Convolutional Neural Networks Memory Optimization Inference with Splitting Image

Weihao Zhuang, Hascoet Tristan, Ryoichi Takashima, Tetsuya Takiguchi, Yasuo Ariki

KOBE UNIVERSITY

Contributions

- ◆ We propose a method to reduce the memory requirement of Convolutional Neural Networks (CNNs) in the inference phase.
- ◆ Specifically, EfficientNet-b7 a state-of-the-art model, reduced memory usage by 26% without accuracy dropping.

Introduction

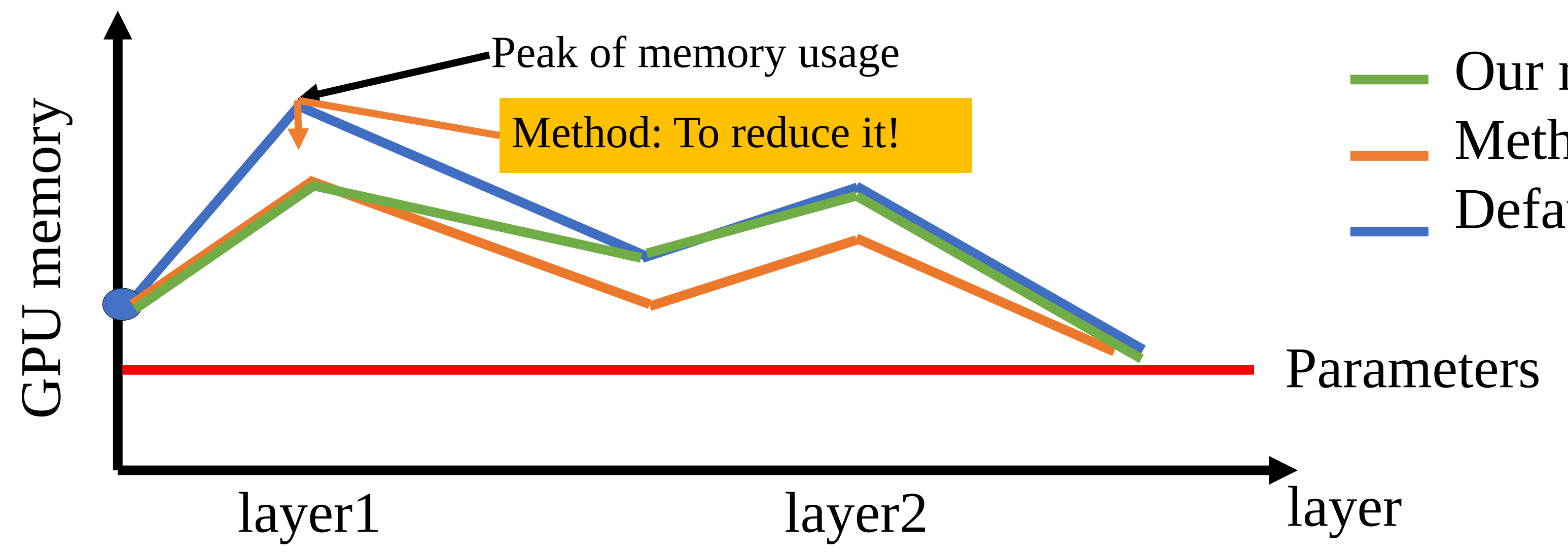


Fig. 1. GPU memory usage of inference.

■ Default	Training: Original input	Inference: Original input
■ Method 1 (resize image)	Training: Original input	Inference: Resized input
■ Method 2 (retrain model)	Training: Resized input	Inference: Resized input
■ **Our method**	Training: Original input	**Inference: Splited input image. Combine activations at specific layer.**

Method

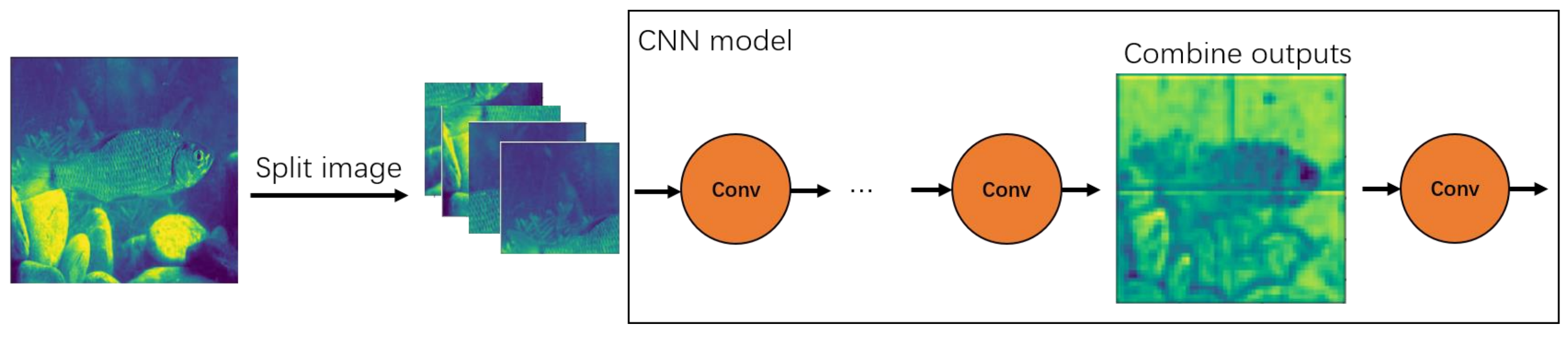
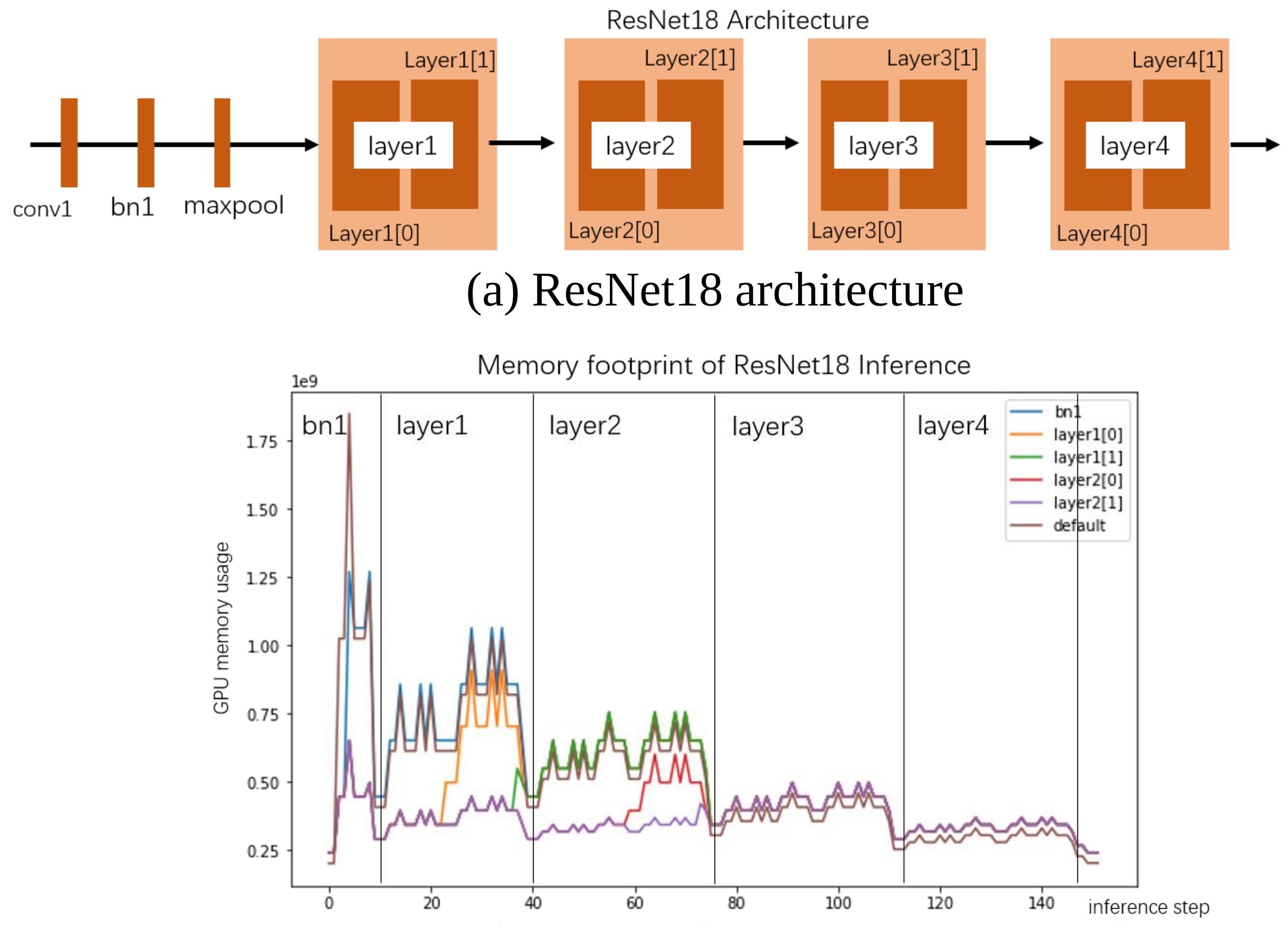


Fig. 2. Splitting input to 4 packs as new input.

- ■ Input images $X \in \mathbb{R}^{c\times h\times w}$,$(c, h, w$ representing the channel, height and width) are split into four parts X_1, X_2, X_3 and X_4, size of each part is $c \times \frac{h}{2} \times \frac{w}{2}$.
- ■ Feed those parts into CNN model one by one.
- ■ After passing through the peak point of memory, combine four outputs activations concatenation and complete the remaining inference.

Experiments



(a) ResNet18 architecture

(b) Memory footprint of ImageNet dataset inference

Fig. 3. GPU memory utilization of different stages of a ResNet18 forward computation

- ■ x axis : Different layers of the network in the order of their execution.
- ■ y axis : Amount of GPU memory usage.
- ■ The brown curve corresponds to a baseline model, without applying our method
- ■ The peak memory usage appears during the execution of the first batch normalization layer (bn1) of the model.
- ■ The blue curve, orange curve, green curve, red curve, and purple curve illustrate the memory usage using our method to combine the output activation after different gradually layers of the network.
- ■ It is important to choose a suitable layer to combine the different activations. The choice of such layer not only affects the memory usage but also the accuracy of the model

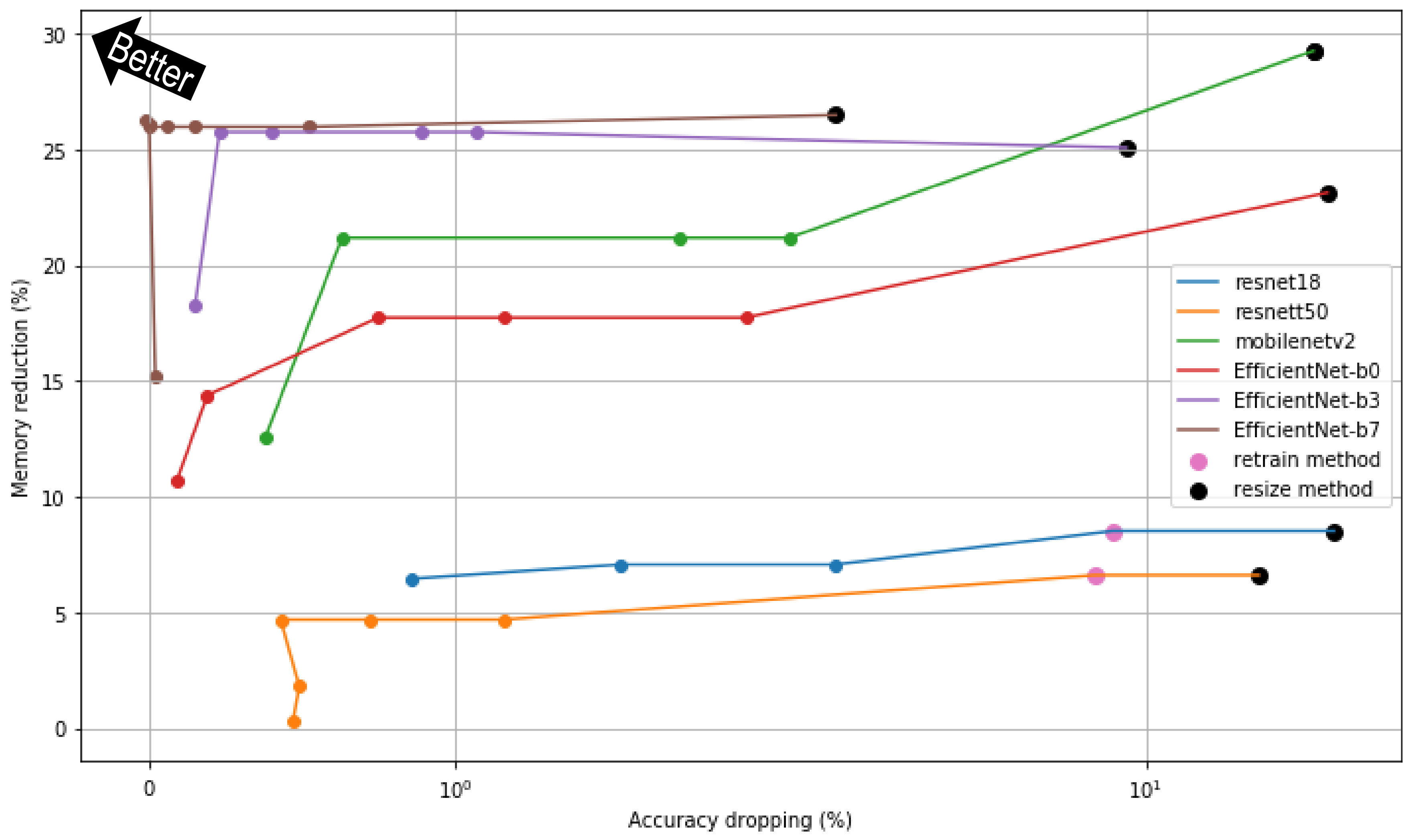


Fig. 4. Relationship between inference accuracy dropping and memory reduction of CNN models inference ImageNet dataset.

- ■ Our method can obtain higher accuracy and the memory reduction is comparable to 'Resize image method' and 'Retrain model method'.
- ■ It is surprising that EfficientNet, a state-of-the-art model, after using our method, the tradeoff between accuracy and memory reduction is better than other models. Specifically, EfficientNet-b7 reduced memory usage by 26% without accuracy dropping.

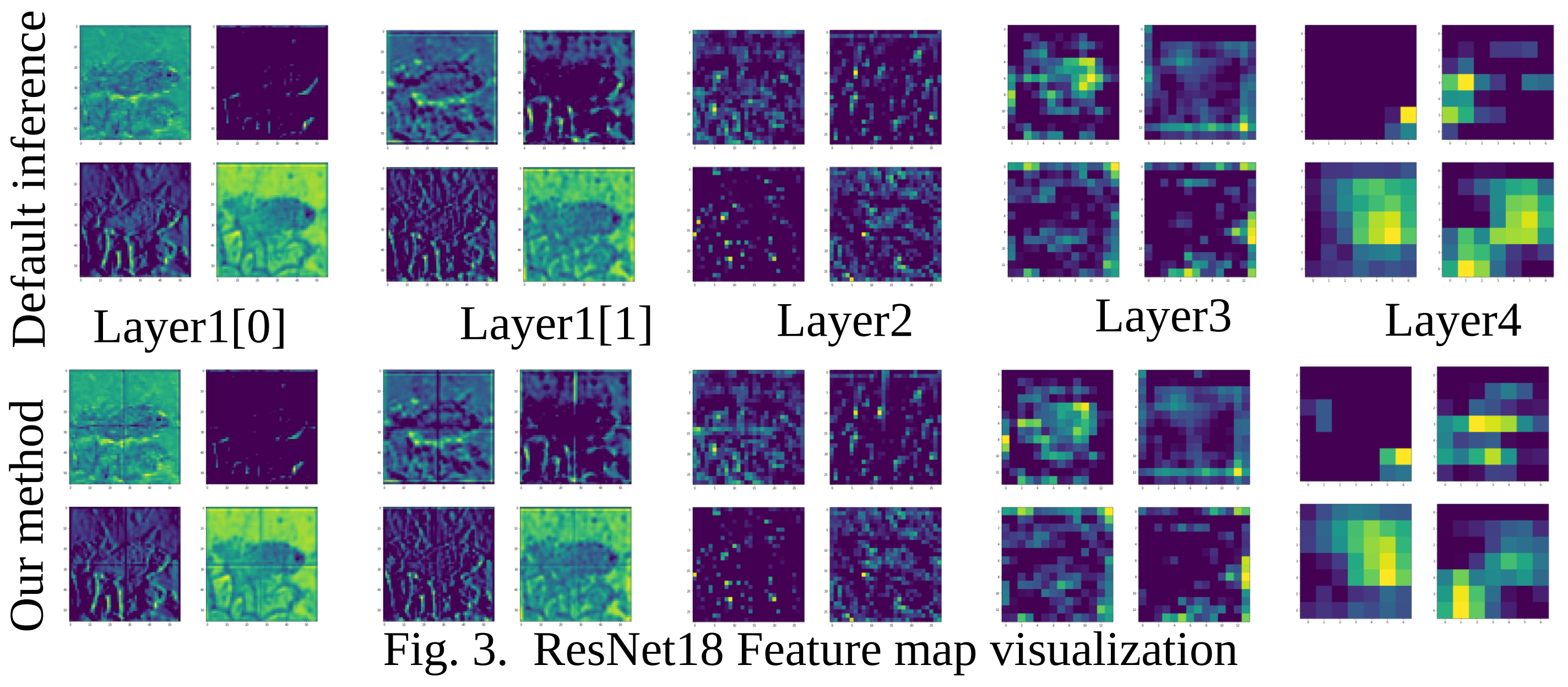


Fig. 3. ResNet18 Feature map visualization

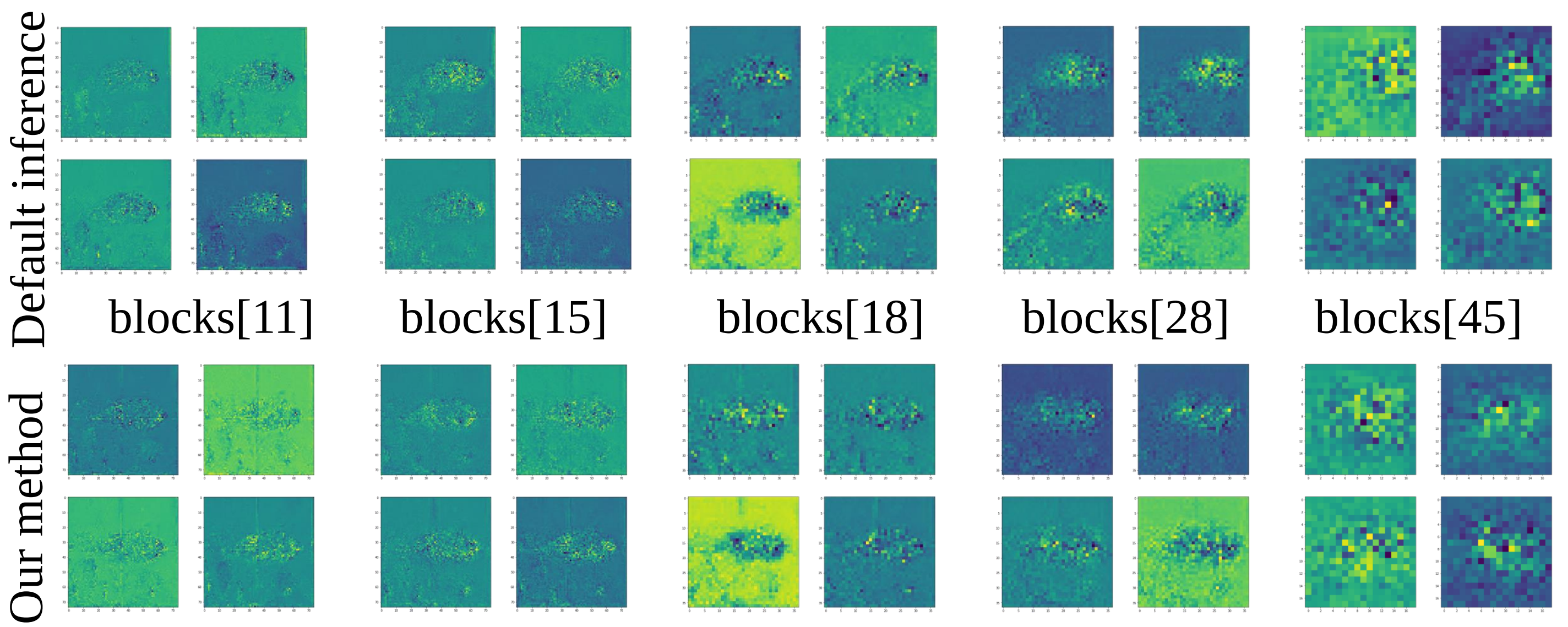


Fig. 4. EfficientNet-b7 Feature map visualization

- ■ ResNet18 input size: (3, 224, 224)
- ■ EfficientNet-b7 input size: (3, 600, 600)
- ■ The sharp inner edges will gradually smooth.
- ■ Our method is more suitable for large resolution input.