/50: 100%|      | 782/782 [00:13<00:00, 57.20it/s, loss=3.0231,
acc=23.49%]

Epoch 16/50 - Loss: 3.0826, Accuracy: 23.49% Time: 13.67s

Epoch 17/50: 100%|      | 782/782 [00:13<00:00, 57.02it/s, loss=3.2997,
acc=24.26%]

Epoch 17/50 - Loss: 3.0461, Accuracy: 24.26% Time: 13.71s

Epoch 18/50: 100%|      | 782/782 [00:13<00:00, 57.15it/s, loss=2.8321,
acc=25.23%]

Epoch 18/50 - Loss: 2.9883, Accuracy: 25.23% Time: 13.68s

Epoch 19/50: 100%|      | 782/782 [00:13<00:00, 57.17it/s, loss=3.0569,
acc=26.02%]

Epoch 19/50 - Loss: 2.9425, Accuracy: 26.02% Time: 13.68s

Epoch 20/50: 100%|      | 782/782 [00:13<00:00, 57.31it/s, loss=2.9145,
acc=27.45%]

Epoch 20/50 - Loss: 2.8737, Accuracy: 27.45% Time: 13.65s

Epoch 21/50: 100%|      | 782/782 [00:13<00:00, 57.26it/s, loss=2.3358,
acc=28.23%]

Epoch 21/50 - Loss: 2.8356, Accuracy: 28.23% Time: 13.66s

Epoch 22/50: 100%|      | 782/782 [00:13<00:00, 57.04it/s, loss=2.0995,
acc=29.83%]

Epoch 22/50 - Loss: 2.7543, Accuracy: 29.83% Time: 13.71s

Epoch 23/50: 100%|      | 782/782 [00:13<00:00, 57.53it/s, loss=3.0954,
acc=30.92%]

Epoch 23/50 - Loss: 2.7059, Accuracy: 30.92% Time: 13.59s

Epoch 24/50: 100%|      | 782/782 [00:13<00:00, 57.82it/s, loss=2.6762,
acc=31.78%]

Epoch 24/50 - Loss: 2.6505, Accuracy: 31.78% Time: 13.53s
```

```
Epoch 25/50: 100%|     | 782/782 [00:13<00:00, 57.56it/s, loss=2.8407,
acc=33.02%]

Epoch 25/50 - Loss: 2.5933, Accuracy: 33.02% Time: 13.59s

Epoch 26/50: 100%|     | 782/782 [00:13<00:00, 57.17it/s, loss=2.5937,
acc=34.38%]

Epoch 26/50 - Loss: 2.5388, Accuracy: 34.38% Time: 13.68s

Epoch 27/50: 100%|     | 782/782 [00:13<00:00, 57.43it/s, loss=3.0914,
acc=35.43%]

Epoch 27/50 - Loss: 2.4803, Accuracy: 35.43% Time: 13.62s

Epoch 28/50: 100%|     | 782/782 [00:13<00:00, 57.47it/s, loss=2.3186,
acc=36.67%]

Epoch 28/50 - Loss: 2.4238, Accuracy: 36.67% Time: 13.61s

Epoch 29/50: 100%|     | 782/782 [00:13<00:00, 57.71it/s, loss=2.2006,
acc=37.54%]

Epoch 29/50 - Loss: 2.3763, Accuracy: 37.54% Time: 13.55s

Epoch 30/50: 100%|     | 782/782 [00:13<00:00, 57.46it/s, loss=2.6054,
acc=38.68%]

Epoch 30/50 - Loss: 2.3155, Accuracy: 38.68% Time: 13.61s

Epoch 31/50: 100%|     | 782/782 [00:13<00:00, 57.66it/s, loss=2.1911,
acc=39.75%]

Epoch 31/50 - Loss: 2.2702, Accuracy: 39.75% Time: 13.56s

Epoch 32/50: 100%|     | 782/782 [00:13<00:00, 56.82it/s, loss=1.3694,
acc=40.75%]

Epoch 32/50 - Loss: 2.2209, Accuracy: 40.75% Time: 13.76s

Epoch 33/50: 100%|     | 782/782 [00:13<00:00, 56.74it/s, loss=2.0665,
acc=41.70%]

Epoch 33/50 - Loss: 2.1767, Accuracy: 41.70% Time: 13.78s

Epoch 34/50: 100%|     | 782/782 [00:13<00:00, 57.02it/s, loss=2.3963,
acc=42.82%]

Epoch 34/50 - Loss: 2.1190, Accuracy: 42.82% Time: 13.71s

Epoch 35/50: 100%|     | 782/782 [00:13<00:00, 56.25it/s, loss=2.2794,
acc=44.01%]

Epoch 35/50 - Loss: 2.0733, Accuracy: 44.01% Time: 13.90s

Epoch 36/50: 100%|     | 782/782 [00:13<00:00, 57.58it/s, loss=2.1592,
acc=44.79%]

Epoch 36/50 - Loss: 2.0281, Accuracy: 44.79% Time: 13.58s
```

```
Epoch 37/50: 100%|      | 782/782 [00:14<00:00, 55.40it/s, loss=1.9899,
acc=45.98%]

Epoch 37/50 - Loss: 1.9754, Accuracy: 45.98% Time: 14.12s

Epoch 38/50: 100%|      | 782/782 [00:13<00:00, 57.55it/s, loss=1.9522,
acc=47.17%]

Epoch 38/50 - Loss: 1.9262, Accuracy: 47.17% Time: 13.59s

Epoch 39/50: 100%|      | 782/782 [00:13<00:00, 58.24it/s, loss=2.2314,
acc=48.25%]

Epoch 39/50 - Loss: 1.8803, Accuracy: 48.25% Time: 13.43s

Epoch 40/50: 100%|      | 782/782 [00:13<00:00, 57.96it/s, loss=2.0673,
acc=49.39%]

Epoch 40/50 - Loss: 1.8369, Accuracy: 49.39% Time: 13.49s

Epoch 41/50: 100%|      | 782/782 [00:13<00:00, 58.52it/s, loss=1.4846,
acc=50.21%]

Epoch 41/50 - Loss: 1.7861, Accuracy: 50.21% Time: 13.36s

Epoch 42/50: 100%|      | 782/782 [00:13<00:00, 58.24it/s, loss=2.2068,
acc=51.55%]

Epoch 42/50 - Loss: 1.7450, Accuracy: 51.55% Time: 13.43s

Epoch 43/50: 100%|      | 782/782 [00:13<00:00, 57.77it/s, loss=1.0602,
acc=52.31%]

Epoch 43/50 - Loss: 1.7060, Accuracy: 52.31% Time: 13.54s

Epoch 44/50: 100%|      | 782/782 [00:13<00:00, 56.94it/s, loss=1.8950,
acc=52.99%]

Epoch 44/50 - Loss: 1.6659, Accuracy: 52.99% Time: 13.73s

Epoch 45/50: 100%|      | 782/782 [00:13<00:00, 57.98it/s, loss=1.2905,
acc=54.28%]

Epoch 45/50 - Loss: 1.6141, Accuracy: 54.28% Time: 13.49s

Epoch 46/50: 100%|      | 782/782 [00:13<00:00, 57.55it/s, loss=2.2394,
acc=55.12%]

Epoch 46/50 - Loss: 1.5798, Accuracy: 55.12% Time: 13.59s

Epoch 47/50: 100%|      | 782/782 [00:13<00:00, 57.11it/s, loss=2.2903,
acc=55.93%]

Epoch 47/50 - Loss: 1.5460, Accuracy: 55.93% Time: 13.69s

Epoch 48/50: 100%|      | 782/782 [00:13<00:00, 59.03it/s, loss=1.5054,
acc=56.92%]

Epoch 48/50 - Loss: 1.4995, Accuracy: 56.92% Time: 13.25s
```

```
Epoch 49/50: 100%|      | 782/782 [00:13<00:00, 58.98it/s, loss=1.6483,
acc=57.32%]

Epoch 49/50 - Loss: 1.4749, Accuracy: 57.32% Time: 13.26s

Epoch 50/50: 100%|      | 782/782 [00:13<00:00, 58.91it/s, loss=0.8294,
acc=58.85%]

Epoch 50/50 - Loss: 1.4148, Accuracy: 58.85% Time: 13.28s
Average epoch training time: 13.63 seconds


Testing started…

Testing: 100%|      | 157/157 [00:02<00:00, 73.21it/s, accuracy=37.89%]

Final Test Loss: 2.7635, Final Test Accuracy: 37.89%


Visualizing results…
```



[7]:
```python
#ViT from scratch - 4x4  patch size - 4 heads 8 layers 256 dim 512 mlp

# Delete model and optimizer variables
del model, optimizer, test_losses, test_accuracies, train_losses,␣
 ↪train_accuracies, epoch_times

# Clear CUDA cache if using GPU
gc.collect()
# Clear CUDA cache
if torch.cuda.is_available():
    torch.cuda.empty_cache()


# Initialize model
model = VisionTransformer(
```

```
    image_size=image_size,
    patch_size=patch_size,
    num_classes=num_classes,
    embed_dim=hidden_dim,
    num_heads=num_heads,
    num_layers=num_layers,
    mlp_dim=mlp_dim
).to(device)

# Loss and optimizer
criterion = nn.CrossEntropyLoss()
optimizer = torch.optim.AdamW(model.parameters(), lr=learning_rate,␣
 ↪weight_decay=0.01)

# Display model summary
summary(model,
        input_size=(batch_size, 3, image_size, image_size),
        col_names=["input_size", "output_size", "kernel_size", "num_params",␣
 ↪"mult_adds", "trainable"],
        col_width=20,
        depth=5,
        verbose=True,
        device=device)

# Hyperparameters
image_size = 32
patch_size = 4
num_classes = 100
num_epochs = 50
batch_size = 64
learning_rate = 0.001
num_heads = 4
num_layers = 8
hidden_dim = 256
mlp_dim = 512

# Run training and testing
if __name__ == '__main__':
    print("Training started...")
    train_losses, train_accuracies, epoch_times = train()
    print("\nTesting started...")
    test_losses, test_accuracies = test()

    # Visualize results
    print("\nVisualizing results...")
    visualize_results(train_losses, train_accuracies, test_losses,␣
 ↪test_accuracies)
```

```
==========================================================================================
==========================================================================================
=====
Layer (type:depth-idx)                        Input Shape          Output Shape
Kernel Shape         Param #              Mult-Adds            Trainable
==========================================================================================
==========================================================================================
=====
VisionTransformer                             [64, 3, 32, 32]      [64, 100]
--                   16,896               --                   True
 PatchEmbedding: 1-1                            [64, 3, 32, 32]      [64, 64, 256]
--                   --                   --                   True
     Conv2d: 2-1                               [64, 3, 32, 32]      [64, 256, 8,
8]      [4, 4]               12,544               51,380,224           True
 Dropout: 1-2                                   [64, 65, 256]        [64, 65, 256]
--                   --                   --                   --
 ModuleList: 1-3                                --                   --
--                   --                   --                   True
     TransformerEncoder: 2-2                   [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
          LayerNorm: 3-1                     [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
          MultiHeadSelfAttention: 3-2        [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
               Linear: 4-1                   [64, 65, 256]        [64, 65, 768]
--                   197,376              12,632,064           True
               Linear: 4-2                   [64, 65, 256]        [64, 65, 256]
--                   65,792               4,210,688            True
          LayerNorm: 3-3                     [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
          MLP: 3-4                           [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
               Linear: 4-3                   [64, 65, 256]        [64, 65, 512]
--                   131,584              8,421,376            True
               GELU: 4-4                     [64, 65, 512]        [64, 65, 512]
--                   --                   --                   --
               Linear: 4-5                   [64, 65, 512]        [64, 65, 256]
--                   131,328              8,404,992            True
     TransformerEncoder: 2-3                   [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
          LayerNorm: 3-5                     [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
          MultiHeadSelfAttention: 3-6        [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
               Linear: 4-6                   [64, 65, 256]        [64, 65, 768]
--                   197,376              12,632,064           True
               Linear: 4-7                   [64, 65, 256]        [64, 65, 256]
--                   65,792               4,210,688            True
```

```
          LayerNorm: 3-7                  [64, 65, 256]       [64, 65, 256]
--                    512                 32,768              True
          MLP: 3-8                        [64, 65, 256]       [64, 65, 256]
--                    --                  --                  True
               Linear: 4-8                [64, 65, 256]       [64, 65, 512]
--                    131,584             8,421,376           True
               GELU: 4-9                  [64, 65, 512]       [64, 65, 512]
--                    --                  --                  --
               Linear: 4-10               [64, 65, 512]       [64, 65, 256]
--                    131,328             8,404,992           True
     TransformerEncoder: 2-4              [64, 65, 256]       [64, 65, 256]
--                    --                  --                  True
          LayerNorm: 3-9                  [64, 65, 256]       [64, 65, 256]
--                    512                 32,768              True
          MultiHeadSelfAttention: 3-10    [64, 65, 256]       [64, 65, 256]
--                    --                  --                  True
               Linear: 4-11               [64, 65, 256]       [64, 65, 768]
--                    197,376             12,632,064          True
               Linear: 4-12               [64, 65, 256]       [64, 65, 256]
--                    65,792              4,210,688           True
          LayerNorm: 3-11                 [64, 65, 256]       [64, 65, 256]
--                    512                 32,768              True
          MLP: 3-12                       [64, 65, 256]       [64, 65, 256]
--                    --                  --                  True
               Linear: 4-13               [64, 65, 256]       [64, 65, 512]
--                    131,584             8,421,376           True
               GELU: 4-14                 [64, 65, 512]       [64, 65, 512]
--                    --                  --                  --
               Linear: 4-15               [64, 65, 512]       [64, 65, 256]
--                    131,328             8,404,992           True
     TransformerEncoder: 2-5              [64, 65, 256]       [64, 65, 256]
--                    --                  --                  True
          LayerNorm: 3-13                 [64, 65, 256]       [64, 65, 256]
--                    512                 32,768              True
          MultiHeadSelfAttention: 3-14    [64, 65, 256]       [64, 65, 256]
--                    --                  --                  True
               Linear: 4-16               [64, 65, 256]       [64, 65, 768]
--                    197,376             12,632,064          True
               Linear: 4-17               [64, 65, 256]       [64, 65, 256]
--                    65,792              4,210,688           True
          LayerNorm: 3-15                 [64, 65, 256]       [64, 65, 256]
--                    512                 32,768              True
          MLP: 3-16                       [64, 65, 256]       [64, 65, 256]
--                    --                  --                  True
               Linear: 4-18               [64, 65, 256]       [64, 65, 512]
--                    131,584             8,421,376           True
               GELU: 4-19                 [64, 65, 512]       [64, 65, 512]
--                    --                  --                  --
```

```
            Linear: 4-20                [64, 65, 512]       [64, 65, 256]
--                  131,328             8,404,992           True
    TransformerEncoder: 2-6             [64, 65, 256]       [64, 65, 256]
--                  --                  --                  True
        LayerNorm: 3-17                 [64, 65, 256]       [64, 65, 256]
--                  512                 32,768              True
        MultiHeadSelfAttention: 3-18    [64, 65, 256]       [64, 65, 256]
--                  --                  --                  True
            Linear: 4-21                [64, 65, 256]       [64, 65, 768]
--                  197,376             12,632,064          True
            Linear: 4-22                [64, 65, 256]       [64, 65, 256]
--                  65,792              4,210,688           True
        LayerNorm: 3-19                 [64, 65, 256]       [64, 65, 256]
--                  512                 32,768              True
        MLP: 3-20                       [64, 65, 256]       [64, 65, 256]
--                  --                  --                  True
            Linear: 4-23                [64, 65, 256]       [64, 65, 512]
--                  131,584             8,421,376           True
            GELU: 4-24                  [64, 65, 512]       [64, 65, 512]
--                  --                  --                  --
            Linear: 4-25                [64, 65, 512]       [64, 65, 256]
--                  131,328             8,404,992           True
    TransformerEncoder: 2-7             [64, 65, 256]       [64, 65, 256]
--                  --                  --                  True
        LayerNorm: 3-21                 [64, 65, 256]       [64, 65, 256]
--                  512                 32,768              True
        MultiHeadSelfAttention: 3-22    [64, 65, 256]       [64, 65, 256]
--                  --                  --                  True
            Linear: 4-26                [64, 65, 256]       [64, 65, 768]
--                  197,376             12,632,064          True
            Linear: 4-27                [64, 65, 256]       [64, 65, 256]
--                  65,792              4,210,688           True
        LayerNorm: 3-23                 [64, 65, 256]       [64, 65, 256]
--                  512                 32,768              True
        MLP: 3-24                       [64, 65, 256]       [64, 65, 256]
--                  --                  --                  True
            Linear: 4-28                [64, 65, 256]       [64, 65, 512]
--                  131,584             8,421,376           True
            GELU: 4-29                  [64, 65, 512]       [64, 65, 512]
--                  --                  --                  --
            Linear: 4-30                [64, 65, 512]       [64, 65, 256]
--                  131,328             8,404,992           True
    TransformerEncoder: 2-8             [64, 65, 256]       [64, 65, 256]
--                  --                  --                  True
        LayerNorm: 3-25                 [64, 65, 256]       [64, 65, 256]
--                  512                 32,768              True
        MultiHeadSelfAttention: 3-26    [64, 65, 256]       [64, 65, 256]
--                  --                  --                  True
```

```
                Linear: 4-31                  [64, 65, 256]        [64, 65, 768]
--                    197,376                 12,632,064           True
                Linear: 4-32                  [64, 65, 256]        [64, 65, 256]
--                    65,792                  4,210,688            True
          LayerNorm: 3-27                       [64, 65, 256]        [64, 65, 256]
--                    512                     32,768               True
          MLP: 3-28                             [64, 65, 256]        [64, 65, 256]
--                    --                      --                   True
                Linear: 4-33                  [64, 65, 256]        [64, 65, 512]
--                    131,584                 8,421,376            True
                GELU: 4-34                    [64, 65, 512]        [64, 65, 512]
--                    --                      --                   --
                Linear: 4-35                  [64, 65, 512]        [64, 65, 256]
--                    131,328                 8,404,992            True
     TransformerEncoder: 2-9                    [64, 65, 256]        [64, 65, 256]
--                    --                      --                   True
          LayerNorm: 3-29                       [64, 65, 256]        [64, 65, 256]
--                    512                     32,768               True
          MultiHeadSelfAttention: 3-30          [64, 65, 256]        [64, 65, 256]
--                    --                      --                   True
                Linear: 4-36                  [64, 65, 256]        [64, 65, 768]
--                    197,376                 12,632,064           True
                Linear: 4-37                  [64, 65, 256]        [64, 65, 256]
--                    65,792                  4,210,688            True
          LayerNorm: 3-31                       [64, 65, 256]        [64, 65, 256]
--                    512                     32,768               True
          MLP: 3-32                             [64, 65, 256]        [64, 65, 256]
--                    --                      --                   True
                Linear: 4-38                  [64, 65, 256]        [64, 65, 512]
--                    131,584                 8,421,376            True
                GELU: 4-39                    [64, 65, 512]        [64, 65, 512]
--                    --                      --                   --
                Linear: 4-40                  [64, 65, 512]        [64, 65, 256]
--                    131,328                 8,404,992            True
 LayerNorm: 1-4                                 [64, 65, 256]        [64, 65, 256]
--                    512                     32,768               True
 Linear: 1-5                                    [64, 256]            [64, 100]
--                    25,700                  1,644,800            True
==========================================================================
==========================================================================
=====
Total params: 4,272,484
Trainable params: 4,272,484
Non-trainable params: 0
Total mult-adds (Units.MEGABYTES): 322.94
==========================================================================
==========================================================================
=====
```

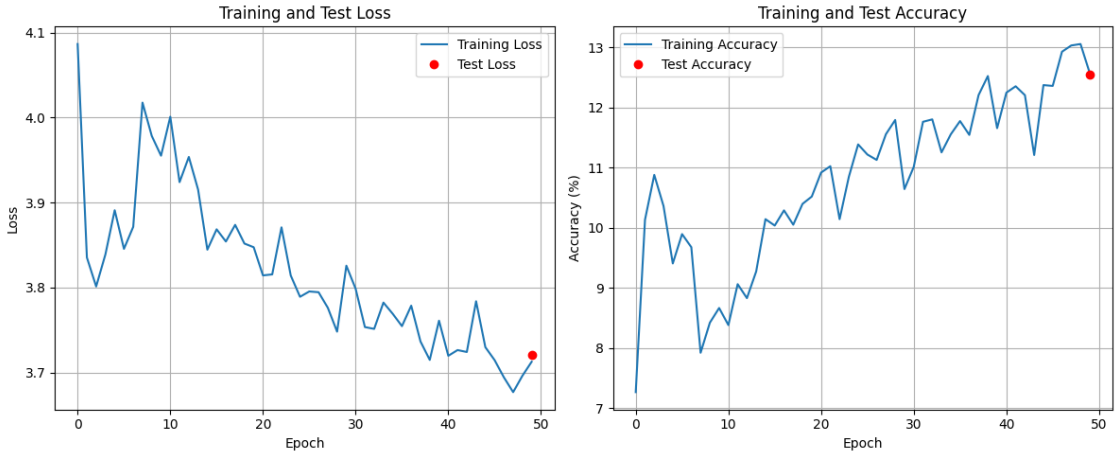
```
Input size (MB): 0.79
Forward/backward pass size (MB): 630.38
Params size (MB): 17.02
Estimated Total Size (MB): 648.19
==========================================================================
==========================================================================
=====
Training started…

Epoch 1/50:   1%|          | 4/782 [00:00<00:22, 34.98it/s, loss=5.1839,
acc=0.89%]

Input images shape: torch.Size([64, 3, 32, 32])
Labels shape: torch.Size([64])
Labels values: tensor([89,  3,  0, 12, 67, 91, 42,  2, 71, 61], device='cuda:0')
Model outputs shape: torch.Size([64, 100])
Expected outputs shape: torch.Size([64, 100])

Epoch 1/50: 100%|     | 782/782 [00:19<00:00, 39.19it/s, loss=3.7347,
acc=7.27%]

Epoch 1/50 - Loss: 4.0865, Accuracy: 7.27% Time: 19.95s

Epoch 2/50: 100%|     | 782/782 [00:19<00:00, 39.95it/s, loss=3.7498,
acc=10.13%]

Epoch 2/50 - Loss: 3.8354, Accuracy: 10.13% Time: 19.57s

Epoch 3/50: 100%|     | 782/782 [00:19<00:00, 39.75it/s, loss=4.1773,
acc=10.88%]

Epoch 3/50 - Loss: 3.8013, Accuracy: 10.88% Time: 19.67s

Epoch 4/50: 100%|     | 782/782 [00:19<00:00, 39.84it/s, loss=3.8591,
acc=10.36%]

Epoch 4/50 - Loss: 3.8391, Accuracy: 10.36% Time: 19.63s

Epoch 5/50: 100%|     | 782/782 [00:19<00:00, 39.90it/s, loss=3.5480,
acc=9.41%]

Epoch 5/50 - Loss: 3.8910, Accuracy: 9.41% Time: 19.60s

Epoch 6/50: 100%|     | 782/782 [00:19<00:00, 39.76it/s, loss=4.1262,
acc=9.89%]

Epoch 6/50 - Loss: 3.8455, Accuracy: 9.89% Time: 19.67s

Epoch 7/50: 100%|     | 782/782 [00:19<00:00, 39.72it/s, loss=4.0893,
acc=9.68%]

Epoch 7/50 - Loss: 3.8714, Accuracy: 9.68% Time: 19.69s

Epoch 8/50: 100%|     | 782/782 [00:19<00:00, 39.81it/s, loss=3.8079,
acc=7.92%]

Epoch 8/50 - Loss: 4.0176, Accuracy: 7.92% Time: 19.65s
```

```
Epoch 9/50: 100%|      | 782/782 [00:19<00:00, 39.80it/s, loss=4.3709,
acc=8.42%]

Epoch 9/50 - Loss: 3.9784, Accuracy: 8.42% Time: 19.65s

Epoch 10/50: 100%|      | 782/782 [00:19<00:00, 39.91it/s, loss=3.7649,
acc=8.67%]

Epoch 10/50 - Loss: 3.9551, Accuracy: 8.67% Time: 19.60s

Epoch 11/50: 100%|      | 782/782 [00:19<00:00, 39.80it/s, loss=4.1625,
acc=8.38%]

Epoch 11/50 - Loss: 4.0010, Accuracy: 8.38% Time: 19.65s

Epoch 12/50: 100%|      | 782/782 [00:19<00:00, 39.75it/s, loss=3.6972,
acc=9.06%]

Epoch 12/50 - Loss: 3.9240, Accuracy: 9.06% Time: 19.68s

Epoch 13/50: 100%|      | 782/782 [00:19<00:00, 39.88it/s, loss=4.1965,
acc=8.83%]

Epoch 13/50 - Loss: 3.9537, Accuracy: 8.83% Time: 19.61s

Epoch 14/50: 100%|      | 782/782 [00:19<00:00, 39.85it/s, loss=3.5563,
acc=9.27%]

Epoch 14/50 - Loss: 3.9152, Accuracy: 9.27% Time: 19.63s

Epoch 15/50: 100%|      | 782/782 [00:19<00:00, 39.77it/s, loss=4.1377,
acc=10.14%]

Epoch 15/50 - Loss: 3.8446, Accuracy: 10.14% Time: 19.66s

Epoch 16/50: 100%|      | 782/782 [00:19<00:00, 39.72it/s, loss=3.6213,
acc=10.04%]

Epoch 16/50 - Loss: 3.8685, Accuracy: 10.04% Time: 19.69s

Epoch 17/50: 100%|      | 782/782 [00:19<00:00, 39.87it/s, loss=4.2073,
acc=10.29%]

Epoch 17/50 - Loss: 3.8542, Accuracy: 10.29% Time: 19.61s

Epoch 18/50: 100%|      | 782/782 [00:19<00:00, 39.50it/s, loss=4.0906,
acc=10.05%]

Epoch 18/50 - Loss: 3.8738, Accuracy: 10.05% Time: 19.80s

Epoch 19/50: 100%|      | 782/782 [00:20<00:00, 37.63it/s, loss=4.2411,
acc=10.40%]

Epoch 19/50 - Loss: 3.8518, Accuracy: 10.40% Time: 20.78s

Epoch 20/50: 100%|      | 782/782 [00:20<00:00, 37.98it/s, loss=4.0969,
acc=10.52%]

Epoch 20/50 - Loss: 3.8475, Accuracy: 10.52% Time: 20.59s
```

```
Epoch 21/50: 100%|      | 782/782 [00:20<00:00, 37.98it/s, loss=3.7125,
acc=10.92%]

Epoch 21/50 - Loss: 3.8143, Accuracy: 10.92% Time: 20.59s

Epoch 22/50: 100%|      | 782/782 [00:20<00:00, 38.12it/s, loss=3.8390,
acc=11.02%]

Epoch 22/50 - Loss: 3.8155, Accuracy: 11.02% Time: 20.51s

Epoch 23/50: 100%|      | 782/782 [00:20<00:00, 38.08it/s, loss=4.1676,
acc=10.14%]

Epoch 23/50 - Loss: 3.8709, Accuracy: 10.14% Time: 20.54s

Epoch 24/50: 100%|      | 782/782 [00:20<00:00, 37.97it/s, loss=3.5102,
acc=10.85%]

Epoch 24/50 - Loss: 3.8141, Accuracy: 10.85% Time: 20.59s

Epoch 25/50: 100%|      | 782/782 [00:20<00:00, 38.00it/s, loss=3.6942,
acc=11.38%]

Epoch 25/50 - Loss: 3.7892, Accuracy: 11.38% Time: 20.58s

Epoch 26/50: 100%|      | 782/782 [00:20<00:00, 38.09it/s, loss=3.1524,
acc=11.22%]

Epoch 26/50 - Loss: 3.7954, Accuracy: 11.22% Time: 20.53s

Epoch 27/50: 100%|      | 782/782 [00:20<00:00, 38.20it/s, loss=3.4301,
acc=11.13%]

Epoch 27/50 - Loss: 3.7945, Accuracy: 11.13% Time: 20.47s

Epoch 28/50: 100%|      | 782/782 [00:20<00:00, 39.03it/s, loss=4.1683,
acc=11.55%]

Epoch 28/50 - Loss: 3.7762, Accuracy: 11.55% Time: 20.04s

Epoch 29/50: 100%|      | 782/782 [00:20<00:00, 38.45it/s, loss=3.5502,
acc=11.79%]

Epoch 29/50 - Loss: 3.7481, Accuracy: 11.79% Time: 20.34s

Epoch 30/50: 100%|      | 782/782 [00:20<00:00, 38.15it/s, loss=4.4795,
acc=10.64%]

Epoch 30/50 - Loss: 3.8257, Accuracy: 10.64% Time: 20.50s

Epoch 31/50: 100%|      | 782/782 [00:20<00:00, 38.09it/s, loss=3.5763,
acc=11.01%]

Epoch 31/50 - Loss: 3.7986, Accuracy: 11.01% Time: 20.53s

Epoch 32/50: 100%|      | 782/782 [00:20<00:00, 38.20it/s, loss=3.0964,
acc=11.76%]

Epoch 32/50 - Loss: 3.7535, Accuracy: 11.76% Time: 20.47s
```

```
Epoch 33/50: 100%|     | 782/782 [00:20<00:00, 38.09it/s, loss=3.7714,
acc=11.80%]

Epoch 33/50 - Loss: 3.7514, Accuracy: 11.80% Time: 20.53s

Epoch 34/50: 100%|     | 782/782 [00:20<00:00, 38.24it/s, loss=3.9451,
acc=11.25%]

Epoch 34/50 - Loss: 3.7823, Accuracy: 11.25% Time: 20.45s

Epoch 35/50: 100%|     | 782/782 [00:20<00:00, 37.85it/s, loss=3.7362,
acc=11.55%]

Epoch 35/50 - Loss: 3.7692, Accuracy: 11.55% Time: 20.66s

Epoch 36/50: 100%|     | 782/782 [00:20<00:00, 38.19it/s, loss=3.8548,
acc=11.77%]

Epoch 36/50 - Loss: 3.7546, Accuracy: 11.77% Time: 20.48s

Epoch 37/50: 100%|     | 782/782 [00:20<00:00, 38.26it/s, loss=4.0510,
acc=11.54%]

Epoch 37/50 - Loss: 3.7787, Accuracy: 11.54% Time: 20.44s

Epoch 38/50: 100%|     | 782/782 [00:20<00:00, 38.12it/s, loss=3.6517,
acc=12.20%]

Epoch 38/50 - Loss: 3.7366, Accuracy: 12.20% Time: 20.52s

Epoch 39/50: 100%|     | 782/782 [00:20<00:00, 38.23it/s, loss=3.4971,
acc=12.52%]

Epoch 39/50 - Loss: 3.7149, Accuracy: 12.52% Time: 20.46s

Epoch 40/50: 100%|     | 782/782 [00:20<00:00, 38.22it/s, loss=4.0286,
acc=11.65%]

Epoch 40/50 - Loss: 3.7610, Accuracy: 11.65% Time: 20.46s

Epoch 41/50: 100%|     | 782/782 [00:20<00:00, 38.09it/s, loss=3.8477,
acc=12.24%]

Epoch 41/50 - Loss: 3.7197, Accuracy: 12.24% Time: 20.53s

Epoch 42/50: 100%|     | 782/782 [00:20<00:00, 38.14it/s, loss=4.2106,
acc=12.35%]

Epoch 42/50 - Loss: 3.7265, Accuracy: 12.35% Time: 20.51s

Epoch 43/50: 100%|     | 782/782 [00:20<00:00, 38.21it/s, loss=3.0471,
acc=12.20%]

Epoch 43/50 - Loss: 3.7242, Accuracy: 12.20% Time: 20.46s

Epoch 44/50: 100%|     | 782/782 [00:19<00:00, 39.34it/s, loss=4.2223,
acc=11.21%]

Epoch 44/50 - Loss: 3.7839, Accuracy: 11.21% Time: 19.88s
```

```
Epoch 45/50: 100%|      | 782/782 [00:19<00:00, 39.34it/s, loss=3.3281,
acc=12.37%]

Epoch 45/50 - Loss: 3.7299, Accuracy: 12.37% Time: 19.88s

Epoch 46/50: 100%|      | 782/782 [00:19<00:00, 39.30it/s, loss=2.9769,
acc=12.36%]

Epoch 46/50 - Loss: 3.7146, Accuracy: 12.36% Time: 19.90s

Epoch 47/50: 100%|      | 782/782 [00:19<00:00, 39.35it/s, loss=4.4285,
acc=12.92%]

Epoch 47/50 - Loss: 3.6944, Accuracy: 12.92% Time: 19.87s

Epoch 48/50: 100%|      | 782/782 [00:19<00:00, 39.40it/s, loss=3.3062,
acc=13.03%]

Epoch 48/50 - Loss: 3.6770, Accuracy: 13.03% Time: 19.85s

Epoch 49/50: 100%|      | 782/782 [00:19<00:00, 39.25it/s, loss=3.2337,
acc=13.05%]

Epoch 49/50 - Loss: 3.6963, Accuracy: 13.05% Time: 19.93s

Epoch 50/50: 100%|      | 782/782 [00:20<00:00, 39.10it/s, loss=3.3562,
acc=12.57%]

Epoch 50/50 - Loss: 3.7128, Accuracy: 12.57% Time: 20.00s
Average epoch training time: 20.12 seconds

Testing started…

Testing: 100%|      | 157/157 [00:02<00:00, 58.42it/s, accuracy=12.54%]

Final Test Loss: 3.7206, Final Test Accuracy: 12.54%

Visualizing results…
```



[8]:
```
#ViT from scratch - 4x4  patch size - 4 heads 4 layers 512 dim 1024 mlp

# Delete model and optimizer variables
del model, optimizer, test_losses, test_accuracies, train_losses,␣
 ↪train_accuracies, epoch_times

# Clear CUDA cache if using GPU
gc.collect()
# Clear CUDA cache
if torch.cuda.is_available():
    torch.cuda.empty_cache()

# Initialize model
model = VisionTransformer(
    image_size=image_size,
    patch_size=patch_size,
    num_classes=num_classes,
    embed_dim=hidden_dim,
    num_heads=num_heads,
    num_layers=num_layers,
    mlp_dim=mlp_dim
).to(device)

# Loss and optimizer
criterion = nn.CrossEntropyLoss()
optimizer = torch.optim.AdamW(model.parameters(), lr=learning_rate,␣
 ↪weight_decay=0.01)

# Display model summary
summary(model,
        input_size=(batch_size, 3, image_size, image_size),
        col_names=["input_size", "output_size", "kernel_size", "num_params",␣
 ↪"mult_adds", "trainable"],
        col_width=20,
        depth=5,
        verbose=True,
        device=device)

# Hyperparameters
image_size = 32
patch_size = 4
num_classes = 100
num_epochs = 50
batch_size = 64
learning_rate = 0.001
num_heads = 4
num_layers = 4
```

```
hidden_dim = 512
mlp_dim = 1024

# Run training and testing
if __name__ == '__main__':
    print("Training started...")
    train_losses, train_accuracies, epoch_times = train()
    print("\nTesting started...")
    test_losses, test_accuracies = test()

    # Visualize results
    print("\nVisualizing results...")
    visualize_results(train_losses, train_accuracies, test_losses,␣
 ↪test_accuracies)
```

```
==========================================================================
==========================================================================
=====
Layer (type:depth-idx)                        Input Shape          Output Shape
Kernel Shape         Param #              Mult-Adds            Trainable
==========================================================================
==========================================================================
=====
VisionTransformer                             [64, 3, 32, 32]      [64, 100]
--                   16,896               --                   True
 PatchEmbedding: 1-1                            [64, 3, 32, 32]      [64, 64, 256]
--                   --                   --                   True
     Conv2d: 2-1                               [64, 3, 32, 32]      [64, 256, 8,
8]      [4, 4]               12,544               51,380,224           True
 Dropout: 1-2                                   [64, 65, 256]        [64, 65, 256]
--                   --                   --                   --
 ModuleList: 1-3                                --                   --
--                   --                   --                   True
     TransformerEncoder: 2-2                   [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
          LayerNorm: 3-1                     [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
          MultiHeadSelfAttention: 3-2        [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
               Linear: 4-1                   [64, 65, 256]        [64, 65, 768]
--                   197,376              12,632,064           True
               Linear: 4-2                   [64, 65, 256]        [64, 65, 256]
--                   65,792               4,210,688            True
          LayerNorm: 3-3                     [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
          MLP: 3-4                           [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
               Linear: 4-3                   [64, 65, 256]        [64, 65, 512]
```

```
--                    131,584               8,421,376           True
              GELU: 4-4                       [64, 65, 512]       [64, 65, 512]
--                    --                    --                  --
              Linear: 4-5                     [64, 65, 512]       [64, 65, 256]
--                    131,328               8,404,992           True
      TransformerEncoder: 2-3                 [64, 65, 256]       [64, 65, 256]
--                    --                    --                  True
          LayerNorm: 3-5                      [64, 65, 256]       [64, 65, 256]
--                    512                   32,768              True
          MultiHeadSelfAttention: 3-6         [64, 65, 256]       [64, 65, 256]
--                    --                    --                  True
              Linear: 4-6                     [64, 65, 256]       [64, 65, 768]
--                    197,376               12,632,064          True
              Linear: 4-7                     [64, 65, 256]       [64, 65, 256]
--                    65,792                4,210,688           True
          LayerNorm: 3-7                      [64, 65, 256]       [64, 65, 256]
--                    512                   32,768              True
          MLP: 3-8                            [64, 65, 256]       [64, 65, 256]
--                    --                    --                  True
              Linear: 4-8                     [64, 65, 256]       [64, 65, 512]
--                    131,584               8,421,376           True
              GELU: 4-9                       [64, 65, 512]       [64, 65, 512]
--                    --                    --                  --
              Linear: 4-10                    [64, 65, 512]       [64, 65, 256]
--                    131,328               8,404,992           True
      TransformerEncoder: 2-4                 [64, 65, 256]       [64, 65, 256]
--                    --                    --                  True
          LayerNorm: 3-9                      [64, 65, 256]       [64, 65, 256]
--                    512                   32,768              True
          MultiHeadSelfAttention: 3-10        [64, 65, 256]       [64, 65, 256]
--                    --                    --                  True
              Linear: 4-11                    [64, 65, 256]       [64, 65, 768]
--                    197,376               12,632,064          True
              Linear: 4-12                    [64, 65, 256]       [64, 65, 256]
--                    65,792                4,210,688           True
          LayerNorm: 3-11                     [64, 65, 256]       [64, 65, 256]
--                    512                   32,768              True
          MLP: 3-12                           [64, 65, 256]       [64, 65, 256]
--                    --                    --                  True
              Linear: 4-13                    [64, 65, 256]       [64, 65, 512]
--                    131,584               8,421,376           True
              GELU: 4-14                      [64, 65, 512]       [64, 65, 512]
--                    --                    --                  --
              Linear: 4-15                    [64, 65, 512]       [64, 65, 256]
--                    131,328               8,404,992           True
      TransformerEncoder: 2-5                 [64, 65, 256]       [64, 65, 256]
--                    --                    --                  True
          LayerNorm: 3-13                     [64, 65, 256]       [64, 65, 256]
```

```
--                   512                  32,768               True
          MultiHeadSelfAttention: 3-14      [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
               Linear: 4-16                 [64, 65, 256]        [64, 65, 768]
--                   197,376              12,632,064           True
               Linear: 4-17                 [64, 65, 256]        [64, 65, 256]
--                   65,792               4,210,688            True
          LayerNorm: 3-15                   [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
          MLP: 3-16                         [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
               Linear: 4-18                 [64, 65, 256]        [64, 65, 512]
--                   131,584              8,421,376            True
               GELU: 4-19                   [64, 65, 512]        [64, 65, 512]
--                   --                   --                   --
               Linear: 4-20                 [64, 65, 512]        [64, 65, 256]
--                   131,328              8,404,992            True
     TransformerEncoder: 2-6                [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
          LayerNorm: 3-17                   [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
          MultiHeadSelfAttention: 3-18      [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
               Linear: 4-21                 [64, 65, 256]        [64, 65, 768]
--                   197,376              12,632,064           True
               Linear: 4-22                 [64, 65, 256]        [64, 65, 256]
--                   65,792               4,210,688            True
          LayerNorm: 3-19                   [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
          MLP: 3-20                         [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
               Linear: 4-23                 [64, 65, 256]        [64, 65, 512]
--                   131,584              8,421,376            True
               GELU: 4-24                   [64, 65, 512]        [64, 65, 512]
--                   --                   --                   --
               Linear: 4-25                 [64, 65, 512]        [64, 65, 256]
--                   131,328              8,404,992            True
     TransformerEncoder: 2-7                [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
          LayerNorm: 3-21                   [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
          MultiHeadSelfAttention: 3-22      [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
               Linear: 4-26                 [64, 65, 256]        [64, 65, 768]
--                   197,376              12,632,064           True
               Linear: 4-27                 [64, 65, 256]        [64, 65, 256]
--                   65,792               4,210,688            True
          LayerNorm: 3-23                   [64, 65, 256]        [64, 65, 256]
```

```
--                   512                  32,768               True
        MLP: 3-24                         [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
            Linear: 4-28                  [64, 65, 256]        [64, 65, 512]
--                   131,584              8,421,376            True
            GELU: 4-29                    [64, 65, 512]        [64, 65, 512]
--                   --                   --                   --
            Linear: 4-30                  [64, 65, 512]        [64, 65, 256]
--                   131,328              8,404,992            True
    TransformerEncoder: 2-8               [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
        LayerNorm: 3-25                   [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
        MultiHeadSelfAttention: 3-26      [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
            Linear: 4-31                  [64, 65, 256]        [64, 65, 768]
--                   197,376              12,632,064           True
            Linear: 4-32                  [64, 65, 256]        [64, 65, 256]
--                   65,792               4,210,688            True
        LayerNorm: 3-27                   [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
        MLP: 3-28                         [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
            Linear: 4-33                  [64, 65, 256]        [64, 65, 512]
--                   131,584              8,421,376            True
            GELU: 4-34                    [64, 65, 512]        [64, 65, 512]
--                   --                   --                   --
            Linear: 4-35                  [64, 65, 512]        [64, 65, 256]
--                   131,328              8,404,992            True
    TransformerEncoder: 2-9               [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
        LayerNorm: 3-29                   [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
        MultiHeadSelfAttention: 3-30      [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
            Linear: 4-36                  [64, 65, 256]        [64, 65, 768]
--                   197,376              12,632,064           True
            Linear: 4-37                  [64, 65, 256]        [64, 65, 256]
--                   65,792               4,210,688            True
        LayerNorm: 3-31                   [64, 65, 256]        [64, 65, 256]
--                   512                  32,768               True
        MLP: 3-32                         [64, 65, 256]        [64, 65, 256]
--                   --                   --                   True
            Linear: 4-38                  [64, 65, 256]        [64, 65, 512]
--                   131,584              8,421,376            True
            GELU: 4-39                    [64, 65, 512]        [64, 65, 512]
--                   --                   --                   --
            Linear: 4-40                  [64, 65, 512]        [64, 65, 256]
```

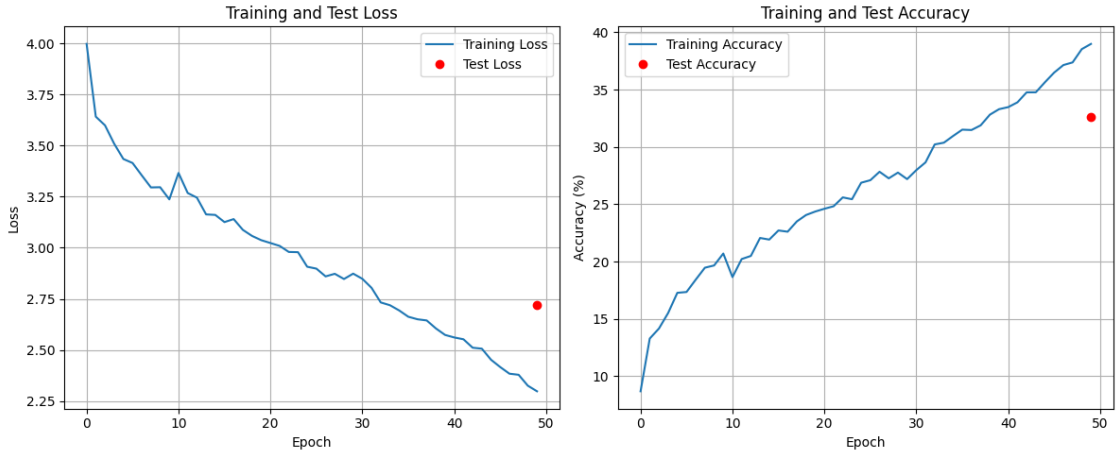
```
--                    131,328              8,404,992            True
 LayerNorm: 1-4                             [64, 65, 256]        [64, 65, 256]
--                    512                  32,768               True
 Linear: 1-5                                [64, 256]            [64, 100]
--                    25,700               1,644,800            True
==========================================================================
==========================================================================
=====
Total params: 4,272,484
Trainable params: 4,272,484
Non-trainable params: 0
Total mult-adds (Units.MEGABYTES): 322.94
==========================================================================
==========================================================================
=====
Input size (MB): 0.79
Forward/backward pass size (MB): 630.38
Params size (MB): 17.02
Estimated Total Size (MB): 648.19
==========================================================================
==========================================================================
=====
Training started…

Epoch 1/50:   1%|          | 4/782 [00:00<00:23, 33.81it/s, loss=5.0138,
acc=1.30%]

Input images shape: torch.Size([64, 3, 32, 32])
Labels shape: torch.Size([64])
Labels values: tensor([83, 55,  8, 10, 91, 25, 99, 57, 87, 66], device='cuda:0')
Model outputs shape: torch.Size([64, 100])
Expected outputs shape: torch.Size([64, 100])

Epoch 1/50: 100%|     | 782/782 [00:21<00:00, 36.49it/s, loss=3.8907,
acc=8.68%]

Epoch 1/50 - Loss: 3.9975, Accuracy: 8.68% Time: 21.43s

Epoch 2/50: 100%|     | 782/782 [00:21<00:00, 36.75it/s, loss=3.5034,
acc=13.28%]

Epoch 2/50 - Loss: 3.6417, Accuracy: 13.28% Time: 21.28s

Epoch 3/50: 100%|     | 782/782 [00:21<00:00, 36.70it/s, loss=4.0097,
acc=14.15%]

Epoch 3/50 - Loss: 3.5992, Accuracy: 14.15% Time: 21.31s

Epoch 4/50: 100%|     | 782/782 [00:21<00:00, 36.77it/s, loss=3.1374,
acc=15.51%]

Epoch 4/50 - Loss: 3.5092, Accuracy: 15.51% Time: 21.27s
```

```
Epoch 5/50: 100%|      | 782/782 [00:21<00:00, 36.76it/s, loss=3.4196,
acc=17.27%]

Epoch 5/50 - Loss: 3.4345, Accuracy: 17.27% Time: 21.27s

Epoch 6/50: 100%|      | 782/782 [00:21<00:00, 36.83it/s, loss=2.8220,
acc=17.34%]

Epoch 6/50 - Loss: 3.4148, Accuracy: 17.34% Time: 21.24s

Epoch 7/50: 100%|      | 782/782 [00:21<00:00, 36.72it/s, loss=4.0905,
acc=18.42%]

Epoch 7/50 - Loss: 3.3544, Accuracy: 18.42% Time: 21.30s

Epoch 8/50: 100%|      | 782/782 [00:21<00:00, 36.76it/s, loss=2.9380,
acc=19.46%]

Epoch 8/50 - Loss: 3.2954, Accuracy: 19.46% Time: 21.28s

Epoch 9/50: 100%|      | 782/782 [00:21<00:00, 36.71it/s, loss=3.1092,
acc=19.66%]

Epoch 9/50 - Loss: 3.2963, Accuracy: 19.66% Time: 21.30s

Epoch 10/50: 100%|      | 782/782 [00:21<00:00, 36.67it/s, loss=3.7064,
acc=20.70%]

Epoch 10/50 - Loss: 3.2370, Accuracy: 20.70% Time: 21.32s

Epoch 11/50: 100%|      | 782/782 [00:21<00:00, 36.67it/s, loss=2.9495,
acc=18.64%]

Epoch 11/50 - Loss: 3.3661, Accuracy: 18.64% Time: 21.33s

Epoch 12/50: 100%|      | 782/782 [00:21<00:00, 36.72it/s, loss=3.3018,
acc=20.22%]

Epoch 12/50 - Loss: 3.2680, Accuracy: 20.22% Time: 21.30s

Epoch 13/50: 100%|      | 782/782 [00:21<00:00, 36.73it/s, loss=3.1222,
acc=20.49%]

Epoch 13/50 - Loss: 3.2455, Accuracy: 20.49% Time: 21.29s

Epoch 14/50: 100%|      | 782/782 [00:21<00:00, 36.73it/s, loss=3.3770,
acc=22.06%]

Epoch 14/50 - Loss: 3.1634, Accuracy: 22.06% Time: 21.29s

Epoch 15/50: 100%|      | 782/782 [00:21<00:00, 36.72it/s, loss=2.8565,
acc=21.91%]

Epoch 15/50 - Loss: 3.1609, Accuracy: 21.91% Time: 21.30s

Epoch 16/50: 100%|      | 782/782 [00:21<00:00, 36.60it/s, loss=3.0487,
acc=22.71%]

Epoch 16/50 - Loss: 3.1258, Accuracy: 22.71% Time: 21.37s
```

```
Epoch 17/50: 100%|      | 782/782 [00:21<00:00, 36.70it/s, loss=3.5242,
acc=22.60%]

Epoch 17/50 - Loss: 3.1405, Accuracy: 22.60% Time: 21.31s

Epoch 18/50: 100%|      | 782/782 [00:21<00:00, 36.75it/s, loss=3.3310,
acc=23.50%]

Epoch 18/50 - Loss: 3.0877, Accuracy: 23.50% Time: 21.28s

Epoch 19/50: 100%|      | 782/782 [00:21<00:00, 36.81it/s, loss=2.3830,
acc=24.06%]

Epoch 19/50 - Loss: 3.0581, Accuracy: 24.06% Time: 21.25s

Epoch 20/50: 100%|      | 782/782 [00:21<00:00, 36.82it/s, loss=3.5386,
acc=24.36%]

Epoch 20/50 - Loss: 3.0373, Accuracy: 24.36% Time: 21.24s

Epoch 21/50: 100%|      | 782/782 [00:21<00:00, 36.71it/s, loss=3.5471,
acc=24.60%]

Epoch 21/50 - Loss: 3.0234, Accuracy: 24.60% Time: 21.30s

Epoch 22/50: 100%|      | 782/782 [00:21<00:00, 36.73it/s, loss=3.2794,
acc=24.82%]

Epoch 22/50 - Loss: 3.0093, Accuracy: 24.82% Time: 21.29s

Epoch 23/50: 100%|      | 782/782 [00:21<00:00, 36.77it/s, loss=2.0467,
acc=25.60%]

Epoch 23/50 - Loss: 2.9797, Accuracy: 25.60% Time: 21.27s

Epoch 24/50: 100%|      | 782/782 [00:21<00:00, 36.76it/s, loss=1.9271,
acc=25.44%]

Epoch 24/50 - Loss: 2.9786, Accuracy: 25.44% Time: 21.27s

Epoch 25/50: 100%|      | 782/782 [00:21<00:00, 36.68it/s, loss=3.5535,
acc=26.88%]

Epoch 25/50 - Loss: 2.9073, Accuracy: 26.88% Time: 21.32s

Epoch 26/50: 100%|      | 782/782 [00:21<00:00, 36.70it/s, loss=2.8421,
acc=27.10%]

Epoch 26/50 - Loss: 2.8981, Accuracy: 27.10% Time: 21.31s

Epoch 27/50: 100%|      | 782/782 [00:21<00:00, 36.75it/s, loss=3.0038,
acc=27.83%]

Epoch 27/50 - Loss: 2.8601, Accuracy: 27.83% Time: 21.28s

Epoch 28/50: 100%|      | 782/782 [00:21<00:00, 36.68it/s, loss=2.9463,
acc=27.26%]

Epoch 28/50 - Loss: 2.8729, Accuracy: 27.26% Time: 21.32s
```

```
Epoch 29/50: 100%|      | 782/782 [00:21<00:00, 36.72it/s, loss=2.3054,
acc=27.76%]

Epoch 29/50 - Loss: 2.8473, Accuracy: 27.76% Time: 21.30s

Epoch 30/50: 100%|      | 782/782 [00:21<00:00, 36.70it/s, loss=2.8155,
acc=27.19%]

Epoch 30/50 - Loss: 2.8736, Accuracy: 27.19% Time: 21.31s

Epoch 31/50: 100%|      | 782/782 [00:21<00:00, 36.69it/s, loss=3.5064,
acc=27.98%]

Epoch 31/50 - Loss: 2.8483, Accuracy: 27.98% Time: 21.31s

Epoch 32/50: 100%|      | 782/782 [00:21<00:00, 36.81it/s, loss=3.0599,
acc=28.65%]

Epoch 32/50 - Loss: 2.8041, Accuracy: 28.65% Time: 21.25s

Epoch 33/50: 100%|      | 782/782 [00:21<00:00, 36.78it/s, loss=3.1293,
acc=30.22%]

Epoch 33/50 - Loss: 2.7327, Accuracy: 30.22% Time: 21.26s

Epoch 34/50: 100%|      | 782/782 [00:21<00:00, 36.67it/s, loss=2.2710,
acc=30.37%]

Epoch 34/50 - Loss: 2.7191, Accuracy: 30.37% Time: 21.33s

Epoch 35/50: 100%|      | 782/782 [00:21<00:00, 36.76it/s, loss=2.4671,
acc=30.95%]

Epoch 35/50 - Loss: 2.6936, Accuracy: 30.95% Time: 21.27s

Epoch 36/50: 100%|      | 782/782 [00:21<00:00, 36.81it/s, loss=2.6896,
acc=31.51%]

Epoch 36/50 - Loss: 2.6626, Accuracy: 31.51% Time: 21.24s

Epoch 37/50: 100%|      | 782/782 [00:21<00:00, 36.66it/s, loss=2.2464,
acc=31.47%]

Epoch 37/50 - Loss: 2.6503, Accuracy: 31.47% Time: 21.33s

Epoch 38/50: 100%|      | 782/782 [00:21<00:00, 36.65it/s, loss=2.3768,
acc=31.88%]

Epoch 38/50 - Loss: 2.6446, Accuracy: 31.88% Time: 21.34s

Epoch 39/50: 100%|      | 782/782 [00:21<00:00, 36.58it/s, loss=2.9805,
acc=32.81%]

Epoch 39/50 - Loss: 2.6055, Accuracy: 32.81% Time: 21.38s

Epoch 40/50: 100%|      | 782/782 [00:21<00:00, 36.72it/s, loss=3.0869,
acc=33.29%]

Epoch 40/50 - Loss: 2.5736, Accuracy: 33.29% Time: 21.30s
```

```
Epoch 41/50: 100%|      | 782/782 [00:21<00:00, 36.63it/s, loss=2.8966,
acc=33.47%]

Epoch 41/50 - Loss: 2.5613, Accuracy: 33.47% Time: 21.35s

Epoch 42/50: 100%|      | 782/782 [00:21<00:00, 36.68it/s, loss=2.4705,
acc=33.89%]

Epoch 42/50 - Loss: 2.5525, Accuracy: 33.89% Time: 21.32s

Epoch 43/50: 100%|      | 782/782 [00:21<00:00, 36.64it/s, loss=2.9227,
acc=34.76%]

Epoch 43/50 - Loss: 2.5112, Accuracy: 34.76% Time: 21.34s

Epoch 44/50: 100%|      | 782/782 [00:21<00:00, 35.60it/s, loss=2.6640,
acc=34.77%]

Epoch 44/50 - Loss: 2.5064, Accuracy: 34.77% Time: 21.96s

Epoch 45/50: 100%|      | 782/782 [00:22<00:00, 35.47it/s, loss=2.6889,
acc=35.65%]

Epoch 45/50 - Loss: 2.4522, Accuracy: 35.65% Time: 22.05s

Epoch 46/50: 100%|      | 782/782 [00:21<00:00, 35.72it/s, loss=3.4575,
acc=36.49%]

Epoch 46/50 - Loss: 2.4164, Accuracy: 36.49% Time: 21.89s

Epoch 47/50: 100%|      | 782/782 [00:21<00:00, 35.83it/s, loss=2.1748,
acc=37.14%]

Epoch 47/50 - Loss: 2.3841, Accuracy: 37.14% Time: 21.83s

Epoch 48/50: 100%|      | 782/782 [00:21<00:00, 36.95it/s, loss=1.6671,
acc=37.37%]

Epoch 48/50 - Loss: 2.3787, Accuracy: 37.37% Time: 21.16s

Epoch 49/50: 100%|      | 782/782 [00:21<00:00, 36.18it/s, loss=2.0253,
acc=38.53%]

Epoch 49/50 - Loss: 2.3254, Accuracy: 38.53% Time: 21.61s

Epoch 50/50: 100%|      | 782/782 [00:21<00:00, 35.88it/s, loss=3.5032,
acc=38.99%]

Epoch 50/50 - Loss: 2.2980, Accuracy: 38.99% Time: 21.80s
Average epoch training time: 21.36 seconds

Testing started…

Testing: 100%|      | 157/157 [00:02<00:00, 59.97it/s, accuracy=32.61%]

Final Test Loss: 2.7216, Final Test Accuracy: 32.61%

Visualizing results…
```



[9]:
```python
#problem 2: swin finetuning and "from scratch" comparison
import torch
import torch.nn as nn
import torchvision
import torchvision.transforms as transforms
from torch.utils.data import DataLoader
from transformers import SwinForImageClassification, SwinConfig,␣
 ↪AutoImageProcessor
from tqdm import tqdm
import time
import pandas as pd
from copy import deepcopy

# Device configuration
device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
print(f"Using device: {device}")

# Hyperparameters
num_epochs = 5
batch_size = 32
learning_rate = 2e-5  # Smaller learning rate for fine-tuning
image_size = 224  # Swin expects 224x224 input by default

# Model configurations
models_config = {
    "swin-tiny-pretrained": {
        "name": "microsoft/swin-tiny-patch4-window7-224",
        "pretrained": True,
        "freeze_backbone": True
    },
    "swin-small-pretrained": {
```

```
        "name": "microsoft/swin-small-patch4-window7-224",
        "pretrained": True,
        "freeze_backbone": True
    },
    "swin-tiny-scratch": {
        "name": "microsoft/swin-tiny-patch4-window7-224",
        "pretrained": False,
        "freeze_backbone": False
    }
}

# Results tracking
results = {
    "model": [],
    "epoch_train_time": [],
    "test_accuracy": []
}

# CIFAR-100 dataset preparation
def prepare_data(model_name):
    # Data preparation with proper preprocessing for Swin
    processor = AutoImageProcessor.from_pretrained(model_name)

    transform = transforms.Compose([
        transforms.Resize((image_size, image_size)),
        transforms.ToTensor(),
        transforms.Normalize(mean=processor.image_mean, std=processor.image_std)
    ])

    # CIFAR-100 dataset
    train_dataset = torchvision.datasets.CIFAR100(root='./data', train=True,
                                                  download=True,␣
 ↪transform=transform)
    test_dataset = torchvision.datasets.CIFAR100(root='./data', train=False,
                                                 download=True, transform=transform)

    train_loader = DataLoader(dataset=train_dataset, batch_size=batch_size,␣
 ↪shuffle=True)
    test_loader = DataLoader(dataset=test_dataset, batch_size=batch_size,␣
 ↪shuffle=False)

    return train_loader, test_loader

# Create and configure model
def setup_model(config):
    if config["pretrained"]:
        print(f"Loading pretrained {config['name']}...")
```

```
        model = SwinForImageClassification.from_pretrained(
            config["name"],
            num_labels=100,  # CIFAR-100 has 100 classes
            ignore_mismatched_sizes=True  # Allows replacing the original␣
↪classifier head
        ).to(device)
    else:
        print(f"Initializing {config['name']} from scratch...")
        # For scratch training, initialize with the same architecture but␣
↪random weights
        swin_config = SwinConfig.from_pretrained(
            config["name"],
            num_labels=100  # CIFAR-100 has 100 classes
        )
        model = SwinForImageClassification(swin_config).to(device)

    # Freeze backbone parameters if specified
    if config["freeze_backbone"]:
        print("Freezing backbone parameters...")
        for param in model.swin.parameters():
            param.requires_grad = False

        # Only the classifier head will be trained
        for param in model.classifier.parameters():
            param.requires_grad = True

        # Configure optimizer for fine-tuning (only classifier parameters)
        optimizer = torch.optim.Adam(model.classifier.parameters(),␣
↪lr=learning_rate)
    else:
        print("Training all parameters...")
        # Configure optimizer for training from scratch (all parameters)
        optimizer = torch.optim.Adam(model.parameters(), lr=learning_rate)

    return model, optimizer

# Training function
def train_model(model, optimizer, train_loader, test_loader, model_name):
    criterion = nn.CrossEntropyLoss()
    epoch_times = []

    for epoch in range(num_epochs):
        model.train()
        start_time = time.time()
        progress_bar = tqdm(train_loader, desc=f'Epoch [{epoch+1}/
↪{num_epochs}]')
```

```
        for i, (images, labels) in enumerate(progress_bar):
            images = images.to(device)
            labels = labels.to(device)

            # Forward pass
            outputs = model(images).logits
            loss = criterion(outputs, labels)

            # Backward and optimize
            optimizer.zero_grad()
            loss.backward()
            optimizer.step()

            # Update progress bar
            if (i+1) % 100 == 0:
                progress_bar.set_postfix({'loss': loss.item()})

        epoch_time = time.time() - start_time
        epoch_times.append(epoch_time)
        print(f"Epoch {epoch+1} training time: {epoch_time:.2f} seconds")

    # Calculate average epoch time
    avg_epoch_time = sum(epoch_times) / len(epoch_times)

    # Test the model
    accuracy = test_model(model, test_loader)

    # Store results
    results["model"].append(model_name)
    results["epoch_train_time"].append(avg_epoch_time)
    results["test_accuracy"].append(accuracy)

    return avg_epoch_time, accuracy

# Testing function
def test_model(model, test_loader):
    model.eval()
    with torch.no_grad():
        correct = 0
        total = 0
        for images, labels in tqdm(test_loader, desc='Testing'):
            images = images.to(device)
            labels = labels.to(device)
            outputs = model(images).logits
            _, predicted = torch.max(outputs.data, 1)
            total += labels.size(0)
            correct += (predicted == labels).sum().item()
```

```
        accuracy = 100 * correct / total
        print(f'Test Accuracy: {accuracy:.2f}%')

        return accuracy

# Main function
def main():
    for model_name, config in models_config.items():
        print(f"\n{'='*50}")
        print(f"Training {model_name}")
        print(f"{'='*50}")

        # Prepare data
        train_loader, test_loader = prepare_data(config["name"])

        # Setup model
        model, optimizer = setup_model(config)

        # Train and test model
        avg_epoch_time, accuracy = train_model(model, optimizer, train_loader,␣
↪test_loader, model_name)

        print(f"Model: {model_name}")
        print(f"Average epoch training time: {avg_epoch_time:.2f} seconds")
        print(f"Test accuracy: {accuracy:.2f}%")

        # Clear GPU memory
        del model, optimizer
        torch.cuda.empty_cache() if torch.cuda.is_available() else None

    # Create and display results table
    results_df = pd.DataFrame(results)
    print("\nResults Summary:")
    print(results_df.to_string(index=False))

    # Save results to CSV
    results_df.to_csv("swin_comparison_results.csv", index=False)
    print("Results saved to swin_comparison_results.csv")

    # Print findings for report
    print("\nKey Findings for Report:")
    print("1. Fine-tuning vs. Training from Scratch:")
    ft_acc = results_df[results_df['model'] ==␣
↪'swin-tiny-pretrained']['test_accuracy'].values[0]
    scratch_acc = results_df[results_df['model'] ==␣
↪'swin-tiny-scratch']['test_accuracy'].values[0]
```

```
    print(f"   - Accuracy difference: {ft_acc - scratch_acc:.2f}%")

    print("2. Swin-Tiny vs. Swin-Small:")
    tiny_acc = results_df[results_df['model'] ==␣
↪'swin-tiny-pretrained']['test_accuracy'].values[0]
    small_acc = results_df[results_df['model'] ==␣
↪'swin-small-pretrained']['test_accuracy'].values[0]
    print(f"   - Accuracy difference: {small_acc - tiny_acc:.2f}%")

    # Note about training times
    tiny_time = results_df[results_df['model'] ==␣
↪'swin-tiny-pretrained']['epoch_train_time'].values[0]
    small_time = results_df[results_df['model'] ==␣
↪'swin-small-pretrained']['epoch_train_time'].values[0]
    scratch_time = results_df[results_df['model'] ==␣
↪'swin-tiny-scratch']['epoch_train_time'].values[0]
    print(f"3. Training Time Comparison:")
    print(f"   - Swin-Tiny (pretrained): {tiny_time:.2f} seconds/epoch")
    print(f"   - Swin-Small (pretrained): {small_time:.2f} seconds/epoch")
    print(f"   - Swin-Tiny (scratch): {scratch_time:.2f} seconds/epoch")

if __name__ == '__main__':
    main()
```

```
/home/dman/.venv/master/lib/python3.12/site-packages/tqdm/auto.py:21:
TqdmWarning: IProgress not found. Please update jupyter and ipywidgets. See
https://ipywidgets.readthedocs.io/en/stable/user_install.html
  from .autonotebook import tqdm as notebook_tqdm
Using a slow image processor as `use_fast` is unset and a slow processor was
saved with this model. `use_fast=True` will be the default behavior in v4.52,
even if the model was saved with a slow processor. This will result in minor
differences in outputs. You'll still be able to use a slow processor with
`use_fast=False`.
```

```
Using device: cuda

==================================================
Training swin-tiny-pretrained
==================================================
```

```
Some weights of SwinForImageClassification were not initialized from the model
checkpoint at microsoft/swin-tiny-patch4-window7-224 and are newly initialized
because the shapes did not match:
- classifier.bias: found shape torch.Size([1000]) in the checkpoint and
torch.Size([100]) in the model instantiated
- classifier.weight: found shape torch.Size([1000, 768]) in the checkpoint and
torch.Size([100, 768]) in the model instantiated
You should probably TRAIN this model on a down-stream task to be able to use it
```

```
for predictions and inference.

Loading pretrained microsoft/swin-tiny-patch4-window7-224…
Freezing backbone parameters…

Epoch [1/5]: 100%|      | 1563/1563 [01:20<00:00, 19.36it/s, loss=3.58]

Epoch 1 training time: 80.74 seconds

Epoch [2/5]: 100%|      | 1563/1563 [01:21<00:00, 19.15it/s, loss=2.83]

Epoch 2 training time: 81.64 seconds

Epoch [3/5]: 100%|      | 1563/1563 [01:23<00:00, 18.65it/s, loss=2.07]

Epoch 3 training time: 83.82 seconds

Epoch [4/5]: 100%|      | 1563/1563 [01:21<00:00, 19.26it/s, loss=1.54]

Epoch 4 training time: 81.15 seconds

Epoch [5/5]: 100%|      | 1563/1563 [01:21<00:00, 19.11it/s, loss=1.52]

Epoch 5 training time: 81.80 seconds

Testing: 100%|      | 313/313 [00:20<00:00, 15.01it/s]

Test Accuracy: 66.33%
Model: swin-tiny-pretrained
Average epoch training time: 81.83 seconds
Test accuracy: 66.33%

==================================================
Training swin-small-pretrained
==================================================
Loading pretrained microsoft/swin-small-patch4-window7-224…

Some weights of SwinForImageClassification were not initialized from the model
checkpoint at microsoft/swin-small-patch4-window7-224 and are newly initialized
because the shapes did not match:
- classifier.weight: found shape torch.Size([1000, 768]) in the checkpoint and
torch.Size([100, 768]) in the model instantiated
- classifier.bias: found shape torch.Size([1000]) in the checkpoint and
torch.Size([100]) in the model instantiated
You should probably TRAIN this model on a down-stream task to be able to use it
for predictions and inference.

Freezing backbone parameters…

Epoch [1/5]: 100%|      | 1563/1563 [02:13<00:00, 11.67it/s, loss=3.43]

Epoch 1 training time: 133.91 seconds

Epoch [2/5]: 100%|      | 1563/1563 [02:16<00:00, 11.48it/s, loss=2.36]

Epoch 2 training time: 136.10 seconds

Epoch [3/5]: 100%|      | 1563/1563 [02:16<00:00, 11.46it/s, loss=1.89]
```

```
Epoch 3 training time: 136.41 seconds

Epoch [4/5]: 100%|     | 1563/1563 [02:15<00:00, 11.52it/s, loss=1.51]

Epoch 4 training time: 135.73 seconds

Epoch [5/5]: 100%|     | 1563/1563 [02:06<00:00, 12.36it/s, loss=1.64]

Epoch 5 training time: 126.42 seconds

Testing: 100%|     | 313/313 [00:28<00:00, 11.11it/s]

Test Accuracy: 70.48%
Model: swin-small-pretrained
Average epoch training time: 133.71 seconds
Test accuracy: 70.48%

==================================================
Training swin-tiny-scratch
==================================================
Initializing microsoft/swin-tiny-patch4-window7-224 from scratch…
Training all parameters…

Epoch [1/5]: 100%|     | 1563/1563 [03:15<00:00,  8.02it/s, loss=3.38]

Epoch 1 training time: 195.01 seconds

Epoch [2/5]: 100%|     | 1563/1563 [03:26<00:00,  7.56it/s, loss=2.9]

Epoch 2 training time: 206.74 seconds

Epoch [3/5]: 100%|     | 1563/1563 [03:31<00:00,  7.41it/s, loss=2.92]

Epoch 3 training time: 211.06 seconds

Epoch [4/5]: 100%|     | 1563/1563 [03:33<00:00,  7.31it/s, loss=2.95]

Epoch 4 training time: 213.95 seconds

Epoch [5/5]: 100%|     | 1563/1563 [03:34<00:00,  7.29it/s, loss=2.47]

Epoch 5 training time: 214.35 seconds

Testing: 100%|     | 313/313 [00:20<00:00, 15.30it/s]

Test Accuracy: 37.09%
Model: swin-tiny-scratch
Average epoch training time: 208.22 seconds
Test accuracy: 37.09%

Results Summary:
                model  epoch_train_time  test_accuracy
 swin-tiny-pretrained         81.827815          66.33
swin-small-pretrained        133.713724          70.48
    swin-tiny-scratch        208.222045          37.09
Results saved to swin_comparison_results.csv
```

```
Key Findings for Report:
1. Fine-tuning vs. Training from Scratch:
   - Accuracy difference: 29.24%
2. Swin-Tiny vs. Swin-Small:
   - Accuracy difference: 4.15%
3. Training Time Comparison:
   - Swin-Tiny (pretrained): 81.83 seconds/epoch
   - Swin-Small (pretrained): 133.71 seconds/epoch
   - Swin-Tiny (scratch): 208.22 seconds/epoch
```

[ ]:
```python
#ResNet100 baseline
import torch
import torch.nn as nn
import torch.optim as optim
from torch.utils.data import DataLoader
import torchvision
import torchvision.transforms as transforms
import time
import numpy as np
import matplotlib.pyplot as plt
from torchinfo import summary
from tqdm import tqdm

# Set random seed for reproducibility
torch.manual_seed(42)
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
print(f"Using device: {device}")

# Helper function to count parameters and calculate FLOPs
def count_params_and_flops(model, input_size=(1, 3, 32, 32)):
    """Count parameters and estimate FLOPs using torchinfo."""
    model_summary = summary(model, input_size=input_size, verbose=0)
    params = model_summary.total_params
    flops = model_summary.total_mult_adds
    return params, flops, model_summary

# Training function
def train(model, train_loader, criterion, optimizer, device):
    model.train()
    running_loss = 0.0
    correct = 0
    total = 0
    start_time = time.time()

    pbar = tqdm(train_loader, desc='Training', leave=True)
    for batch_idx, (inputs, targets) in enumerate(pbar):
        inputs, targets = inputs.to(device), targets.to(device)
```

```
        optimizer.zero_grad()
        outputs = model(inputs)
        loss = criterion(outputs, targets)
        loss.backward()
        optimizer.step()

        running_loss += loss.item()
        _, predicted = outputs.max(1)
        total += targets.size(0)
        correct += predicted.eq(targets).sum().item()

        # Update progress bar with current loss and accuracy
        current_loss = running_loss / (batch_idx + 1)
        current_acc = 100.0 * correct / total
        pbar.set_postfix({'loss': f'{current_loss:.4f}', 'acc': f'{current_acc:.
↪2f}%'})

    train_loss = running_loss / len(train_loader)
    train_acc = 100.0 * correct / total
    epoch_time = time.time() - start_time

    return train_loss, train_acc, epoch_time

# Evaluation function
def evaluate(model, test_loader, criterion, device):
    model.eval()
    running_loss = 0.0
    correct = 0
    total = 0

    with torch.no_grad():
        pbar = tqdm(test_loader, desc='Evaluating', leave=True)
        for batch_idx, (inputs, targets) in enumerate(pbar):
            inputs, targets = inputs.to(device), targets.to(device)
            outputs = model(inputs)
            loss = criterion(outputs, targets)

            running_loss += loss.item()
            _, predicted = outputs.max(1)
            total += targets.size(0)
            correct += predicted.eq(targets).sum().item()

            # Update progress bar
            current_loss = running_loss / (batch_idx + 1)
            current_acc = 100.0 * correct / total
```

```
            pbar.set_postfix({'loss': f'{current_loss:.4f}', 'acc':␣
↪f'{current_acc:.2f}%'})

    test_loss = running_loss / len(test_loader)
    test_acc = 100.0 * correct / total

    return test_loss, test_acc

# Main function to run the ResNet experiment
def main():
    # Data preprocessing
    transform_train = transforms.Compose([
        transforms.RandomCrop(32, padding=4),
        transforms.RandomHorizontalFlip(),
        transforms.ToTensor(),
        transforms.Normalize((0.5071, 0.4867, 0.4408), (0.2675, 0.2565, 0.
↪2761)),
    ])

    transform_test = transforms.Compose([
        transforms.ToTensor(),
        transforms.Normalize((0.5071, 0.4867, 0.4408), (0.2675, 0.2565, 0.
↪2761)),
    ])

    # Load CIFAR-100 dataset
    trainset = torchvision.datasets.CIFAR100(
        root='./data', train=True, download=True, transform=transform_train)
    trainloader = torch.utils.data.DataLoader(
        trainset, batch_size=64, shuffle=True, num_workers=2)

    testset = torchvision.datasets.CIFAR100(
        root='./data', train=False, download=True, transform=transform_test)
    testloader = torch.utils.data.DataLoader(
        testset, batch_size=64, shuffle=False, num_workers=2)

    # Create ResNet-18 model
    resnet18 = torchvision.models.resnet18(weights=None)
    # Modify the first layer for CIFAR-100 (smaller images)
    resnet18.conv1 = nn.Conv2d(3, 64, kernel_size=3, stride=1, padding=1,␣
↪bias=False)
    resnet18.maxpool = nn.Identity()  # Remove maxpool layer
    # Modify the final FC layer for 100 classes
    resnet18.fc = nn.Linear(512, 100)

    # Calculate parameters and FLOPs
    params, flops, model_summary = count_params_and_flops(resnet18)
```

```
    print(f"ResNet-18 Model Summary:")
    print(model_summary)
    print(f"Number of parameters: {params:,}")
    print(f"Estimated FLOPs per forward pass: {flops:,}")

    # Set up training
    num_epochs = 30
    criterion = nn.CrossEntropyLoss()
    optimizer = optim.Adam(resnet18.parameters(), lr=0.001)

    # Training loop
    resnet18.to(device)
    results = {
        "train_time_per_epoch": [],
        "train_loss": [],
        "train_acc": [],
        "test_loss": [],
        "test_acc": []
    }

    for epoch in range(num_epochs):
        # Train
        train_loss, train_acc, epoch_time = train(resnet18, trainloader,␣
↪criterion, optimizer, device)

        # Evaluate
        test_loss, test_acc = evaluate(resnet18, testloader, criterion, device)

        # Record results
        results["train_time_per_epoch"].append(epoch_time)
        results["train_loss"].append(train_loss)
        results["train_acc"].append(train_acc)
        results["test_loss"].append(test_loss)
        results["test_acc"].append(test_acc)

        print(f"Epoch {epoch+1}/{num_epochs}, Train Loss: {train_loss:.4f},␣
↪Train Acc: {train_acc:.2f}%, "
              f"Test Loss: {test_loss:.4f}, Test Acc: {test_acc:.2f}%, Time:␣
↪{epoch_time:.2f}s")

    # Calculate average metrics
    avg_train_time = np.mean(results["train_time_per_epoch"])
    final_test_acc = results["test_acc"][-1]

    print("\nFinal Results:")
    print(f"Average training time per epoch: {avg_train_time:.2f}s")
    print(f"Final test accuracy: {final_test_acc:.2f}%")
```

```python
    # Visualize results
    epochs = range(1, num_epochs+1)

    # Plot accuracy
    plt.figure(figsize=(12, 5))
    plt.subplot(1, 2, 1)
    plt.plot(epochs, results["train_acc"], label="Train Accuracy")
    plt.plot(epochs, results["test_acc"], label="Test Accuracy")
    plt.xlabel("Epoch")
    plt.ylabel("Accuracy (%)")
    plt.title("Training and Test Accuracy")
    plt.legend()
    plt.grid(True)

    # Plot loss
    plt.subplot(1, 2, 2)
    plt.plot(epochs, results["train_loss"], label="Train Loss")
    plt.plot(epochs, results["test_loss"], label="Test Loss")
    plt.xlabel("Epoch")
    plt.ylabel("Loss")
    plt.title("Training and Test Loss")
    plt.legend()
    plt.grid(True)

    plt.tight_layout()
    plt.savefig("resnet100_performance.png")
    plt.show()

if __name__ == "__main__":
    main()
```

```
Using device: cuda
ResNet-18 Model Summary:
==========================================================================================
==========
Layer (type:depth-idx)                   Output Shape              Param #
==========================================================================================
==========
ResNet                                    [1, 100]                  --
 Conv2d: 1-1                             [1, 64, 32, 32]           1,728
 BatchNorm2d: 1-2                        [1, 64, 32, 32]           128
 ReLU: 1-3                               [1, 64, 32, 32]           --
 Identity: 1-4                           [1, 64, 32, 32]           --
 Sequential: 1-5                         [1, 64, 32, 32]           --
     BasicBlock: 2-1                    [1, 64, 32, 32]           --
          Conv2d: 3-1                   [1, 64, 32, 32]           36,864
          BatchNorm2d: 3-2              [1, 64, 32, 32]           128
```

```
        ReLU: 3-3                [1, 64, 32, 32]           --
        Conv2d: 3-4              [1, 64, 32, 32]           36,864
        BatchNorm2d: 3-5         [1, 64, 32, 32]           128
        ReLU: 3-6                [1, 64, 32, 32]           --
    BasicBlock: 2-2              [1, 64, 32, 32]           --
        Conv2d: 3-7              [1, 64, 32, 32]           36,864
        BatchNorm2d: 3-8         [1, 64, 32, 32]           128
        ReLU: 3-9                [1, 64, 32, 32]           --
        Conv2d: 3-10             [1, 64, 32, 32]           36,864
        BatchNorm2d: 3-11        [1, 64, 32, 32]           128
        ReLU: 3-12               [1, 64, 32, 32]           --
Sequential: 1-6                  [1, 128, 16, 16]          --
    BasicBlock: 2-3              [1, 128, 16, 16]          --
        Conv2d: 3-13             [1, 128, 16, 16]          73,728
        BatchNorm2d: 3-14        [1, 128, 16, 16]          256
        ReLU: 3-15               [1, 128, 16, 16]          --
        Conv2d: 3-16             [1, 128, 16, 16]          147,456
        BatchNorm2d: 3-17        [1, 128, 16, 16]          256
        Sequential: 3-18         [1, 128, 16, 16]          8,448
        ReLU: 3-19               [1, 128, 16, 16]          --
    BasicBlock: 2-4              [1, 128, 16, 16]          --
        Conv2d: 3-20             [1, 128, 16, 16]          147,456
        BatchNorm2d: 3-21        [1, 128, 16, 16]          256
        ReLU: 3-22               [1, 128, 16, 16]          --
        Conv2d: 3-23             [1, 128, 16, 16]          147,456
        BatchNorm2d: 3-24        [1, 128, 16, 16]          256
        ReLU: 3-25               [1, 128, 16, 16]          --
Sequential: 1-7                  [1, 256, 8, 8]            --
    BasicBlock: 2-5              [1, 256, 8, 8]            --
        Conv2d: 3-26             [1, 256, 8, 8]            294,912
        BatchNorm2d: 3-27        [1, 256, 8, 8]            512
        ReLU: 3-28               [1, 256, 8, 8]            --
        Conv2d: 3-29             [1, 256, 8, 8]            589,824
        BatchNorm2d: 3-30        [1, 256, 8, 8]            512
        Sequential: 3-31         [1, 256, 8, 8]            33,280
        ReLU: 3-32               [1, 256, 8, 8]            --
    BasicBlock: 2-6              [1, 256, 8, 8]            --
        Conv2d: 3-33             [1, 256, 8, 8]            589,824
        BatchNorm2d: 3-34        [1, 256, 8, 8]            512
        ReLU: 3-35               [1, 256, 8, 8]            --
        Conv2d: 3-36             [1, 256, 8, 8]            589,824
        BatchNorm2d: 3-37        [1, 256, 8, 8]            512
        ReLU: 3-38               [1, 256, 8, 8]            --
Sequential: 1-8                  [1, 512, 4, 4]            --
    BasicBlock: 2-7              [1, 512, 4, 4]            --
        Conv2d: 3-39             [1, 512, 4, 4]            1,179,648
        BatchNorm2d: 3-40        [1, 512, 4, 4]            1,024
        ReLU: 3-41               [1, 512, 4, 4]            --
```

```
          Conv2d: 3-42                  [1, 512, 4, 4]            2,359,296
          BatchNorm2d: 3-43             [1, 512, 4, 4]            1,024
          Sequential: 3-44              [1, 512, 4, 4]            132,096
          ReLU: 3-45                    [1, 512, 4, 4]            --
     BasicBlock: 2-8                    [1, 512, 4, 4]            --
          Conv2d: 3-46                  [1, 512, 4, 4]            2,359,296
          BatchNorm2d: 3-47             [1, 512, 4, 4]            1,024
          ReLU: 3-48                    [1, 512, 4, 4]            --
          Conv2d: 3-49                  [1, 512, 4, 4]            2,359,296
          BatchNorm2d: 3-50             [1, 512, 4, 4]            1,024
          ReLU: 3-51                    [1, 512, 4, 4]            --
 AdaptiveAvgPool2d: 1-9                 [1, 512, 1, 1]            --
 Linear: 1-10                           [1, 100]                  51,300
==========================================================================
==========
Total params: 11,220,132
Trainable params: 11,220,132
Non-trainable params: 0
Total mult-adds (Units.MEGABYTES): 555.48
==========================================================================
==========
Input size (MB): 0.01
Forward/backward pass size (MB): 9.83
Params size (MB): 44.88
Estimated Total Size (MB): 54.72
==========================================================================
==========
Number of parameters: 11,220,132
Estimated FLOPs per forward pass: 555,478,500

Training: 100%|     | 782/782 [00:15<00:00, 50.75it/s, loss=3.6675,
acc=13.47%]
Evaluating: 100%|     | 157/157 [00:01<00:00, 147.98it/s, loss=3.2582,
acc=20.97%]

Epoch 1/30, Train Loss: 3.6675, Train Acc: 13.47%, Test Loss: 3.2582, Test Acc:
20.97%, Time: 15.41s

Training: 100%|     | 782/782 [00:15<00:00, 50.85it/s, loss=2.8093,
acc=28.06%]
Evaluating: 100%|     | 157/157 [00:01<00:00, 132.33it/s, loss=2.6548,
acc=31.94%]

Epoch 2/30, Train Loss: 2.8093, Train Acc: 28.06%, Test Loss: 2.6548, Test Acc:
31.94%, Time: 15.38s

Training: 100%|     | 782/782 [00:15<00:00, 51.88it/s, loss=2.2971,
acc=38.68%]
Evaluating: 100%|     | 157/157 [00:01<00:00, 142.54it/s, loss=2.2716,
acc=40.37%]
```

```
Epoch 3/30, Train Loss: 2.2971, Train Acc: 38.68%, Test Loss: 2.2716, Test Acc:
40.37%, Time: 15.07s

Training: 100%|     | 782/782 [00:15<00:00, 50.96it/s, loss=1.9459,
acc=46.47%]
Evaluating: 100%|     | 157/157 [00:01<00:00, 134.95it/s, loss=1.9328,
acc=48.26%]

Epoch 4/30, Train Loss: 1.9459, Train Acc: 46.47%, Test Loss: 1.9328, Test Acc:
48.26%, Time: 15.35s

Training: 100%|     | 782/782 [00:17<00:00, 44.44it/s, loss=1.7067,
acc=52.16%]
Evaluating: 100%|     | 157/157 [00:01<00:00, 125.40it/s, loss=1.8149,
acc=50.83%]

Epoch 5/30, Train Loss: 1.7067, Train Acc: 52.16%, Test Loss: 1.8149, Test Acc:
50.83%, Time: 17.60s

Training: 100%|     | 782/782 [00:16<00:00, 46.80it/s, loss=1.5265,
acc=56.56%]
Evaluating: 100%|     | 157/157 [00:01<00:00, 140.69it/s, loss=1.6401,
acc=54.78%]

Epoch 6/30, Train Loss: 1.5265, Train Acc: 56.56%, Test Loss: 1.6401, Test Acc:
54.78%, Time: 16.71s

Training: 100%|     | 782/782 [00:15<00:00, 52.09it/s, loss=1.3795,
acc=60.27%]
Evaluating: 100%|     | 157/157 [00:01<00:00, 143.54it/s, loss=1.5255,
acc=57.67%]

Epoch 7/30, Train Loss: 1.3795, Train Acc: 60.27%, Test Loss: 1.5255, Test Acc:
57.67%, Time: 15.01s

Training: 100%|     | 782/782 [00:14<00:00, 52.14it/s, loss=1.2550,
acc=63.48%]
Evaluating: 100%|     | 157/157 [00:01<00:00, 146.26it/s, loss=1.4952,
acc=59.21%]

Epoch 8/30, Train Loss: 1.2550, Train Acc: 63.48%, Test Loss: 1.4952, Test Acc:
59.21%, Time: 15.00s

Training: 100%|     | 782/782 [00:15<00:00, 52.06it/s, loss=1.1427,
acc=66.26%]
Evaluating: 100%|     | 157/157 [00:01<00:00, 146.19it/s, loss=1.4326,
acc=60.48%]

Epoch 9/30, Train Loss: 1.1427, Train Acc: 66.26%, Test Loss: 1.4326, Test Acc:
60.48%, Time: 15.02s

Training: 100%|     | 782/782 [00:15<00:00, 51.69it/s, loss=1.0480,
acc=68.70%]
```

```
Evaluating: 100%|      | 157/157 [00:01<00:00, 143.21it/s, loss=1.3494,
acc=63.26%]

Epoch 10/30, Train Loss: 1.0480, Train Acc: 68.70%, Test Loss: 1.3494, Test Acc:
63.26%, Time: 15.13s

Training: 100%|      | 782/782 [00:15<00:00, 51.08it/s, loss=0.9605,
acc=70.91%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 131.85it/s, loss=1.3277,
acc=63.85%]

Epoch 11/30, Train Loss: 0.9605, Train Acc: 70.91%, Test Loss: 1.3277, Test Acc:
63.85%, Time: 15.31s

Training: 100%|      | 782/782 [00:16<00:00, 48.38it/s, loss=0.8730,
acc=73.37%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 129.39it/s, loss=1.3069,
acc=64.49%]

Epoch 12/30, Train Loss: 0.8730, Train Acc: 73.37%, Test Loss: 1.3069, Test Acc:
64.49%, Time: 16.17s

Training: 100%|      | 782/782 [00:16<00:00, 48.77it/s, loss=0.7916,
acc=75.58%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 141.27it/s, loss=1.3999,
acc=63.95%]

Epoch 13/30, Train Loss: 0.7916, Train Acc: 75.58%, Test Loss: 1.3999, Test Acc:
63.95%, Time: 16.04s

Training: 100%|      | 782/782 [00:14<00:00, 52.33it/s, loss=0.7234,
acc=77.55%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 141.92it/s, loss=1.3015,
acc=66.04%]

Epoch 14/30, Train Loss: 0.7234, Train Acc: 77.55%, Test Loss: 1.3015, Test Acc:
66.04%, Time: 14.94s

Training: 100%|      | 782/782 [00:15<00:00, 51.40it/s, loss=0.6519,
acc=79.35%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 142.97it/s, loss=1.3629,
acc=65.36%]

Epoch 15/30, Train Loss: 0.6519, Train Acc: 79.35%, Test Loss: 1.3629, Test Acc:
65.36%, Time: 15.22s

Training: 100%|      | 782/782 [00:15<00:00, 51.15it/s, loss=0.5944,
acc=81.11%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 128.06it/s, loss=1.3980,
acc=65.43%]

Epoch 16/30, Train Loss: 0.5944, Train Acc: 81.11%, Test Loss: 1.3980, Test Acc:
65.43%, Time: 15.29s
```

```
Training: 100%|      | 782/782 [00:16<00:00, 46.40it/s, loss=0.5416,
acc=82.56%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 131.26it/s, loss=1.3809,
acc=66.44%]

Epoch 17/30, Train Loss: 0.5416, Train Acc: 82.56%, Test Loss: 1.3809, Test Acc:
66.44%, Time: 16.86s

Training: 100%|      | 782/782 [00:16<00:00, 48.43it/s, loss=0.4918,
acc=84.12%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 117.71it/s, loss=1.3767,
acc=67.10%]

Epoch 18/30, Train Loss: 0.4918, Train Acc: 84.12%, Test Loss: 1.3767, Test Acc:
67.10%, Time: 16.15s

Training: 100%|      | 782/782 [00:16<00:00, 46.48it/s, loss=0.4396,
acc=85.83%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 133.46it/s, loss=1.5155,
acc=65.61%]

Epoch 19/30, Train Loss: 0.4396, Train Acc: 85.83%, Test Loss: 1.5155, Test Acc:
65.61%, Time: 16.83s

Training: 100%|      | 782/782 [00:16<00:00, 48.15it/s, loss=0.3978,
acc=86.97%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 141.60it/s, loss=1.4619,
acc=66.81%]

Epoch 20/30, Train Loss: 0.3978, Train Acc: 86.97%, Test Loss: 1.4619, Test Acc:
66.81%, Time: 16.24s

Training: 100%|      | 782/782 [00:15<00:00, 52.07it/s, loss=0.3722,
acc=87.76%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 140.70it/s, loss=1.4725,
acc=68.05%]

Epoch 21/30, Train Loss: 0.3722, Train Acc: 87.76%, Test Loss: 1.4725, Test Acc:
68.05%, Time: 15.02s

Training: 100%|      | 782/782 [00:15<00:00, 49.45it/s, loss=0.3341,
acc=89.15%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 143.66it/s, loss=1.5085,
acc=67.39%]

Epoch 22/30, Train Loss: 0.3341, Train Acc: 89.15%, Test Loss: 1.5085, Test Acc:
67.39%, Time: 15.82s

Training: 100%|      | 782/782 [00:15<00:00, 49.75it/s, loss=0.3074,
acc=89.83%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 142.68it/s, loss=1.5694,
acc=66.46%]
```

```
Epoch 23/30, Train Loss: 0.3074, Train Acc: 89.83%, Test Loss: 1.5694, Test Acc:
66.46%, Time: 15.72s

Training: 100%|      | 782/782 [00:16<00:00, 48.41it/s, loss=0.2810,
acc=90.70%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 137.02it/s, loss=1.6118,
acc=67.21%]

Epoch 24/30, Train Loss: 0.2810, Train Acc: 90.70%, Test Loss: 1.6118, Test Acc:
67.21%, Time: 16.15s

Training: 100%|      | 782/782 [00:15<00:00, 49.56it/s, loss=0.2639,
acc=91.39%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 142.74it/s, loss=1.6592,
acc=67.13%]

Epoch 25/30, Train Loss: 0.2639, Train Acc: 91.39%, Test Loss: 1.6592, Test Acc:
67.13%, Time: 15.78s

Training: 100%|      | 782/782 [00:14<00:00, 52.31it/s, loss=0.2473,
acc=91.80%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 144.64it/s, loss=1.7298,
acc=66.70%]

Epoch 26/30, Train Loss: 0.2473, Train Acc: 91.80%, Test Loss: 1.7298, Test Acc:
66.70%, Time: 14.95s

Training: 100%|      | 782/782 [00:15<00:00, 50.85it/s, loss=0.2318,
acc=92.31%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 142.44it/s, loss=1.7080,
acc=66.66%]

Epoch 27/30, Train Loss: 0.2318, Train Acc: 92.31%, Test Loss: 1.7080, Test Acc:
66.66%, Time: 15.38s

Training: 100%|      | 782/782 [00:14<00:00, 52.34it/s, loss=0.2139,
acc=92.83%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 145.11it/s, loss=1.7011,
acc=67.28%]

Epoch 28/30, Train Loss: 0.2139, Train Acc: 92.83%, Test Loss: 1.7011, Test Acc:
67.28%, Time: 14.94s

Training: 100%|      | 782/782 [00:14<00:00, 52.38it/s, loss=0.1996,
acc=93.33%]
Evaluating: 100%|      | 157/157 [00:01<00:00, 142.62it/s, loss=1.7256,
acc=67.39%]

Epoch 29/30, Train Loss: 0.1996, Train Acc: 93.33%, Test Loss: 1.7256, Test Acc:
67.39%, Time: 14.93s

Training: 100%|      | 782/782 [00:14<00:00, 52.63it/s, loss=0.1896,
acc=93.71%]
```

```
Evaluating: 100%|          | 157/157 [00:01<00:00, 145.08it/s, loss=1.7530,
acc=67.92%]

Epoch 30/30, Train Loss: 0.1896, Train Acc: 93.71%, Test Loss: 1.7530, Test Acc:
67.92%, Time: 14.86s

Final Results:
Average training time per epoch: 15.61s
Final test accuracy: 67.92%
```