

bare_model

April 25, 2025

1 Doing Normalization and Setting up The batch size.

```
[3]: from torchvision import transforms
      from torchvision.datasets import ImageFolder
      from torch.utils.data import DataLoader

      # Define normalization transformations
      val_transform = transforms.Compose([
          transforms.ToTensor(), # Convert image to tensor
          transforms.Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.
          ↪225]), # Normalize
      ])

      # Load datasets with normalization (no augmentation)
      train_dataset = ImageFolder("processed_data/train", transform=val_transform)
      val_dataset = ImageFolder("processed_data/val", transform=val_transform)
      test_dataset = ImageFolder("processed_data/test", transform=val_transform)

      # Create data loaders
      train_loader = DataLoader(train_dataset, batch_size=32, shuffle=True)
      val_loader = DataLoader(val_dataset, batch_size=32)
      test_loader = DataLoader(test_dataset, batch_size=32)

      # Verify dataset sizes
      print(f"Training dataset size: {len(train_dataset)} images")
      print(f"Validation dataset size: {len(val_dataset)} images")
      print(f"Test dataset size: {len(test_dataset)} images")
```

Training dataset size: 4800 images
Validation dataset size: 600 images
Test dataset size: 600 images

```
[5]: import torch
      import torch.nn as nn
      import torch.optim as optim
      from torch.utils.data import DataLoader
      from torchvision import datasets, transforms
      import matplotlib.pyplot as plt
```

```
import timm
import os
```

2 Define the bare EfficientNet-B0 model

```
[28]: class BareEfficientNetB0(nn.Module):
        def __init__(self, num_classes=4):
            super().__init__()
            self.model = timm.create_model("efficientnet_b0", pretrained=True,
            ↪ num_classes=num_classes)

        def forward(self, x):
            return self.model(x)
```

```
[30]: # Training setup model 1
import torch.optim as optim
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
bare2 = BareEfficientNetB0(num_classes=4).to(device)
criterion = nn.CrossEntropyLoss()
optimizer = optim.Adam(bare2.parameters(), lr=0.001)

print("DONE")
```

DONE

```
[33]: # model 2
print("Starting")
num_epochs = 20
for epoch in range(num_epochs):
    bare2.train()
    running_loss = 0.0
    for images, labels in train_loader:
        images, labels = images.to(device), labels.to(device)
        optimizer.zero_grad()
        outputs = bare2(images)
        loss = criterion(outputs, labels)
        loss.backward()
        optimizer.step()
        running_loss += loss.item()
    print(f"Epoch {epoch+1}, Loss: {running_loss/len(train_loader)}")

    # Validation
    bare2.eval()
    correct = 0
    total = 0
    with torch.no_grad():
        for images, labels in val_loader:
```

```

        images, labels = images.to(device), labels.to(device)
        outputs = bare2(images)
        _, predicted = torch.max(outputs, 1)
        total += labels.size(0)
        correct += (predicted == labels).sum().item()
    accuracy = 100 * correct / total
    print(f"Validation Accuracy: {accuracy:.2f}%")

```

Starting

```

Epoch 1, Loss: 0.07188148938447739
Validation Accuracy: 99.67%
Epoch 2, Loss: 0.11954412172675348
Validation Accuracy: 99.67%
Epoch 3, Loss: 0.09582909294365284
Validation Accuracy: 99.67%
Epoch 4, Loss: 0.026464333012966867
Validation Accuracy: 100.00%
Epoch 5, Loss: 0.00844317286483753
Validation Accuracy: 100.00%
Epoch 6, Loss: 0.014769503286235401
Validation Accuracy: 99.83%
Epoch 7, Loss: 0.01976312504391558
Validation Accuracy: 99.83%
Epoch 8, Loss: 0.019273661120255525
Validation Accuracy: 100.00%
Epoch 9, Loss: 0.07283284577810263
Validation Accuracy: 99.50%
Epoch 10, Loss: 0.10914428449061234
Validation Accuracy: 99.83%
Epoch 11, Loss: 0.016352600168453742
Validation Accuracy: 100.00%
Epoch 12, Loss: 0.01425813367928034
Validation Accuracy: 99.83%
Epoch 13, Loss: 0.02178748345256281
Validation Accuracy: 100.00%
Epoch 14, Loss: 0.0065334327878463225
Validation Accuracy: 99.67%
Epoch 15, Loss: 0.005221256741663941
Validation Accuracy: 100.00%
Epoch 16, Loss: 0.006534091316225386
Validation Accuracy: 100.00%
Epoch 17, Loss: 0.004397848067418788
Validation Accuracy: 99.83%
Epoch 18, Loss: 0.005037086614725771
Validation Accuracy: 99.17%
Epoch 19, Loss: 0.014025342957597027
Validation Accuracy: 95.83%
Epoch 20, Loss: 0.04287365573836041

```

Validation Accuracy: 97.17%

```
[35]: # Save Bare EfficientNet-B0 model
torch.save(bare2.state_dict(), 'efficientnet_b0_bare_2.pth')
print("Model saved as 'efficientnet_b0_bare_2.pth'")
```

Model saved as 'efficientnet_b0_bare_2.pth'

```
[36]: import torch

# Assuming `model` is your trained or defined model
num_params = sum(p.numel() for p in bare2.parameters())
print(f"Total number of parameters: {num_params}")
```

Total number of parameters: 4012672

```
[37]: pip install torchsummary
```

Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: torchsummary in
/home/rjaswal1634/.local/lib/python3.12/site-packages (1.5.1)
Note: you may need to restart the kernel to use updated packages.

```
[38]: from torchsummary import summary

# Assuming `model` is your trained or defined model
summary(bare2, (3, 224, 224)) # (3, 224, 224) for a 224x224 RGB image
```

Layer (type)	Output Shape	Param #
Conv2d-1	[-1, 32, 112, 112]	864
Identity-2	[-1, 32, 112, 112]	0
SiLU-3	[-1, 32, 112, 112]	0
BatchNormAct2d-4	[-1, 32, 112, 112]	64
Conv2d-5	[-1, 32, 112, 112]	288
Identity-6	[-1, 32, 112, 112]	0
SiLU-7	[-1, 32, 112, 112]	0
BatchNormAct2d-8	[-1, 32, 112, 112]	64
Identity-9	[-1, 32, 112, 112]	0
Conv2d-10	[-1, 8, 1, 1]	264
SiLU-11	[-1, 8, 1, 1]	0
Conv2d-12	[-1, 32, 1, 1]	288
Sigmoid-13	[-1, 32, 1, 1]	0
SqueezeExcite-14	[-1, 32, 112, 112]	0
Conv2d-15	[-1, 16, 112, 112]	512
Identity-16	[-1, 16, 112, 112]	0
Identity-17	[-1, 16, 112, 112]	0
BatchNormAct2d-18	[-1, 16, 112, 112]	32
DepthwiseSeparableConv-19	[-1, 16, 112, 112]	0

Conv2d-20	[-1, 96, 112, 112]	1,536
Identity-21	[-1, 96, 112, 112]	0
SiLU-22	[-1, 96, 112, 112]	0
BatchNormAct2d-23	[-1, 96, 112, 112]	192
Conv2d-24	[-1, 96, 56, 56]	864
Identity-25	[-1, 96, 56, 56]	0
SiLU-26	[-1, 96, 56, 56]	0
BatchNormAct2d-27	[-1, 96, 56, 56]	192
Identity-28	[-1, 96, 56, 56]	0
Conv2d-29	[-1, 4, 1, 1]	388
SiLU-30	[-1, 4, 1, 1]	0
Conv2d-31	[-1, 96, 1, 1]	480
Sigmoid-32	[-1, 96, 1, 1]	0
SqueezeExcite-33	[-1, 96, 56, 56]	0
Conv2d-34	[-1, 24, 56, 56]	2,304
Identity-35	[-1, 24, 56, 56]	0
Identity-36	[-1, 24, 56, 56]	0
BatchNormAct2d-37	[-1, 24, 56, 56]	48
InvertedResidual-38	[-1, 24, 56, 56]	0
Conv2d-39	[-1, 144, 56, 56]	3,456
Identity-40	[-1, 144, 56, 56]	0
SiLU-41	[-1, 144, 56, 56]	0
BatchNormAct2d-42	[-1, 144, 56, 56]	288
Conv2d-43	[-1, 144, 56, 56]	1,296
Identity-44	[-1, 144, 56, 56]	0
SiLU-45	[-1, 144, 56, 56]	0
BatchNormAct2d-46	[-1, 144, 56, 56]	288
Identity-47	[-1, 144, 56, 56]	0
Conv2d-48	[-1, 6, 1, 1]	870
SiLU-49	[-1, 6, 1, 1]	0
Conv2d-50	[-1, 144, 1, 1]	1,008
Sigmoid-51	[-1, 144, 1, 1]	0
SqueezeExcite-52	[-1, 144, 56, 56]	0
Conv2d-53	[-1, 24, 56, 56]	3,456
Identity-54	[-1, 24, 56, 56]	0
Identity-55	[-1, 24, 56, 56]	0
BatchNormAct2d-56	[-1, 24, 56, 56]	48
Identity-57	[-1, 24, 56, 56]	0
InvertedResidual-58	[-1, 24, 56, 56]	0
Conv2d-59	[-1, 144, 56, 56]	3,456
Identity-60	[-1, 144, 56, 56]	0
SiLU-61	[-1, 144, 56, 56]	0
BatchNormAct2d-62	[-1, 144, 56, 56]	288
Conv2d-63	[-1, 144, 28, 28]	3,600
Identity-64	[-1, 144, 28, 28]	0
SiLU-65	[-1, 144, 28, 28]	0
BatchNormAct2d-66	[-1, 144, 28, 28]	288
Identity-67	[-1, 144, 28, 28]	0

Conv2d-68	[-1, 6, 1, 1]	870
SiLU-69	[-1, 6, 1, 1]	0
Conv2d-70	[-1, 144, 1, 1]	1,008
Sigmoid-71	[-1, 144, 1, 1]	0
SqueezeExcite-72	[-1, 144, 28, 28]	0
Conv2d-73	[-1, 40, 28, 28]	5,760
Identity-74	[-1, 40, 28, 28]	0
Identity-75	[-1, 40, 28, 28]	0
BatchNormAct2d-76	[-1, 40, 28, 28]	80
InvertedResidual-77	[-1, 40, 28, 28]	0
Conv2d-78	[-1, 240, 28, 28]	9,600
Identity-79	[-1, 240, 28, 28]	0
SiLU-80	[-1, 240, 28, 28]	0
BatchNormAct2d-81	[-1, 240, 28, 28]	480
Conv2d-82	[-1, 240, 28, 28]	6,000
Identity-83	[-1, 240, 28, 28]	0
SiLU-84	[-1, 240, 28, 28]	0
BatchNormAct2d-85	[-1, 240, 28, 28]	480
Identity-86	[-1, 240, 28, 28]	0
Conv2d-87	[-1, 10, 1, 1]	2,410
SiLU-88	[-1, 10, 1, 1]	0
Conv2d-89	[-1, 240, 1, 1]	2,640
Sigmoid-90	[-1, 240, 1, 1]	0
SqueezeExcite-91	[-1, 240, 28, 28]	0
Conv2d-92	[-1, 40, 28, 28]	9,600
Identity-93	[-1, 40, 28, 28]	0
Identity-94	[-1, 40, 28, 28]	0
BatchNormAct2d-95	[-1, 40, 28, 28]	80
Identity-96	[-1, 40, 28, 28]	0
InvertedResidual-97	[-1, 40, 28, 28]	0
Conv2d-98	[-1, 240, 28, 28]	9,600
Identity-99	[-1, 240, 28, 28]	0
SiLU-100	[-1, 240, 28, 28]	0
BatchNormAct2d-101	[-1, 240, 28, 28]	480
Conv2d-102	[-1, 240, 14, 14]	2,160
Identity-103	[-1, 240, 14, 14]	0
SiLU-104	[-1, 240, 14, 14]	0
BatchNormAct2d-105	[-1, 240, 14, 14]	480
Identity-106	[-1, 240, 14, 14]	0
Conv2d-107	[-1, 10, 1, 1]	2,410
SiLU-108	[-1, 10, 1, 1]	0
Conv2d-109	[-1, 240, 1, 1]	2,640
Sigmoid-110	[-1, 240, 1, 1]	0
SqueezeExcite-111	[-1, 240, 14, 14]	0
Conv2d-112	[-1, 80, 14, 14]	19,200
Identity-113	[-1, 80, 14, 14]	0
Identity-114	[-1, 80, 14, 14]	0
BatchNormAct2d-115	[-1, 80, 14, 14]	160

InvertedResidual-116	[-1, 80, 14, 14]	0
Conv2d-117	[-1, 480, 14, 14]	38,400
Identity-118	[-1, 480, 14, 14]	0
SiLU-119	[-1, 480, 14, 14]	0
BatchNormAct2d-120	[-1, 480, 14, 14]	960
Conv2d-121	[-1, 480, 14, 14]	4,320
Identity-122	[-1, 480, 14, 14]	0
SiLU-123	[-1, 480, 14, 14]	0
BatchNormAct2d-124	[-1, 480, 14, 14]	960
Identity-125	[-1, 480, 14, 14]	0
Conv2d-126	[-1, 20, 1, 1]	9,620
SiLU-127	[-1, 20, 1, 1]	0
Conv2d-128	[-1, 480, 1, 1]	10,080
Sigmoid-129	[-1, 480, 1, 1]	0
SqueezeExcite-130	[-1, 480, 14, 14]	0
Conv2d-131	[-1, 80, 14, 14]	38,400
Identity-132	[-1, 80, 14, 14]	0
Identity-133	[-1, 80, 14, 14]	0
BatchNormAct2d-134	[-1, 80, 14, 14]	160
Identity-135	[-1, 80, 14, 14]	0
InvertedResidual-136	[-1, 80, 14, 14]	0
Conv2d-137	[-1, 480, 14, 14]	38,400
Identity-138	[-1, 480, 14, 14]	0
SiLU-139	[-1, 480, 14, 14]	0
BatchNormAct2d-140	[-1, 480, 14, 14]	960
Conv2d-141	[-1, 480, 14, 14]	4,320
Identity-142	[-1, 480, 14, 14]	0
SiLU-143	[-1, 480, 14, 14]	0
BatchNormAct2d-144	[-1, 480, 14, 14]	960
Identity-145	[-1, 480, 14, 14]	0
Conv2d-146	[-1, 20, 1, 1]	9,620
SiLU-147	[-1, 20, 1, 1]	0
Conv2d-148	[-1, 480, 1, 1]	10,080
Sigmoid-149	[-1, 480, 1, 1]	0
SqueezeExcite-150	[-1, 480, 14, 14]	0
Conv2d-151	[-1, 80, 14, 14]	38,400
Identity-152	[-1, 80, 14, 14]	0
Identity-153	[-1, 80, 14, 14]	0
BatchNormAct2d-154	[-1, 80, 14, 14]	160
Identity-155	[-1, 80, 14, 14]	0
InvertedResidual-156	[-1, 80, 14, 14]	0
Conv2d-157	[-1, 480, 14, 14]	38,400
Identity-158	[-1, 480, 14, 14]	0
SiLU-159	[-1, 480, 14, 14]	0
BatchNormAct2d-160	[-1, 480, 14, 14]	960
Conv2d-161	[-1, 480, 14, 14]	12,000
Identity-162	[-1, 480, 14, 14]	0
SiLU-163	[-1, 480, 14, 14]	0

BatchNormAct2d-164	[-1, 480, 14, 14]	960
Identity-165	[-1, 480, 14, 14]	0
Conv2d-166	[-1, 20, 1, 1]	9,620
SiLU-167	[-1, 20, 1, 1]	0
Conv2d-168	[-1, 480, 1, 1]	10,080
Sigmoid-169	[-1, 480, 1, 1]	0
SqueezeExcite-170	[-1, 480, 14, 14]	0
Conv2d-171	[-1, 112, 14, 14]	53,760
Identity-172	[-1, 112, 14, 14]	0
Identity-173	[-1, 112, 14, 14]	0
BatchNormAct2d-174	[-1, 112, 14, 14]	224
InvertedResidual-175	[-1, 112, 14, 14]	0
Conv2d-176	[-1, 672, 14, 14]	75,264
Identity-177	[-1, 672, 14, 14]	0
SiLU-178	[-1, 672, 14, 14]	0
BatchNormAct2d-179	[-1, 672, 14, 14]	1,344
Conv2d-180	[-1, 672, 14, 14]	16,800
Identity-181	[-1, 672, 14, 14]	0
SiLU-182	[-1, 672, 14, 14]	0
BatchNormAct2d-183	[-1, 672, 14, 14]	1,344
Identity-184	[-1, 672, 14, 14]	0
Conv2d-185	[-1, 28, 1, 1]	18,844
SiLU-186	[-1, 28, 1, 1]	0
Conv2d-187	[-1, 672, 1, 1]	19,488
Sigmoid-188	[-1, 672, 1, 1]	0
SqueezeExcite-189	[-1, 672, 14, 14]	0
Conv2d-190	[-1, 112, 14, 14]	75,264
Identity-191	[-1, 112, 14, 14]	0
Identity-192	[-1, 112, 14, 14]	0
BatchNormAct2d-193	[-1, 112, 14, 14]	224
Identity-194	[-1, 112, 14, 14]	0
InvertedResidual-195	[-1, 112, 14, 14]	0
Conv2d-196	[-1, 672, 14, 14]	75,264
Identity-197	[-1, 672, 14, 14]	0
SiLU-198	[-1, 672, 14, 14]	0
BatchNormAct2d-199	[-1, 672, 14, 14]	1,344
Conv2d-200	[-1, 672, 14, 14]	16,800
Identity-201	[-1, 672, 14, 14]	0
SiLU-202	[-1, 672, 14, 14]	0
BatchNormAct2d-203	[-1, 672, 14, 14]	1,344
Identity-204	[-1, 672, 14, 14]	0
Conv2d-205	[-1, 28, 1, 1]	18,844
SiLU-206	[-1, 28, 1, 1]	0
Conv2d-207	[-1, 672, 1, 1]	19,488
Sigmoid-208	[-1, 672, 1, 1]	0
SqueezeExcite-209	[-1, 672, 14, 14]	0
Conv2d-210	[-1, 112, 14, 14]	75,264
Identity-211	[-1, 112, 14, 14]	0

Identity-212	[-1, 112, 14, 14]	0
BatchNormAct2d-213	[-1, 112, 14, 14]	224
Identity-214	[-1, 112, 14, 14]	0
InvertedResidual-215	[-1, 112, 14, 14]	0
Conv2d-216	[-1, 672, 14, 14]	75,264
Identity-217	[-1, 672, 14, 14]	0
SiLU-218	[-1, 672, 14, 14]	0
BatchNormAct2d-219	[-1, 672, 14, 14]	1,344
Conv2d-220	[-1, 672, 7, 7]	16,800
Identity-221	[-1, 672, 7, 7]	0
SiLU-222	[-1, 672, 7, 7]	0
BatchNormAct2d-223	[-1, 672, 7, 7]	1,344
Identity-224	[-1, 672, 7, 7]	0
Conv2d-225	[-1, 28, 1, 1]	18,844
SiLU-226	[-1, 28, 1, 1]	0
Conv2d-227	[-1, 672, 1, 1]	19,488
Sigmoid-228	[-1, 672, 1, 1]	0
SqueezeExcite-229	[-1, 672, 7, 7]	0
Conv2d-230	[-1, 192, 7, 7]	129,024
Identity-231	[-1, 192, 7, 7]	0
Identity-232	[-1, 192, 7, 7]	0
BatchNormAct2d-233	[-1, 192, 7, 7]	384
InvertedResidual-234	[-1, 192, 7, 7]	0
Conv2d-235	[-1, 1152, 7, 7]	221,184
Identity-236	[-1, 1152, 7, 7]	0
SiLU-237	[-1, 1152, 7, 7]	0
BatchNormAct2d-238	[-1, 1152, 7, 7]	2,304
Conv2d-239	[-1, 1152, 7, 7]	28,800
Identity-240	[-1, 1152, 7, 7]	0
SiLU-241	[-1, 1152, 7, 7]	0
BatchNormAct2d-242	[-1, 1152, 7, 7]	2,304
Identity-243	[-1, 1152, 7, 7]	0
Conv2d-244	[-1, 48, 1, 1]	55,344
SiLU-245	[-1, 48, 1, 1]	0
Conv2d-246	[-1, 1152, 1, 1]	56,448
Sigmoid-247	[-1, 1152, 1, 1]	0
SqueezeExcite-248	[-1, 1152, 7, 7]	0
Conv2d-249	[-1, 192, 7, 7]	221,184
Identity-250	[-1, 192, 7, 7]	0
Identity-251	[-1, 192, 7, 7]	0
BatchNormAct2d-252	[-1, 192, 7, 7]	384
Identity-253	[-1, 192, 7, 7]	0
InvertedResidual-254	[-1, 192, 7, 7]	0
Conv2d-255	[-1, 1152, 7, 7]	221,184
Identity-256	[-1, 1152, 7, 7]	0
SiLU-257	[-1, 1152, 7, 7]	0
BatchNormAct2d-258	[-1, 1152, 7, 7]	2,304
Conv2d-259	[-1, 1152, 7, 7]	28,800

Identity-260	[-1, 1152, 7, 7]	0
SiLU-261	[-1, 1152, 7, 7]	0
BatchNormAct2d-262	[-1, 1152, 7, 7]	2,304
Identity-263	[-1, 1152, 7, 7]	0
Conv2d-264	[-1, 48, 1, 1]	55,344
SiLU-265	[-1, 48, 1, 1]	0
Conv2d-266	[-1, 1152, 1, 1]	56,448
Sigmoid-267	[-1, 1152, 1, 1]	0
SqueezeExcite-268	[-1, 1152, 7, 7]	0
Conv2d-269	[-1, 192, 7, 7]	221,184
Identity-270	[-1, 192, 7, 7]	0
Identity-271	[-1, 192, 7, 7]	0
BatchNormAct2d-272	[-1, 192, 7, 7]	384
Identity-273	[-1, 192, 7, 7]	0
InvertedResidual-274	[-1, 192, 7, 7]	0
Conv2d-275	[-1, 1152, 7, 7]	221,184
Identity-276	[-1, 1152, 7, 7]	0
SiLU-277	[-1, 1152, 7, 7]	0
BatchNormAct2d-278	[-1, 1152, 7, 7]	2,304
Conv2d-279	[-1, 1152, 7, 7]	28,800
Identity-280	[-1, 1152, 7, 7]	0
SiLU-281	[-1, 1152, 7, 7]	0
BatchNormAct2d-282	[-1, 1152, 7, 7]	2,304
Identity-283	[-1, 1152, 7, 7]	0
Conv2d-284	[-1, 48, 1, 1]	55,344
SiLU-285	[-1, 48, 1, 1]	0
Conv2d-286	[-1, 1152, 1, 1]	56,448
Sigmoid-287	[-1, 1152, 1, 1]	0
SqueezeExcite-288	[-1, 1152, 7, 7]	0
Conv2d-289	[-1, 192, 7, 7]	221,184
Identity-290	[-1, 192, 7, 7]	0
Identity-291	[-1, 192, 7, 7]	0
BatchNormAct2d-292	[-1, 192, 7, 7]	384
Identity-293	[-1, 192, 7, 7]	0
InvertedResidual-294	[-1, 192, 7, 7]	0
Conv2d-295	[-1, 1152, 7, 7]	221,184
Identity-296	[-1, 1152, 7, 7]	0
SiLU-297	[-1, 1152, 7, 7]	0
BatchNormAct2d-298	[-1, 1152, 7, 7]	2,304
Conv2d-299	[-1, 1152, 7, 7]	10,368
Identity-300	[-1, 1152, 7, 7]	0
SiLU-301	[-1, 1152, 7, 7]	0
BatchNormAct2d-302	[-1, 1152, 7, 7]	2,304
Identity-303	[-1, 1152, 7, 7]	0
Conv2d-304	[-1, 48, 1, 1]	55,344
SiLU-305	[-1, 48, 1, 1]	0
Conv2d-306	[-1, 1152, 1, 1]	56,448
Sigmoid-307	[-1, 1152, 1, 1]	0

SqueezeExcite-308	[-1, 1152, 7, 7]	0	
Conv2d-309	[-1, 320, 7, 7]	368,640	
Identity-310	[-1, 320, 7, 7]	0	
Identity-311	[-1, 320, 7, 7]	0	
BatchNormAct2d-312	[-1, 320, 7, 7]	640	
InvertedResidual-313	[-1, 320, 7, 7]	0	
Conv2d-314	[-1, 1280, 7, 7]	409,600	
Identity-315	[-1, 1280, 7, 7]	0	
SiLU-316	[-1, 1280, 7, 7]	0	
BatchNormAct2d-317	[-1, 1280, 7, 7]	2,560	
AdaptiveAvgPool2d-318	[-1, 1280, 1, 1]	0	
Flatten-319	[-1, 1280]	0	
SelectAdaptivePool2d-320	[-1, 1280]	0	0
Linear-321	[-1, 4]	5,124	
EfficientNet-322	[-1, 4]	0	

=====

Total params: 4,012,672

Trainable params: 4,012,672

Non-trainable params: 0

Input size (MB): 0.57

Forward/backward pass size (MB): 247.01

Params size (MB): 15.31

Estimated Total Size (MB): 262.90

[]: