# Planckian Jitter: countering the color-crippling effects of color jitter on self-supervised training (SUPPLEMENTARY MATERIAL)

**Anonymous Author(s)**
Affiliation
Address
email

## 1 Planckian Jitter

Figure 4 illustrates the illuminants sampled from the distribution of a black body radiator, with correlated color temperature $T$ in the interval between $3000K$ and $15000K$. The resulting spectra are visualized on the left and in the middle, while the resulting distribution of illuminants is visualized in the Angle-Retaining Chromaticity diagram on the right.

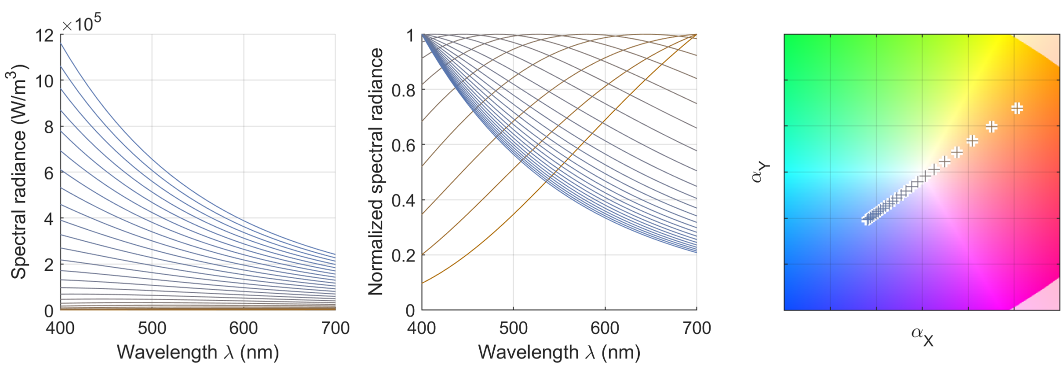


Figure 4: Spectral power distributions (left) and corresponding ARC chromaticities (right) of the sampled black body radiator, used to generate Planckian jittering.

Figure 5 shows a comparison between default color jitter (left) and Planckian jitter (right), replicating Figure 1 in xy chromaticity.

## 2 Datasets details

In section 4.4 of the main paper we analyzed the impact of our data augmentation when using the features extracted from the backbone trained on ImageNet on new datasets. The datsets used in the finetuning step are:

- Flowers-102 [3]: Dataset consisting of 102 flower categories commonly occurring in the United Kingdom. Each class consists of between 40 and 258 images, for a total 8,189 images.
- VegFru [5]: Dataset consisting of more than 160,000 images of vegetables and fruits divided in 292 classes.
- CUB-200 [6]: Dataset made of 6,033 images of 200 bird species.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

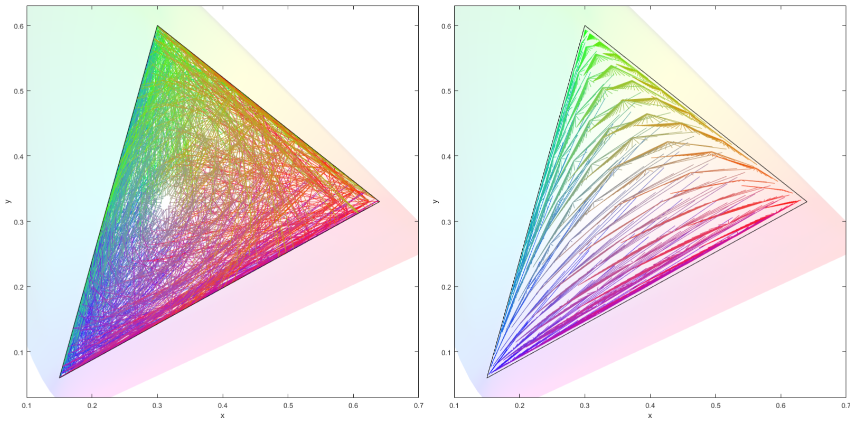

Figure 5: Default color jitter (left) and Planckian jitter (right) in xy chromaticity.

- T1K+ [1]: Dataset of textures divided into 1129 classes and organized in 5 groups of 266 super classes. We adopted the 266 class labeling to finetune and test our models.

A few example images for each of the color task datasets are given in Figure 6.

Figure 6: Example images from the datasets used as downstream classification tasks. From left to right: FLOWERS-102, CUB-200, VEGFRU, and T1K+.

Additionally, in section 6 of this supplementary material TINY-IMAGENET [2] is used. It contains 100,000 images of 200 classes (500 for each class) at $64 \times 64$ pixel resolution.

## 3 Color selectivity index

Color selectivity is defined in [4] as the property of a neuron that activates strongly when a specific color appears in the input image, and does not when the color is absent. It is computed by estimating the ratio between the neuron's global activation with color input images and the global activation with corresponding grayscale images:

$$\alpha(n^{L,i}) = 1 - \frac{\sum_{j=1}^{N} w'_{j,i,L}}{\sum_{j=1}^{N} w_{j,i,L}}. \tag{1}$$

Here $w_{j,i,L}$ refers to the activation of an image patch $j$ for the $i$-th neuron $n^{L,i}$ at layer $L$, normalized for the maximum activation value across all possible image patches. $w'_{j,i,L}$ is the equivalent formulation for a grayscale version of the images. The set of considered image patches is restricted to the top-$N$ regions from a given dataset that maximally activate the neuron of interest.

We can distinguish between neurons that are colorblind or neurons that highly rely on color information by looking at the $\alpha$ value obtained: an $\alpha$ value more than $0.25$ means that the neuron is high color selective, while an alpha value less than $0.1$ means that the neuron is basically colorblind. These thresholds were selected based on the analysis in [4]. We collected alpha values for the neurons in the

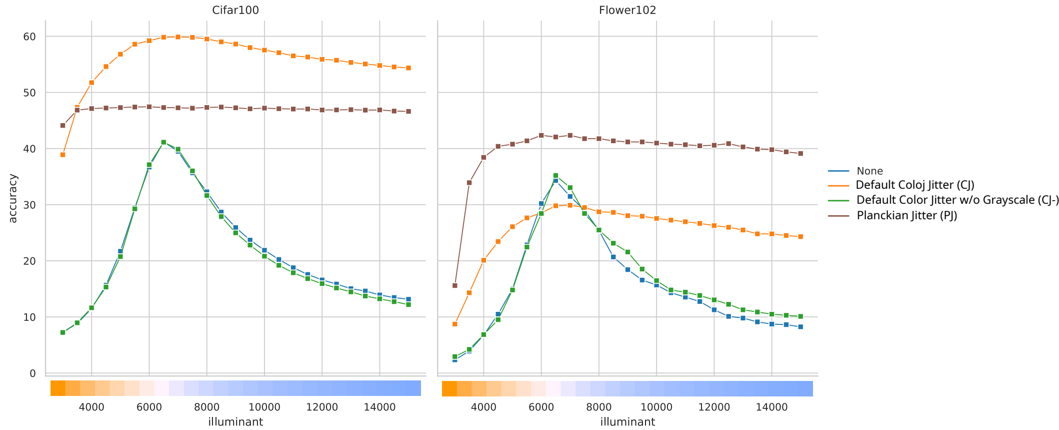


Figure 7: Illumimant robustness analysis. To assess feature invariance to realistic color changes in images, for each method we evaluate classification accuracy on 25 different, re-illumintated versions of the datsets. The images of the two datasets (CIFAR-100 on the left and FLOWERS-102 on the right) have been modified with the illumimants from temperature 3000 K to 15000 K using the Planckian Jitter transform.

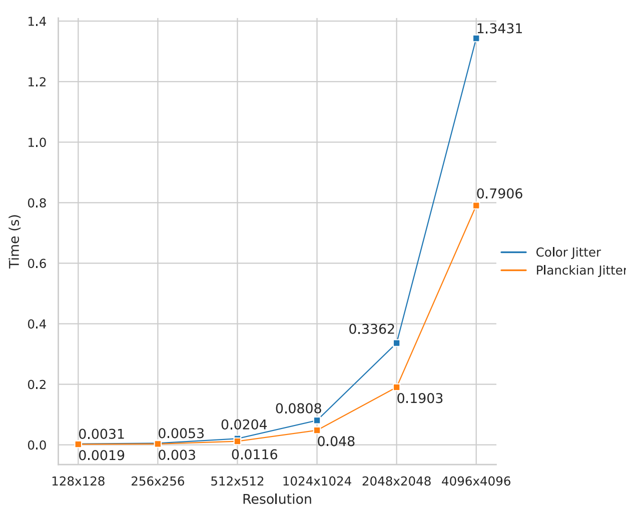


Figure 8: Comparison of execution time between the proposed *Plackian Jitter* transform and the Color Jitter implementation in Pytorch Torchvision v0.9.1. For each resolution we executed both the algorithms 40 times.

last layer of the encoders trained with different data augmentation configurations in order to compare the models sensitivity to color and how it changes in relation to the training procedure adopted.

## 4 Color sensitivity

To analyze feature robustness to different illuminants, we tested the models with different, re-illuminated versions of the CIFAR-100 and FLOWERS-102 datasets. We applied *Planckian Jitter* on the two datasets, generating 25 different versions of each, one for each illuminant sampled. Using these different versions of the datasets we then test the models for each illuminant and collect the classification accuracies. The results on both CIFAR-100 and FLOWERS-102 are given in Figure 7.

Table 5: Additional analysis on downstream tasks. Self-supervised training is performed on TINY-IMAGENET at $(64 \times 64)$.

| DATA AUGMENTATION | TINY-IMAGENET | FLOWERS-102 | CUB200 | VEGFRU | T1K+ |
|---|---|---|---|---|---|
| None | 27.06% | 37.65% | 18.76% | 24.07% | 35.82% |
| Default Color Jitter (CJ) | 33.12% | 46.27% | 19.36% | 23.92% | 26.01% |
| Default Color Jitter w/o Grayscale (CJ-) | 31.62% | 40.39% | 21.90% | 27.39% | 32.50% |
| Planckian Jitter (PJ) | 30.95% | 52.35% | 25.12% | 28.94% | 32.51% |
| LSC: [CJ,CJ-] | 39.02% | 58.33% | 26.82% | 36.43% | 37.20% |
| LSC: [CJ,PJ] | 39.23% | 61.57% | 30.45% | 39.65% | 38.20% |

# 5 Execution time comparison

Here we provide an execution time comparison performed to assess the usability of the proposed data augmentation compared to the already standard Color Jitter data augmentation algorithm. We executed the two algorithms: the Color Jitter image transform from PyTorch Torchvision package (respectively at versions v1.8.1 and v0.9.1) and the proposed *Planckian Jitter* at different image resolutions. For each resolution we ran the code 40 times and averaged the execution time. Results are shown in Figure 8. All augmentations were performed in CPU using an Intel i7-8700 processor. As can be seen, the proposed *Planckian Jitter* is faster with respect to the standard color Color Jitter algorithm.

# 6 Additional downstream results on Tiny-Imagenet

We also performed experiments for several other configurations of the downstream tasks with the representation trained on Tiny-ImageNet. In Table 5 we report results for the main task and downstream task (as in section 4.4 of the main paper ImageNet, but here all images are at $64 \times 64$ pixel resolution.

These additional comparisons confirm the conclusions described in section 4.4 of the main paper. For all of the considered downstream tasks the application of the proposed data augmentation procedure improves the results even in comparison with other combinations of the originally used data augmentations. Moreover, the comparison with the latent space combination with the two versions of the default color jitter shows how exploiting features extracted by the model trained using the proposed Planckian Jitter augmentation enriches the expressive power of the final model.

# References

[1] Claudio Cusano, Paolo Napoletano, and Raimondo Schettini. T1k+: A database for benchmarking color texture classification and retrieval methods. *Sensors*, 21(3):1010, 2021.

[2] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

[3] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE, 2008.

[4] Ivet Rafegas and Maria Vanrell. Color encoding in biologically-inspired convolutional neural networks. *Vision research*, 151:7–17, 2018.

[5] Yushan Feng Saihui Hou and Zilei Wang. Vegfru: A domain-specific dataset for fine-grained visual categorization. In *IEEE International Conference on Computer Vision*, 2017.

[6] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010.