Learning an Adaptation Function to Assess Image Visual Similarities

Olivier Risser-Maroix, Amine Marzouki, Hala Djeghim, Camille Kurtz, Nicolas Lomenie

Learning an adaptation function to assess image visual similarities

O. Risser-Maroix[1,2] A. Marzouki[1] H. Djeghim[1] C. Kurtz[1] N. Loménie[1]

[1] LIPADE, Université de Paris – Paris, France

[2]orissermaroix@gmail.com

Abstract

Human perception is routinely assessing the similarity between images, both for decision making and creative thinking. But the underlying cognitive process is not really well understood yet, hence difficult to be mimicked by computer vision systems. State-of-the-art approaches using deep architectures are often based on the comparison of images described as feature vectors learned for image categorization task. As a consequence, such features are powerful to compare semantically related images but not really efficient to compare images visually similar but semantically unrelated. Inspired by previous works on neural features adaptation to psycho-cognitive representations, we focus here on the specific task of learning visual image similarities when analogy matters. We propose to compare different supervised, semi-supervised and self-supervised networks, pre-trained on distinct scales and contents datasets (such as ImageNet-21k, ImageNet-1K or VGGFace2) to conclude which model may be the best to approximate the visual cortex and learn only an adaptation function corresponding to the approximation of the the primate IT cortex through the metric learning framework. Our experiments conducted on the Totally Looks Like image dataset highlight the interest of our method, by increasing the retrieval scores of the best model @1 by 2.25×. This research work was recently accepted for publication at the ICIP 2021 international conference [1]. In this new article, we expand on this previous work by using and comparing new pre-trained feature extractors on other datasets.

Keywords

Visual Similarity, Features Adaptation, Image Retrieval, Analogies

1 Motivation

Analogies are constantly employed by humans to find connections / similarities between images, when learning concepts or for creative purposes. Our perception being extremely complex to model, it remains difficult to imitate by machines. However, capturing image similarities is a cornerstone in various computer vision tasks, such as retrieval, classification, spotting, etc. This task is challenging since an image has many more interpretations than its textual description and the sought similarity may depend on a hidden intention.

State-of-the-art approaches often rely on the comparison of images described as vectors of features learned via convolutional neural networks (CNNs) with objectives close to classification task. Since such features are generally learned for image categorization, they are biased by the semantics of the decision process of the classifiers. In this article, we focus on the task of learning image visual similarities and we involve this in an image retrieval context. Inspired by previous works on neural features adaptation to psycho-cognitive representations, we propose a way to learn a perceptual similarity function between images that share visual connections but that are semantically unrelated. To our knowledge, we are among the first to propose a methodology to face this issue of understanding the underlying psycho-visual process of matching images of different natures.

2 Background

In human perception, the notion of similarity between concepts or even images has been studied for a while and remains extremely difficult to define [2, 3, 4, 5, 6]. During the last decade, scientists from cognitive and computer sciences started to analyze the differences between human and machine perception and to investigate on how modern neural architectures could help to capture human judgments of similarity ; such a similarity can be either guided by general concepts [7, 8, 9] or performed by strictly visual correspondences [10, 11, 12, 13, 14]. As already stated by [15], strategies based on visual stimuli yield good results with very simple images (with well segmented background) [11] or when focusing on a narrow class [7, 12, 16] (such as only faces, animals, arts).

The *Totally Looks Like* image dataset (denoted as *TLL*) is an interesting example of this research area [15]. *TLL* is composed of 6k+ image pairs resulting from human propositions, where similarity between images is not based on semantic categorization (as this is the case in most classical image retrieval datasets) but only based on visual clues derived from the image contents. The images belong to different domains such as photography, cartoons, sketches, logos, etc. making the task even harder but more represen-

tative of human ability to make connections between semantically unrelated objects (some illustrative samples are provided in Fig. 1). From this dataset, one can note that the process of human visual similarity mixes multiple different levels of analysis ranging from color, texture, to shape, layout, etc. in which context, cultural aspect and possibly humor and irony can play an important role.

Ones could argue that similarity is a too subjective and an ill-posed problem. But given a dataset of image pairs such as *TLL*, authors from [15] conducted human experiments and found that when other image candidates were proposed to form pairs, humans remained consistent in their choices and selected invariably the right target image to make the pair. Authors from [5] found that results in different visual studies are highly influenced by the task requirements ; in our case the task is defined by the visual similarity without taking into account the semantic similarity. In addition, [4] showed that, for the task of grading the similarity between images, the quantitative analysis of the similarity scores reported by subjects reached a consensus.

Authors from [15] tried to reproduce the human similarity judgments from a given image pairs dataset with different neural architectures and, as mentioned earlier, they found that the learned convolutional features were not adapted for this task. Later [17] improved the similarity scores by using even more descriptors crafted for different uses (color only, shape only, etc.), and by using the right descriptor for each pair as oracle. None of [15, 17] worked on features adaptation, neither learned a model on a part of the dataset, making us the first to propose a baseline for this task. As comparative work, starting from the observation that traditional metrics (L2, PSNR, etc.) disagree with human judgments, authors from [13] already learned a low-level perceptual measure of similarity from image patches affected by distortions. The poor results obtained by [17] on *TLL* by using the low level perceptual learned similarity from [13], as well as with higher level semantic features confirm the previous study of [6] stating that human visual similarity was not based in the visual cortex but may be the result of processing done in the primate inferior temporal (IT) cortex. In our case, we hypothesize that even if neural networks are learned to be robust to cases where images are semantically similar but visually dissimilar, they still carry useful (and reusable) information about texture, shapes, etc. When dealing with relatively small datasets such as *TLL*, based on a strategy originally introduced in [6], we propose to use different layers of pre-trained networks with the opposite objective of categorization as a rough approximation of the visual cortex and learn only an adaptation function corresponding to the approximation of the the primate IT cortex through the *metric learning* framework.

Similarity learning is closely related to *distance metric learning* where the goal is to learn a distance function over objects that measures how similar two objects are. In our case, we look for a model able to bring closer each training image pair, while moving away all other images that would form a less good pair than the ground-truth one. For each image, we only have a single solution which makes our problem close to one-shot learning where metric learning has been considered [18] by using *Siamese networks* [19]. Another wildely used architecture in deep metric learning is the *triplet network* [20] where for each positive pair, a negative image is also provided to learn simultaneously to gather the positive images while increasing the distance to the negative image. In a similar fashion, [9] was able to provide powerful representations capturing human behavior by asking participants to select the *one-odd-out* from a triplet of images and thus learning a model to mimic it. It has been shown that those architectures suffer from sampling issues that could be partially solved with complex mining strategies [21, 22].

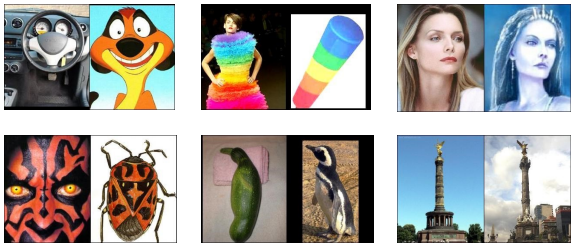


FIGURE 1 – Examples of image pairs from the "Totally Looks Like" dataset. Similarity can be based on the color, shape, texture, layout, facial similarity, etc.

For the sake of simplicity, we rely here on the hypothesis that each image in a pair has a stronger connection than with any other image in the dataset. We then learn a function able to bring closer the right image to the query than any other image. By doing so, we aim to bypass the sampling issue and use a generalization of the triplet loss similar to the *N-Loss* [23]. Inspired from previous works in cognitive sciences, we propose the first baseline for the task of visual similarity between images without taking into account semantic similarity.

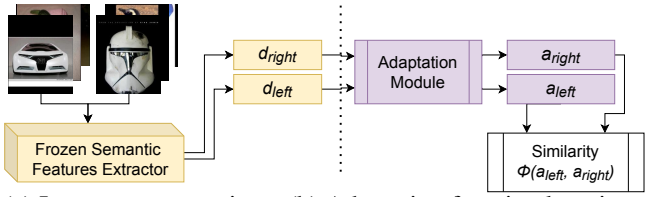
This research work was recently accepted for publication at the ICIP 2021 international conference. In this new article, we expand on this previous work by using and comparing new pre-trained feature extractors on other datasets.

3 Learning image visual similarities

We propose to learn an adaptation function able to compute visual similarity from image pairs and to involve it in an image retrieval task (Fig. 2). Inspired by [6], the features extraction part (Fig.2 (a)) could be interpreted as the visual cortex while the learned adaptation matrix and the measure deciding on the visual similarity between two images (Fig.2 (b)) can be viewed as the primate IT cortex.

3.1 Image retrieval task

We assume that we have an image dataset, structured as pairs of visually similar images (considered as the ground-truth (GT)). We consider each pair as the juxtaposition of



(a) Image representations (b) Adaptation function learning

FIGURE 2 – Pipeline to learn an adaptation function able to compute visual similarity from image pairs.

a *left* image and and a *right* image, leading to two sets of images. As in [15], we formulate this problem as an image retrieval task. For each query in the left set of images, we rank all the images in the set of right, according to a given similarity measure $\phi(\cdot, \cdot)$, and reciprocally. From this ranking, we want that for each query, the best returned candidate is the one expected by the GT pair.
We know from [24] that asymmetry in human judgment of similarity is important. For example, in Fig. 1, most humans will think that the zucchini looks like the penguin rather than the penguin looks like the zucchini. In our case, as we do not have the direction information of which image from left or right is looking to the other one, we cannot use an asymmetric function such as the *Tversky Ratio Model* [24] and $\phi(\cdot, \cdot)$ is here a simple *cosine similarity*. However, we will embed this asymmetry in the evaluation function.

3.2 Image representation

As pre-processing, we extract visual features for each image (right, left) from pre-trained networks. Different layers were used for some architectures based on Residual Networks while only the last layer was adopted for more complex ones due to the difficulty to localize the middle layers (FaceNet and transfomers). For more details, please refer to Table 1 in the annex.

We reduce the feature maps of each layer to a simple vector with the same dimension as the number of channels, by averaging features maps on the spatial dimensions followed by *L2* normalization. By doing so, each image is represented as the concatenation of features extracted from different layers to capture information, from low to high levels. To avoid overfitting, we reduce the number of dimension of the original vectors (up to 15k+) with a PCA. Left and right vectors are respectively named d_{left}, d_{right} and form the image representations.
This pre-processing step (Fig. 2 (a)) is done once before training and can be interpreted as a rough approximation of visual features extracted by the human visual cortex. Indeed, the object recognition and visual similarity assessment is done in the Ventral Stream which is composed of the Visual Cortex and the Primate IT Cortex [25]. The Visual Cortex is decomposed into sub-regions (V1-to-V4); V4 is particularly known to respond to orientation, color, disparity and simple shapes and it is directly connected to the Primate IT Cortex. Since the different features V4 responds to are located in different levels of a CNN (color and orientation are lower levels and extracted from early layers while shapes are higher layer levels) we propose to approximate V4 region by taking as input the different layers of a pre-trained CNN (aiming to extract color, orientation, shape, etc.) and directly pass them to our adaptation module. It then becomes obvious that this learned adaptation module directly wired to the V4 visual features simulated by the frozen pre-trained CNN is our model of the Primate IT cortex we propose to learn in a contrastive setting.

3.3 Adaptation function learning

We look for an adaptation function able to bring closer the left embeddings (d_{left}) to the right ones (d_{right}) of the corresponding pairs closer than any other data. We model our adaptation module as the multiplication between a (learnable) weight matrix W and input features d_{left}, d_{right} followed by a ReLU activation. We will refer to the adaptation of the d_{left}, d_{right} features as a_{left}, a_{right} (Fig. 2 (b)).
Previous works used contrastive or triplet loss [19, 20] when learning from pairs or triplets. To avoid sampling issue, as the whole pre-processed embeddings are light in memory, we build a similarity matrix by measuring similarity of each left image to each right image with the $\phi(\cdot, \cdot)$ similarity function. The corresponding GT is the identity matrix. We can thus learn to classify by using a softmax activation, followed by a Cross-Entropy loss function $\mathcal{L}$. We learn to find the right image from left queries and left images from right queries simultaneously, by averaging the directed $\mathcal{L}_{\text{left to right}}$ and $\mathcal{L}_{\text{right to left}}$ losses as in Eq. 1.

$$\begin{aligned}\mathcal{L}_{\text{left to right}} &= \text{CrossEntropy}(\text{softmax}(\phi(a_{left}, a_{right}) \cdot \sigma), \mathbb{1}) \\ \mathcal{L}_{\text{right to left}} &= \text{CrossEntropy}(\text{softmax}(\phi(a_{right}, a_{left}) \cdot \sigma), \mathbb{1}) \\ \mathcal{L} &= (\mathcal{L}_{\text{left to right}} + \mathcal{L}_{\text{right to left}})/2\end{aligned} \tag{1}$$

We used a parameter σ in Eq. 1, often referred as a temperature parameter in the literature. This specific parameter σ has already been used with success in previous works such as in [26]. In our case, by using a large σ, we are able to peak the softmax distribution to near binary values. This can be viewed as a form of regularization. Let us consider the case where the pair is successfully matched, without σ the loss would still continue to try to bring the adapted embeddings even closer while they already are the best matches. But, when a large σ is used, the correct best match will have a value close to 1 not penalizing anymore the loss for right classification and thus not trying to continue to learn on adapted embeddings which are already optimal. We show the positive influence on learning and generalization by using a large σ in Table. 1.

4 Experimental study

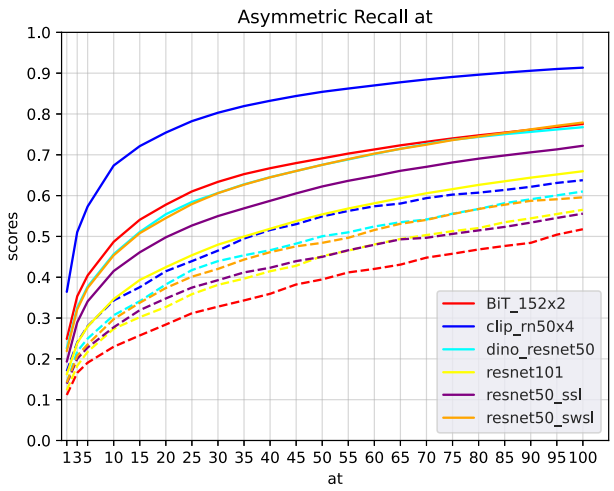


FIGURE 3 – Comparison of *Asymmetric Recall curves at different ranks between baselines with* $\sigma = 15$. Dashed line correspond to performances before adaptation, plain ones correspond to the adapted ones.

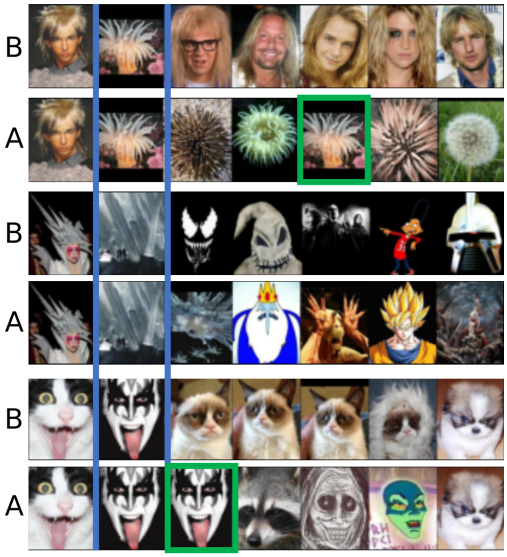


FIGURE 4 – Results before (B) and after (A) adaptation. The first column corresponds to the query, the second to the GT and the other images correspond to the returned candidates.

4.1 TTL Dataset

We considered in our experimental study the *Totally Looks Like* (*TLL*) image dataset introduced earlier. It is composed of 6016 image pairs, perceptually similar but semantically unrelated (some illustrative samples are provided in Fig. 1). The images come from very different domains such as photography, cartoons, paintings, sketches, logos, etc.

The dataset can be split into two sub-datasets : one of 1817 pairs containing only well centered faces (noted TLL_{faces}) and one of 4199 pairs of images captured from the wild (noted TLL_{obj}). Since [12] already focused on the particular case of facial similarity with data richer than pairs on a bigger dataset, we will focus on the more general case of the 4199 remaining pairs. To subtract the TLL_{faces} subset from the whole *TLL*, we labeled as face-pairs only pairs where faces were detected in both right and left images with a Haar Cascade classifier. In this context, the remaining set (TLL_{obj}) still contains pairs where faces are compared to other animals, objects, paintings, etc. due to strange facial expressions or other features.

4.2 Protocol

For each experiment, the TLL_{obj} dataset of 4199 pairs was divided into 75-25 train-test sets. We did not use a validation set due to the non significativness of the scores obtained on too small validation or testing bases, furthermore, if we decrease the training set size, the exhaustivity of the different ways that human compare images is too partial.

We used a PCA with 256 dimensions to compress the precomputed embeddings into d_{left}, d_{right}. Experimentally, we found that 256 was able to remove the noise reducing both, the input space and overfitting of our adaptation. Thus we will use those compressed vectors both as a second baseline and as inputs of our method. We used then an adaptation vector size of 1024, thus the matrix W is sized from 256 to 1024. As we do not have a validation set, we experimentally set the number of epochs to 150 and we selected the optimal number of epochs by bootstrapping for each model. This provides a good compromise between low overfitting and the best possible model if we had an oracle to select it.

Due to the small number of examples in the testing base and the stochasticity induced by the random split and matrix weights initialization, we run our experiments 20 times with different splits and matrix initializations.

4.3 Evaluation and results

As discussed previously, human similarity judgment has been found to be asymmetric [24]. Since we do not have the asymmetry indications in the *TLL* dataset, we propose to evaluate the results in an optimistic way : if the pairing is found in one direction or another, the pairing is considered as successful. We refer to this optimistic recall as *Asymmetric Recall* (aR).

Results are reported on Fig. 3 and Tab. 1. We computed the *Asymmetric Recall* on 20 random tests splits and display the mean scores surrounded by two standard deviations. From these results, we observe the strong influence of the temperature σ on the results. Our method is compared to the baselines (with and without) dimension reduction. We can see a $2.25\times$, $3.55\times$ and $4.53\times$ average improvement for $aR@\{1, 5, 20\}$ by only using the *Clip_RN50×4* features.

We provide a few qualitative results in Fig. 4 from the test set. The second case shows that none of the previous neither our method is able to find the right image, however our method was able to capture the sharp hairstyle and gla-

cial aspect of the picture. The first and last ones show the limitation of purely semantic features for visual retrieval, while our method leads to more diverse results. On the last example, it can be noticed that while a semantically biased features extractor will firstly output cats when querying a cat, our method finds image from very different classes, still sharing connection on the dark eyes surrounding.

Comparative study. Datasets scales and contents influence the results of each learning approach, which made us question the importance of the pre-trained network chosen for the features extraction module. In the Table 1 in annex we compare Residual networks of different depths : ResNet18, ResNet50, ResNet101, ResNet152, [27], the self-supervised DINO framework pre-trained with ResNet50 [28], FaceNet pre-trained on VGGFace2 [21], BiT-M of different sizes : ResNet-50x1, ResNet-50x3, ResNet-101x1, ResNet-101x3, and ResNet-152x4 [29], Barlow Twins Self-Supervised model pre-trained on ResNet50 [30] and CLIP, a framework that works on learning visual concepts from natural language supervision [31].

In [32], they found that the unsupervised models as well as self-supervised ones were the best at modelling and approximating the ventral visual stream, though their similarity task is different from ours. We compared different learning types to confirm that self-supervised models are the best ones for approximating the visual ventral system, which is coherent with their findings.

5 Discussions

In this article, we focus on the specific task of learning visual image similarities. Inspired by previous works on neural features adaptation to psycho-cognitive representations, we proposed a method to adapt semantic neural representations to visual ones. In average, our method improves the retrieval score up to 4.53×. We observed as well a qualitative improvement on the returned candidates. In addition, we saw that the CLIP framework was the best in our case, and we assume that it is due to two things : (1) First, such model has been trained on a dataset which consists of 400M images gathered from the internet ; (2) Second, because of the robustness of the CLIP model and its ability to generalize the concepts as well as its flexibility when it comes to semantic bias. Despite those improvements, some retrieval results are still not meaningful. Accordingly to [14], in contrary to humans, neural features may be less suited to achieve viewpoints invariance. Moreover, as for several pairs the shared visual similarity relies on few pixels, this problem could be tackled in a fine-grained framework. Since pairs in the TLL dataset are not equally sampled from the meta-features of an image (color, texture, layout, etc.), identifying those different meta-features used by human to judge about the visual similarity between images could help to re-balance learning. Further investigations are ongoing.

Références

[1] Olivier Risser-Maroix et al., "Learning an adaptation function to assess image visual similaritie," in *ICIP*, 2021, pp. XX–XX.

[2] Amos Tversky and Itamar Gati, "Studies of similarity," *Cognition and Categorization*, 1978.

[3] Bernice Rogowitz et al., "Perceptual image similarity experiments," in *HVEI*, 1998, pp. 576–590.

[4] Pierre Tirilly, Xiangming Mu, Chunsheng Huang, Iris Xie, Wooseob Jeong, and Jin Zhang, "On the consistency and features of image similarity," in *IIiX*, 2012, pp. 164–173.

[5] Taosheng Liu and Lynn A. Cooper, "The influence of task requirements on priming in object decision and matching," *Memory & Cognition*, vol. 29, no. 6, pp. 874–882, 2001.

[6] Marieke Mur, Mirjam Meys, Jerzy Bodurka, Rainer Goebel, Peter A. Bandettini, and Nikolaus Kriegeskorte, "Human object-similarity judgments reflect and transcend the primate-IT object representation," *Frontiers in Psychology*, vol. 4, pp. 128, 2013.

[7] Joshua C. Peterson, Joshua T. Abbott, and Thomas L. Griffiths, "Evaluating (and improving) the correspondence between deep neural networks and human representations," *Cognitive Science*, vol. 42, no. 8, pp. 2648–2669, 2018.

[8] Charles Y. Zheng, Francisco Pereira, Chris I. Baker, and Martin N. Hebart, "Revealing interpretable object representations from human behavior," in *ICLR*, 2019.

[9] Martin N. Hebart, Charles Y. Zheng, Francisco Pereira, and Chris I. Baker, "Revealing the multidimensional mental representations of natural objects underlying human similarity judgements," *Nature Human Behaviour*, vol. 4, no. 11, pp. 1173–1185, 2020.

[10] Jonas Kubilius, Stefania Bracci, and Hans P. Op de Beeck, "Deep neural networks as a computational model for human shape sensitivity," *PLOS Computational Biology*, vol. 12, no. 4, pp. e1004896, 2016.

[11] R. T. Pramod and S. P. Arun, "Do computational models differ systematically from human object perception ?," in *CVPR*, 2016, pp. 1601–1609.

[12] Amir Sadovnik, Wassim Gharbi, Thanh Vu, and Andrew C. Gallagher, "Finding your lookalike : Measuring face similarity rather than face identity," in *CVPR Workshops*, 2018, pp. 2345–2353.

[13] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *CVPR*, 2018, pp. 586–595.

[14] Joseph Scott German and Robert A. Jacobs, "Can machine learning account for human visual object shape similarity judgments ?," *Vision Research*, vol. 167, pp. 87–99, 2020.

[15] Amir Rosenfeld, Markus D. Solbach, and John K. Tsotsos, "Totally looks like - how humans compare, compared to machines," in *ACCV*, 2018, pp. 282–297.

[16] Mark Hamilton, Stephanie Fu, William T Freeman, and Mindren Lu, "Conditional image retrieval," *CoRR*, vol. abs/2007.07177, 2020.

[17] Amir Rosenfeld, Richard S. Zemel, and John K. Tsotsos, "High-level perceptual similarity is enabled by learning diverse tasks," *CoRR*, vol. abs/1903.10920, 2019.

[18] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *ICML Workshops*, 2015, pp. 1–8.

[19] Sumit Chopra, Raia Hadsell, and Yann LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *CVPR*, 2005, pp. 539–546.

[20] Elad Hoffer and Nir Ailon, "Deep metric learning using triplet network," in *SIMBAD*, 2015, pp. 84–92.

[21] Florian Schroff, Dmitry Kalenichenko, and James Philbin, "FaceNet : A unified embedding for face recognition and clustering," in *CVPR*, 2015, pp. 815–823.

[22] Chao-Yuan Wu, R. Manmatha, Alexander J. Smola, and Philipp Krahenbuhl, "Sampling matters in deep embedding learning," in *ICCV*, 2017, pp. 2859–2867.

[23] Kihyuk Sohn, "Improved deep metric learning with multi-class n-pair loss objective," in *NIPS*, 2016, pp. 1849–1857.

[24] Amos Tversky, "Features of similarity.," *Psychological review*, vol. 84, no. 4, pp. 327–335, 1977.

[25] Grace W Lindsay, "Convolutional neural networks as a model of the visual system : past, present, and future," *Journal of cognitive neuroscience*, pp. 1–15, 2020.

[26] Zhirong Wu, Alexei A. Efros, and Stella X. Yu, "Improving generalization via scalable neighborhood component analysis," in *ECCV*, pp. 712–728. 2018.

[27] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.

[28] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin, "Emerging properties in self-supervised vision transformers," 2021.

[29] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby, "Large scale learning of general visual representations for transfer," *CoRR*, vol. abs/1912.11370, 2019.

[30] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny, "Barlow twins : Self-supervised learning via redundancy reduction," *CoRR*, vol. abs/2103.03230, 2021.

[31] Alec Radford, Jong Wook Kim, ..., Gretchen Krueger, and Ilya Sutskever, "Learning transferable visual models from natural language supervision," *CoRR*, vol. abs/2103.00020, 2021.

[32] Chengxu Zhuang, Siming Yan, Aran Nayebi, Martin Schrimpf, Michael C Frank, James J DiCarlo, and Daniel LK Yamins, "Unsupervised neural network models of the ventral visual stream," *Proceedings of the National Academy of Sciences*, vol. 118, no. 3, 2021.

Annexe A

TABLE 1 – Asymmetric Recall scores of each model before and after adaptation (our method). We only extracted features from one of the last layer in models marked with a $*$. Scores are averaged on 20 runs and reported with the standard deviation.

Model	Learning type	Dataset	Number of images (M)	Concatenated Size	PCA Variance Sum	Concatenated Score	PCA Score	Our aR@1 $\sigma = 1$	Our aR@1 $\sigma = 15$
barlow	self-supervised	ImageNet-1k	1.28	3840	0.7555	0.1389 $\pm$0.0106	0.1357	0.1781 $\pm$0.0041	0.2065 $\pm$0.0045
deits16 *	self-supervised	ImageNet-1k	1.28	384	0.9352	0.1392 $\pm$0.0107	0.1362	0.1456 $\pm$0.0044	0.1826 $\pm$0.0044
deits8 *	self-supervised	ImageNet-1k	1.28	384	0.9451	0.149 $\pm$0.0109	0.1477	0.1623 $\pm$0.0057	0.2035 $\pm$0.0057
dino_resnet50	self-supervised	ImageNet-1k	1.28	3840	0.7732	0.136 $\pm$0.0089	0.1288	0.1947 $\pm$0.0034	**0.2435** $\pm$0.0040
dino_vitb16 *	self-supervised	ImageNet-1k	1.28	768	0.8505	0.1456 $\pm$0.0100	0.1394	0.1745 $\pm$0.0039	0.2009 $\pm$0.0048
dino_vitb8 *	self-supervised	ImageNet-1k	1.28	768	0.8199	0.1334 $\pm$0.0122	0.1314	0.1763 $\pm$0.0041	0.2012 $\pm$0.0047
facenet *	supervised	VGGFace2	3.31	1792	0.9722	0.0533 $\pm$0.0050	0.0611	0.0739 $\pm$0.0035	0.1173 $\pm$0.0038
alexnet *	supervised	ImageNet-1k	1.28	256	1.0000	0.1023 $\pm$0.0080	0.1041	0.1061 $\pm$0.0043	0.1321 $\pm$0.0045
resnet18	supervised	ImageNet-1k	1.28	960	0.9118	0.1283 $\pm$0.0097	0.1259	0.1323 $\pm$0.0035	0.1684 $\pm$0.0044
resnet34	supervised	ImageNet-1k	1.28	960	0.9011	0.1236 $\pm$0.0091	0.1304	0.1426 $\pm$0.0037	0.1469 $\pm$0.0047
resnet50	supervised	ImageNet-1k	1.28	3840	0.8832	0.1211 $\pm$0.0073	0.1258	0.1340 $\pm$0.0051	**0.1913** $\pm$0.0034
resnet101	supervised	ImageNet-1k	1.28	3840	0.8974	0.1209 $\pm$0.0073	0.1212	0.1557 $\pm$0.0042	0.1686 $\pm$0.0046
resnet152	supervised	ImageNet-1k	1.28	3840	0.8971	0.125 $\pm$0.0094	0.1214	0.1472 $\pm$0.0044	0.1892 $\pm$0.0040
efficientnet_b0 *	supervised	ImageNet-1k	1.28	1280	0.7725	0.1164 $\pm$0.0093	0.1149	0.1232 $\pm$0.0039	0.1552 $\pm$0.0032
efficientnet_b1 *	supervised	ImageNet-1k	1.28	1280	0.7630	0.1243 $\pm$0.0067	0.1242	0.1384 $\pm$0.0033	0.1387 $\pm$0.0034
efficientnet_b2 *	supervised	ImageNet-1k	1.28	1408	0.7423	0.1165 $\pm$0.0071	0.1170	0.1205 $\pm$0.0029	0.1602 $\pm$0.0042
efficientnet_b3 *	supervised	ImageNet-1k	1.28	1536	0.7188	0.1162 $\pm$0.0090	0.1126	0.1205 $\pm$0.0033	0.1410 $\pm$0.0046
ViT *	supervised	ImageNet-1k	1.28	768	0.8001	0.121 $\pm$0.0084	0.1173	0.1377 $\pm$0.0032	0.1570 $\pm$0.0035
BiT-M-R50x1	supervised	ImageNet-21k	14.19	3840	0.9212	0.1165 $\pm$0.0076	0.1142	0.1722 $\pm$0.0031	0.2332 $\pm$0.0042
BiT-M-R50x3	supervised	ImageNet-21k	14.19	11520	0.8768	0.1165 $\pm$0.0099	0.1143	0.2049 $\pm$0.0046	0.2659 $\pm$0.0052
BiT-M-R101x1	supervised	ImageNet-21k	14.19	3840	0.9174	0.1114 $\pm$0.0087	0.1054	0.2030 $\pm$0.0051	0.2509 $\pm$0.0055
BiT-M-R101x3	supervised	ImageNet-21k	14.19	11520	0.8789	0.1099 $\pm$0.0087	0.1075	0.1800 $\pm$0.0040	0.2437 $\pm$0.0039
BiT-M-R152x2	supervised	ImageNet-21k	14.19	7680	0.8877	0.1171 $\pm$0.0058	0.1101	0.2215 $\pm$0.0060	**0.2732** $\pm$0.0049
BiT-M-R152x4	supervised	ImageNet-21k	14.19	15360	0.8033	0.1174 $\pm$0.0102	0.1143	0.2145 $\pm$0.0024	0.2540 $\pm$0.0052
resnet50_ssl	semi-supervised	ImageNet-1k	1.28	3840	0.7973	0.1396 $\pm$0.0075	0.1401	0.1883 $\pm$0.0039	**0.1995** $\pm$0.0051
resnext101_32x8d_ssl	semi-supervised	ImageNet-1k	1.28	3840	0.8539	0.1255 $\pm$0.0089	0.1276	0.1589 $\pm$0.0033	0.1550 $\pm$0.0040
resnet50_swsl	weakly semi-supervised	ImageNet-1k	1.28	3840	0.7912	0.1548 $\pm$0.0099	0.1506	0.1929 $\pm$0.0035	**0.2191** $\pm$0.0038
resnext101_32x16d_swsl	weakly semi-supervised	ImageNet-1k	1.28	3840	0.8729	0.1213 $\pm$0.0068	0.1270	0.1414 $\pm$0.0031	0.1986 $\pm$0.0049
resnext101_32x8d_swsl	weakly semi-supervised	ImageNet-1k	1.28	3840	0.8702	0.1295 $\pm$0.0067	0.1297	0.1544 $\pm$0.0029	0.2005 $\pm$0.0041
resnext101_32x8d_wsl	weakly supervised	ImageNet-1k	1.28	3840	0.7861	0.1297 $\pm$0.0094	0.1192	0.1642 $\pm$0.0035	0.1917 $\pm$0.0041
resnext101_32x16d_wsl	weakly supervised	ImageNet-1k	1.28	3840	0.7731	0.1292 $\pm$0.0079	0.1204	0.1670 $\pm$0.0035	0.2061 $\pm$0.0047
resnext101_32x32d_wsl	weakly supervised	ImageNet-1k	1.28	3840	0.7651	0.1406 $\pm$0.0091	0.1323	0.1639 $\pm$0.0047	0.2295 $\pm$0.0052
resnext101_32x48d_wsl	weakly supervised	ImageNet-1k	1.28	3840	0.7629	0.1458 $\pm$0.0085	0.1419	0.1696 $\pm$0.0037	**0.2318** $\pm$0.0046
clip_rn101	natural language supervision	Internet	400.0	4352	0.8426	0.1674 $\pm$0.0104	0.1658	0.2814 $\pm$0.0050	0.3825 $\pm$0.0046
clip_rn50	natural language supervision	Internet	400.0	4864	0.8679	0.1575 $\pm$0.0102	0.1606	0.2834 $\pm$0.0059	0.3607 $\pm$0.0047
clip_rn50x4	natural language supervision	Internet	400.0	5440	0.8199	0.1743 $\pm$0.0070	0.1700	0.3282 $\pm$0.0041	**0.3939** $\pm$0.0071
clip_vit32_b *	natural language supervision	Internet	400.0	768	0.8600	0.1714 $\pm$0.0074	0.1797	0.2655 $\pm$0.0068	0.3540 $\pm$0.0046