Image Preprocessing

Bert Gollnick

Image Preprocessing

- Original image
- Represented by pixels
- Dimensions: (C, H, W)...color, height, width

3 for color image
1 for grayscale

number of
horizontal pixels

number of
vertical pixels

- Pixels are represented as numerical values with [0, ..., 255]
- Image conversion required by PyTorch
- Augmentations can improve model performance

Original Image

Image Preprocessing

- Scales an image (usually down); identical tensor shapes required for all images

Original Image

Processed Image
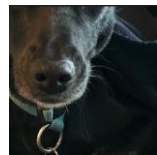
Image Preprocessing

- crops an image at the center

Original Image

Processed Image

Image Preprocessing

- creates grayscale image, reduces color channel dimension from 3 to 1

Original Image

Processed Image

Image Preprocessing

- randomly rotates image within bounds, usually combined with crop.

Original Image

Processed Image

Image Preprocessing

- flips an image with given probability

Original Image

Processed Image

Image Preprocessing

- converts a PIL image to tensor with dimensions (C, H, W) and value range [0.0, 1.0]
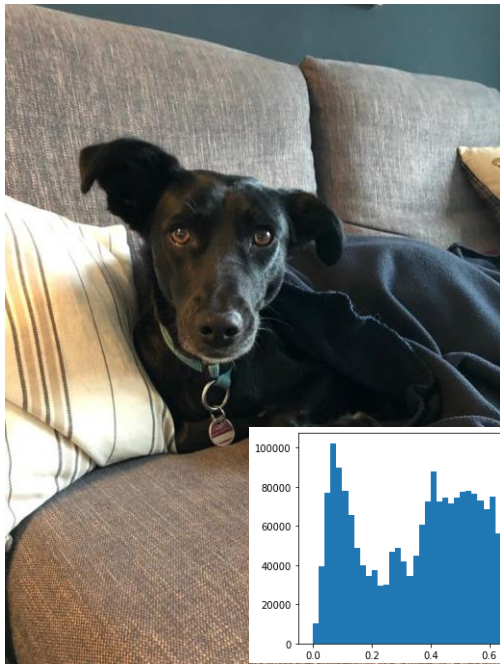

Original Image

Processed Image

Image Preprocessing

- normalizes an image with mean and standard deviation

Batch of images
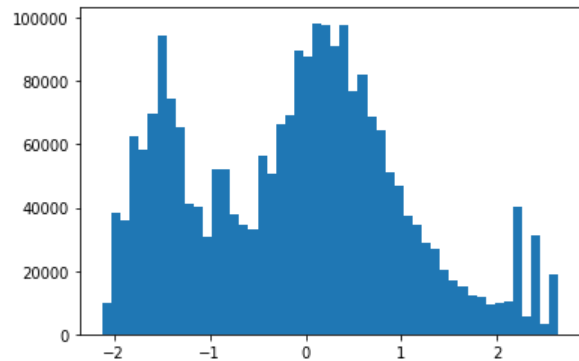e.g. ImageNet
mean = (0.485, 0.456, 0.406)
Sd = (0.229, 0.224, 0.225)

Image is centered around given mean and has given standard deviation.

Original Image (created from tensor)

Processed Image

Image Preprocessing

- Performs several transformations at once
- composed steps are applied to all images

```python
preprocess_steps = transforms.Compose([
    transforms.Resize(300),   # better (300, 300)
    transforms.RandomRotation(50),
    transforms.CenterCrop(500),
    transforms.Grayscale(),
    transforms.RandomVerticalFlip(),
    transforms.ToTensor(),
    transforms.Normalize((0.485, 0.456, 0.406), (0.229, 0.224, 0.225)),
])
x = preprocess_steps(img)
```

Let's find out in coding lecture.