The Effectiveness of Data Augmentation in Image Classification using Deep Learning

Extension of AutoAugment

Combining Two Images

Extension of AutoAugment

Related Word / Background

Affine Transformation == geometric and color augmentations:
reflect, crop, translate, change color palette

Applied affine transformation at the feature map of each layer

GAN to generate new images: counterfeit vs distinguish

CycleGAN for style transfer: day → night drive

Augmentation types

1. Traditional Transformation

Shifted, zoomed in/out, rotated, flipped, distorted, shaded with hue

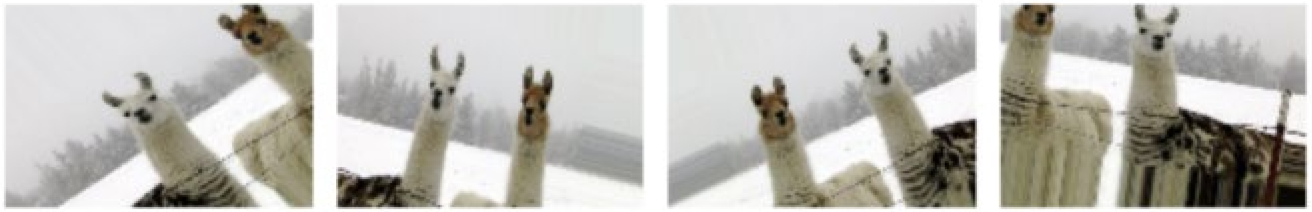

Figure I: Traditional Transformations

Augmentation types

2. GAN

Style transfer: Cezanne, Enhance, Monet, Ukiyoe, Van Gogh, Winter

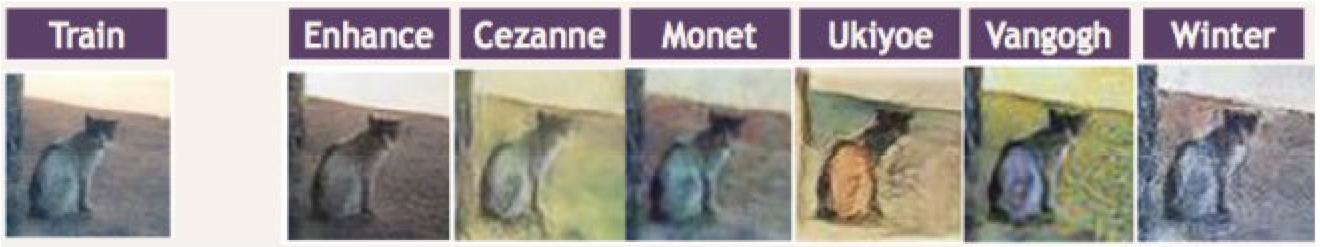

Figure II: Style Transformations via GANs

Augmentation types

3. Augmentation Network

1 + 1 = 3 + augmentation loss

Figure VI: Goldfish sample II

Figure VII: Dog sample I

Augmentation types

3. Augmentation Network

1 + 1 = 3 + augmentation loss

Figure VI: Goldfish sample II

Figure VII: Dog sample I

Augmentation Network

1. Conv with 16 channels and 3x3 filters. Relu activations.
2. Conv with 16 channels and 3x3 filters. Relu activations.
3. Conv with 16 channels and 3x3 filters. Relu activations.
4. Conv with 16 channels and 3x3 filters. Relu activations.
5. Conv with 3 channels and 3x3 filters.

Dataset

8 : 2 = train : valid

tiny-imagenet-200

500 dogs
500 cats
500 goldfish

→ 80k training images

MNIST

1k 0's
1k 8's

→ 640k training images

Experiment Setting

10 experiment

40 epochs

0.0001 Adam Opt

SmallNet

Specific net is not very important

Any net that can reliably predict the classes suffices

Replace SmallNet → VGG16

SmallNet

1. Conv with 16 channels and 3x3 filters. Relu activations.
2. Batch normalization.
3. Max pooling with 2x2 filters and 2x2 stride.
4. Conv with 32 channels and 3x3 filters. Relu activations.
5. Conv with 32 channels and 3x3 filters. Relu activations.
6. Batch normalization.
7. Max pooling with 2x2 filters and 2x2 stride.
8. Fully connected with output dimension 1024. Dropout.
9. Fully connected layer with output dimension 2.

Loss

$$\alpha L_c + \beta L_a$$

$$L_a^{content} = \frac{1}{D^2}\sum_{i,j}(A_{ij} - T_{ij})$$

$$L_a^{style} = \frac{1}{C^2}\sum_{i,j}(G_{ij}^A - G_{ij}^T)$$

$$G_{ij} = \sum_k F_{ik}F_{jk}$$

Note:

Content loss never converged

But

Act as regularization so the "Augmentation net" doesn't generate images too different from original images

Overview

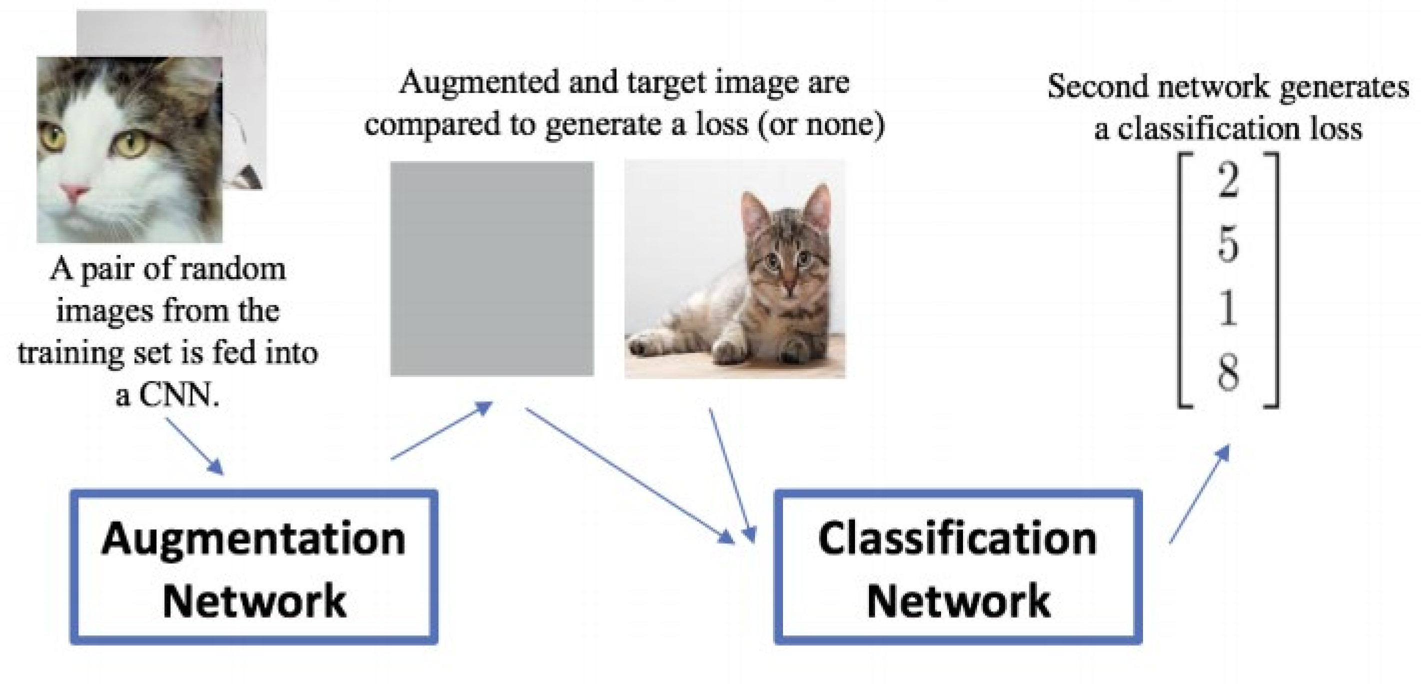

Figure III: Training model

Overview

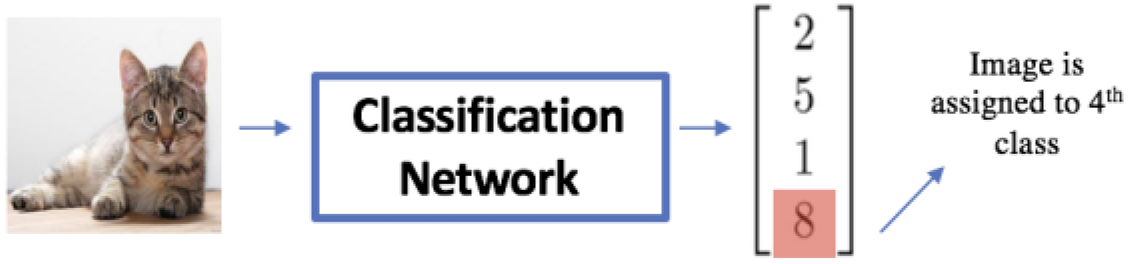

Figure IV: Testing/Validation model

Result

And… additional points:

MNIST == structured data
ImageNet == unstructured data

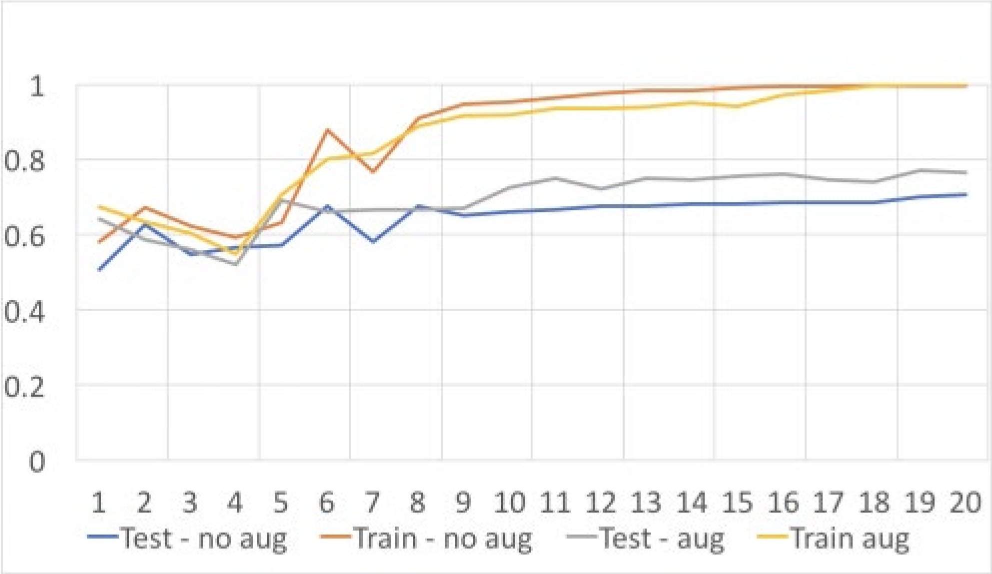

Figure XI: Accuracy plots

Dogs vs Goldfish	
Augmentation	Val. Acc.
None	0.855
Traditional	0.890
GANs	0.865
Neural + No Loss	**0.915**
Neural + Content Loss	0.900
Neural + Style	0.890
Control	0.840

Table I: Quantitative Results on Dogs vs Goldfish

Dogs vs Cat	
Augmentation	Val. Acc.
None	0.705
Traditional	**0.775**
GANs	0.720
Neural + No Loss	0.765
Neural + Content Loss	0.770
Neural + Style	0.740
Control	0.710

Table II: Quantitative Results on Dogs vs Cats

MNIST 0's and 8's	
Augmentation	Val. Acc.
None	0.972
Neural + No Loss	**0.975**
Neural + Content Loss	0.968

Table III: MNIST