Minimally Invasive Surgery for Sparse Neural Networks in Contrastive Manner Appendix

Chong Yu[1,2]
[1]NVIDIA [2]Fudan University
chongy@nvidia.com

1. Introduction and motivation

In the main paper, we analyze the potential problems of traditional model compression and knowledge distillation methods. Inspired by the principle of minimally invasive surgery, we propose a brand-new model compression method named **M**inimally **I**nvasive **S**urgery. **MIS** learns the principal features from a pair of dense and compressed models in a contrastive manner. We prove that **MIS** changes the learning effectiveness ratio and the probability distribution between easy and hard learning objects from information entropy and Bayes perspectives. With the comparison and ablation experiments, we show the success of **MIS** relies on learning the inherent discrepancy between the representation capacities of the dense and the compressed model, and the discrepancy introduced by hardware acceleration restrictions between two compressed models. With **MIS**, we can compress the models for various tasks into efficient forms and can get considerable acceleration in general-purpose GPUs.

The motivation of **MIS** method combines two main points in the contribution list.

- **MIS** is designed to have better performance than traditional knowledge distillation and the other network compression methods.
- **MIS** needs to provide the end-to-end compression for neural networks to meet the specific hardware acceleration requirements.

When making the investigation of the network compression methods aiming at model sparsity, we can divide them into two categories, i.e., coarse-grained sparsity and fine-grained sparsity. For the coarse-grained sparsity like filter-sparsity and channel-sparsity, the regular sparse pattern is easy to achieve acceleration on general-purpose processors because it is equivalent to a smaller dense model. For the fine-grained sparsity, the acceleration on general-purpose hardware is very limited due to the irregular sparse pattern caused by the compression methods. Because there are many papers focus on the coarse-grained sparsity (e.g., filter pruning researches), so the main focus of the proposed **MIS** method is on the fine-grained sparse model.

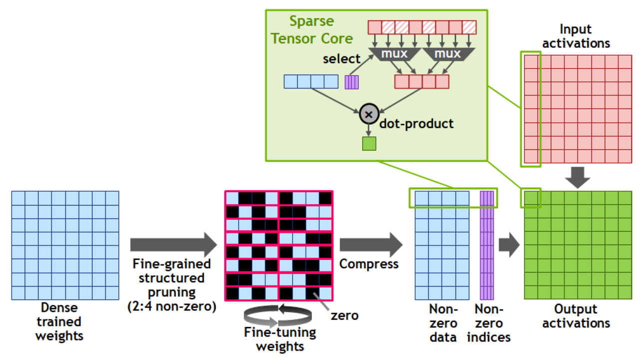

Figure 1. A100 fine-grained structured sparsity feature.

A100 GPU [19] has the new feature to support the fine-grained structured sparsity by enforcing through a 2:4 sparse matrix definition that allows two non-zero values in every four-entry vector, as shown in Figure 1. Due to the well-defined structure of the matrix, it can be compressed efficiently and reduce memory storage and bandwidth by almost 2**X**. The sparse Tensor Cores can exploit 2:4 structured sparsity to double the compute throughput of standard Tensor Core operations for neural networks. So if we can make tiny changes on the irregular fine-grained sparse pattern like minimally invasive surgery, and match the 2:4 fine-grained sparse requirements. Then we can fully utilize this feature to provide extra acceleration to the fine-grained sparse model.

In this **Appendix**, we will provide some supplementary materials and more experimental results for the proposed **MIS** algorithm.

2. Experimental results

In the experiments section of the main paper, we have investigated these issues:

- **MIS** effectiveness and performance across most of the comment networks and applications.

- Quantitative comparison with state-of-the-art methods.
- Ablation experiments.
- Acceleration on general-purpose hardware.

In the **Appendix**, we will provide more results with different parameters settings. For the experiments in this section, we choose PyTorch [21] to implement all algorithms. Most of the training and fine-tuning experimental results are obtained with V100 GPU clusters [18]. The acceleration performance results are obtained with A100 GPU clusters [19] to fully utilize its Tensor Core [20] support for fine-grained structured sparsity and irregularly-compressed models. Because V100 and A100 GPUs could provide much larger math throughput of FP16 than FP32 data type, we also combine **MIS** with the mixed-precision training [17] provided by ***APEX***[1] to compress the models into a more hardware-efficient format. So all the accuracy results reported by **MIS** are using FP16 as the default data type. All the reference algorithms use the default data type provided in public repositories. (Almost all use FP32 except where noted.)

2.1. Effectiveness experiments for classification task

To evaluate the effectiveness of the **MIS** method on the image classification task, we take the ResNet-50 [8], ResNeXt-101 [28] and MobileNet-V2 [23] from ***TorchVision***[2] are chosen as the experiment target models.

In the main paper, the loss adjustment parameters among the surgical prediction loss (α), the healthy-surgical distillation loss (β) and the recovered-surgical distillation loss (γ) apply 1, 10, 50, respectively. More results with different adjustment parameters can refer to Table 1. (The variance is within ± 0.17 for Top-1 accuracy, and ± 0.15 for Top-5 accuracy with different random seeds.)

The original sparse models serve as M_R are trained with the public ***Distiller*** library[3] [33]. ****-PRE*** represents the pre-trained model, ****-FINE*** represents the fine-grained sparse model obtained by adopting a gradual pruning technique (**AGP**) to sparsify the model during the training process[4] [31], ****-BLK*** represents the block-grained [32] sparse model, ****-SUR*** represents the fine-grained [6] sparse model by applying pruning and splicing in a dynamical manner, ****-SNIP*** represents the single-shot pruned [13] model by analyzing the connection sensitivity. In this experiment, **MIS** does not use the ground truth label provided by **ImageNet** [4] dataset. It takes the predicted label from M_H to calculate the surgical prediction loss.

[1]https://github.com/NVIDIA/apex.

[2]https://github.com/pytorch/vision.

[3]https://github.com/NervanaSystems/distiller.

[4]Notice some of the sparse ResNet-50 models and all of the sparse ResNeXt-101 models have higher accuracy than the pre-trained dense models provided by ***TorchVision***.

Models	Original Sparsity	Recovered Model Accuracy		Finetuned Sparsity	Surgical Model Accuracy		Surgical vs. Healthy	Surgical vs. Recovered	Surgical vs. Fake Label
		Top-1 (%)	Top-5 (%)		Top-1 (%)	Top-5 (%)			
ResNet50-RPE	0%	76.130	92.862	N/A	N/A	N/A	N/A	N/A	N/A
					75.910	92.650	10	50	1
ResNet-50-FINE	70%	76.496	93.080	70%	75.892	92.681	10	25	1
					75.854	92.704	10	10	1
					75.198	92.280	10	50	1
ResNet-50-FINE	85%	75.670	92.682	85%	75.175	92.301	10	25	1
					75.134	92.334	10	10	1
					74.156	91.874	10	50	1
ResNet-50-FINE	90%	74.680	92.298	90%	74.125	91.810	10	25	1
					74.078	91.764	10	10	1
					71.414	90.288	10	50	1
ResNet-50-FINE	95%	71.830	90.646	95%	71.455	90.293	10	25	1
					71.518	90.292	10	10	1
					76.224	92.852	10	50	1
ResNet-50-BLK	70%	76.452	92.990	70%	76.231	92.841	10	25	1
					76.240	92.838	10	10	1
					75.162	92.390	10	50	1
ResNet-50-SUR	80%	75.538	92.670	80%	75.160	92.384	10	25	1
					75.156	92.374	10	10	1
ResNeXt101-RPE	0%	78.188	93.886	N/A	N/A	N/A	N/A	N/A	N/A
					79.254	**94.544**	10	50	1
ResNeXt-101-FINE	75%	79.078	94.468	75%	**79.297**	**94.505**	10	25	1
					79.322	**94.488**	10	10	1
					78.880	**94.398**	10	50	1
ResNeXt-101-FINE	85%	78.764	94.368	85%	**79.952**	**94.374**	10	25	1
					79.002	**94.354**	10	10	1
					78.584	**94.154**	10	50	1
ResNeXt-101-FINE	90%	78.530	94.110	90%	**78.624**	**94.135**	10	25	1
					78.648	**94.120**	10	10	1
					77.058	**93.596**	10	50	1
ResNeXt-101-FINE	95%	76.922	93.574	95%	**76.995**	**93.591**	10	25	1
					76.928	**93.584**	10	10	1
					79.173	**94.471**	10	50	1
ResNeXt-101-BLK	75%	79.063	94.404	75%	**79.220**	**94.462**	10	25	1
					79.269	**94.453**	10	10	1
					78.845	**94.502**	10	50	1
ResNeXt-101-SUR	80%	78.631	94.356	80%	**78.891**	**94.497**	10	25	1
					78.939	**94.488**	10	10	1
MobileNet-V2-RPE	0%	71.880	90.290	N/A	N/A	N/A	N/A	N/A	N/A
					70.804	**88.918**	10	50	1
MobileNet-V2-FINE	50%	69.023	88.765	50%	**70.236**	**88.874**	10	25	1
					70.006	**88.804**	10	10	1
					68.500	**88.412**	10	50	1
MobileNet-V2-FINE	75%	68.371	88.303	75%	**68.472**	**88.398**	10	25	1
					68.442	**88.400**	10	10	1
					65.422	**86.676**	10	50	1
MobileNet-V2-FINE	85%	65.303	86.519	85%	**65.406**	**86.638**	10	25	1
					65.380	**86.544**	10	10	1

Table 1. **MIS** effectiveness on image classification task.

2.2. Effectiveness experiments for detection task

To evaluate the effectiveness of the **MIS** on the detection task, we take the Faster R-CNN [22], RetinaNet [14], Mask R-CNN [7] from ***Detectron***[5], and SSD [16] from ***NVIDIA repository***[6] as the experiment target models. In the main paper, the loss adjustment parameters among the surgical prediction loss (α), the healthy-surgical distillation loss (β) and the recovered-surgical distillation loss (γ) apply 1, 10, 15, respectively. More results with different adjustment parameters can refer to Table 2. (The variance is within ± 0.20 for average precision, and ± 0.24 for average recall with different random seeds.)

The original sparse models serve as M_R are compressed with **AGP** method and trained with the ***Distiller*** library[3]. ***R50***, ***R101*** and ***X101*** in the brackets represent the ResNet-50, ResNet-101 and ResNeXt-101 models served as the backbone of the detection networks. ***1x*** and ***3x*** represent the different learning rate schedulers which are applied when training the backbone models. ***AP*** and ***AR*** represent the average precision and average recall metrics. In this experiment, **MIS** uses the ground truth info provided by **COCO** [15] dataset.

[5]https://github.com/facebookresearch/detectron2.

[6]https://github.com/NVIDIA/DeepLearningExamples.

Model	Healthy Model Box AP	Healthy Model Box AR	Sparsity Ratio	Recovered Model Box AP	Recovered Model Box AR	Surgical Model Box AP	Surgical Model Box AR	Surgical vs. Healthy	Surgical vs. Recovered	Surgical vs. Fake Label
Faster R-CNN(R50-1x)	37.65	52.14	50%	38.58	53.04	38.82	53.07	10	25	1
						38.76	53.05	10	15	1
						38.65	52.97	10	5	1
			75%	36.67	51.31	36.66	51.47	10	25	1
						36.57	51.42	10	15	1
						36.51	51.39	10	5	1
Faster R-CNN(R50-3x)	39.79	52.14	50%	39.96	53.97	39.93	53.95	10	25	1
						39.89	53.92	10	15	1
						39.81	53.90	10	5	1
			75%	38.85	52.92	38.99	53.26	10	25	1
						38.94	53.21	10	15	1
						38.88	53.14	10	5	1
Faster R-CNN(R101-3x)	41.92	55.55	50%	42.03	55.53	42.11	55.77	10	25	1
						42.01	55.65	10	15	1
						41.93	55.58	10	5	1
			75%	41.12	55.11	41.15	55.29	10	25	1
						41.11	55.23	10	15	1
						41.03	55.17	10	5	1
Faster R-CNN(X101-3x)	43.08	55.63	50%	42.59	55.74	42.79	55.88	10	25	1
						42.68	55.83	10	15	1
						42.57	55.81	10	5	1
			75%	42.52	55.63	42.70	55.82	10	25	1
						42.63	55.74	10	15	1
						42.58	55.67	10	5	1
RetinaNet(R50-1x)	36.45	53.36	50%	37.43	53.82	37.49	54.23	10	25	1
						37.42	54.11	10	15	1
						37.28	54.03	10	5	1
			75%	34.85	51.84	34.90	51.98	10	25	1
						34.81	51.93	10	15	1
						34.76	51.85	10	5	1
RetinaNet(R50-3x)	38.45	54.34	50%	37.44	53.71	37.71	53.94	10	25	1
						37.55	53.81	10	15	1
						37.39	53.66	10	5	1
			75%	37.40	53.33	37.57	53.42	10	25	1
						37.43	53.28	10	15	1
						37.29	53.19	10	5	1
RetinaNet(R101-3x)	40.04	55.61	50%	39.33	55.22	39.34	55.21	10	25	1
						39.27	55.07	10	15	1
						39.21	54.95	10	5	1
			75%	39.22	54.32	39.18	54.47	10	25	1
						39.06	54.33	10	15	1
						38.97	54.19	10	5	1
SSD(R50)	25.11	36.13	50%	25.83	36.91	25.84	36.97	10	25	1
						25.72	36.80	10	15	1
						25.60	36.61	10	5	1
			75%	24.90	35.88	24.95	36.05	10	25	1
						24.86	35.93	10	15	1
						24.72	35.71	10	5	1
Mask R-CNN(R50-1x)	39.91	54.42	50%	39.79	53.92	40.37	54.75	10	25	1
						40.21	54.62	10	15	1
						40.10	54.48	10	5	1
			75%	37.27	52.01	37.41	52.22	10	25	1
						37.41	52.13	10	15	1
						37.41	52.01	10	5	1
Mask R-CNN(R50-3x)	40.62	54.53	50%	40.70	54.63	40.97	54.71	10	25	1
						40.84	54.50	10	15	1
						40.68	54.34	10	5	1
			75%	39.90	54.24	39.85	54.43	10	25	1
						39.75	54.22	10	15	1
						39.62	54.10	10	5	1
Mask R-CNN(R101-3x)	42.92	56.51	50%	43.21	56.83	43.14	56.68	10	25	1
						43.01	56.55	10	15	1
						42.87	56.32	10	5	1
			75%	42.04	56.01	42.24	56.21	10	25	1
						42.16	56.03	10	15	1
						42.07	55.98	10	5	1
Mask R-CNN(X101-3x)	44.13	56.92	50%	43.95	55.81	43.96	55.84	10	25	1
						43.89	55.74	10	15	1
						43.77	55.68	10	5	1
			75%	43.62	56.32	43.91	56.43	10	25	1
						43.80	56.29	10	15	1
						43.68	56.14	10	5	1

Table 2. **MIS** effectiveness on detection task.

2.3. Effectiveness experiments for translation task

To evaluate the effectiveness of the **MIS** on the translation task, we take the GNMT [27] from ***NVIDIA repository***[6] and Transformer [25] from ***Fairseq***[7] as the experiment target models. In the main paper, the loss adjustment parameters among the surgical prediction loss (α), the healthy-surgical distillation loss (β) and the recovered-surgical distillation loss (γ) apply 1, 2, 5, respectively. More results with different adjustment parameters can refer to Table 3. (The variance is within $\pm$0.15 for BLEU score with different random seeds.)

The original sparse models serve as M_R are compressed with the pruning method [3]. **WMT14 En-Ge** and **WMT16 En-Ge** in the brackets represent the WMT14 and WMT16 English-German dataset[8], respectively. In this experiment, **MIS** uses the ground truth info provided by WMT datasets.

Model	Healthy Model BLEU Score	Sparsity Ratio	Recovered Model BLEU Score	Surgical Model BLEU Score	Surgical vs. Healthy	Surgical vs. Recovered	Surgical vs. Fake Label
GNMT(WMT16 En-Ge)	24.37	50%	24.77	24.75	2	10	1
				24.73	2	5	1
				24.72	2	3	1
		75%	24.67	24.72	2	10	1
				24.69	2	5	1
				24.66	2	3	1
		90%	24.30	24.34	2	10	1
				24.31	2	5	1
				24.29	2	3	1
Transformer(WMT14 En-Ge)	28.65	50%	28.89	28.94	2	10	1
				28.91	2	5	1
				28.87	2	3	1
		75%	28.79	28.80	2	10	1
				28.77	2	5	1
				28.75	2	3	1
		90%	28.15	28.25	2	10	1
				28.21	2	5	1
				28.18	2	3	1
Transformer(WMT16 En-Ge)	27.79	50%	28.01	28.06	2	10	1
				28.03	2	5	1
				28.00	2	3	1
		75%	27.99	28.01	2	10	1
				27.97	2	5	1
				27.95	2	3	1
		90%	27.65	27.74	2	10	1
				27.70	2	5	1
				27.66	2	3	1

Table 3. **MIS** effectiveness on translation task.

2.4. Effectiveness experiments for super resolution

To evaluate the effectiveness of the **MIS** on the super resolution task, we take the SRResNet[9] [12] as the experiment target model. In the main paper, the loss adjustment parameters among the surgical prediction loss (α), the healthy-surgical distillation loss (β) and the recovered-surgical distillation loss (γ) apply 1, 1.5, 3, respectively. More results with different adjustment parameters can refer to Table 4. (The variance is within $\pm$0.13 for ***PSNR***, and $\pm$0.045 for ***SSIM*** with different random seeds.) The representative super-resolution outputs are shown in Figure 2.

Dataset	Healthy Model PSNR	Healthy Model SSIM	Sparsity Ratio	Recovered Model PSNR	Recovered Model SSIM	Surgical Model PSNR	Surgical Model SSIM	Surgical vs. Healthy	Surgical vs. Recovered	Surgical vs. Fake Label
Set5	31.803	0.863	50%	31.234	0.870	**31.496**	**0.873**	1.5	5	1
						31.484	**0.872**	1.5	3	1
						31.476	**0.870**	1.5	2	1
			75%	31.145	0.862	**31.421**	**0.863**	1.5	5	1
						31.301	0.861	1.5	3	1
						31.159	0.860	1.5	2	1
			90%	30.989	0.854	**31.071**	**0.860**	1.5	5	1
						31.004	**0.856**	1.5	3	1
						30.993	0.852	1.5	2	1
Set14	28.643	0.726	50%	28.315	0.755	**28.423**	**0.757**	1.5	5	1
						28.417	0.754	1.5	3	1
						28.411	0.750	1.5	2	1
			75%	28.275	0.750	**28.381**	**0.756**	1.5	5	1
						28.369	**0.753**	1.5	3	1
						28.354	**0.751**	1.5	2	1
			90%	28.012	0.743	**28.146**	**0.749**	1.5	5	1
						28.134	**0.747**	1.5	3	1
						28.127	**0.745**	1.5	2	1
DIV2K	29.256	0.788	50%	28.926	0.811	**29.134**	**0.812**	1.5	5	1
						29.025	0.810	1.5	3	1
						29.003	0.807	1.5	2	1
			75%	28.795	0.793	**28.932**	**0.803**	1.5	5	1
						28.918	**0.798**	1.5	3	1
						28.901	**0.794**	1.5	2	1
			90%	28.423	0.735	**28.515**	**0.746**	1.5	5	1
						28.506	**0.740**	1.5	3	1
						28.495	**0.736**	1.5	2	1

Table 4. **MIS** effectiveness on super resolution task.

The original sparse models serve as M_R are compressed with the pruning method [9]. SRResNet is trained on the

[7] https://github.com/pytorch/fairseq.

[8] http://www.statmt.org/wmt16/translation-task.html.

[9] https://github.com/twtygqyy/pytorch-SRResNet.

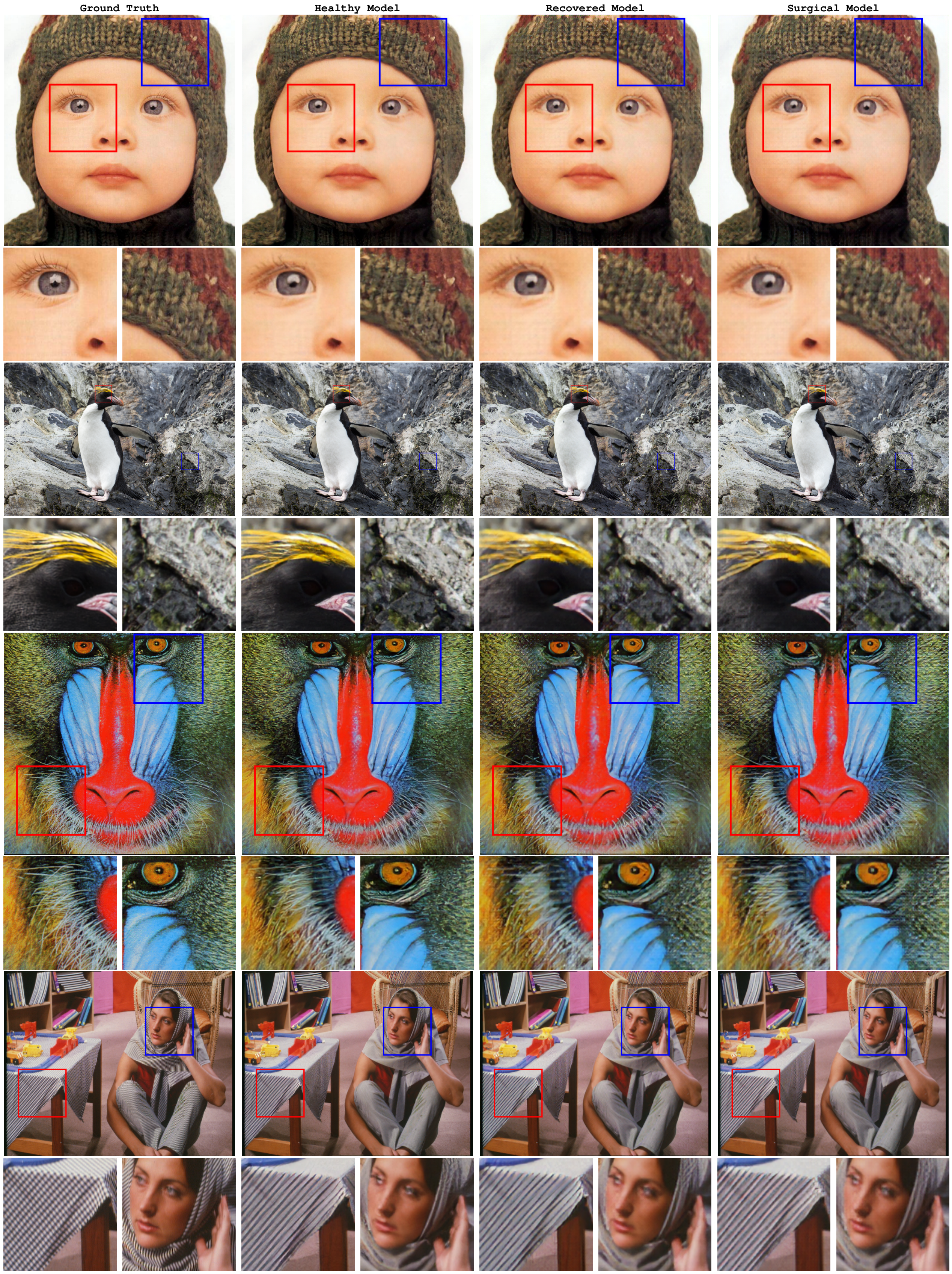

Figure 2. Representative super resolution results with enlargements of boxed areas (The Recovered Model and Surgical Model are compressed to 50% sparse level).

DIV2K dataset [1]. The **DIV2K** validation images, as well as **Set5** [2] and **Set14** [29] datasets are used to report deployment quality. In the super resolution task, image quality is often evaluated by two metrics: Peak Signal-to-Noise Ratio (***PSNR***) [10] and Structural Similarity (***SSIM***) [26].

2.5. Ablation experiments and insights

2.5.1 More accurate healthy model

We change the healthy model with a more accurate one to verify whether it can further improve the effect of **MIS**. We use the pre-trained ResNeXt-101 from ***TorchVision***[2] as the healthy model. The results are shown in Table 5.

Model	Sparsity Ratio	Recovered Model Accuracy		Surgical Model Accuracy	
		Top-1 (%)	Top-5 (%)	Top-1 (%)	Top-5 (%)
ResNet-50	0%	76.130	92.862	N/A	N/A
ResNeXt-101	0%	78.188	93.886	N/A	N/A
ResNet50	70%-FINE	76.496	93.080	**77.038**	**93.240**
	85%-FINE	75.670	92.682	**75.836**	**92.704**
	90%-FINE	74.680	92.298	**74.796**	92.208
	95%-FINE	71.830	90.646	**71.964**	90.638
	70%-BLK	76.452	92.990	**77.112**	**93.304**
	80%-SUR	75.538	92.670	**75.820**	**92.738**

Table 5. **MIS** with more accurate healthy model.

From the results, we can conclude a more accurate healthy model can bring extra benefit in accuracy. It also proves that **MIS** can be used when dense and compressed models have different structures. This is not realizable for the model compression methods which rely on distillation from pure feature maps, like **LIT** [11].

2.5.2 Contribution of each component

In this experiment, we want to check the contribution of each component in **MIS** to the final model compression effect. Then we can have a deep insight into why **MIS** can outperform state-of-the-art methods. Apart from **AGP** and **KD** methods we have discussed, we also involve the **R**esidual **K**nowledge **D**istillation [5] (**RKD**) and **C**ontrastive **R**epresentation **D**istillation [24] (**CRD**) methods in the comparison. The results with different sparsity ratio are shown in Table 6. ***Unsupervised*** and ***Supervised*** in the brackets represent **MIS** does not use & use the ground truth info provided by **ImageNet**, respectively.

MIS introduces two distillation loss items to learn the inherent discrepancy between the representation capacities of the dense and the compressed model, and the discrepancy introduced by hardware acceleration restrictions between two compressed models. From the results, we can see these key differences from **KD**, **RKD**, and **CRD** contribute to the good effect of **MIS**.

We can also find even without the ground truth info from the training set, **MIS** can still achieve satisfactory accuracy. The distillation between the different representation capacities of the dense and the compressed model helps **MIS** to improve the generalization without ground truth info.

Model	Algorithm	Sparsity Ratio	Model Accuracy	
			Top-1 (%)	Top-5 (%)
ResNet-50	Baseline	0%	76.130	92.862
	BLK	70%	76.452	92.990
	AGP	70%	76.496	93.080
	KD	70%	75.950	92.710
	RKD	70%	75.474	93.124
	CRD	70%	76.432	93.190
	MIS(Unsupervised)	70%	75.910	92.650
	MIS(Supervised)	70%	**76.558**	**93.188**
	AGP	85%	**75.670**	92.682
	KD	85%	75.094	92.294
	RKD	85%	75.546	92.746
	CRD	85%	75.538	92.762
	MIS(Unsupervised)	85%	75.198	92.280
	MIS(Supervised)	85%	75.576	**92.774**
	AGP	90%	**74.680**	92.298
	KD	90%	74.014	91.794
	RKD	90%	75.574	92.288
	CRD	90%	74.682	92.258
	MIS(Unsupervised)	90%	74.156	91.874
	MIS(Supervised)	90%	74.584	**92.308**
	AGP	95%	71.830	90.646
	KD	95%	71.484	90.298
	RKD	95%	71.934	90.748
	CRD	95%	**71.972**	90.746
	MIS(Unsupervised)	95%	71.414	90.288
	MIS(Supervised)	95%	71.938	**90.792**
ResNeXt-101	Baseline	0%	78.188	93.886
	BLK	75%	79.063	94.404
	AGP	75%	79.078	94.468
	KD	75%	79.114	94.466
	RKD	75%	78.954	94.482
	CRD	75%	78.958	94.462
	MIS(Unsupervised)	75%	79.254	94.544
	MIS(Supervised)	75%	**79.348**	**94.682**
	AGP	85%	78.764	94.368
	KD	85%	78.956	94.340
	RKD	85%	78.866	94.384
	CRD	85%	78.734	94.372
	MIS(Unsupervised)	85%	78.880	94.398
	MIS(Supervised)	85%	**78.966**	**94.422**
	AGP	90%	78.530	94.110
	KD	90%	78.560	94.150
	RKD	90%	78.564	94.154
	CRD	90%	78.566	94.168
	MIS(Unsupervised)	90%	78.584	94.154
	MIS(Supervised)	90%	**78.664**	**94.264**
	AGP	95%	76.922	93.574
	KD	95%	76.910	93.438
	RKD	95%	77.024	93.566
	CRD	95%	77.044	93.542
	MIS(Unsupervised)	95%	77.058	93.596
	MIS(Supervised)	95%	**77.124**	**93.636**

Table 6. Ablation experiment on contribution of each component. (Use the image classification task as an example.)

2.5.3 Visualization

We apply the Class Activation Mapping (**CAM**) tool [30] to the healthy model M_H, the recovered model M_R and the surgical model M_S for ResNet-50. **CAM** can highlight the importance of the image region to the final prediction. The visualization results are shown in Figure 3.

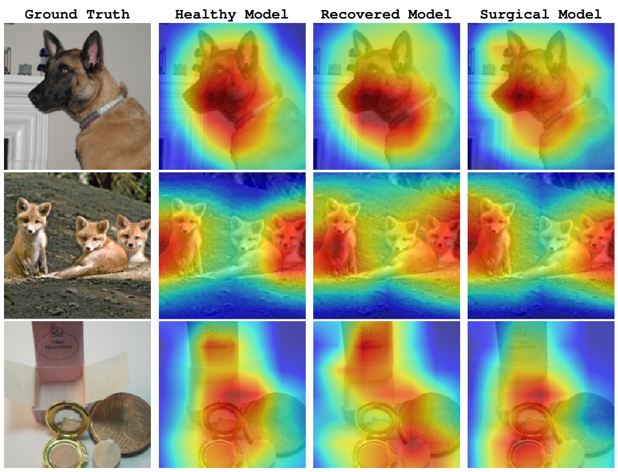

Figure 3. Class activation mapping visualization. (The Recovered Model and Surgical Model are compressed to 80% sparse level).

For **CAM**, the red color highlight the "attention" area of each model. Though the surgical model is restricted by the hardware acceleration requirements, the **CAM**s of M_H, M_R and M_S all focus on the inherent features of the Malinois, red fox and face powder in the ground truth images, which leading to the right classification.

3. Conclusion and ethics statement

For the open-source community, our experimental observations and the proposed compression technique could be inspiring to the model compression field. Our study also provides good guidance for people who want to try the latest features for the newly announced A100 GPU.

Mobile applications performing object detection or super-resolution on the client to save bandwidth can benefit from simpler models. Using efficient models in the data centers can leave more resources available to train much more complex networks.

From the societal impact aspect, the neural models are widely used to daily tasks like autonomous driving, medical imaging, etc. Our proposed compression technique can bring beneficial impacts on various applications. So compressed models with higher deployment efficiency will help in pedestrian detection, emergency protection, medical analysis, and diagnosis. And eventually protecting people's safety and saving more lives.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 126–135, 2017. 5

[2] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. 2012. 5

[3] Robin Cheong and Robel Daniel. transformers. zip: Compressing transformers with pruning and quantization. Technical report, tech. rep., Stanford University, Stanford, California, 2019. 3

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 2

[5] Mengya Gao, Yujun Shen, Quanquan Li, and Chen Change Loy. Residual knowledge distillation. *arXiv preprint arXiv:2002.09168*, 2020. 5

[6] Yiwen Guo, Anbang Yao, and Yurong Chen. Dynamic network surgery for efficient dnns. In *Advances in neural information processing systems*, pages 1379–1387, 2016. 2

[7] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 2

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[9] Zejiang Hou and Sun-Yuan Kung. Efficient image super resolution via channel discriminative deep neural network pruning. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3647–3651. IEEE, 2020. 3

[10] Quan Huynh-Thu and Mohammed Ghanbari. Scope of validity of psnr in image/video quality assessment. *Electronics letters*, 44(13):800–801, 2008. 5

[11] Animesh Koratana, Daniel Kang, Peter Bailis, and Matei Zaharia. Lit: Learned intermediate representation training for model compression. In *International Conference on Machine Learning*, pages 3509–3518, 2019. 5

[12] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017. 3

[13] Namhoon Lee, Thalaiyasingam Ajanthan, and Philip HS Torr. Snip: Single-shot network pruning based on connection sensitivity. *arXiv preprint arXiv:1810.02340*, 2018. 2

[14] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 2

[15] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 2

[16] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016. 2

[17] Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. Mixed precision training. *arXiv preprint arXiv:1710.03740*, 2017. 2

[18] NVIDIA. NVIDIA Tesla V100 GPU Architecture, 2017. 2

[19] NVIDIA. NVIDIA A100 Tensor Core GPU Architecture, 2020. 1, 2

[20] NVIDIA. NVIDIA Tensor Core, 2020. 2

[21] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017. 2

[22] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 2

[23] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018. 2

[24] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. *arXiv preprint arXiv:1910.10699*, 2019. 5

[25] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 3

[26] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 5

[27] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016. 3

[28] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017. 2

[29] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *International conference on curves and surfaces*, pages 711–730. Springer, 2010. 5

[30] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016. 6

[31] Michael Zhu and Suyog Gupta. To prune, or not to prune: exploring the efficacy of pruning for model compression. In *International Conference on Learning Representations*, 2018. 2

[32] Zmora. Block pruning using L1-norm ranking and AGP, 2019. 2

[33] Neta Zmora, Guy Jacob, and Gal Novik. Neural network distiller. *URL https://zenodo. org/record/1297430*, 2018. 2