A. FPN based models: Disentangling Capacity

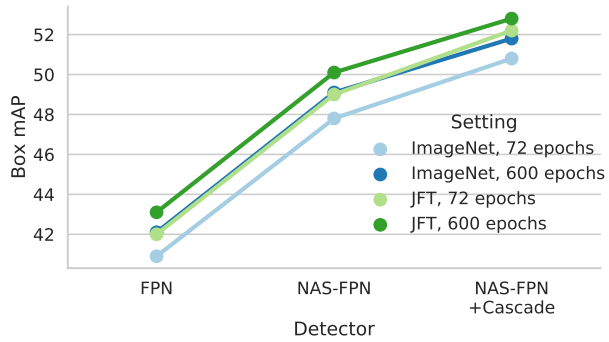

Figure 6. Performance of different detectors preserving the same set of weights (ResNet-101). As the backbones are frozen, it is possible to disentangle the relative benefit from pre-training on ImageNet *versus* JFT-300M from other confounding factors. Pre-training on a larger image classification dataset (JFT-300M) has a clear (and similar) benefit across detectors with different compositions (FPN, NAS-FPN, NAS-FPN + Cascade). Training longer benefits all variations.

Model	Pretraining	mAP	AP @ 50
FPN			
+ResNet-50	ImageNet	46.2	68.5
	+ Freeze backbone	(−6.5) 39.7	(−6.2) 62.3
	JFT-300M	46.4	68.2
	+ Freeze backbone	(−6.0) 40.4	(−4.1) 64.1
+ResNet-101	ImageNet	47.6	69.0
	+ Freeze backbone	(−6.7) 40.9	(−5.0) 64.0
	JFT-300M	48.1	69.8
	+ Freeze backbone	(−6.1) 42.0	(−3.6) 66.2
FPN + Cascade			
+ResNet-50	ImageNet	48.5	66.1
	+ Freeze backbone	(−6.1) 42.4	(−6.3) 59.8
	JFT-300M	49.4	67.3
	+ Freeze backbone	(−6.1) 43.3	(−6.1) 61.4
+ResNet-101	ImageNet	49.7	67.8
	+ Freeze backbone	(−5.5) 44.2	(−5.8) 62.0
	JFT-300M	50.3	68.6
	+ Freeze backbone	(−5.1) 45.2	(−4.6) 64.0

Table 8. Impact of freezing backbone on FPN based models. Training for shorter (72 epochs) results. Models adopting FPN do not benefit from feature preservation. The addition of Cascade heads do not change the observed results. The performance decrease is similar for models using ResNet-50 and ResNet-101 frozen backbones.

The experiments contained in this appendix further investigate the role of capacity in detectors adopting FPNs.

As presented in the main text, knowledge preservation improves the performance of models with strong detector components (NAS-FPN and NAS-FPN + Cascade) using both ResNet (subsection 4.2) and EfficientNet-B7 (subsection 4.3) backbones. Longer training schemes are able to change the backbone weights further away from a good initial representation, but the relative increase in performance from pre-training on a large dataset is clarified by comparing their frozen counterparts. As shown in Figure 6, the the relative benefit from preserving the knowledge from larger classification datasets is similar across different detectors. This is shown in the visualization as the lines comparing models pre-trained on ImageNet are close to parallel to those of models pre-trained on JFT-300M. Individual performances can be found in Table 1, Table 8 and Table 12.

On the other hand, the absolute performances of models using FPN show that those with frozen backbone lag behind their corresponding fine-tuned or trained from scratch (for longer) counterparts (Table 8). They also show that the

Pretraining	conv. layers	mAP	mAP @ 50
From scratch	1	48.9	70.7
	2	(−0.5) 48.4	(−0.6) 70.1
– Fine-tune backbone			
ImageNet	1	48.4	69.7
	2	(+0.2) 48.6	(+0.8) 70.5
JFT-300M	1	48.6	70.1
	2	(+0.1) 48.7	(+0.4) 70.5
– With frozen backbone			
ImageNet	1	41.3	64.4
	2	(+0.8) 42.1	(+0.9) 65.3
JFT-300M	1	42.2	66.4
	2	(+0.9) 43.1	(+0.8) 67.2

Table 9. Impact on FPN based models performance from decreasing their RPN capacity by reducing its convolutional layers from two (baseline value) to one layer. Results taken using a Resnet-101 backbone (training for longer). Training from scratch performance is increased with the capacity reduction while the opposite happens for models with pre-trained initialization. Models with frozen backbone have the largest decrease in performance from reducing RPN's capacity.

Pretraining	conv. layers	mAP	mAP @ 50
From scratch	2	45.6	67.0
	4	(+0.3) 45.9	(+0.5) 67.5
– Fine-tune backbone			
ImageNet	2	47.5	69.0
	4	(+0.3) 47.8	(+0.7) 69.7
JFT-300M	2	48.1	69.8
	4	(−0.2) 47.9	(< 0.1) 69.8
– With frozen backbone			
ImageNet	2	40.9	64.0
	4	(+1.0) 41.9	(+1.2) 65.2
JFT-300M	2	42.0	66.2
	4	(+0.9) 42.9	(+0.5) 66.7

Table 10. Impact on FPN based models performance from augmenting RPN capacity by doubling its convolutional layers (from two to four layers). Results taken using a Resnet-101 backbone (training for 72 epochs). The gap between frozen and trained models remains large after increasing RPN's capacity.

Pretraining	#filters	mAP	AP @ 50
– Full model fine-tuning			
ImageNet	256	47.6	69.0
	512	(+0.1)47.7	(+0.1)69.1
JFT-300M	256	48.1	69.8
	512	(< 0.1) 48.1	(< 0.1) 69.8
– With frozen backbone			
ImageNet	256	40.9	64.0
	512	(+2.0) 42.9	(+1.9) 65.9
JFT-300M	256	42.0	66.2
	512	(+1.4) 43.4	(+1.0) 67.2

Table 11. Impact on FPN based models performance from increasing detector components hidden representation form 256 up to 512. Results taken using a ResNet-101 backbone (training for 72 epochs). Fine-tuned models show close to no improvement in performance. Models with frozen backbone benefit from extra capacity with a relative improvement. Their absolute performance, on the other hand, shows that extra capacity on filters alone is not enough to fully benefit from knowledge preservation.

addition of Cascade heads alone does not reduce the gap between the fine-tuned or frozen counterparts.

The experiments in this appendix aim to disentangle the role of capacity from other confounding factors, in the gap observed on FPN results. With that goal in mind, we ablate extra experiments that preserve the overall architecture of the FPN based detector, but change the number of trainable parameters available.

First, we evaluate the impact of a small change in capacity, by decreasing the number of convolutional layers available on the RPN. As Table 9 shows, this reduction in capacity impacts the performance of models with pre-trained backbones and those trained from scratch differently. While pre-trained ones have their performance reduced, the trained from scratch version benefits from it. The results also show that frozen models are more harmed by the decrease in capacity than the fine-tuned ones. Table 10 presents the results of the opposite experiment, of increasing RPN's capacity by doubling the number of convolutional filters in this component. The increase in performance observed in frozen models is still small compared to the performance of fully trained counterparts.

Next, we evaluate a larger change in number of parameters, by increasing the number of filters in the detector components from 256 to 512. More specifically we increase the hidden representation size on the RPN, Decoder and Detection head (see Figure 2). As Table 11 shows, increasing capacity of the hidden representations does not impact fine-tuned models significantly, while improved the frozen counterpart.

While the experiments explored in this section reduce the gap between the tuned and frozen FPN models, they also show that changes in capacity alone do not fully explain the performance increase observed on stronger detectors. Next, we review the performance of NAS-FPN based models.

Model	Pretraining	mAP	AP @ 50
NAS-FPN			
+ResNet-50	ImageNet	47.0	68.0
	+ Freeze backbone	(+0.1) 47.1	(+0.3) 68.3
	JFT-300M	47.5	68.9
	+ Freeze backbone	(+0.4) 47.9	(+0.7) 69.6
ResNet-101	ImageNet	48.2	69.6
	+ Freeze backbone	(−0.4) 47.8	(−0.3) 69.3
	JFT-300M	48.5	69.2
	+ Freeze backbone	(+0.5) 49.0	(+1.3) 70.5
NAS-FPN + Cascade			
+ResNet-50	ImageNet	49.4	66.8
	+ Freeze backbone	(+0.5) 49.9	(+0.3) 67.1
	JFT-300M	49.9	67.6
	+ Freeze backbone	(+1.1) 51.0	(+1.2) 68.8
+ResNet-101	ImageNet	51.1	68.7
	+ Freeze backbone	(−0.3) 50.8	(−0.3) 68.4
	JFT-300M	50.9	68.5
	+ Freeze backbone	(+1.3) 52.2	(+1.5) 70.0

Table 12. Impact on performance from frozen representation associated with the use of NAS-FPN under shorter schedule regime (72 epochs). Freezing from ImageNet takes longer to converge as the increase in accuracy observed on NAS-FPN and NAS-FPN +Cascade models is smaller than the observed in longer training regimes. Models composed with NAS-FPN and frozen backbone produce matching or superior performance while consuming fewer resources during training.

B. NAS-FPN based models

This section presents additional results with a shorter training schedule than the 600 epochs schedule used in the main text. We aim to address the impact of training time on our results and review [47]'s observations in light of more recent tricks and practices for training object detectors. In this appendix specifically, we extend their observations to models using NAS-FPN backbones, not covered by the original work. Adding to that, by the time [47] was published, object detection batch size was considerably smaller than the 256 size used in recent literature [14]. In [47], MSCOCO training scheme adopted a batch size of 9 images and a maximum of $3M$ steps. The only data augmentation used by them is random flipping, while recent findings show the importance of large scale jittering (LSJ) [50] on detector training. Thus, Tables 12 and 8 observe the impact of training for 72 epochs with larger batch size (64) and the use of stronger data augmentation (LSJ). We also note that NAS-FPN was proposed after [47], thus, our results extend their observations to the use of more recent components. In summary, by replicating the comparison using more recent findings, we confirm that [47]'s findings about FPN based models on shorter training scheme are still valid. At the same time, our results show that *the benefit from the reuse of the knowledge from large scale image classification datasets is dependent on the choice of the feature pyramid*

Model	Pretraining	mAP	AP @ 50
NAS-FPN + ResNet-101	ImageNet	44.0	63.0
	+ Freeze backbone	(+0.0) 44.0	(+0.2) 63.2
	JFT-300M	44.8	63.7
	+ Freeze backbone	(+0.8) 45.6	(+1.5) 65.3
FPN + ResNet-101	ImageNet	42.6	62.2
	+ Freeze backbone	(−7.3) 35.3	(−6.0) 56.2
	JFT-300M	43.7	63.0
	+ Freeze backbone	(−8.4) 35.3	(−4.8) 58.2

Table 13. One-stage detector performance (RetinaNet, 72 epochs). Similarly to the two-stage detectors, models with frozen features based on NAS-FPN obtain similar performance e significantly reducing resources used during training.

Model	Pretraining	mAP	AP @ 50
NAS-FPN + ResNet-101	From scratch	42.6	61.7
	ImageNet	44.3	63.3
	+ Freeze backbone	(+1.5) 45.8	(+1.9) 65.2
	JFT-300M	44.0	62.8
	+ Freeze backbone	(+2.4) 46.4	(+3.5) 66.3
FPN + ResNet-101	From scratch	43.5	63.2
	ImageNet	43.8	63.6
	+ Freeze backbone	(−7.6) 36.2	(−6.2) 57.4
	JFT-300M	44.3	63.7
	+ Freeze backbone	(−8.3) 36.0	(−4.6) 59.1

Table 14. One-stage detector performance (RetinaNet) under longer training (600 epochs) shows similar trends to the two-stage detectors results: models with Nas-FPN benefit from feature preservation. Training for longer improved the results and relative gain of feature preservation for models based on NAS-FPN but did not help closing the performance gap on those based on FPN.

network architecture more than any of the other detector components.

C. NAS-FPN Single Stage Detectors

This appendix complements the observations of the main text taken using two-stage detectors with ablations adopting single stage detectors, more specifically RetinaNet [29]. Table Table 13 presents the results obtained by training RetinaNet detectors using a ResNet-101 backbones composed with FPN and NAS-FPN backbones trained for 72 epochs. Table Table 14 presents the ablations using similar models but trained for 600 epochs. Similarly to the observations taken using two-stage detectors, single stage models using NAS-FPN also benefit from feature preservation.

D. LVIS: with no data augmentation

This section presents results on the LVIS dataset without the use of Copy-Paste [12] augmentation. Results from Table 15 show that preserving the features obtained on large classification datasets improves performance of objects with different numbers of annotations, while results from Table 16 show that the benefit is also observed across objects of different sizes. The tables show positive impact for both detection and segmentation tasks.

E. Feature preservation and adaptation

In this appendix we extend the results presented in subsection 4.5 to further explore the use of Residual Adapters [40, 41] as a mechanism to balance preserving knowledge obtained from the larger dataset and maintaining some amount of adaptability in the backbone while learning on the target task. We present the performance delta with respect to full backbone fine-tuning (δ_{Ft}) and backbone freezing (δ_{Fz}) to better highlight the impact of residual adapters on results presented in Table 1.

We ablate the use of Residual Adapters across detectors using different compositions (FPN, NAS-FPN, NAS-FPN +Cascade) and a fixed backbone (ResNet-101). As shown in Table 17, the use of residual adapters increased the performance of frozen models in all compositions (positive δ_{Fz}). Detectors adopting FPN presented the largest gain from the use of residual adapters, but their absolute performance still lags behind their fine-tuned counterpart (negative δ_{Ft}).

The largest absolute performance is obtained with NAS-FPN-based detectors (with gains over both the fine-tuned and frozen baselines), at the cost of increased computational resource requirements.

Next, Table 18 presents our results on the effect of varying the backbone while fixing the detector components on the stronger detector composition explored (NAS-FPN +Cascade). Using this detector composition, the gain obtained by the use of adapters is positive no matter if using the smaller (ResNet-50) or larger (ResNet-101) backbone.

LVIS	Box				Mask			
	mAP	**mAP$_r$**	**mAP$_c$**	**mAP$_f$**	**mAP**	**mAP$_r$**	**mAP$_c$**	**mAP$_f$**
First stage results: regular training								
[12]'s baseline	35.0	12.7	34.0	45.9	32.2	13.4	32.2	40.4
+ Freeze backbone	(+0.9) 35.9	(+0.1) 12.8	(+1.3) 35.3	(+0.9) 46.8	(+1.3) 33.5	(+0.0) 13.4	(+2.0) 34.2	(+1.1) 41.5
Second stage: tunes detection-classifier final layer using class-balanced loss								
[12]'s baseline	37.6	23.2	36.0	45.7	34.9	24.6	34.2	40.3
+ Freeze backbone	(+1.8) 39.4	(+1.2) 24.4	(+2.7) 38.7	(+1.1) 46.8	(+2.3) 37.2	(+2.0) 26.6	(+3.3) 37.5	(+1.3) 41.6

Table 15. Performance using EfficientNet-B7 + NAS-FPN per number of annotations groups. Freezing the backbone matches or surpasses fine-tuning performance in all cases (both detection and segmentation) . Balanced loss improves frozen backbone performance more than the fine-tuned one. As opposed to fine-tuning [12], preserving features induces an increase in performance on rare ($\mathrm{mAP_r}$) and common ($\mathrm{mAP_c}$) classes, while still improving frequent ($\mathrm{mAP_f}$) classes. Original first phase results are provided by the authors of [12]. Results with Copy-Paste augmentations can be found in Table 5.

LVIS	Box				Mask			
	mAP	**mAP$_s$**	**mAP$_m$**	**mAP$_l$**	**mAP**	**mAP$_s$**	**mAP$_m$**	**mAP$_l$**
First stage results: regular training								
[12]'s baseline	35.0	28.5	43.8	50.3	32.2	24.4	42.2	49.0
+ Freeze backbone	(+0.9) 35.9	(+0.2) 28.7	(+2.4) 46.2	(+2.4) 52.7	(+1.3) 33.5	(+0.4) 24.8	(+2.3) 44.5	(+2.8) 51.8
Second stage: tunes detection-classifier final layer using class-balanced loss								
[12]'s baseline	37.6	30.8	46.9	53.4	34.9	26.4	45.3	52.2
+ Freeze backbone	(+1.8) 39.4	(+0.6) 31.4	(+2.9) 49.8	(+4.0) 57.4	(+2.3) 37.2	(+0.6) 27.0	(+3.3) 48.6	(+4.5) 56.7

Table 16. Performance using EfficientNet-B7 + NAS-FPN per object size groups. Freezing the backbone surpasses fine-tuning performance for object of all sizes (both detection and segmentation). It has the strongest positive performance impact on large objects ($\mathrm{mAP_l}$), then medium-sized objects ($\mathrm{mAP_m}$), and finally small objects ($\mathrm{mAP_s}$). Balanced loss improves frozen backbone performance more than the fine-tuned one. Original first phase results are provided by the authors of [12]. Comparison results with Copy-Paste augmentations can be found in Table 6.

Model	**Pretraining**	**Epochs**	**mAP**	δ_{Ft}	δ_{Fz}	**mAP50**	δ_{Ft}	δ_{Fz}
FPN + fbb + ra	ImageNet	72	43.9	(−3.7)	(+3.0)	66.8	(−2.2)	(+2.8)
		600	45.0	(−3.6)	(+2.9)	67.8	(−2.6)	(+2.5)
	JFT	72	46.0	(−2.2)	(+3.9)	69.2	(−0.6)	(+3.0)
		600	46.7	(−2.0)	(+3.6)	69.9	(−0.6)	(+2.7)
NAS-FPN + fbb + ra	ImageNet	72	48.6	(+0.4)	(+0.8)	70.0	(+0.4)	(+0.7)
		600	49.9	(+0.9)	(+0.8)	71.4	(+1.4)	(+1.1)
	JFT	72	49.8	(+1.3)	(+0.8)	71.5	(+2.3)	(+1.0)
		600	51.1	(+1.9)	(+1.0)	73.1	(+2.9)	(+1.3)
NAS-FPN, Cascade + fbb + ra	ImageNet	72	51.8	(+0.7)	(+1.0)	69.7	(+1.0)	(+1.3)
		600	53.0	(+1.9)	(+1.2)	71.2	(+2.0)	(+1.2)
	JFT	72	53.0	(+2.1)	(+0.8)	71.2	(+2.7)	(+1.2)
		600	53.6	(+2.5)	(+0.9)	72.0	(+3.0)	(+1.0)

Table 17. Residual adapters across detectors with increasing capacity and fixed backbone (training for shorter): Adapting feature backbones while preserving original knowledge by freezing the original ResNet weights. Columns show mAP and mAP@50 and their difference to their corresponding model trained with fine-tuned features (δ_{Ft}) and frozen features (δ_{Fz}). All models using Resnet-101. fbb: backbone frozen on classification features, ra: residual adapters.

Model	**Pretraining**	**Epochs**	**mAP**	δ_{Ft}	δ_{Fz}	**mAP50**	δ_{Ft}	δ_{Fz}
ResNet-50	ImageNet	72	50.5	(+1.1)	(+0.7)	68.3	(+1.5)	(+1.2)
		600	51.4	(+1.1)	(+0.3)	69.4	(+1.4)	(+0.3)
	JFT	72	51.7	(+1.8)	(+0.7)	69.8	(+2.2)	(+1.1)
		600	52.8	(+2.4)	(+0.7)	71.2	(+3.1)	(+0.8)
ResNet-101	ImageNet	72	51.8	(+0.7)	(+1.0)	69.7	(+1.0)	(+1.3)
		600	53.0	(+1.9)	(+1.2)	71.2	(+2.0)	(+1.2)
	JFT	72	53.0	(+2.1)	(+0.8)	71.2	(+2.7)	(+1.2)
		600	53.6	(+2.5)	(+0.9)	72.0	(+3.0)	(+1.0)

Table 18. Residual adapters over backbones in two sizes and same detector structure (training for longer): Adapting feature backbones while preserving original knowledge by freezing the original ResNet weights. Table shows results on Fast-RCNN combined with NAS-FPN and Cascade heads. Columns show mAP and mAP@50 and their difference to the corresponding model with fine-tuned features (δ_{Ft}) and frozen features (δ_{Fz}). fbb: backbone frozen on classification features, ra: residual adapters.