AutoAugment: Learning Augmentation Strategies from Data

Recent trends in Automated Machine Learning (AutoML)
Technische Universität München
Bikem Çamli
June 2nd, 2021

OUTLINE

- Problem Definition
- Related Works
- Implementation
- Experiments and Results
- AutoAugment-Transfer

AUGMENTATION

Improve the accuracy of the classifiers

Increase the amount and diversity of data

Teach a model about invariances

PROBLEM DEFINITION

The best augmentation methods needs to be designed manually and they are dataset-specific

As these methods are designed manually, they require expert knowledge and time.

«For instance, on ImageNet, the data augmentation approach is introduced in 2012, remains the standard with small changes.»

RELATED WORKS

Smart Augmentation

Merges multiple samples from the same class to produce augmented data

Bayesian Approach

Generates data based on the distribution learned from the training set

Devries and Taylor

Used simple transformations in the learned feature space to
augment data

GANs - Ratner

Generate sequences that describe data augmentation strategies

IMPLEMENTATION

SEARCH ALGORITHM

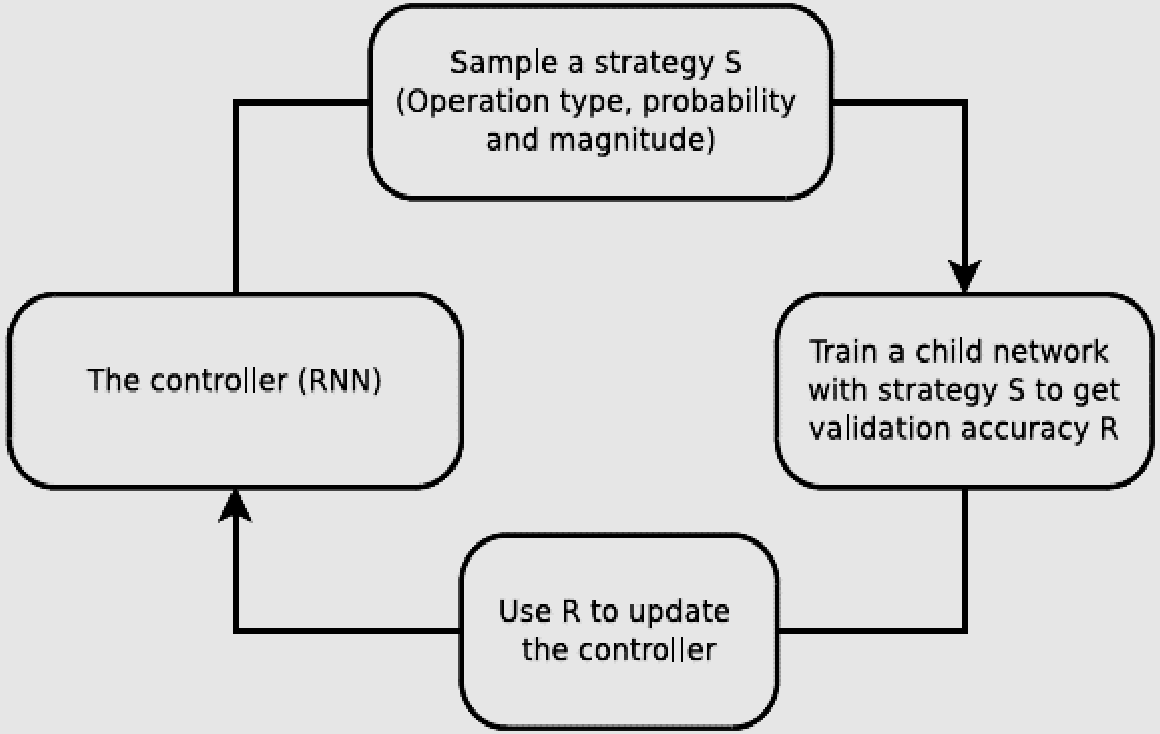
- A controller - a recurrent neural network,
- A training algorithm - Proximal Policy Optimization algorithm

SEARCH SPACE

- A policy consists of 5 sub-policies with two image operations to be applied in sequence.
- Each operation is with two hyperparameters:
  - the probability of applying the operation,
  - the magnitude of the operation.

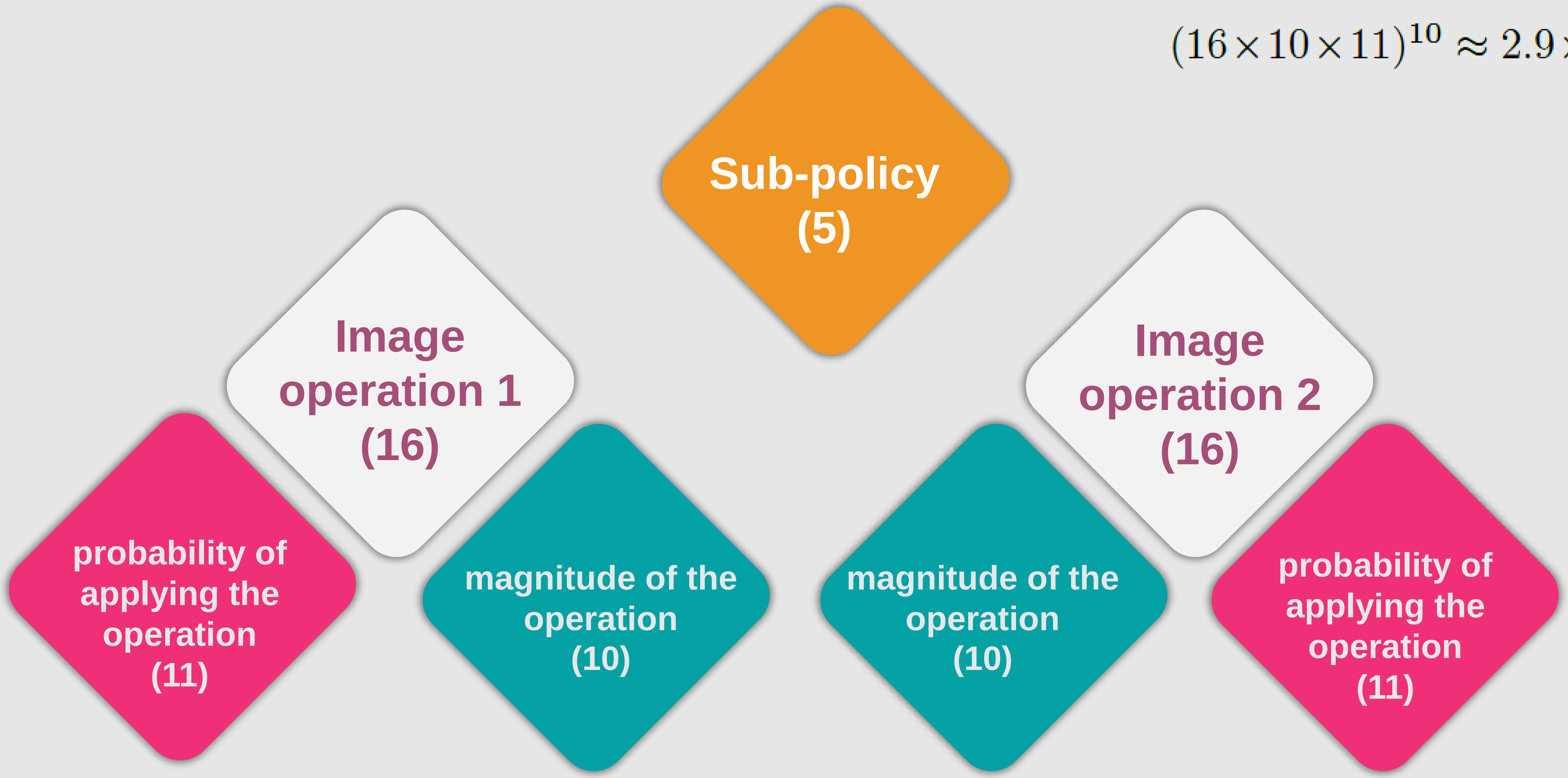
SEARCH ALGORITHM

- A controller RNN predicts an augmentation policy (can be viewed as a list of actions to design an architecture for a child network) from the search space.
- These actions applied to the dataset (environment) and a child network is trained until the convergence, achieving accuracy R.
- Autoaugment use this accuracy R as the reward signal with the policy gradient method (since the reward signal R is non-differentiable) to update the controller to generate better policies in next iteration.

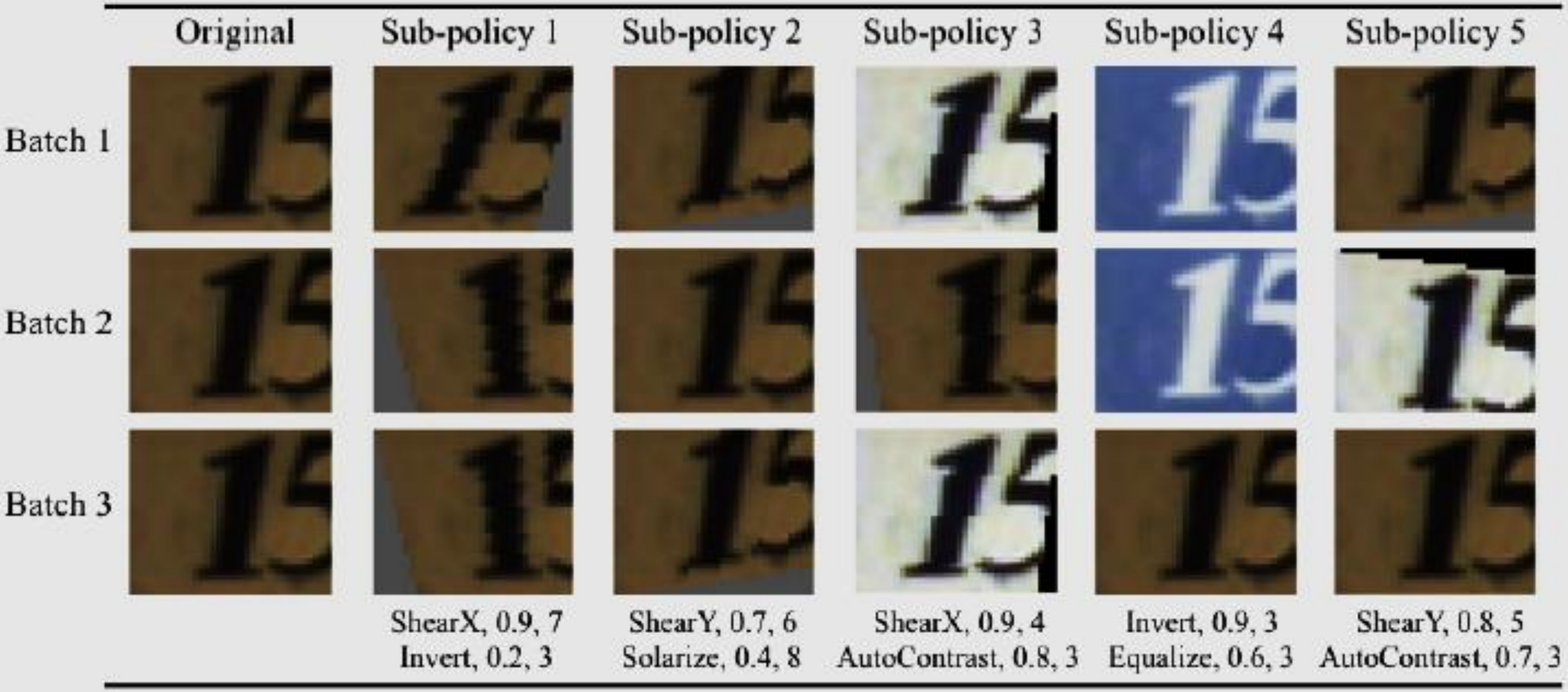


SEARCH SPACE

$$(16 \times 10 \times 11)^{10} \approx 2.9 \times 10^{32}$$



Augmentation Example



Operations

PIL + Cutout [12] and SamplePairing

Operation Name	Description	Range of magnitudes
ShearX(Y)	Shear the image along the horizontal (vertical) axis with rate *magnitude*.	[-0.3,0.3]
TranslateX(Y)	Translate the image in the horizontal (vertical) direction by *magnitude* number of pixels.	[-150,150]
Rotate	Rotate the image *magnitude* degrees.	[-30,30]
AutoContrast	Maximize the the image contrast, by making the darkest pixel black and lightest pixel white.	
Invert	Invert the pixels of the image.	
Equalize	Equalize the image histogram.	
Solarize	Invert all pixels above a threshold value of *magnitude*.	[0,256]
Posterize	Reduce the number of bits for each pixel to *magnitude* bits.	[4,8]
Contrast	Control the contrast of the image. A *magnitude*=0 gives a gray image, whereas *magnitude*=1 gives the original image.	[0.1,1.9]
Color	Adjust the color balance of the image, in a manner similar to the controls on a colour TV set. A *magnitude*=0 gives a black & white image, whereas *magnitude*=1 gives the original image.	[0.1,1.9]
Brightness	Adjust the brightness of the image. A *magnitude*=0 gives a black image, whereas *magnitude*=1 gives the original image.	[0.1,1.9]
Sharpness	Adjust the sharpness of the image. A *magnitude*=0 gives a blurred image, whereas *magnitude*=1 gives the original image.	[0.1,1.9]
Cutout [12, 69]	Set a random square patch of side-length *magnitude* pixels to gray.	[0,60]
Sample Pairing [24, 68]	Linearly add the image with another image (selected at random from the same mini-batch) with weight *magnitude*, without changing the label.	[0, 0.4]

EXPERIMENT DETAILS

- Reduced Datasets
- More Epochs
- Stochasticity
- Baseline Pre-processing

Results

Dataset	Model	Baseline	Cutout [12]	AutoAugment
CIFAR-10	Wide-ResNet-28-10 [67]	3.9	3.1	2.6±0.1
	Shake-Shake (26 2x32d) [17]	3.6	3.0	2.5±0.1
	Shake-Shake (26 2x96d) [17]	2.9	2.6	2.0±0.1
	Shake-Shake (26 2x112d) [17]	2.8	2.6	1.9±0.1
	AmoebaNet-B (6,128) [48]	3.0	2.1	1.8±0.1
	PyramidNet+ShakeDrop [65]	2.7	2.3	**1.5 ± 0.1**
Reduced CIFAR-10	Wide-ResNet-28-10 [67]	18.8	16.5	14.1±0.3
	Shake-Shake (26 2x96d) [17]	17.1	13.4	**10.0 ± 0.2**
CIFAR-100	Wide-ResNet-28-10 [67]	18.8	18.4	17.1±0.3
	Shake-Shake (26 2x96d) [17]	17.1	16.0	14.3±0.2
	PyramidNet+ShakeDrop [65]	14.0	12.2	**10.7 ± 0.2**
SVHN	Wide-ResNet-28-10 [67]	1.5	1.3	1.1
	Shake-Shake (26 2x96d) [17]	1.4	1.2	**1.0**
Reduced SVHN	Wide-ResNet-28-10 [67]	13.2	32.5	8.2
	Shake-Shake (26 2x96d) [17]	12.3	24.2	**5.9**

«The results show that a direct application of AutoAugment improves significantly the baseline models and produces state-of-the-art accuracies on these challenging datasets.»

AutoAugment vs. Related Works

Method	Baseline	Augmented	Improvement Δ
LSTM [47]	7.7	6.0	1.6
MF [47]	7.7	5.6	2.1
AutoAugment (ResNet-32)	7.7	4.5	**3.2**
AutoAugment (ResNet-56)	6.6	3.6	**3.0**

Results for CIFAR-10 and ImageNet

	Operation 1	Operation 2
Sub-policy 0	(Invert,0.1,7)	(Contrast,0.2,6)
Sub-policy 1	(Rotate,0.7,2)	(TranslateX,0.3,9)
Sub-policy 2	(Sharpness,0.8,1)	(Sharpness,0.9,3)
Sub-policy 3	(ShearY,0.5,8)	(TranslateY,0.7,9)
Sub-policy 4	(AutoContrast,0.5,8)	(Equalize,0.9,2)
Sub-policy 5	(ShearY,0.2,7)	(Posterize,0.3,7)
Sub-policy 6	(Color,0.4,3)	(Brightness,0.6,7)
Sub-policy 7	(Sharpness,0.3,9)	(Brightness,0.7,9)
Sub-policy 8	(Equalize,0.6,5)	(Equalize,0.5,1)
Sub-policy 9	(Contrast,0.6,7)	(Sharpness,0.6,5)
Sub-policy 10	(Color,0.7,7)	(TranslateX,0.5,8)
Sub-policy 11	(Equalize,0.3,7)	(AutoContrast,0.4,8)
Sub-policy 12	(TranslateY,0.4,3)	(Sharpness,0.2,6)
Sub-policy 13	(Brightness,0.9,6)	(Color,0.2,8)
Sub-policy 14	(Solarize,0.5,2)	(Invert,0.0,3)
Sub-policy 15	(Equalize,0.2,0)	(AutoContrast,0.6,0)
Sub-policy 16	(Equalize,0.2,8)	(Equalize,0.6,4)
Sub-policy 17	(Color,0.9,9)	(Equalize,0.6,6)
Sub-policy 18	(AutoContrast,0.8,4)	(Solarize,0.2,8)
Sub-policy 19	(Brightness,0.1,3)	(Color,0.7,0)
Sub-policy 20	(Solarize,0.4,5)	(AutoContrast,0.9,3)
Sub-policy 21	(TranslateY,0.9,9)	(TranslateY,0.7,9)
Sub-policy 22	(AutoContrast,0.9,2)	(Solarize,0.8,3)
Sub-policy 23	(Equalize,0.8,8)	(Invert,0.1,3)
Sub-policy 24	(TranslateY,0.7,9)	(AutoContrast,0.9,1)

Table 7. AutoAugment policy found on reduced CIFAR-10.

	Operation 1	Operation 2
Sub-policy 0	(ShearX,0.9,4)	(Invert,0.2,3)
Sub-policy 1	(ShearY,0.9,8)	(Invert,0.7,5)
Sub-policy 2	(Equalize,0.6,5)	(Solarize,0.6,6)
Sub-policy 3	(Invert,0.9,3)	(Equalize,0.6,3)
Sub-policy 4	(Equalize,0.6,1)	(Rotate,0.9,3)
Sub-policy 5	(ShearX,0.9,4)	(AutoContrast,0.8,3)
Sub-policy 6	(ShearY,0.9,8)	(Invert,0.4,5)
Sub-policy 7	(ShearY,0.9,5)	(Solarize,0.2,6)
Sub-policy 8	(Invert,0.9,6)	(AutoContrast,0.8,1)
Sub-policy 9	(Equalize,0.6,3)	(Rotate,0.9,3)
Sub-policy 10	(ShearX,0.9,4)	(Solarize,0.3,3)
Sub-policy 11	(ShearY,0.8,8)	(Invert,0.7,4)
Sub-policy 12	(Equalize,0.9,5)	(TranslateY,0.6,6)
Sub-policy 13	(Invert,0.9,4)	(Equalize,0.6,7)
Sub-policy 14	(Contrast,0.3,3)	(Rotate,0.8,4)
Sub-policy 15	(Invert,0.8,5)	(TranslateY,0.0,2)
Sub-policy 16	(ShearY,0.7,6)	(Solarize,0.4,8)
Sub-policy 17	(Invert,0.6,4)	(Rotate,0.8,4)
Sub-policy 18	(ShearY,0.3,7)	(TranslateX,0.9,3)
Sub-policy 19	(ShearX,0.1,6)	(Invert,0.6,5)
Sub-policy 20	(Solarize,0.7,2)	(TranslateY,0.6,7)
Sub-policy 21	(ShearY,0.8,4)	(Invert,0.8,8)
Sub-policy 22	(ShearX,0.7,9)	(TranslateY,0.8,3)
Sub-policy 23	(ShearY,0.8,5)	(AutoContrast,0.7,3)
Sub-policy 24	(ShearX,0.7,2)	(Invert,0.1,5)

Table 8. AutoAugment policy found on reduced SVHN.

AutoAugment-Transfer

Transfer augmentation policies from one dataset to another

Clear evidence that AutoAugment does not “overfit”

Resource requirements

Transfer Results

Dataset	Train Size	Classes	Baseline	AutoAugment-transfer
Oxford 102 Flowers [43]	2,040	102	6.7	**4.6**
Caltech-101 [15]	3,060	102	19.4	**13.1**
Oxford-IIIT Pets [14]	3,680	37	13.5	**11.0**
FGVC Aircraft [38]	6,667	100	9.1	**7.3**
Stanford Cars [27]	8,144	196	6.4	**5.2**

«Policies learned on data distributions closest to the target yield the best performance»

Strengths and Weaknesses

NEGATIVE

- Requires high computational power
- Long computation time
- Dataset specific AutoAugment gives the best results

POSSITIVE

- Does not require expert knowledge
- Improvement in accuracies
- Transferable – does not overfit to a dataset

THANK YOU!

References

E. Cubuk, B. Zoph, D. Mane, V. Vasudevan, Q. Le. AutoAugment: Learning Augmentation Strategies From Data. 113-123. 10.1109/CVPR.2019.00020. 2019.
[14] Y. Em, F. Gag, Y. Lou, S. Wang, T. Huang, and L.-Y. Duan. Incorporating intra-class variance to fine-grained visual recognition. In Multimedia and Expo (ICME), 2017 IEEE International Conference on, pages 1452–1457. IEEE,2017.
[15] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. Computer vision and Image understanding, 106(1):59–70, 2007.
[17] X. Gastaldi. Shake-shake regularization. arXiv preprint arXiv:1705.07485, 2017.
[27] J. Krause, J. Deng, M. Stark, and L. Fei-Fei. Collecting a large-scale dataset of fine-grained cars. In Second Workshop on Fine-Grained Visual Categorization, 2013.
[38] S. Maji, E. Rahtu, J. Kannala, M. Blaschko, and A. Vedaldi.Fine-grained visual classification of aircraft. arXiv preprint arXiv:1306.5151, 2013.
[43] M.-E. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In Computer Vision, Graphics & Image Processing, 2008. ICVGIP'08. Sixth Indian Conference on, pages 722–729. IEEE, 2008.
[47] A. J. Ratner, H. Ehrenberg, Z. Hussain, J. Dunnmon, and C. R´e. Learning to compose domain-specific transformations for data augmentation. In Advances in Neural Information Processing Systems, pages 3239–3249, 2017.
[48] E. Real, A. Aggarwal, Y. Huang, and Q. V. Le. Regularized evolution for image classifier architecture search. arXiv preprint arXiv:1802.01548, 2018.
[65] Y. Yamada, M. Iwamura, and K. Kise. Shakedrop regularization. arXiv preprint arXiv:1802.02375, 2018.
[67] S. Zagoruyko and N. Komodakis. Wide residual networks. In British Machine Vision Conference, 2016.