

## ✓ Colab training notebook - Diabetic Retinopathy Classification

This notebook runs the training code from the repository on Google Colab (GPU runtime). It supports both ResNet50 and Vision Transformer (ViT) models.

### Model Options:

- `resnet50`: ResNet50 backbone (baseline)
- `vit_base_patch16_224`: Vision Transformer Base (recommended for better accuracy)
- `vit_small_patch16_224`: Vision Transformer Small (faster, less memory)
- `vit_tiny_patch16_224`: Vision Transformer Tiny (fastest, least memory)

### Notes:

- Make sure your dataset folder has `dr_labels.csv` and a `DR_images/` subfolder.
- You can either upload the `data/` folder to Colab session storage, or mount Google Drive and point `--data-dir` to a folder on Drive.
- If you prefer to run from a GitHub repo, upload this workspace to a public GitHub and use the git clone cell below.
- The code now includes improved data augmentation for better accuracy.

```
from google.colab import drive
drive.mount('/content/drive')
```

Drive already mounted at /content/drive; to attempt to forcibly remount, call `drive.mount("/content/drive", force_remount=True)`

```
%cd /content
```

```
/content
```

```
mkdir /content/Diabetic-Retinopathy
```

```
# Install PyTorch with CUDA support
!pip install -q torch torchvision --extra-index-url https://download.pytorch.org/whl/cu118
```

```
# Install other requirements (includes timm for Vision Transformer)
```

```
!pip install -q -r requirements_colab.txt
```

```
print("Installation complete!")
```

```
!rm -rf /content/Diabetic-Retinopathy/  
!git clone https://github.com/Ojasvsakhi/Diabetic-Retinopathy.git /content/Diabetic-Retinopathy  
%cd /content/Diabetic-Retinopathy
```

```
import os  
DATA_DIR = '/content/drive/MyDrive/DR Dataset'  
if not os.path.exists(DATA_DIR):  
    print('Warning: expected DATA_DIR not found:', DATA_DIR)  
else:  
    print('Using DATA_DIR =', DATA_DIR)
```

```
Cloning into '/content/Diabetic-Retinopathy'...  
remote: Enumerating objects: 142, done.  
remote: Counting objects: 100% (142/142), done.  
remote: Compressing objects: 100% (99/99), done.  
remote: Total 142 (delta 81), reused 95 (delta 40), pack-reused 0 (from 0)  
Receiving objects: 100% (142/142), 239.18 KiB | 6.83 MiB/s, done.  
Resolving deltas: 100% (81/81), done.  
/content/Diabetic-Retinopathy  
Using DATA_DIR = /content/drive/MyDrive/DR Dataset
```

```
import sys, os  
repo_src = os.path.join(os.getcwd(), 'src')  
if os.path.exists(repo_src):  
    sys.path.insert(0, repo_src)  
else:  
    sys.path.append('src')  
  
from argparse import Namespace  
from src.train import train  
  
# ===== CONFIGURATION =====  
# Choose your model: 'resnet50', 'vit_base_patch16_224', 'vit_small_patch16_224', 'vit_tiny_patch16_224'  
MODEL_NAME = 'vit_base_patch16_224' # Change this to switch models  
  
# Training hyperparameters  
EPOCHS = 20
```

```

BATCH_SIZE = 16 # Reduce to 8 if you run out of memory with ViT
IMG_SIZE = 224
LEARNING_RATE = 1e-4
NUM_WORKERS = 2

# ===== TRAINING =====
args = Namespace(
    data_dir=DATA_DIR,
    epochs=EPOCHS,
    batch_size=BATCH_SIZE,
    img_size=IMG_SIZE,
    lr=LEARNING_RATE,
    num_workers=NUM_WORKERS,
    model_name=MODEL_NAME
)

print(f"Starting training with model: {MODEL_NAME}")
print(f"Batch size: {BATCH_SIZE}, Epochs: {EPOCHS}, Learning rate: {LEARNING_RATE}")

# Training
train(args)
Epoch 3/20 [train]: 100%|██████████| 25/25 [00:38<00:00, 1.54s/it, loss=1.14]
Validation: 100%|██████████| 7/7 [00:10<00:00, 1.44s/it]
Epoch 3 validation -- acc: 0.5051 macro-F1: 0.4948
Epoch 4/20 [train]: 100%|██████████| 25/25 [00:38<00:00, 1.55s/it, loss=0.775]
Validation: 100%|██████████| 7/7 [00:09<00:00, 1.37s/it]
Epoch 4 validation -- acc: 0.5960 macro-F1: 0.5678
Epoch 5/20 [train]: 100%|██████████| 25/25 [00:37<00:00, 1.50s/it, loss=0.521]
Validation: 100%|██████████| 7/7 [00:09<00:00, 1.42s/it]
Epoch 5 validation -- acc: 0.5354 macro-F1: 0.4115
Epoch 6/20 [train]: 100%|██████████| 25/25 [00:38<00:00, 1.55s/it, loss=0.57]

```

```

Epoch 10 validation -- acc: 0.6061 macro-F1: 0.5535
Epoch 11/20 [train]: 100%|██████████| 25/25 [00:38<00:00, 1.55s/it, loss=0.197]
Validation: 100%|██████████| 7/7 [00:08<00:00, 1.23s/it]
Epoch 11 validation -- acc: 0.5758 macro-F1: 0.5167
Learning rates reduced: [0.0001] -> [5e-05]
Epoch 12/20 [train]: 100%|██████████| 25/25 [00:39<00:00, 1.56s/it, loss=0.195]
Validation: 100%|██████████| 7/7 [00:09<00:00, 1.41s/it]
Epoch 12 validation -- acc: 0.6566 macro-F1: 0.6200
Epoch 13/20 [train]: 100%|██████████| 25/25 [00:37<00:00, 1.52s/it, loss=0.124]
Validation: 100%|██████████| 7/7 [00:08<00:00, 1.26s/it]
Epoch 13 validation -- acc: 0.6364 macro-F1: 0.6035
Epoch 14/20 [train]: 100%|██████████| 25/25 [00:39<00:00, 1.56s/it, loss=0.104]
Validation: 100%|██████████| 7/7 [00:10<00:00, 1.45s/it]
Epoch 14 validation -- acc: 0.5960 macro-F1: 0.5508
Epoch 15/20 [train]: 100%|██████████| 25/25 [00:39<00:00, 1.58s/it, loss=0.0866]
Validation: 100%|██████████| 7/7 [00:08<00:00, 1.21s/it]
Epoch 15 validation -- acc: 0.6566 macro-F1: 0.6131
Learning rates reduced: [5e-05] -> [2.5e-05]
Epoch 16/20 [train]: 100%|██████████| 25/25 [00:38<00:00, 1.55s/it, loss=0.0433]
Validation: 100%|██████████| 7/7 [00:10<00:00, 1.43s/it]
Epoch 16 validation -- acc: 0.6465 macro-F1: 0.6043
Epoch 17/20 [train]: 100%|██████████| 25/25 [00:39<00:00, 1.56s/it, loss=0.0754]
Validation: 100%|██████████| 7/7 [00:08<00:00, 1.21s/it]
Epoch 17 validation -- acc: 0.6263 macro-F1: 0.5782
Epoch 18/20 [train]: 100%|██████████| 25/25 [00:38<00:00, 1.54s/it, loss=0.0592]
Validation: 100%|██████████| 7/7 [00:09<00:00, 1.42s/it]
Epoch 18 validation -- acc: 0.6364 macro-F1: 0.5816
Learning rates reduced: [2.5e-05] -> [1.25e-05]
Epoch 19/20 [train]: 100%|██████████| 25/25 [00:39<00:00, 1.56s/it, loss=0.0379]
Validation: 100%|██████████| 7/7 [00:09<00:00, 1.31s/it]
Epoch 19 validation -- acc: 0.6768 macro-F1: 0.6198
Epoch 20/20 [train]: 100%|██████████| 25/25 [00:37<00:00, 1.51s/it, loss=0.0343]
Validation: 100%|██████████| 7/7 [00:10<00:00, 1.44s/it]Epoch 20 validation -- acc: 0.6465 macro-F1: 0.5901

```