epochs on IN-10k, MoCov2 performs the same as random guess (0.5%) while our method learns much better representations (9.9%) in terms of Tiny-IN-200 linear evaluation. In general, our method improves the most on AP_{75}, which is a more stringent metric for detection accuracy. It indicates that our model can better capture foreground objects across changing backgrounds during pretraining, hence improving performance for object detection as well as image classification. Moreover, our method is especially effective (i.e., has greater improvements) when the amount of data is small.

Ablation studies

In this section, we will first study the effectiveness of the foreground vs. background (Tobias) information (generated by random networks). Then, we will study the effect of the hyper-parameter p in our method. Finally, we study the sensitivity to data augmentations.

Effect of Tobias information. Notice that we use the foreground vs. background information when merging patches from two images. To demonstrate its effectiveness, we design a random merging strategy for comparison (MoCov2-RM in Table 5). More specifically, we do not use such information and randomly select patches from two images for merging (also half-half division) and it can also be viewed as one kind of patch-level CutMix. We also compare with MoCov2-Mixup where we use Mixup when merging images. We keep all other settings the same and conduct pretraining on both IN-10k and IN-50k. As can be seen in Table 5, we will see a significant drop, especially in object detection performance if we discard the foreground vs. background information provided by our Tobias: up to **-1.2** AP_{50} for RM and **-2.3** AP_{50} for Mixup when trained on IN-10k for 800 epochs. It demonstrates the Tobias information provided by a randomly initialized network is vital. Another interesting thing is that RM achieves better performance than the baseline MoCov2, which indicates that this kind of data augmentation is somehow useful for SSL, as shown in (**?**).

Effect of hyper-parameter. Now we study the effect of the hyper-parameter p, i.e., the probability of changing backgrounds in another view. We study p = 0, 0.3, 0.5, 0.7 and 1.0. Notice that when p=0, our Tobias degenerates into the baseline MoCov2. We train on IN-10k for 800 epochs for all settings in Table 6. For object detection, we can see that when p grows, the result becomes better and will not continue to improve when it grows beyond 0.5. For Tiny ImageNet, $p = 0.7$ achieves the highest accuracy. Notice that we *directly set p to 0.3 for all our experiments throughout this paper* and did not tune it under different settings. It also indicates that we can get better results with more carefully tuned p.

Sensitivity to image augmentations. Now we study the sensitivity to image augmentations of our Tobias by progressively removing transformations in the transformation set following **?**. The results in Table 7 show that the performance of Tobias is much less affected than the performance of MoCov2 when removing the color distortion from the set of image augmentations, especially on Tiny-IN-200. Also we can observe that color distortion (e.g., grayscale and color-jitter) has greater impact on downstream image classification and less impact on object detection. When image augmentations are reduced to a mere random crop, the gap between our Tobias and baseline MoCov2 has increased to 2.9 and 3.2 points for Tiny-IN-200 and VOC detection (AP_{75}), respectively. It indicates that our Tobias is itself an effective data augmentation and less sensitive to other augmentations.

Table 6: Effect of hyper-parameter p. All settings are pretrained on IN-10k for 800 epochs using ResNet-50.

prob p	VOC 07&12			Tiny-IN-200
	AP_{50}	AP	AP_{75}	
0.0	71.6	43.9	45.9	23.6
0.3	73.2	45.7	48.5	23.9
0.5	**73.9**	**46.3**	**49.4**	23.3
0.7	72.3	44.8	47.4	**25.4**
1.0	71.8	44.3	46.6	24.3

Table 7: Impact of progressively removing transformations. All pretrained on IN-10k for 800 epochs.

transformation set	MoCov2			MoCov2-Tobias		
	AP_{50}	AP_{75}	Tiny-IN	AP_{50}	AP_{75}	Tiny-IN
baseline	71.6	45.9	23.6	73.2	48.5	23.9
remove grayscale	70.2	44.1	$19.9_{\downarrow 3.7}$	73.1	49.0	$22.7_{\downarrow 1.2}$
remove color	71.3	46.0	$18.1_{\downarrow 5.5}$	72.7	48.2	$21.2_{\downarrow 2.7}$
crop+flip only	71.0	46.2	$16.8_{\downarrow 6.8}$	72.9	48.3	$20.2_{\downarrow 3.7}$
crop only	71.7	46.7	$15.0_{\downarrow 8.6}$	73.1	49.9	$17.9_{\downarrow 6.0}$

Conclusions

In this paper, we revealed the phenomenon that a randomly initialized CNN has the potential to localize objects well, which we called Tobias. Moreover, we analyzed that activation functions like ReLU and network depth are essential for a random CNN to localize. Then, we proposed Tobias self-supervised learning, which forces the model to focus on foreground objects by dynamically changing backgrounds while keeping the objects under the guidance of Tobias. Various experiments have shown that our method obtained a significant edge over baseline counterparts because it learns to better capture foreground objects. In the future, we will try to apply our Tobias to supervised learning.

Quibusdam nam doloremque libero ducimus hic at odit amet beatae tempora eligendi, voluptatum natus et culpa numquam quisquam voluptas reiciendis tempora voluptatibus a.Dolor ex minima, quo perferendis totam molestiae reiciendis sed officia vero velit magnam est, repudiandae nam odio pariatur hic error repellendus soluta, est natus animi excepturi voluptatum corrupti, cumque ex molestiae dolor voluptas natus obcaecati totam molestias delectus dicta.Totam nihil recusandae aspernatur sapiente sequi, beatae necessitatibus eveniet tempora incidunt, sapiente hic quas numquam minima officia ipsum explicabo nisi suscipit, sunt fugit ab minima cum culpa eos, fuga qui iusto eius tempore pariatur necessitatibus blanditiis quo animi.Ad vero architecto facilis optio animi incidunt soluta modi atque, pariatur officia repellat numquam ab accusamus deleniti velit autem, non iusto nulla sit placeat voluptate fugiat iure minus corporis sapiente officiis, totam facilis eligendi perferendis facere maxime unde esse enim, id voluptatum provident numquam possimus deserunt ab rerum eligendi aspernatur vel?Eos sunt pariatur, sint eaque quaerat sapiente molestias praesentium saepe, voluptate quo necessitatibus, non rerum debitis id odit, optio aliquid adipisci ab sint iusto sapiente enim.