Mix & Match: Training ConvNets With Mixed Image Sizes For Improved Accuracy, Speed And Scale Resiliency

Elad Hoffer[1], Berry Weinstein[1], Itay Hubara[1], Tal Ben-Nun[2], Torsten Hoefler[2], Daniel Soudry[3]
{ehoffer, bweinstein, ihubara}@habana.ai, daniel.soudry@gmail.com, {talbn,htor}@inf.ethz.ch
(1) Habana Labs (2) ETH Zurich (3) Technion - Israel Institute of Technology

Introduction

Convolutional neural networks (CNNs) are commonly trained using a fixed spatial image size predetermined for a given model. Although trained on images of a specific size, it is well established that CNNs can be used to evaluate a wide range of image sizes at test time, by adjusting the size of intermediate feature maps.

In this work, we describe and evaluate a novel mixed-size training regime that mixes several image sizes at training time. We demonstrate that models trained using our method are more resilient to image size changes and generalize well even on small images. This allows faster inference by using smaller images at test time. Furthermore, for a given image size used at test time, we show this method can either accelerate training, or improve the final test accuracy.

MixSize: Training with multiple image scales

We suggest "MixSize", a stochastic training regime, where input sizes can vary in each optimization step. In this regime, we modify the spatial dimensions H, W (height and width) of the input image size as well as the batch size. The batch size is changed either by the number of samples used, denoted B, or the number of batch-augmentations for each sample (Hoffer et al., 2019), denoted D ("duplicates"). We will follow the common practice of training on square images and use $S = H = W$. Formally, in the MixSize regime, these sizes can be described as random variables sharing a single discrete distribution

$$(\hat{S}, \hat{B}, \hat{D}) = \{(S, B, D)_i \quad w.p. \quad p_i\}, \quad \text{where} \quad \forall i : p_i \geq 0 \quad \text{and} \quad \sum_i p_i = 1.$$

As the computational cost of each training step is approximately proportional to $S^2 \cdot B \cdot D$, we choose these sizes to reflect an approximately fixed budget for any choice i such that $S_i^2 B_i D_i \approx Const$. Thus the computational and memory requirements for each step are constant.

Benefits and Trade-offs

Benefits. MixSize regime improves trained networks **resiliency to the image size used at evaluation**. That is, mixed-size networks will be shown to have better accuracy across a wide range of sizes. This entails a considerable saving in computations needed for inference, especially when using smaller models.

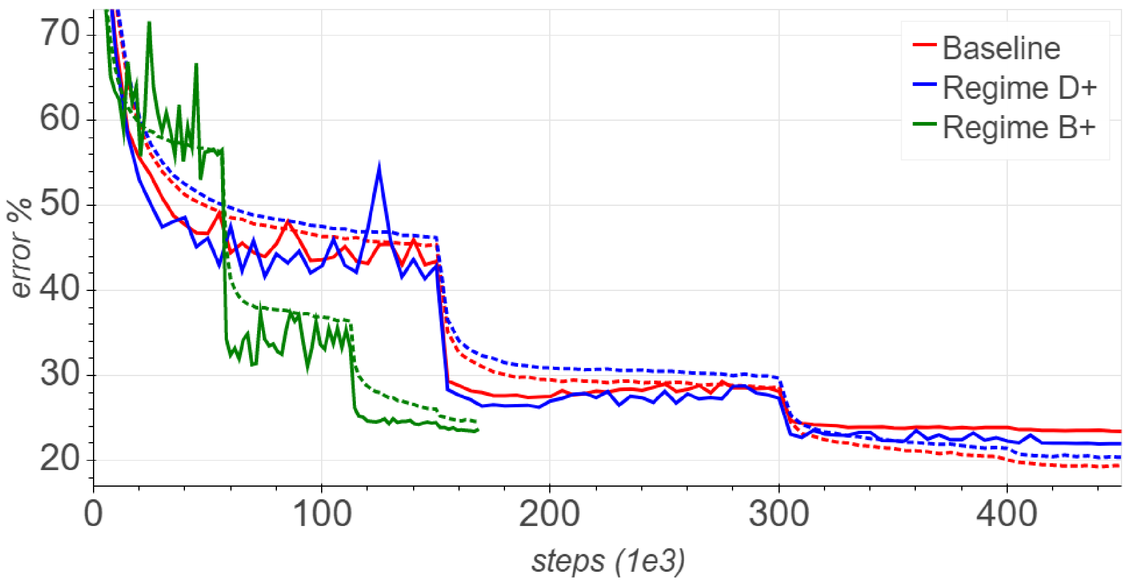
Tradeoffs. Given a fixed budget of computational and time resources (per step), we modify our regime along spatial and batch axes, yeilding two trade-offs:

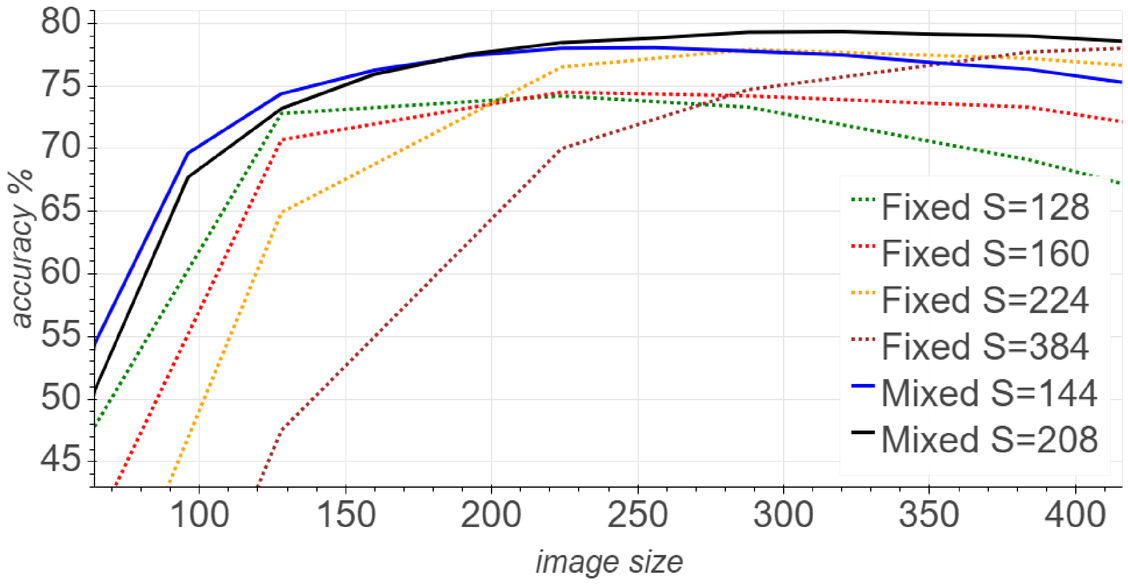
- **Decrease number of steps per epoch** – by enlarging B at the expense of S (denoted B^+).
- **Improve generalization per epoch** – by enlarging D at the expense of S (denoted D^+).

Experiments

We performed several experiments using the Cifar and ImageNet datasets to confirm our findings. For the ImageNet dataset, we use the following stochastic regime found by cross-validation on several alternatives. Our proposed regime makes for an average image size of $\bar{S} \times \bar{S} = 144 \times 144$ and designed so that the reduced spatial size can be used to increase the corresponding batch size or the number of BA duplicates.

$$S^{(144)}: \quad S = \begin{cases} 256, & \text{w.p} \quad p = 0.1 \\ 224, & \text{w.p} \quad p = 0.1 \\ 128, & \text{w.p} \quad p = 0.6 \\ 96, & \text{w.p} \quad p = 0.2 \end{cases}$$





Our MixSize regime yields a **76.43**% accuracy using ResNet50 with an image size of 160, which matches the accuracy of the baseline model with **2× fewer computations**. Furthermore, we are able to reach a **79.27**% accuracy when evaluating at a 288 spatial size for a relative improvement of 14% over baseline. These training speedups and accuracy gains were also observed in additional models and datasets:

Network	Dataset	Steps			Accuracy		
		Baseline	B^+	D^+	Baseline	B^+	D^+
ResNet-44	CIFAR10	156K	**109K**	156K	92.84%	94.30%	**94.37%**
WRN-28-10	CIFAR10	156K	**109K**	156K	96.60%	97.28%	**97.68%**
AmoebaNet	CIFAR10	469K	**328K**	469K	98.16%	98.14%	**98.32%**
ResNet-44	CIFAR100	156K	**109K**	156K	70.36%	72.19%	**73.10%**
WRN-28-10	CIFAR100	156K	**109K**	156K	79.85%	83.08%	**83.52%**
ResNet-50	ImageNet	450K	**169K**	450K	76.40%	76.61%	**78.04%**
EfficientNet-B0	ImageNet	1000K	**376K**	1000K	76.32%	76.29%	**76.53%**

References

Hoffer, E., Ben-Nun, T., Hubara, I., et al. Augment your batch: better training with larger batches. *arXiv preprint arXiv:1901.09335*, 2019.

Tan, M. and Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pp. 6105–6114, 2019.

Touvron, H., Vedaldi, A., Douze, M., and Jégou, H. Fixing the train-test resolution discrepancy. *CoRR*, abs/1906.06423, 2019.