Supplementary material

"Grafit: Learning fine-grained image representations with coarse labels"

In this supplementary material we report additional analyses, results and examples that complement our paper. Appendix A presents two experiments on the impact of self-supervised losses on the granularity and distribution of embeddings. Appendix B contains a detailed description of the experimental settings and complementary experimental analysis and Tables that we have pointed out in our main paper. Appendix C presents additional visualizations of the ranking obtained with Grafit.

A. About granularity

Is it possible to create representations that discriminate between classes finer than the available coarse labels? Considering that we have seen only coarse labels at training time, how can we exploit the coarse classifier for fine-grained classification, if useful at all? In this section we discuss these two questions and construct an experiment to analyze the role of the losses and of the coarse classifier. We then provide empirical observations.

Practical setup. In the following two experiments, we consider the CIFAR-100 benchmark that has two granularity levels with 20 and 100 classes (see Section 4.1).

We denote by f the Resnet-18 trunk mapping from the image space to an embedding space. We train the neural network trunk f with three possible losses:

- Baseline: regular cross-entropy classification training $\mathcal{L}_{\text{CE}}$ with coarse or fine classes;
- Triplet loss: training a triplet loss $\mathcal{L}_{\text{Triplet}}$ to differentiate between image instances (does not use the labels);
- $\mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{Triplet}}$: sum of the two losses. This is intended to be a simple proxy of Grafit.

A.1. Experiment: separating arbitrary fine labels

This experiment is inspired both by the Rademacher complexity [8] and by the self-supervised learning (SSL) literature [4]. In SSL, the standard way to evaluate the quality of a feature extractor f is to measure the accuracy of the network after learning a linear classifier l for the target classes on top of f. The Rademacher complexity measures how a class of functions (*i.e.* $l \circ f$, with f fixed and l learned) is able to classify a set of images with random binary labels.

For this experiment we train the trunk f jointly with a (coarse class) classifier with $\mathcal{L}_{\text{CE}}$ using coarse labels. We hope to improve the granularity of f, i.e. improve the network trunk such that a (finer-grained) classifier l trained on top of f performs better at discriminating between instances that have the same coarse label.

Table 9: Separability experiment on CIFAR-100. The trunk is trained with coarse labels only. Images with the same coarse label are randomly grouped into two distinct fine-grain labels (40 distinct labels in total). Then we fine-tune a linear classifier on this synthetic labels and measure the top-1 accuracy on fine-labels. When conditioning, the estimator exploits the hierarchy: we first predict the coarse class and condition on it to make the final prediction. We report results with three training losses.

Training loss	Top-1 (%) no cond.	Top-1 (%) coarse cond.
$\mathcal{L}_{\text{CE}}$	53.7 ±0.3	54.5 ±0.3
$\mathcal{L}_{\text{Triplet}}$	26.4 ±0.3	—
$\mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{Triplet}}$	57.1 ±0.2	**58.5** ±0.3
Random network	8.7 ±0.3	—

Random labels. We generate synthetic fine labels by the following process: for each coarse label, we randomly and evenly split the training images into two subcategories, yielding 40 classes in total. Inspired by the empirical Rademacher estimation, we sample 10 distinct splits of random labels. For each split, we learn a linear classifier l on top of f_i. We then compute the mean accuracy (top-1, %) of $l \circ f_i$ on the training examples for the three losses. By evaluating to what extent one can fit a linear classifier l on top of f, this experiment measures how well the data are spread in the representation spaces.

Impact of the loss terms. We report the results in Table 9. We can see that, to distinguish between our synthetic fine labels, training with the triplet loss $\mathcal{L}_{\text{Triplet}}$ in combination with the classification loss $\mathcal{L}_{\text{CE}}$ is essential: the sum of losses outperforms each individual loss.

Conditioning. We also measure the impact of *conditioning on coarse classes*: we first predict the coarse label with the coarse classifier, and leverage its softmax output to classify the fine class. This clearly improves the accuracy, which motivates our fusion strategy inspired by this conditioning in Section 3.2.

Table 10: Top-1 accuracy of a ResNet-18 on CIFAR-100 for different training schemes. We report the results after finetuning of the linear classifier on the fine labels (see Section A). The Triplet training is unsupervised, therefore the results for the two columns are identical.

Method	Train coarse	Train fine
$\mathcal{L}_{\text{CE}}$	80.4 ± 0.2	80.6 ± 0.2
$\mathcal{L}_{\text{Triplet}}$	76.5 ± 0.2	76.5 ± 0.2
$\mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{Triplet}}$	**80.9** ± 0.2	**81.3** ± 0.2

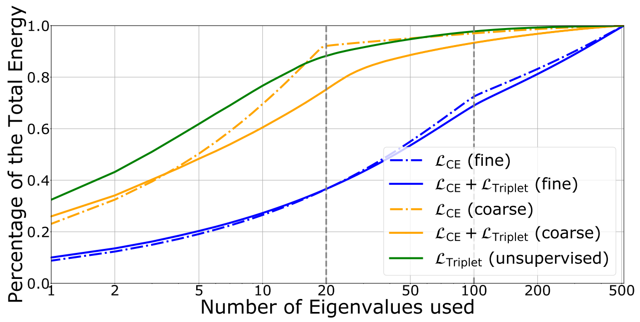

Figure 5: Cumulative energy of the PCA decomposition of CIFAR-100 image embeddings, depending on the granularity of the training labels (20 or 100 classes).

A.2. Experiment: varying the training granularity

In this section we make empirical observations related to the training granularity in the embedding space.

We train f with one of the three losses and either coarse or fine labels as supervision. In a second stage, we train a linear classifier l on the Resnet-18 trunk with fine class supervision, and evaluate its accuracy on the test set.

Accuracies. We first quantify the quality of the representation space. The accuracies are reported in Table 10. We observe that the coarse labels are almost as good as the fine labels as a pre-training. The unsupervised $\mathcal{L}_{\text{Triplet}}$ loss performs significantly worse, which concurs with our previous separability experiment. Combining this loss with the $\mathcal{L}_{\text{CE}}$ loss improves is, both with coarse and fine supervisions.

Size of the representation space. We quantify the information content of embedding vectors by computing their principal components analysis (PCA). This is a reasonable proxy for information content given that the features are separated by linear classifiers afterwards. We observe the cumulative energy of the PCA components ordered by decreasing energy. We assume that a more uniform energy distribution (and thus lower curves) means that the representation is richer, since a few vector components cannot summarize it.

Figure 5 shows the results. When training with $\mathcal{L}_{\text{CE}}$ loss, the most uniform distribution for the principal components is obtained for the fine supervision. This is expected since it is a finer-grain separation of entities and that can not be summarized with a subspace as small as the one associated with a relatively small number of categories. Notice that the training granularity (20 or 100 classes) can be read as an inflexion point on the PCA decomposition curves. The loss $\mathcal{L}_{\text{Triplet}}$ is not very informative on its own but does improve the cross-entropy representation when combined with it.

Discussion. This simple preliminary experiment shows that the label granularity has a strong impact on very basic statistics of the embedding distribution. It is the basic intuition behind Grafit: a rich representation can be obtained using just coarse labels, if we combine them with a self-supervised loss.

B. Additions to the experiments

This section details the training procedure of Grafit and provides more extensive experimental results.

B.1. Training settings

As described in the main part, our training procedure is inspired by Tong et al. [28]: we use SGD with Nesterov momentum and cosine learning rates decay. We follow Goyal et al.'s [24] recommendation for the learning rate magnitude: $\text{lr} = \frac{0.1}{256} \times \text{batchsize}$. The augmentations include random resized crop, RandAugment [14] and Erasing [70]. We train for 600 epochs with batches of 1024 images of resolution 224×224 pixels (except for CIFAR-100 where the resolution is 32×32). For Grafit with $\mathcal{L}_{\text{inst}}$ we use $T = 4$ different data-augmentations on ImageNet and $T = 8$ on CIFAR-100. For the supervised loss we use one data-augmentation in order to have the same training procedure as our supervised baseline.

Weighting of the losses. We investigate the impact of weighting the losses $\mathcal{L}_{\text{knn}}$ and $\mathcal{L}_{\text{inst}}$. For example, on CIFAR-100 classification, Table 11 shows that an equal weighting gives the best or near-best results. Therefore, to avoid adding a hyper-parameter and in order to simplify the method, we chose to not use weighting, *i.e.* we just sum up the two losses.

A strong Baseline. Our training procedure improves the ResNet-50 performance and thus is a strong baseline against which we can compare Grafit. Therefore, Table 12 compares our baseline on ImageNet with other ResNet-50 training procedures. We observe that our training procedure gives better results than many other approaches. This makes it possible to isolate the contribution of our improved training practices and that of the Grafit loss.

Table 11: Category-level (mAP, %) and one-the-fly kNN classification (top-1, %) performance in a coarse-to-fine setting on CIFAR-100 with different loss weighting. Our total loss is defined as $\mathcal{L}_{\rm tot}(x) = \mathcal{L}_{\rm knn}(g_\theta(x), y) + \lambda \mathcal{L}_{\rm inst}(x)$ with λ being a real-valued coefficient.

λ	0.0	0.2	0.4	0.6	0.8	1.0	1.2	1.4
mAP	35.9	46.3	49.6	51.4	52.4	52.9	52.8	52.4
kNN	72.2	70.0	73.2	74.8	75.8	77.7	77.4	77.7

Table 12: Performance comparison (top-1 accuracy) with our ResNet-50 baseline and state of the art ResNet-50 on ImageNet. All results are with single center crop evaluation with image resolution 224×224.

Method	Extra-data	Top-1 (%)
ResNet-50 [27] PyTorch	-	76.2
RandAugment [14]	-	77.6
CutMix [68]	-	78.6
Noisy-Student [63]	JFT-300M [63]	78.9
Billion Scale [66]	YFCC100M [52]	79.1
Our Baseline	-	**79.3**

B.2. {coarse,fine}-to-{coarse,fine}: evaluation

We compare our main baselines and Grafit's performance in the 4 following scenarios: coarse-to-coarse, coarse-to-fine, fine-to-fine and fine-to-coarse. The evaluations are performed with two classifiers: a kNN classifier (kNN) on top of the embeddings and a linear classifier with the fine-tuned network (FT) with a cross-entropy loss.

The results in Table 13 show that Grafit training improves the accuracy in almost all settings, including the fine-to-fine setting, which is just regular classification with the vanilla labels for Imagenet.

B.3. Coarse-to-Fine with different taxonomic rank

Datasets. We carry out evaluations on iNaturalist-2018, and with iNaturalist-2019 [31], which is a subset of iNaturalist-2018 [30] where classes with too few images have been removed. iNaturalist 2019 dataset is thus composed of 268,243 images divided into 1,010 classes at the finest level. From the coarse to the finest level, we have 3 classes for Kingdom, 4 classes for Phylum, 9 classes for Class, 34 classes for Order, 57 classes for Family, 72 classes for Genus and 1,010 classes for Species.

Results. We report exhaustive results with our two coarse-to-fine evaluation protocols with all our baselines on iNaturalist-2018 [30] and iNaturalist-2019 [31] in Table 14.

We comment more specifically the kNN classification accuracy (left) because for retrieval, Grafit outperform all

Table 13: Performance comparison (top-1 accuracy) when learning and testing at different granularities (ResNet50). For CIFAR-100, there are 100 fine and 20 coarse concepts. ImageNet covers 1000 fine and 127 coarse concepts. We report the results of both the kNN classifier and of a linear classifier fine-tuned with the target granularity (FT).

	Method	↓ Test	Train Coarse		Train Fine	
			kNN	FT	kNN	FT
CIFAR-100	Baseline	Coarse	89.3 $\pm$0.1	89.4 $\pm$0.2	90.3 $\pm$0.1	90.5 $\pm$0.2
	SNCA+		88.4 $\pm$0.3	88.9 $\pm$0.3	88.8 $\pm$0.1	90.2 $\pm$0.1
	Grafit		**90.6** $\pm$0.1	**90.6** $\pm$0.1	**90.6** $\pm$0.3	**90.9** $\pm$0.2
	Baseline	Fine	71.8 $\pm$0.3	82.3 $\pm$0.2	82.7 $\pm$0.2	82.7 $\pm$0.2
	SNCA+		72.2 $\pm$0.3	82.0 $\pm$0.4	81.7 $\pm$0.1	82.9 $\pm$0.1
	Grafit		**77.7** $\pm$0.2	**83.7** $\pm$0.2	**83.2** $\pm$0.3	**83.7** $\pm$0.2
ImageNet-1k	Baseline	Coarse	87.0 $\pm$0.1	**87.6** $\pm$0.1	87.4 $\pm$0.1	87.9 $\pm$0.1
	SNCA+		87.7 $\pm$0.1	87.5 $\pm$0.1	88.9 $\pm$0.1	87.2 $\pm$0.1
	Grafit		**88.4** $\pm$0.1	87.3 $\pm$0.1	**89.2** $\pm$0.1	**87.7** $\pm$0.1
	Baseline	Fine	54.7 $\pm$0.2	**78.1** $\pm$0.1	78.0 $\pm$0.1	**79.3** $\pm$0.1
	SNCA+		55.4 $\pm$0.2	77.9 $\pm$0.1	79.1 $\pm$0.1	77.4 $\pm$0.1
	Grafit		**69.1** $\pm$0.2	77.9 $\pm$0.1	**79.6** $\pm$0.1	78.0 $\pm$0.1

the baselines by a large margin. The table on the right is divided in 10 matrices each containing results for one combination of a method and a dataset (iNaturalist 2018 or 2019).

The diagonal values in the matrices correspond to a traditional setting where the training and the test granularity are the same. Even in this case, the Grafit descriptors outperforms the baseline methods most often. On iNaturalist 2018, for Species, the finest and most challenging level, the additional Grafit loss improves the top-1 accuracy by 7% absolute. The gain is more marginal for iNaturalist 2019 (+0.9%), which shows that Grafit is especially useful for unbalanced class distributions where some classes are in a low-shot training regime.

The lower triangle of each matrix reports the coarse-to-fine results, which is the setting in which we focus in our paper. Grafit obtains the best results for most combinations, with accuracy gains of around 10 points with respect to the baseline and by a few points for ClusterFit+. It is interesting to look at the $\varnothing$ column, which is the unsupervised case. In that case, the baseline training reduces to a random network, but Grafit is able to extract signal from the kNN loss.

The upper triangle is the fine-to-coarse setting, where finer labels are available for the training images than what is used at test time. This is obviously not the setting of the paper but it is worth discussing these results. A natural baseline for fine-to-coarse is to discard the fine labels and train only with the coarse labels induced by the fine annotation. This would yield the same accuracy as on the corresponding entry of the diagonal of the matrix. Irrespective of the method, the fine-to-coarse training does not necessarily outperform this simple strategy.

Table 14: Evaluation on iNaturalist-2018/2019 with all combinations of training / testing semantic levels. *Left:* on-the-fly k-NN classification accuracy (top-1, %) *Right:* category-level retrieval (mAP, %). We highlight the best and second-best result across methods for each train-test granularity combination. The diagonals (test = train granularity) are in **bold**. Lower triangles are coarse-to-fine combinations, handled in the paper. Upper triangles (fine-to-coarse) are reported for reference but not formally addressed by our approach: better strategies would exploit the hierarchy of concepts more explicitly.

	Test \ Train	∅	King.	Phyl.	Class	Order	Fam.	Gen.	Spec.
					iNaturalist-2018				
	# classes:	1	6	25	57	272	1118	4401	8142
Baseline	Kingdom	70.9	**97.6**	98.0	98.1	98.2	98.2	97.9	97.5
	Phylum	48.8	88.0	**96.3**	96.4	96.6	96.7	96.2	95.2
	Class	40.4	77.1	86.7	**94.1**	94.7	94.8	94.1	92.9
	Order	17.1	43.6	55.0	61.0	**85.6**	86.6	85.5	82.6
	Family	5.6	23.0	32.8	36.7	62.0	**80.7**	79.7	76.1
	Genus	0.9	10.0	17.3	20.1	41.7	63.0	**72.5**	68.3
	Species	0.3	6.3	11.5	13.6	31.2	51.3	61.6	**60.2**
SNCA+	Kingdom	71.2	**97.7**	97.9	98.1	97.9	98.0	98.2	98.3
	Phylum	48.0	68.7	**96.1**	96.4	96.4	96.5	96.7	96.7
	Class	39.4	56.7	84.8	**93.9**	94.3	94.6	94.7	94.7
	Order	16.2	23.3	47.4	59.0	**85.4**	86.2	86.7	86.7
	Family	5.2	7.8	23.2	33.2	57.8	**80.2**	81.1	81.3
	Genus	0.9	1.3	10.4	17.6	36.9	56.8	**74.2**	74.1
	Species	0.3	0.5	6.3	11.9	26.2	42.4	58.9	**64.6**
ClusterFit+	Kingdom	70.9	**94.7**	95.0	95.3	95.6	96.2	96.3	96.1
	Phylum	48.8	87.4	**90.3**	90.7	91.1	92.6	92.6	92.2
	Class	40.4	80.2	83.8	**85.7**	86.7	88.8	88.8	88.2
	Order	17.1	54.5	59.0	61.4	**70.8**	73.9	74.3	72.3
	Family	5.6	38.3	42.1	44.4	54.3	**63.0**	64.2	61.9
	Genus	0.9	26.7	29.5	31.5	40.1	49.4	**53.9**	51.7
	Species	0.3	21.8	23.7	25.2	32.7	40.3	44.7	**43.4**
Grafit FC	Kingdom	91.1	**97.8**	98.1	98.4	98.3	98.4	98.5	98.4
	Phylum	81.7	93.0	**96.4**	96.9	97.0	96.9	97.1	96.8
	Class	71.9	86.0	90.7	**94.8**	95.0	95.1	95.3	95.0
	Order	41.8	58.5	66.8	72.2	**86.8**	87.1	87.3	87.2
	Family	22.4	38.8	48.4	54.4	70.4	**81.1**	81.6	81.7
	Genus	11.4	24.6	33.1	38.6	53.0	63.9	**73.8**	74.2
	Species	8.13	18.8	25.6	29.9	41.5	50.9	60.9	**65.9**
Grafit	Kingdom	95.5	**98.1**	98.2	98.2	98.2	98.2	98.4	98.3
	Phylum	90.0	94.1	**96.6**	96.7	96.8	96.7	96.9	96.7
	Class	82.2	87.5	90.9	**94.5**	94.9	94.9	95.0	95.0
	Order	54.0	61.7	66.9	72.7	**87.1**	87.5	87.6	87.3
	Family	33.7	42.1	48.7	55.1	70.9	**81.8**	82.4	82.1
	Genus	20.5	27.0	33.5	39.5	54.2	64.6	**75.6**	75.5
	Species	15.9	20.4	25.5	30.8	42.7	51.2	61.9	**67.7**
					iNaturalist-2019				
	# classes:	1	3	4	9	34	57	72	1010
Baseline	Kingdom	77.0	**98.9**	98.9	99.0	99.3	99.4	99.3	98.9
	Phylum	73.8	97.1	**98.7**	98.9	99.2	99.2	99.2	98.7
	Class	63.3	87.6	90.3	**98.0**	98.5	98.6	98.6	98.0
	Order	17.9	49.6	56.4	70.8	**95.6**	95.5	96.0	95.2
	Family	12.4	42.1	50.4	65.0	89.4	**94.8**	95.1	94.4
	Genus	9.6	39.2	46.5	62.1	86.1	91.5	**94.8**	93.9
	Species	1.5	9.8	13.5	20.6	34.5	39.9	42.4	**75.0**
SNCA+	Kingdom	76.9	**98.6**	98.9	99.2	99.2	99.3	99.1	99.0
	Phylum	73.3	87.1	**98.8**	99.1	99.1	99.1	98.9	99.0
	Class	62.3	74.9	84.1	**98.2**	98.6	98.3	98.1	97.8
	Order	17.6	19.7	30.2	55.4	**95.3**	95.2	95.2	94.2
	Family	12.2	12.7	20.7	45.5	88.2	**94.5**	94.6	93.5
	Genus	9.3	9.2	17.1	41.6	85.0	91.2	**94.0**	93.1
	Species	1.3	1.0	1.8	10.4	36.0	40.8	42.3	**74.7**
ClusterFit+	Kingdom	77.0	**96.4**	96.1	95.8	95.7	95.7	95.4	97.0
	Phylum	73.8	94.2	**95.0**	94.6	94.3	94.4	93.8	95.5
	Class	63.3	88.7	90.1	**91.3**	90.1	90.9	90.6	93.5
	Order	17.9	65.5	67.9	70.9	**76.8**	79.0	78.1	83.2
	Family	12.4	59.5	62.0	65.4	71.7	**75.6**	75.3	80.4
	Genus	9.6	56.9	59.3	62.7	68.7	72.6	**73.9**	78.6
	Species	1.5	24.5	25.6	27.3	31.1	33.6	33.9	**49.6**
Grafit FC	Kingdom	93.1	**98.9**	99.0	99.2	99.2	99.4	99.4	99.2
	Phylum	90.9	98.2	**98.9**	99.1	99.1	99.3	99.2	99.0
	Class	82.6	94.9	96.4	**98.2**	98.3	98.7	98.7	98.3
	Order	52.5	80.0	83.5	89.5	**95.8**	96.0	95.9	95.3
	Family	45.3	74.6	78.9	86.0	93.4	**95.2**	95.4	94.7
	Genus	41.7	71.7	76.3	84.0	91.7	93.4	**95.0**	94.3
	Species	12.0	29.5	32.6	40.4	51.8	53.2	53.9	**75.9**
Grafit	Kingdom	96.9	**99.2**	99.1	99.2	99.2	99.0	99.0	99.0
	Phylum	96.4	98.8	**98.9**	99.0	99.0	98.9	98.9	98.7
	Class	93.0	97.0	97.1	**98.2**	98.4	98.3	98.1	97.8
	Order	81.3	89.0	89.3	91.2	**95.9**	95.3	95.3	94.5
	Family	76.5	85.2	85.2	87.8	93.1	**94.5**	94.5	93.8
	Genus	73.8	82.7	83.1	85.8	91.3	92.6	**94.2**	93.4
	Species	31.0	41.6	41.4	46.0	51.8	53.5	55.3	**75.3**

	Test \ Train	King.	Phyl.	Class	Order	Fam.	Gen.	Spec.
					iNaturalist-2018			
	# classes:	6	25	57	272	1118	4401	8142
Baseline	Kingdom	**97.8**	86.3	81.0	76.4	65.9	62.1	61.5
	Phylum	64.2	**96.6**	82.1	63.1	45.9	39.8	38.1
	Class	46.0	72.1	**93.8**	60.1	39.2	31.2	28.5
	Order	12.2	24.1	34.3	**74.5**	35.4	20.1	15.6
	Family	3.69	7.02	10.1	32.6	**51.3**	20.9	14.5
	Genus	1.30	3.06	4.47	16.6	30.4	**33.3**	24.0
	Species	1.18	2.63	3.63	12.8	25.7	31.4	**27.9**
SNCA+	Kingdom	**97.6**	83.3	75.9	59.2	56.0	54.9	55.0
	Phylum	59.8	**91.7**	79.4	49.1	35.0	32.3	32.2
	Class	41.3	73.1	**89.9**	49.2	28.1	23.6	23.0
	Order	9.09	24.9	35.7	**77.9**	35.3	18.0	15.0
	Family	2.24	6.43	11.2	35.7	**68.4**	29.1	21.7
	Genus	0.39	2.47	5.03	18.1	36.6	**60.5**	46.0
	Species	0.19	1.86	3.80	12.8	26.4	46.0	**54.9**
ClusterFit+	Kingdom	**55.5**	55.5	55.7	56.4	57.0	57.6	57.7
	Phylum	31.6	**32.1**	32.1	32.4	33.1	33.9	34.0
	Class	21.0	21.6	**22.0**	22.2	23.0	23.7	23.8
	Order	6.8	7.4	7.8	**9.4**	9.9	10.3	10.1
	Family	2.9	3.5	3.9	5.5	**7.8**	7.9	7.3
	Genus	3.6	4.3	4.8	7.1	10.8	**13.3**	12.0
	Species	4.7	5.4	5.9	8.6	12.5	15.3	**14.5**
Grafit FC	Kingdom	**98.5**	88.3	80.6	61.6	57.7	56.0	56.0
	Phylum	69.6	**97.2**	83.1	50.5	37.9	33.9	33.3
	Class	52.4	75.8	**95.7**	51.3	31.3	25.5	24.5
	Order	18.6	31.4	41.6	**88.0**	41.6	20.4	16.4
	Family	7.68	12.9	17.7	44.7	**82.4**	33.4	23.6
	Genus	4.97	8.82	11.9	27.3	45.0	**75.5**	52.2
	Species	4.95	8.25	10.7	21.4	33.6	53.8	**68.1**
Grafit	Kingdom	**98.6**	88.3	79.7	60.8	58.0	55.9	55.5
	Phylum	67.8	**97.2**	82.1	50.9	38.9	34.2	33.0
	Class	50.1	74.9	**95.4**	51.2	32.3	25.9	24.1
	Order	17.7	30.7	42.7	**88.3**	42.3	21.1	16.2
	Family	8.70	13.2	18.0	43.9	**83.1**	34.8	24.2
	Genus	6.78	9.72	13.5	29.0	46.9	**77.2**	53.9
	Species	6.45	9.02	12.1	23.6	35.6	55.4	**70.0**
					iNaturalist-2019			
	# classes:	3	4	9	34	57	72	1010
Baseline	Kingdom	**99.0**	98.2	88.9	73.6	65.8	67.4	58.6
	Phylum	87.1	**98.9**	90.8	71.7	59.8	61.6	51.7
	Class	67.2	77.6	**98.2**	68.8	55.1	56.3	42.8
	Order	15.1	21.1	33.7	**94.8**	68.6	57.6	26.2
	Family	9.72	13.8	24.2	70.7	**94.2**	80.6	31.5
	Genus	7.77	11.0	21.3	59.6	81.4	**93.9**	34.8
	Species	1.09	1.55	3.60	10.8	14.8	16.6	**57.0**
SNCA+	Kingdom	**98.4**	90.1	82.0	63.5	60.9	60.3	55.0
	Phylum	84.1	**97.7**	87.7	62.6	55.9	55.3	49.3
	Class	63.2	75.6	**95.5**	59.0	50.0	49.1	38.5
	Order	11.5	17.2	32.4	**83.0**	64.3	54.4	15.7
	Family	6.53	10.0	20.1	75.2	**90.9**	78.8	19.5
	Genus	5.08	7.61	18.1	71.5	84.6	**92.8**	22.0
	Species	0.40	0.65	2.11	15.4	17.1	18.6	**72.3**
ClusterFit+	Kingdom	**55.1**	55.0	54.7	54.4	54.5	54.6	55.5
	Phylum	49.0	**49.1**	48.8	48.2	48.3	48.4	49.3
	Class	36.8	36.9	**37.1**	36.3	36.4	36.5	37.6
	Order	7.5	7.7	8.2	**8.4**	8.5	8.4	9.7
	Family	5.6	5.9	6.4	6.8	**7.4**	7.3	8.9
	Genus	4.9	5.2	5.8	6.1	6.8	**6.9**	8.6
	Species	2.4	2.5	2.9	3.5	4.1	4.2	**10.1**
Grafit FC	Kingdom	**99.2**	93.2	86.5	63.8	62.3	62.2	56.1
	Phylum	88.6	**99.2**	90.7	63.0	58.9	57.9	50.5
	Class	70.4	80.9	**98.5**	61.6	53.9	52.5	39.9
	Order	25.1	32.6	45.4	**96.3**	70.3	58.6	18.4
	Family	20.8	26.7	38.2	84.5	**95.8**	82.2	23.0
	Genus	18.9	24.7	34.0	78.3	88.4	**95.7**	25.7
	Species	6.83	9.63	14.7	28.4	29.7	31.5	**78.4**
Grafit	Kingdom	**99.4**	93.1	85.9	62.7	61.5	60.9	56.6
	Phylum	88.8	**99.2**	90.2	62.6	58.4	57.2	51.0
	Class	71.8	81.9	**98.6**	61.3	53.2	52.0	40.4
	Order	30.7	36.6	48.1	**96.4**	69.3	58.3	19.1
	Family	28.2	30.8	41.8	82.5	**95.1**	81.5	23.7
	Genus	28.0	29.8	40.5	76.2	87.5	**94.8**	26.3
	Species	18.7	18.9	21.8	32.5	33.3	34.7	**77.5**

B.4. Transfer Learning Tasks

This section details the experimental settings for the transfer learning and reports more results and comparisons.

Fine-tuning settings As described in Section 4.6 We initialize the network trunk with ImageNet pre-trained weights and fine-tune the model.

For our method, we keep the pre-trained network trunk f_θ. But the projector P_θ is discarded. For all methods we fine-tune during 240 epochs with a cosine learning rate schedule starting at 0.01, and batches of 512 images.

For fine-tuning results in Table 8 we additionally use Cutmix [68] and FixRes [53] during fine-tuning and we fine-tune with more epochs (1000 for Flowers [41] and Cars [35], 300 for Food-101 [7] and iNaturalist [30, 31]). These choices improve the performance for all the methods.

Results. Table 15 compares the performance obtained with Grafit for different architectures. We report results with Grafit topped with either a multi-layers perceptron (MLP) or a linear classifier (FC). The accuracy increases for larger models. This shows that, although ResNet-50 serves as a running example architecture for Grafit, the method applies without modifications to other architectures.

Table 16 compares the performances= obtained with Grafit and baselines with MLP and FC classifier. For all settings, the flexibility of the MLP is useful to outperform the linear classifier (FC). The transfer learning results are better or as good for Grafit variants. The gap with the baseline methods is higher for the iNaturalist variants. This is because the datasets are more challenging, as evidenced by the relatively low accuracies reported.

C. Visualization

CIFAR. Figure 6 shows for a giving test image in CIFAR-100 the 10 nearest neighbours in the training according the cosine similarity in the embedding space. In Figure 6 models are trained on the 20 CIFAR-100 coarse classes. The correct classes are indicated in green. For example, in the first row, Grafit correctly identifies a butterfly query in 9 out of 10 results, while the baseline method succeeds only 5 times. The second row is a relative failure case, because Grafit confuses a van with a pickup truck. However, it correctly matches the colors of the vehicles.

iNaturalist. Figure 7, 8 and Figure 9 present similar results for three examples for iNaturalist-2018, but with several levels of granularity for the training set, which allow one to vizualize the importance of the training granularity as well. Each granularity level is identified with a color. The frame color around the image indicates which at which granularity the match is correct: for example, light orange means it is correct at the order level and green means that the result is correct at the finest granularity (Species).

We can observe from the colors and the image content that the level at which Grafit is correct is almost systematically better than the baseline[2]. For example, the baseline model trained at the genus granularity in Figure 7 matches the deer query with a moose (rank 3).

In Figure 8, the butterfly is matched relatively easily with other butterflies by both classifiers, even when they are trained with coarse granularity. This is because butterflies have quite distinctive textures. However, Grafit slightly outperforms the baseline for finer granularity levels.

Figure 9 shows an orca query, which is quite distinctive with its black-and-white skin. The baseline method is unable to distinguish it from other marine mammals, even when trained at the finest granularity. Grafit is able to distinguish these textures more accurately, so it gets perfect retrieval results even when trained at the genus granularity.

[2]These examples are representative of typical comparisons, as we have not cherry-picked to show cases where our method is better.

Table 15: Transfer learning task with various architectures pretrained on ImageNet with Grafit. We report the Top-1 accuracy (%) for the evaluation with a single center crop at resolution 224×224.

	ResNeXt50-32x4 [64]		ResNeXt50D-32x4 [28]		ResNeXt50D-32x8 [28]		RegNety-4GF [43]		RegNety-8GF [43]	
# Params	25M		25M		48M		21M		39M	
Dataset	FC	MLP	FC	MLP	FC	MLP	FC	MLP	FC	MLP
Flowers-102 [41]	95.5	**98.3**	95.9	**98.6**	96.3	**98.7**	98.1	**98.6**	**99.0**	98.8
Stanford Cars [35]	91.6	**92.9**	88.7	**93.3**	90.9	**93.8**	**93.3**	92.7	**94.0**	93.4
Food101 [7]	89.6	**89.9**	90.2	**90.3**	90.9	**91.1**	91.2	91.3	92.1	**92.4**
iNaturalist 2018 [30]	67.7	**71.2**	68.9	**72.4**	71.4	**74.4**	73.8	**74.2**	76.4	**76.8**
iNaturalist 2019 [31]	75.3	**76.3**	75.8	**77.6**	77.8	**78.7**	**78.1**	77.9	79.8	**80.0**

Table 16: Transfer learning with ResNet-50 pretrained on ImageNet. Comparison between different pre-training methods and two different classifiers trained on the target domain (a linear FC or an MLP). We report the top-1 accuracy (%) with a single center crop evaluation at resolution 224×224.

	Baseline		SNCA+		ClusterFit+		Grafit		Grafit FC	
Dataset	FC	MLP	FC	MLP	FC	MLP	FC	MLP	FC	MLP
Flowers-102 [41]	96.2	95.7	94.3	**98.2**	96.2	96.1	97.6	**98.2**	94.3	97.6
Stanford Cars [35]	90.0	89.8	91.6	**92.5**	89.4	89.3	91.4	**92.5**	91.4	**92.7**
Food101 [7]	88.2	88.9	88.7	88.8	88.5	88.9	88.9	**89.5**	88.5	88.7
iNaturalist 2018 [30]	65.0	68.4	64.7	69.2	64.2	67.5	65.6	**69.8**	65.2	68.5
iNaturalist 2019 [31]	72.8	73.7	73.1	74.5	71.8	73.8	74.1	**75.9**	73.9	74.6

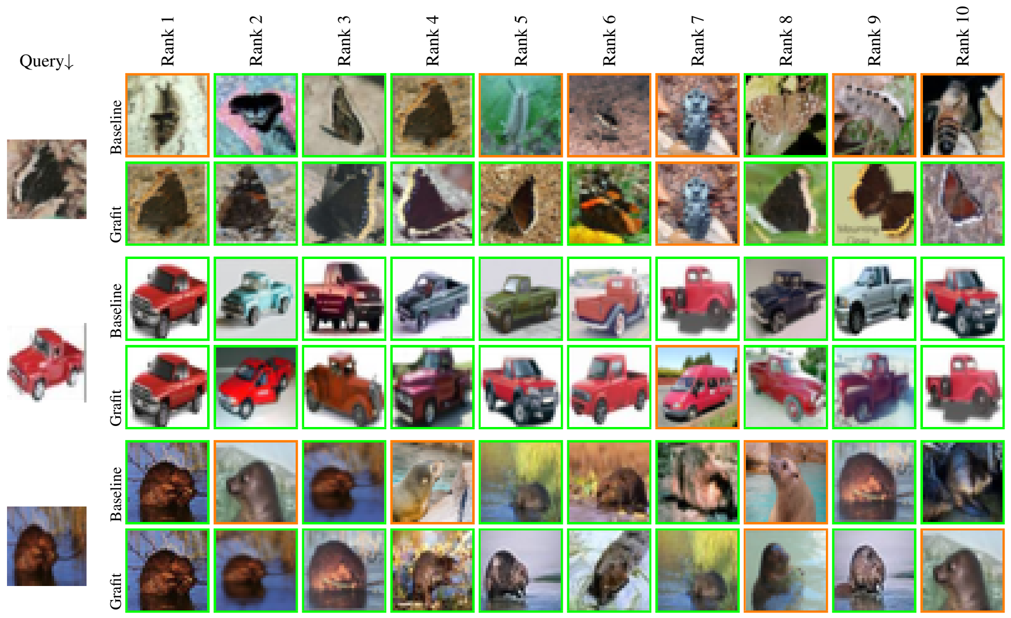

Figure 6: CIFAR-100: For given test images (*top*), we present the ranked list of train images most similar with embeddings obtained with a baseline method (top) and our method (bottom) train with coarse labels. Images in green indicate that the image belongs to the correct fine class; orange indicates the correct coarse class but incorrect fine class. In this example, all results are correct w.r.t. coarse granularity.

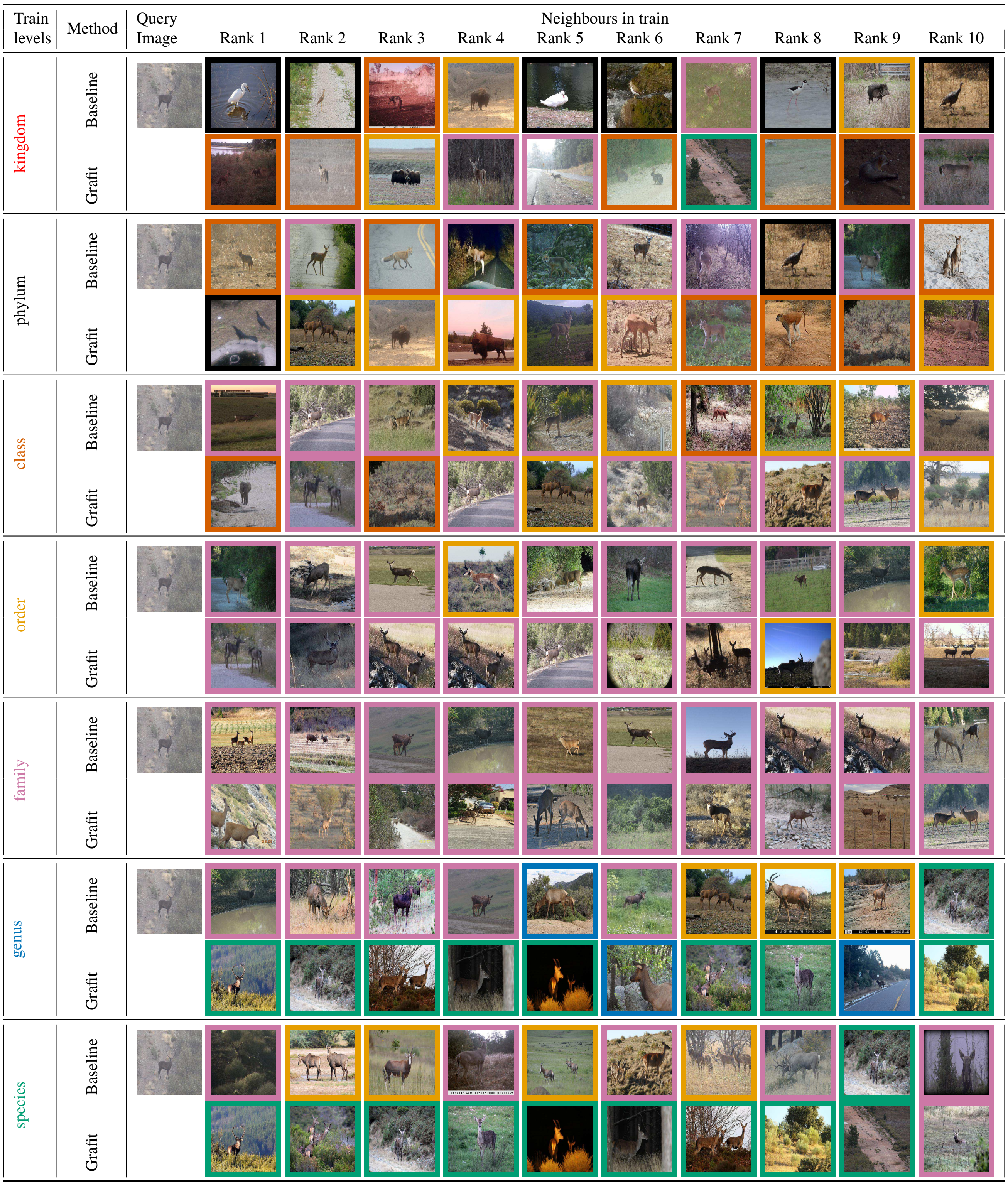

Figure 7: We compare Grafit and Baseline for different training granularity. We rank the 10 closest images in the iNaturalist-2018 train set for a query image in the test set. The ranking is obtained with a cosine similarity on the features space of each of the two approaches. See Table 17 for authors and image copyrights.

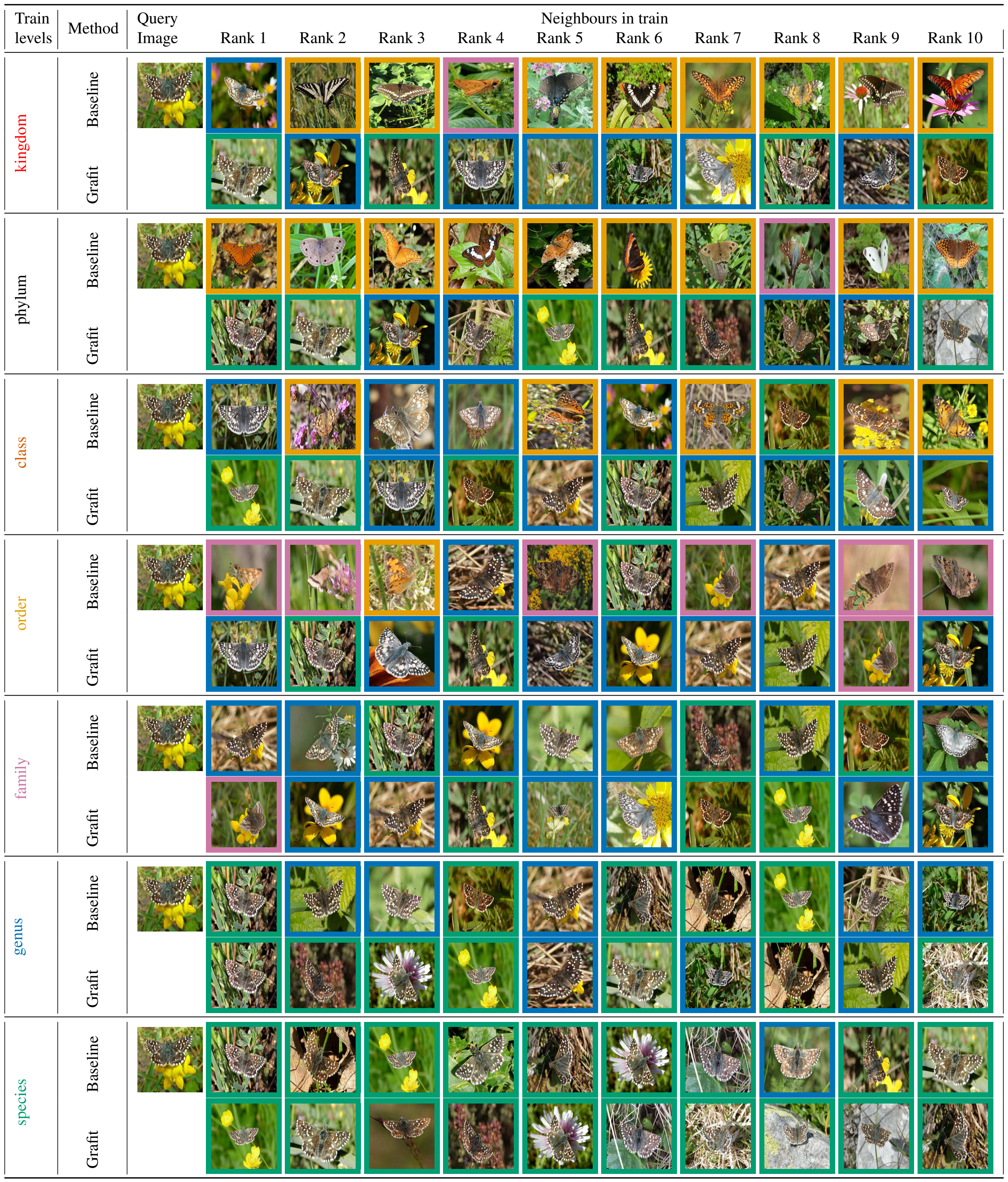

Figure 8: We compare Grafit and Baseline for different training granularity. We rank the 10 closest images in the iNaturalist-2018 train set for a query image in the test set. The ranking is obtained with a cosine similarity on the features space of each of the two approaches. See Table 18 for authors and image copyrights.

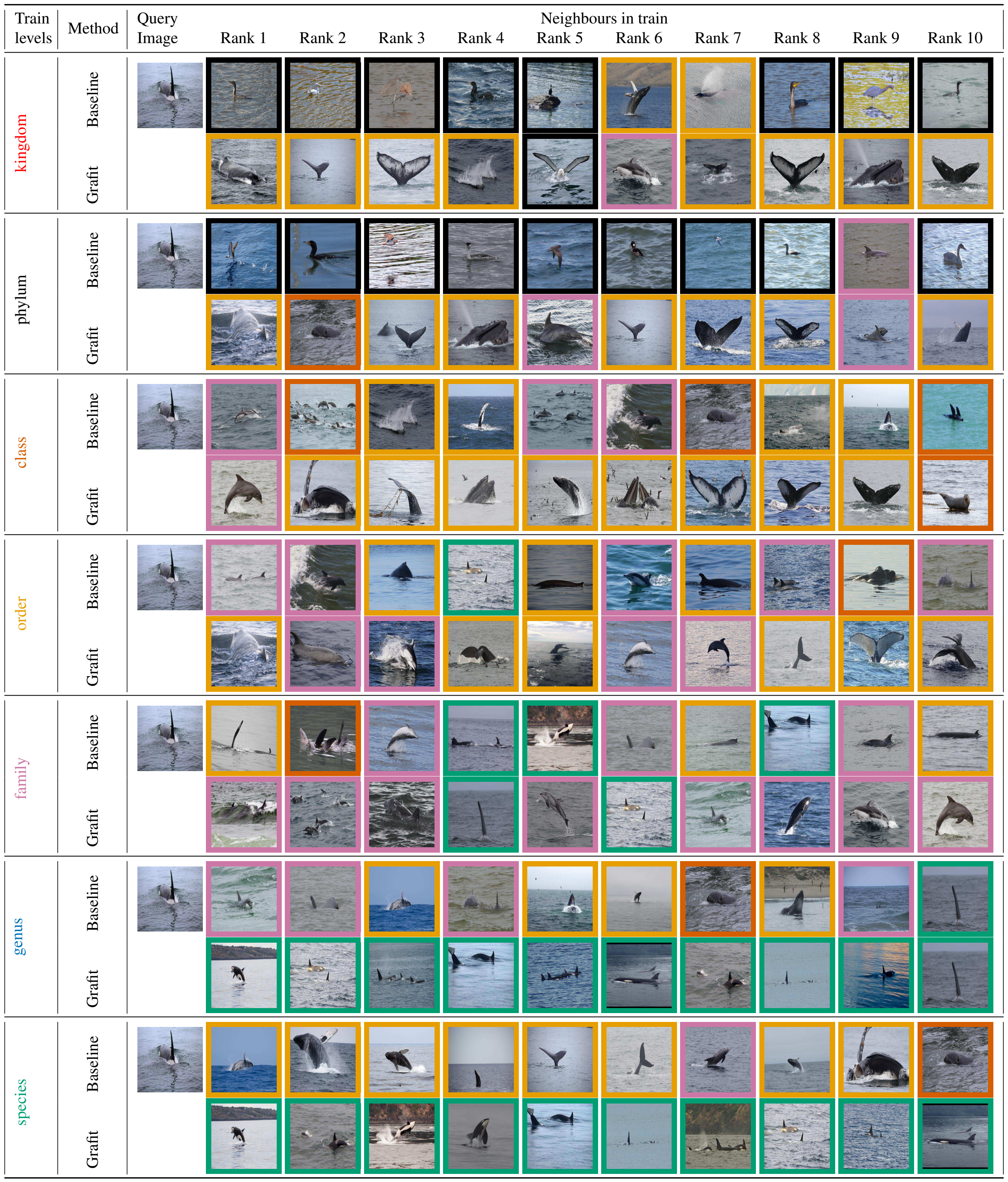

Figure 9: We compare Grafit and Baseline for different training granularity. We rank the 10 closest images in the iNaturalist-2018 train set for a query image in the test set. The ranking is obtained with a cosine similarity on the features space of each of the two approaches. See Table 19 for authors and image copyrights.

Table 17: Author and Creative Commons Copyright notice for images in Figure 7.

sequeirajluis: CC BY-NC 4.0, 101724807574796438015: CC BY-NC 4.0, Juan Carlos Pérez Magaña: CC BY-NC 4.0, ONG OeBenin: CC BY-NC 4.0, Damon Tighe: CC BY-NC 4.0, colinmorita: CC BY-NC 4.0, Bob Hislop: CC BY-NC 4.0, Skyla Slemp: CC BY-NC 4.0, Sam Kieschnick: CC BY-NC 4.0, Derek Broman: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, dfwuw: CC BY-NC 4.0, maractwin: CC BY-NC-SA 4.0, pfaucher: CC BY-NC 4.0, Ryan Blankenship: CC BY-NC 4.0, Gary Chang: CC BY-NC-SA 4.0, Linda Jo: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, pwiedenkeller: CC BY-NC 4.0, dfwuw: CC BY-NC 4.0, dfwuw: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, CC BY-NC 4.0, Bruno Durand: CC BY-NC 4.0, Kyle Jones: CC BY-NC 4.0, Jakub Pełka: CC BY-NC 4.0, ekoaraba: CC BY-NC 4.0, Shane: CC BY-NC 4.0, dfwuw: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, gyrrlfalcon: CC BY-NC 4.0, dengland81: CC BY-NC 4.0, mafzam: CC BY-NC 4.0, ONG OeBenin: CC BY-NC 4.0, Damon Tighe: CC BY-NC 4.0, Cathy Bell: CC BY-NC-ND 4.0, ONG OeBenin: CC BY-NC 4.0, ONG OeBenin: CC BY-NC 4.0, fiddleman: CC BY-NC 4.0, ONG OeBenin: CC BY-NC 4.0, Brian Gratwicke: CC BY 4.0, ONG OeBenin: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, gyrrlfalcon: CC BY-NC 4.0, Simon Kingston: CC BY-NC 4.0, Mark Freeman: CC BY-NC 4.0, Mike: CC BY-NC 4.0, tiyumq: CC BY-NC 4.0, Kyle Jones: CC BY-NC 4.0, abn: CC BY-NC 4.0, James Bailey: CC BY-NC 4.0, Allan Finlayson: CC BY-NC 4.0, joysglobal: CC BY-NC 4.0, sea-kangaroo: CC BY-NC-ND 4.0, Ken-ichi Ueda: CC BY-NC 4.0, driles5: CC BY-NC 4.0, driles5: CC BY-NC 4.0, 112692329998402018828: CC BY-NC-SA 4.0, Andy Jones: CC BY-NC-SA 4.0, almanzacamille: CC BY-NC 4.0, CC BY-NC 4.0, J. Maughn: CC BY-NC 4.0, jameseatonbio1020: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, bio22003jeaton: CC BY-NC 4.0, 112692329998402018828: CC BY-NC-SA 4.0, Sterling Sheehy: CC BY 4.0, Guillermo Debandi: CC BY-NC 4.0, James Bailey: CC BY-NC 4.0, sliverman: CC BY-NC 4.0, Joanne Siderius: CC BY-NC 4.0, janaohrner: CC BY-NC 4.0, ONG OeBenin: CC BY-NC 4.0, mcodellwildlife: CC BY-NC 4.0, Johnny Wilson: CC BY-NC 4.0, sea-kangaroo: CC BY-NC-ND 4.0, Brian Gratwicke: CC BY 4.0, 112692329998402018828: CC BY-NC-SA 4.0, ONG OeBenin: CC BY-NC 4.0, summersilence: CC BY-NC 4.0, efarias1: CC BY-NC 4.0, redhat: CC BY-NC 4.0, Dale Hameister: CC BY-NC 4.0, 116916927065934112165: CC BY-NC-SA 4.0, sequeirajluis: CC BY-NC 4.0, Zac Cota: CC BY-NC 4.0, J Brew: CC BY-SA 4.0, Cullen Hanks: CC BY-NC 4.0, Allan Finlayson: CC BY-NC 4.0, dfwuw: CC BY-NC 4.0, Mark Freeman: CC BY-NC 4.0, mmski303: CC BY-NC 4.0, driles5: CC BY-NC 4.0, driles5: CC BY-NC 4.0, Marcus Garvie: CC BY-NC 4.0, ryanubrown: CC BY-NC-ND 4.0, efarias1: CC BY-NC 4.0, amarena: CC BY-NC 4.0, ttempel: CC BY-NC 4.0, tnewman: CC BY-NC 4.0, Juan Cruzado Cortés: CC BY-NC-SA 4.0, lonnyholmes: CC BY-NC 4.0, Audrey Kremer: CC BY-NC 4.0, texasblonde: CC BY-NC 4.0, Dale Hameister: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, Allan Finlayson: CC BY-NC 4.0, pfaucher: CC BY-NC 4.0, Edward George: CC BY-NC 4.0, Cullen Hanks: CC BY-NC 4.0, CK Kelly: CC BY 4.0, mustardlypig: CC BY-NC 4.0, ONG OeBenin: CC BY-NC 4.0, ONG OeBenin: CC BY-NC 4.0, ONG OeBenin: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, CK Kelly: CC BY 4.0, sequeirajluis: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, CK Kelly: CC BY 4.0, sequeirajluis: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, nighthawk0083: CC BY-NC 4.0, pfaucher: CC BY-NC 4.0, Johnny Wilson: CC BY-NC 4.0, ncowey: CC BY-NC 4.0, Johnny Wilson: CC BY-NC 4.0, redhat: CC BY-NC 4.0, ONG OeBenin: CC BY-NC 4.0, owlentine: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, Jakub Pełka: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, sequeirajluis: CC BY-NC 4.0, Robert J. ”Bob” Nuelle, Jr. AICEZS: CC BY-NC 4.0

Table 18: Author and Creative Commons Copyright notice for images in Figure 8.

stefanovet: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, selwynq: CC BY-NC 4.0, emily-a: CC BY-NC 4.0, carlaag: CC BY-NC 4.0, carroll: CC BY-NC 4.0, kevinhintsa: CC BY-NC 4.0, Lindsey Smith: CC BY-NC 4.0, kevinhintsa: CC BY-NC 4.0, Bob Heitzman: CC BY-NC 4.0, rmorgan: CC BY-NC 4.0, Giuseppe Cagnetta: CC BY-NC 4.0, cwwood: CC BY-SA 4.0, Fernando de Juana: CC BY-NC 4.0, Chris Evers: CC BY-NC 4.0, Marion Zöller: CC BY-NC 4.0, Marion Zöller: CC BY-NC 4.0, Chuck Sexton: CC BY-NC 4.0, Ronald Werson: CC BY-NC-ND 4.0, Chris Evers: CC BY-NC 4.0, Chris van Swaay: CC BY-NC 4.0, stefanovet: CC BY-NC 4.0, Richard Barnes: CC BY-NC 4.0, Fluff Berger: CC BY-SA 4.0, Cheryl Harleston: CC BY-NC-SA 4.0, Steven Chong: CC BY-NC 4.0, juliayl: CC BY-NC 4.0, Kent McFarland: CC BY-NC 4.0, Bruno Durand: CC BY-NC 4.0, José Belem Hernández Díaz: CC BY-NC 4.0, Nolan Eggert: CC BY-NC 4.0, Lee Elliott: CC BY-NC-SA 4.0, Ronald Werson: CC BY-NC-ND 4.0, Giuseppe Cagnetta: CC BY-NC 4.0, cwwood: CC BY-SA 4.0, Donna Pomeroy: CC BY-NC 4.0, martinswarren: CC BY-NC 4.0, Fernando de Juana: CC BY-NC 4.0, Chris van Swaay: CC BY-NC 4.0, Monica Krancevic: CC BY-NC 4.0, brentano: CC BY-NC 4.0, martinswarren: CC BY-NC 4.0, stefanovet: CC BY-NC 4.0, Mark Rosenstein: CC BY-NC-SA 4.0, Mark Rosenstein: CC BY-NC-SA 4.0, Mark Nenadov: CC BY-NC 4.0, Robin Agarwal: CC BY-NC 4.0, Bryan: CC BY-NC 4.0, Ronald Werson: CC BY-NC-ND 4.0, Chris van Swaay: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Mark Rosenstein: CC BY-NC-SA 4.0, Monica Krancevic: CC BY-NC 4.0, Chris Evers: CC BY-NC 4.0, Ronald Werson: CC BY-NC-ND 4.0, Don Loarie: CC BY 4.0, Fernando de Juana: CC BY-NC 4.0, Chris Evers: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Chris van Swaay: CC BY-NC 4.0, cwwood: CC BY-SA 4.0, stefanovet: CC BY-NC 4.0, Chris Evers: CC BY-NC 4.0, J. Maughn: CC BY-NC 4.0, Liam O'Brien: CC BY-NC 4.0, Greg Lasley: CC BY-NC 4.0, Steven Bach: CC BY-NC-SA 4.0, Donna Pomeroy: CC BY-NC 4.0, Judith Lopez Sikora: CC BY-NC 4.0, Chris van Swaay: CC BY-NC 4.0, J. Maughn: CC BY-NC 4.0, 112692329998402018828: CC BY-NC-SA 4.0, martinswarren: CC BY-NC 4.0, Giuseppe Cagnetta: CC BY-NC 4.0, Chris Evers: CC BY-NC 4.0, Chris van Swaay: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Ronald Werson: CC BY-NC-ND 4.0, Donna Pomeroy: CC BY-NC 4.0, Monica Krancevic: CC BY-NC 4.0, smwhite: CC BY-NC 4.0, dwest: CC BY-NC 4.0, stefanovet: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, greglasley: CC BY-NC 4.0, Ronald Werson: CC BY-NC-ND 4.0, Donna Pomeroy: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Chris van Swaay: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Chris van Swaay: CC BY-NC 4.0, beschwar: CC BY-NC 4.0, Chris van Swaay: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Fernando de Juana: CC BY-NC 4.0, Marion Zöller: CC BY-NC 4.0, Chuck Sexton: CC BY-NC 4.0, Chris van Swaay: CC BY-NC 4.0, martinswarren: CC BY-NC 4.0, Kathy Richardson: CC BY-NC 4.0, cwwood: CC BY-SA 4.0, stefanovet: CC BY-NC 4.0, Ronald Werson: CC BY-NC-ND 4.0, Donna Pomeroy: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Chris van Swaay: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Fernando de Juana: CC BY-NC 4.0, Ronald Werson: CC BY-NC-ND 4.0, martinswarren: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Marion Zöller: CC BY-NC 4.0, Ronald Werson: CC BY-NC-ND 4.0, Chris van Swaay: CC BY-NC 4.0, Giuseppe Cagnetta: CC BY-NC 4.0, martinswarren: CC BY-NC 4.0, Robin Agarwal: CC BY-NC 4.0, Giuseppe Cagnetta: CC BY-NC 4.0, Marion Zöller: CC BY-NC 4.0, Ronald Werson: CC BY-NC-ND 4.0, Donna Pomeroy: CC BY-NC 4.0, martinswarren: CC BY-NC 4.0, stefanovet: CC BY-NC 4.0, Ronald Werson: CC BY-NC-ND 4.0, Ronald Werson: CC BY-NC-ND 4.0, martinswarren: CC BY-NC 4.0, Giuseppe Cagnetta: CC BY-NC 4.0, Fernando de Juana: CC BY-NC 4.0, Giuseppe Cagnetta: CC BY-NC 4.0, rada: CC BY-NC 4.0, Philip Mark Osso: CC BY-NC 4.0, Fernando de Juana: CC BY-NC 4.0, Giuseppe Cagnetta: CC BY-NC 4.0, martinswarren: CC BY-NC 4.0, Giuseppe Cagnetta: CC BY-NC 4.0, Chris van Swaay: CC BY-NC 4.0, Chris van Swaay: CC BY-NC 4.0, Giuseppe Cagnetta: CC BY-NC 4.0, rada: CC BY-NC 4.0, martinswarren: CC BY-NC 4.0, martinswarren: CC BY-NC 4.0, martinswarren: CC BY-NC 4.0, Fernando de Juana: CC BY-NC 4.0.

Table 19: Author and Creative Commons Copyright notice for images in Figure 9.

slsfirefight: CC BY-NC 4.0, J. Maughn: CC BY-NC 4.0, Tim Hite: CC BY 4.0, Mikael Behrens: CC BY-NC 4.0, colinmorita: CC BY-NC 4.0, Mary Joyce: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, icosahedron: CC BY 4.0, Donna Pomeroy: CC BY-NC 4.0, tnewman: CC BY-NC 4.0, phylocode: CC BY-NC 4.0, Marisa or Robin Agarwal: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Tom Benson: CC BY-NC-ND 4.0, tam topes: CC BY-NC 4.0, James Maughn: CC BY-NC 4.0, Mark Rosenstein: CC BY-NC-SA 4.0, Marisa or Robin Agarwal: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Robin Agarwal: CC BY-NC 4.0, slsfirefight: CC BY-NC 4.0, Thomas Koffel: CC BY-NC 4.0, Chris Evers: CC BY-NC 4.0, jaliya: CC BY-NC 4.0, Chris Evers: CC BY-NC 4.0, Victor W Fazio III: CC BY-NC-ND 4.0, Chris Evers: CC BY-NC 4.0, J. Maughn: CC BY-NC 4.0, Andrew Cannizzaro: CC BY 4.0, 116916927065934112165: CC BY-NC-ND 4.0, gyrrlfalcon: CC BY-NC 4.0, kolasafamily: CC BY-NC 4.0, summermule: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Robin Agarwal: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Robin Agarwal: CC BY-NC 4.0, Robin Agarwal: CC BY-NC 4.0, David R: CC BY-NC-ND 4.0, summermule: CC BY-NC 4.0, slsfirefight: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, petecorradino: CC BY-NC 4.0, Mike Leveille: CC BY-NC 4.0, greglasley: CC BY-NC 4.0, tegmort: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Tom Benson: CC BY-NC-ND 4.0, flyfisherking: CC BY-NC 4.0, Jean-Lou Justine: CC BY 4.0, Judith Lopez Sikora: CC BY-NC 4.0, kolasafamily: CC BY-NC 4.0, nudibranchmom: CC BY-NC 4.0, Robin Agarwal: CC BY-NC 4.0, R.J. Adams: CC BY-NC 4.0, monicamares: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, nudibranchmom: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Mike Leveille: CC BY-NC 4.0, nudibranchmom: CC BY-NC 4.0, slsfirefight: CC BY-NC 4.0, J. Maughn: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Tom Benson: CC BY-NC-ND 4.0, JJ Johnson: CC BY-NC 4.0, James Maughn: CC BY-NC 4.0, petecorradino: CC BY-NC 4.0, summermule: CC BY-NC 4.0, pfaucher: CC BY-NC 4.0, Amy: CC BY-NC 4.0, enzedfred: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, BJ Stacey: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Robin Agarwal: CC BY-NC 4.0, Robin Agarwal: CC BY-NC 4.0, Ken-ichi Ueda: CC BY-NC 4.0, Robin Agarwal: CC BY-NC 4.0, Robin Agarwal: CC BY-NC 4.0, Brian Gratwicke: CC BY 4.0, slsfirefight: CC BY-NC 4.0, Robin Agarwal: CC BY-NC 4.0, Robin Agarwal: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, summermule: CC BY-NC 4.0, KK: CC BY-NC 4.0, John Karges: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Javier Solís: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Mike Leveille: CC BY-NC 4.0, Ken-ichi Ueda: CC BY-NC 4.0, J. Maughn: CC BY-NC 4.0, Robin Agarwal: CC BY-NC 4.0, David J Barton: CC BY-NC 4.0, James Maughn: CC BY-NC 4.0, greglasley: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Robin Agarwal: CC BY-NC 4.0, James Maughn: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, slsfirefight: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, John Karges: CC BY-NC 4.0, 104623964081378888743: CC BY-NC-ND 4.0, Judith Lopez Sikora: CC BY-NC 4.0, Amy: CC BY-NC 4.0, Marisa or Robin Agarwal: CC BY-NC 4.0, summermule: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Jennifer Rycenga: CC BY-NC 4.0, David J Barton: CC BY-NC 4.0, thylacine: CC BY-NC 4.0, greglasley: CC BY-NC 4.0, J. Maughn: CC BY-NC 4.0, Javier Solís: CC BY-NC 4.0, redhat: CC BY-NC 4.0, timputtre: CC BY-NC 4.0, icosahedron: CC BY 4.0, rbbrummitt: CC BY-NC 4.0, icosahedron: CC BY 4.0, David J Barton: CC BY-NC 4.0, slsfirefight: CC BY-NC 4.0, 104623964081378888743: CC BY-NC-ND 4.0, Ken-ichi Ueda: CC BY-NC-SA 4.0, Donna Pomeroy: CC BY-NC 4.0, sakuraisomi: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, Donna Pomeroy: CC BY-NC 4.0, J. Maughn: CC BY-NC 4.0, kestrel: CC BY-NC 4.0, BJ Stacey: CC BY-NC 4.0, summermule: CC BY-NC 4.0, thylacine: CC BY-NC 4.0, icosahedron: CC BY 4.0, KK: CC BY-NC 4.0, James Maughn: CC BY-NC 4.0, Javier Solís: CC BY-NC 4.0, rbbrummitt: CC BY-NC 4.0, J. Maughn: CC BY-NC 4.0, greglasley: CC BY-NC 4.0, greglasley: CC BY-NC 4.0, timputtre: CC BY-NC 4.0.