TRAINING ACCURATE BINARY NEURAL NETWORKS FROM SCRATCH

Joseph Bethge, Haojin Yang, Christoph Meinel
Hasso Plattner Institute, University of Potsdam, Germany

INTRODUCTION

- Binary neural networks can run on **mobile** and **embedded** devices
- Can we train binary neural networks **without fine-tuning pretrained** full-precision models?
- How can we **adapt new architectures** for binary networks?

BINARY NEURAL NETWORK

- We use the sign function with a straight-through-estimator to binarize values, similar to [1,2,3]

$$\text{sign}(x) = \begin{cases} +1 \text{ if } x \geq 0, \\ -1 \text{ otherwise.} \end{cases}$$

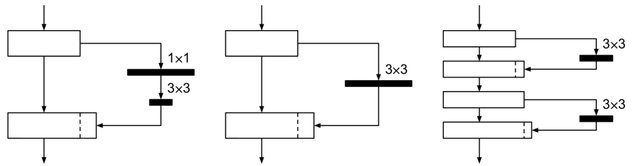
$$\text{Forward: } r_o = \text{sign}(r_i) \ .$$

$$\text{Backward: } \frac{\partial c}{\partial r_i} = \frac{\partial c}{\partial r_o} 1_{|r_i| \leq t_{\text{clip}}} \ .$$

- Gradient clipping threshold t_{clip} was often chosen as 1
- Prevents too large or too small absolute values but **reduces gradients available** during training

IDEA

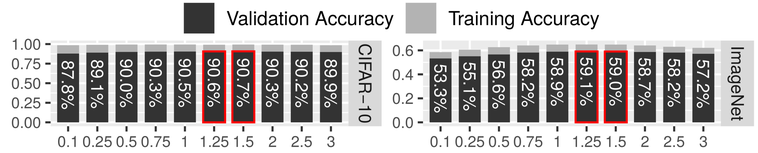
- Dense shortcut connections can reduce the information loss caused by binarization



(a) DenseNet (bottleneck) **(b)** DenseNet (no bottleneck) **(c)** BinaryDenseNet (our suggestion)

- Replace bottlenecks, since they reduce information
- Doubling the number of blocks but halving the size doubles the number of connections, see (b) → (c)
- Use full-precision for downsampling layers, since they have no shortcuts (low amount of weights)

GRADIENT CLIPPING THRESHOLD



- 1 is not optimal, 1.25 to 1.5 are performing better

RESULTS

- Comparison on ImageNet by top 1/top 5 accuracy
- Increasing number of connections is very efficient:

Blocks	Block Size	Model Size	Accuracy
8	256	3.31 MB	50.2%/73.7%
16	128	3.39 MB	52.7%/75.7%
32	64	3.45 MB	**54.3%/77.3%**

- Using full-precision for downsampling layers allows us to use higher reduction rates with accuracy gain:

Blocks, Block Size	Model Size	Downsampl., Reduction rate	Accuracy
16, 128	3.39 MB	binary, low	52.7%/75.7%
	3.03 MB	FP, high	**55.9%/78.5%**
32, 64	3.45 MB	binary, low	54.3%/77.3%
	3.08 MB	FP, high	**57.1%/80.0%**

- Comparison to state-of-the-art:

Approach	~4.0 MB ResNet-18	~5.1 MB ResNet-34
XNOR-Net [1]	51.2%/73.2%	-
TBN [2]	55.6%/74.2%	58.2%/81.0%
BiReal-Net[3]	56.4%/79.5%	62.2%/**83.9%**
Ours	**57.7%/80.0%**	60.2%/82.3%
Ours	**59.4%/81.5%** (BinaryDenseNet-21)	**62.4%/83.9%** (BinaryDenseNet-37)

CONCLUSION

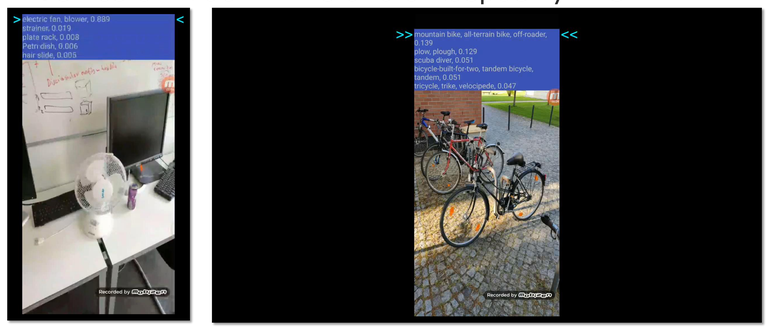
- Binary neural networks can be trained directly from scratch without finetuning
- Three techniques to optimize binary networks:
  - Remove bottlenecks
  - Use full-precision downsampling
  - Increase the number of connections
- Our *BinaryDenseNet* is based on these techniques and surpasses state-of-the-art accuracy

CODE

- BMXNet 2: an open-source framework for binary and quantized networks:
  - https://github.com/hpi-xnor/BMXNet-v2

DEMOS

- These efficient binary networks can run on low power devices in real-time
- Demos on Xiaomi Mi 8 and a Raspberry Pi 3:



REFERENCES

[1] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi, "Xnor-net: Imagenet classification using binary convolutional neural networks," in European Conference on Computer Vision. Springer, 2016, pp. 525–542.
[2] Diwen Wan, Fumin Shen, Li Liu, Fan Zhu, Jie Qin, Ling Shao, and Heng Tao Shen, "Tbn: Convolutional neural network with ternary inputs and binary weights," in The European Conference on Computer Vision (ECCV), September 2018.
[3] Zechun Liu, Baoyuan Wu, Wenhan Luo, Xin Yang, Wei Liu, and Kwang-Ting Cheng, "Bi-real net: Enhancing the performance of 1-bit cnns with improved representational capability and advanced training algorithm," in The European Conference on Computer Vision (ECCV), September 2018.