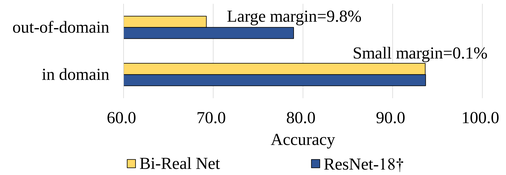
Domain Generalization Capability Enhancement for Binary Neural Networks

Jianming Ye[1], Shunan Mao[1], Shiliang Zhang[1]
[1]Institute of Digital Media, EECS, Peking University
jmye@pku.edu.cn, snmao@pku.edu.cn,slzhang.jdl@pku.edu.cn

Motivation

- The degraded modeling capability for visual cues of BNNs weakens their domain generalization capability.
- Domain Generalization (DG) capability is important for computer vision tasks, this paper aims to improve the DG capability of BNNs through enhancing their robustness to domain shifts..



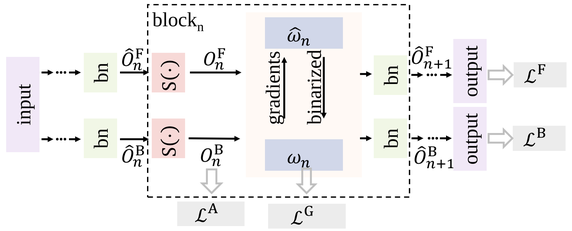
BNN and real-valued networks performance on in domain and out-of-domain dataset on PACS

Contribution

- **Weights**: We optimize the distributions to seek flat minima which show robustness to domain shift.
- **Activations**: We optimize the distributions by reducing quantization error and producing even-distribution activations to preserve the DG capability of real-valued activations.
- **Results**: Experiment results on different network architectures on both DG and traditional datasets show better performance comparing with recent DG methods and BNNs methods.

Network Architecture

The framework of our methods



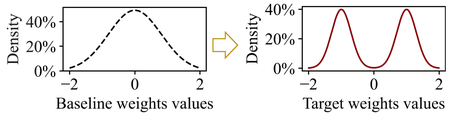
The overall loss

$$\mathcal{L} = \mathcal{L}^{\mathrm{B}} + \beta\mathcal{L}^{\mathrm{F}} + \alpha\mathcal{L}^{G} + \gamma\mathcal{L}^{A},$$

$\mathcal{L}^{B}$: the task specific loss calculated using binarized weights.
$\mathcal{L}^{F}$: the task specific loss calculated using real-valued weights with disturbance to find a flat minimum.
$\mathcal{L}^{G}$: the gap loss to measure the distance between binarized weights and real-valued weights.
$\mathcal{L}^{A}$: the activation regularization loss to optimize the distribution of activations.

Formulation

Flat Minima Optimization on BNN Weights



A solution to a flat minimum can be found by solving $\arg\min_{\hat{W}} \mathcal{L}^{\mathrm{F}}_{\mu}(\hat{W})$ within a neighborhoods bounded by μ, *i.e.*,

$$\arg\min_{\hat{W}} \mathcal{L}^{\mathrm{F}}_{\mu}(\hat{W}) = \arg\min_{\hat{W}} \max_{\|\Delta\|_2 \le \mu} \mathcal{L}^{\mathrm{F}}(\hat{W}+\Delta),$$

We apply Δ as a disturbance for $\hat{W}$. The network is end-to-end trained with task specific loss $\mathcal{L}^{\mathrm{F}}$ calculated on FNN outputs to find flat minima. $\hat{O}^{\mathrm{F}}_{n+1} = \mathrm{bn}(\mathrm{Conv}(\hat{\omega}_n + \Delta, O^{\mathrm{F}}_n))$,

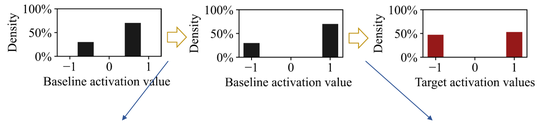
We minimizes the differences between binarized and real-valued weights.

$$\mathcal{L}^{G} = \sum_{n=1}^{N} \|(\hat{\omega}_n - \omega_n)\|_2$$

Regularization on BNN Activations

$$\mathcal{L}^{A} = \mathbf{E} + \mathbf{G},$$

E and **G** denote quantization error loss and even-distribution loss respectively.



E is applied to decrease the quantization error:

$$\mathbf{E} = \frac{1}{P \cdot R}\sum_{p=1}^{P}\sum_{r=1}^{R} -(\hat{\sigma}^{r}(p))^2,$$

G is applied to encode more meaningful cues at each.

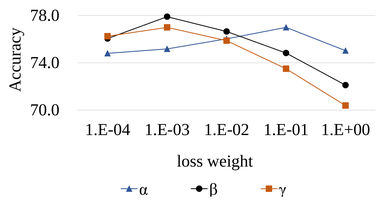
$$\mathbf{G} = \frac{1}{P}\sum_{p=1}^{P}(\frac{1}{R}\sum_{r=1}^{R}\hat{\sigma}^{r}(p))^2.$$

$$\mathcal{L}^{A} = \frac{1}{P}\sum_{p=1}^{P}(\frac{1}{R}\sum_{r=1}^{R} -(\hat{\sigma}^{r}(p))^2 + (\frac{1}{R}\sum_{r=1}^{R}\hat{\sigma}^{r}(p))^2) = -\frac{1}{P}\sum_{p=1}^{P}\sigma^2(p),$$

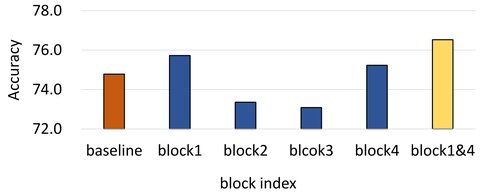
The target can be achieved by maximizing the variance at each location of the activation over the training batch.

Experiments

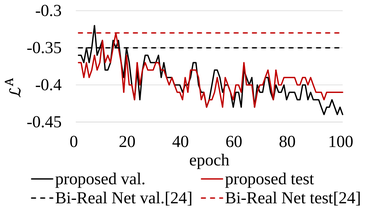
Ablation study



Performance with different α, β and γ .



Performance with $\mathcal{L}^{\mathrm{A}}$ applied on different blocks.



Value of $\mathcal{L}^{\mathrm{A}}$ on validation and test sets.

Comparison with state-of-the-arts

Methods	A	C	P	S	Mean
ResNet-18 [16]†	83.4	80.3	96.0	80.9	85.1
IR-Net [30]	70.4	72.4	87.8	73.5	76.0
ReCU [38]	70.5	73.1	87.0	71.2	75.45
Bi-Real Net [24]	69.2	72.6	86.7	70.6	74.8
+RSC [16]	65.1	71.5	85.2	67.2	72.3
+SWAD [3]	67.3	72.9	87.0	74.0	75.3
+MixStyle [42]	69.5	72.3	87.0	70.9	74.9
+MIRO [4]	69.9	72.9	87.3	71.2	75.3
$+\mathcal{L}^{A}$	69.6	72.9	88.9	74.7	76.5
$+\mathcal{L}^{G}+\mathcal{L}^{\mathrm{F}}$	72.0	73.5	88.7	74.9	77.3
ours	**72.4**	**73.7**	**89.8**	**75.5**	**77.8**
ReActNet [25]	66.4	68.5	85.6	75.7	74.0
$+\mathcal{L}^{G}+\mathcal{L}^{\mathrm{F}}$	**72.3**	72.9	**90.4**	74.2	77.5
ours	72.2	**73.9**	89.3	**75.7**	**77.8**

Comparison on PACS, with A (Art-painting), C (Cartoon), P (Photo) and S (Sketch).

Methods	R	P	C	A	Mean
ResNet-18 [32]†	73.2	71.8	44.2	58.7	62.0
Bi-Real Net [24]	64.9	65.6	41.3	43.2	53.8
$+\mathcal{L}^{G}+\mathcal{L}^{\mathrm{F}}$	64.8	65.6	43.1	43.7	54.3
ours	**66.0**	**66.1**	**43.3**	**44.6**	**55.0**
ReActNet [25]	63.0	63.8	44.6	40.6	53.0
$+\mathcal{L}^{G}+\mathcal{L}^{\mathrm{F}}$	67.0	**67.9**	45.5	46.6	56.7
ours	**67.1**	67.6	**45.6**	**47.9**	**57.0**

Comparison on OfficeHome with R(Real), P(Product), C(Clipart) and A(Art).

Methods	S	P	L	C	Mean
ResNet-18†	67.0	69.7	60.6	94.6	73.0
Bi-Real Net [24]	59.6	64.7	59.7	92.3	69.1
$+\mathcal{L}^{G}+\mathcal{L}^{\mathrm{F}}$	61.7	67.1	60.7	95.8	71.3
ours	**62.1**	**67.8**	**62.4**	**96.2**	**72.1**
ReActNet [25]	61.4	60.4	61.2	93.2	69.1
$+\mathcal{L}^{G}+\mathcal{L}^{\mathrm{F}}$	**62.6**	**67.2**	62.2	95.3	71.8
ours	62.1	66.9	**62.9**	**96.0**	**72.0**

Comparison on VLCS with S (Sun), P (Pascal), L (LabelMe) and C (Caltech).

Method	Top-1 Acc (%)
Bi-Real Net [5]	68.78
+HOW [26]	68.90
$+\mathcal{L}^{A}$	70.15
$+\mathcal{L}^{G}+\mathcal{L}^{\mathrm{F}}$	70.40
ours	**70.53**

Comparison on CIFAR-100 with Bi-Real Net following [5].

Method	GFLOPs	Top-1 Acc (%)
CI-BCNN [36]	0.154	56.7
R2B Net [27]	0.165	65.4
MeliusNet29 [1]	0.214	65.8
ReActNet [25]	0.087	69.4
ours	0.087	68.9

Comparison on ImageNet with recent SOTA BCNN methods.

Visualization