HCSC: Hierarchical Contrastive Selective Coding

Yuanfan Guo[1,5][*,†]   Minghao Xu[2,3][*,‡]   Jiawen Li[4]   Bingbing Ni[1]
Xuanyu Zhu[4]   Zhenbang Sun[4]   Yi Xu[1,5][§]

[*]equal contribution   [†]technique lead   [‡]project lead   [§]corresponding author

[1]MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University
[2]Mila - Québec AI Institute   [3]University of Montréal   [4]ByteDance
[5]Chongqing Research Institute of Shanghai Jiao Tong University

contacts: Yuanfan Guo <gyfastas@sjtu.edu.cn>, Minghao Xu <minghao.xu@umontreal.ca>, Yi Xu <xuyi@sjtu.edu.cn>

Abstract

*Hierarchical semantic structures naturally exist in an image dataset, in which several semantically relevant image clusters can be further integrated into a larger cluster with coarser-grained semantics. Capturing such structures with image representations can greatly benefit the semantic understanding on various downstream tasks. Existing contrastive representation learning methods lack such an important model capability. In addition, the negative pairs used in these methods are not guaranteed to be semantically distinct, which could further hamper the structural correctness of learned image representations. To tackle these limitations, we propose a novel contrastive learning framework called **H**ierarchical **C**ontrastive **S**elective **C**oding (HCSC). In this framework, a set of hierarchical prototypes are constructed and also dynamically updated to represent the hierarchical semantic structures underlying the data in the latent space. To make image representations better fit such semantic structures, we employ and further improve conventional instance-wise and prototypical contrastive learning via an elaborate pair selection scheme. This scheme seeks to select more diverse positive pairs with similar semantics and more precise negative pairs with truly distinct semantics. On extensive downstream tasks, we verify the state-of-the-art performance of HCSC and also the effectiveness of major model components. We are continually building a comprehensive model zoo (see supplementary material). Our source code and model weights are available at* `https://github.com/gyfastas/HCSC`.

1. Introduction

In the past few years, self-supervised image representation learning has witnessed great progresses, in which
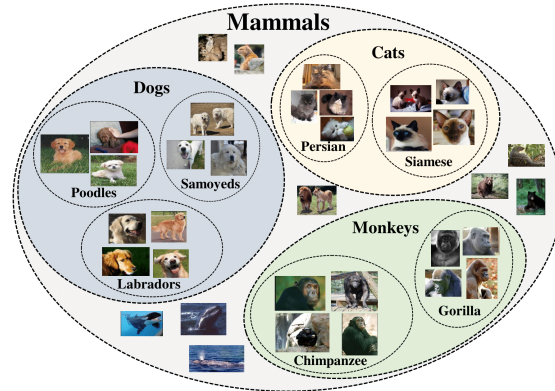

Figure 1. An image dataset always contains multiple semantic hierarchies, *e.g.* "mammals → dogs → Labradors" in the order from coarse-grained semantics to fine-grained semantics.

the traditional methods based on solving informative puzzles [12, 16, 32, 33, 45] are obviously surpassed by contrastive learning [6–8, 19, 34]. These contrastive methods succeed in deriving useful and interpretable representations for various downstream tasks. In particular, under the standard linear evaluation protocol [38], they have achieved inspiring results that approach fully-supervised learning.

Existing contrastive methods mainly lie in two categories, *instance-wise contrastive learning* [7, 19, 34] and *prototypical contrastive learning* [6,26]. Instance-wise contrast seeks to map similar instances nearby in the latent space while mapping dissimilar ones far apart, which guarantees reasonable local structures among different image representations. Prototypical contrast aims to derive compact image representations gathering around corresponding cluster centers, which captures basic semantic structures that can be represented by a single hierarchy of clusters.

However, these approaches lag in representation power when modeling **a large-scale image dataset which could always possess multiple semantic hierarchies**. For exam-

ple, in an extensive species database, the cluster of dogs summarizes the common characters of Labradors, Poodles, Samoyeds, *etc.* and should be placed on a higher hierarchy; similarly, dogs together with cats, monkeys, whales, *etc.* are further summarized by an even higher-level cluster, mammals (see Fig. 1 for a more intuitive illustration). Learning image representations that embrace such hierarchical semantic structures can greatly benefit the semantic understanding on various downstream tasks. How to achieve this by contrastive learning is still an open problem.

In addition, existing contrastive methods commonly construct negative pairs by exhaustive sampling from some noise distribution, and all the sampled negative pairs are used without selection. There is no guarantee that the negative pairs obtained in this way own truly distinct semantics. Therefore, some samples with similar semantics may be wrongly embedded far apart by these methods, which hampers the quality of learned image representations.

To tackle the limitations above, we propose a novel contrastive learning framework called **H**ierarchical **C**ontrastive **S**elective **C**oding (*HCSC*). In this framework, we propose to capture the hierarchical semantic structures underlying the data with *hierarchical prototypes*, a set of tree-structured representative embeddings in the latent space. Along the training process, these prototypes are dynamically updated to fit the current image representations. Under the guidance of such hierarchical semantic structures, we seek to improve both instance-wise and prototypical contrastive learning by *selecting high-quality positive and negative pairs that are semantically correct*. Specifically, for each query sample, we search for its most similar prototype on each semantic hierarchy to build more abundant positive pairs. Moreover, for each candidate of negative pair, we conduct a Bernoulli sampling to keep/discard it if the semantic correlation of the pair is low/high. By using these selected pairs for instance-wise and prototypical contrast, the semantic constraints from hierarchical prototypes can be embedded into the objective of representation learning.

We summarize the contributions of this work as follows:

- We novelly propose to represent the hierarchical semantic structures of image representations by dynamically maintaining hierarchical prototypes.

- We propose a novel contrastive learning framework, Hierarchical Contrastive Selective Coding (HCSC), which improves conventional instance-wise and prototypical contrastive learning by selecting semantically correct positive and negative pairs.

- Our HCSC approach consistently achieves superior performance over state-of-the-art contrastive learning algorithms on various downstream tasks. Also, the effectiveness of key model components are verified by extensive ablation and visualization analysis.

2. Related Work

2.1. Self-supervised Representation Learning

Solving Pretext Puzzles. Most early works for self-supervised image representation learning aim to solve pretext puzzles, *e.g.* counting objects [33], solving the jigsaw puzzle [32], recovering colors from gray-scale images [25, 45], rotation prediction [16], *etc.* These pretext tasks are not guaranteed to derive discriminative feature representations for different downstream tasks.

Instance-wise Contrastive Learning. The instance-wise contrast approaches seek to embed similar instances nearby in the latent space while embed dissimilar ones far apart. Standard instance-wise contrastive methods [7, 19, 38] achieve this goal by optimizing with an InfoNCE loss [34], which is improved by employing inter-instance positive pairs [13], introducing stronger augmentation functions [37] or designing predictive pretext tasks [9, 17, 44]. However, these methods are not aware of the global semantics underlying the whole dataset.

Prototypical Contrastive Learning. Another series of contrastive approaches seek to explicitly exploit semantic structures by utilizing the prototype representations of image clusters. They either contrast between correlated and uncorrelated prototype pairs [6, 27] or between associated and unassociated instance-prototype pairs [26, 36], which derives more semantically compact image representations. However, all these methods represent semantic clusters at a single hierarchy, which neglects the important fact that an image dataset naturally possesses hierarchical semantics.

Improvements over existing works. In this work, we novelly propose to construct and maintain hierarchical semantic structures of image representations, which aligns with a recent effort [41] on learning hierarchical molecular representations. In addition, we improves both instance-wise and prototypical contrastive learning by selecting high-quality positive and negative pairs under precise semantic guidance.

2.2. Deep Clustering

Our work also relates to deep clustering, *i.e.* performing clustering in latent space. A line of research [18, 22] aims at more accurate clustering by leveraging a clustering-favored latent space. Another line of research [1, 4, 5, 26, 28, 40, 42, 43] jointly learns clustering assignments and image representations. Most of these methods [4, 40, 43] utilize some standard clustering algorithm, like K-means [30] or agglomerative clustering [46], to establish a single hierarchy of semantic clusters, which is not sufficient to represent the semantic hierarchies underlying a set of images.

Improvements over existing works. Compared to most existing works that learn a single semantic hierarchy, in this work, we seek to learn image representations with multiple semantic hierarchies.

3. Problem Definition and Preliminaries

3.1. Problem Definition

Given a set $X = \{x_1, x_2, \cdots, x_N\}$ of N unlabeled images, we aim to learn a low-dimensional vector $z_n \in \mathbb{R}^\delta$ for each $x_n \in X$. Besides image representations, we together maintain a set of hierarchical prototypes $C = \{\{c_i^l\}_{i=1}^{M_l}\}_{l=1}^L$ to characterize the hierarchical semantic structures underlying the data, where L stands for the number of semantic hierarchies, and M_l is the number of prototypes in the l-th hierarchy. Each prototype $c_i^l \in C$ is also represented as a δ-dimensional vector. Following the self-supervised learning protocol, image representations $Z = \{z_1, z_2, \cdots, z_N\}$ and hierarchical prototypes C are both learned or maintained under the guidance of the data itself.

3.2. Preliminaries

Instance-wise contrastive learning. To achieve the goal of self-supervised representation learning, a widely-used way is to contrast between a positive instance pair with some negative pairs. Specifically, given the representations (z, z') of a pair of correlated instances, a standard InfoNCE loss [34] is defined to maximize the similarity between this positive pair and minimize the similarities between some randomly sampled negative pairs:

$$\mathcal{L}_{\text{InfoNCE}}(z, z', \mathcal{N}, \tau) = -\log \frac{\exp(z \cdot z'/\tau)}{\sum_{z_j \in \{z'\} \cup \mathcal{N}} \exp(z \cdot z_j/\tau)}, \tag{1}$$

where $\mathcal{N}$ is a set of negative samples for z, and τ denotes a temperature parameter.

Prototypical contrastive learning. In this contrastive representation learning manner, each positive pair consists of an instance and its associated semantic prototype, and negative pairs are formed by pairing instances with irrelevant semantic prototypes. In this way of pair construction, given a positive pair (z, c), the ProtoNCE loss [26] is defined based on the same rationale of InfoNCE loss:

$$\mathcal{L}_{\text{ProtoNCE}}(z, c, \mathcal{N}_c, \{\tau_c\}) = -\log \frac{\exp(z \cdot c/\tau_c)}{\sum_{c_j \in \{c\} \cup \mathcal{N}_c} \exp(z \cdot c_j/\tau_{c_j})}, \tag{2}$$

where $\mathcal{N}_c$ stands for a set of negative prototypes for instance representation z, and τ_c is a prototype-specific temperature parameter which can be adaptively determined by some clustering statistics.

4. Method

4.1. Motivation and Overview

Many recent efforts have been paid on learning informative visual representations with contrastive methods. Most of these works [7, 19, 34] focused on probing instance-wise relationships, and several works [6, 26] further attempted

---

Algorithm 1 Hierarchical K-means.

Input: Image representations Z, # semantic hierarchies L, # prototypes at the l-th hierarchy M_l.
Output: Hierarchical prototypes $C = \{\{c_i^l\}_{i=1}^{M_l}\}_{l=1}^L$, the undirected edges E between different prototypes.
  $\{c_i^1\}_{i=1}^{M_1} \leftarrow$ K-means(Z).
  for $l = 2$ **to** L **do**
    $\{c_i^l\}_{i=1}^{M_l} \leftarrow$ K-means$(\{c_i^{l-1}\}_{i=1}^{M_{l-1}})$.
    for $i = 1$ **to** M_{l-1} **do**
      $E \leftarrow E \cup \{(c_i^{l-1}, \text{Parent}(c_i^{l-1}))\}$.
    end for
  end for

---

to discover the semantic structures within the data during contrastive learning. These approaches represented semantic structures with a group or several independent groups of cluster centers, which cannot represent the semantic hierarchies that naturally exist in an image dataset. For example, the images of Labradors, Poodles, Samoyeds, *etc.* form a higher-level cluster of dogs, and dogs together with cats, monkeys, whales, *etc.* can be further clustered as mammals. Capturing such hierarchical semantics promises the stronger representation power of an image encoder, while it has not been achieved by previous methods. Under the context of contrastive learning in particular, these semantic structures can give helpful guidance to select positive pairs with similar semantics and negative pairs with distinct semantics, which is also less explored by existing works.

Motivated by these limitations, we propose a novel contrastive learning framework called **H**ierarchical **C**ontrastive **S**elective **C**oding (*HCSC*). In a nutshell, we represent the semantic structures of the data with *hierarchical prototypes* and dynamically update these prototypes along the training process (Sec. 4.2). Based on such hierarchical semantic representations, we seek to boost conventional instance-wise and prototypical contrastive learning by selecting better positive and negative pairs that fit the semantic structures (Secs. 4.3 and 4.4), and our overall objective combines both learning manners (Sec. 4.5). The graphical summary of our approach is shown in Fig. 2.

4.2. Hierarchical Semantic Representation

The core of HCSC framework is to construct and maintain the hierarchical semantic structures of the data in the latent space. Compared to previous methods [6, 26] that can only capture a single semantic hierarchy or several independent hierarchies, our approach is able to capture the hierarchical relations among various semantic clusters. We achieve this by a simple yet effective *hierarchical K-means* algorithm that performs in a bottom-up manner.

The detailed algorithm is summarized in Alg. 1. First,

(a) Hierarchical Prototypes   (b) Instance-wise & Prototypical Contrastive Selective Coding
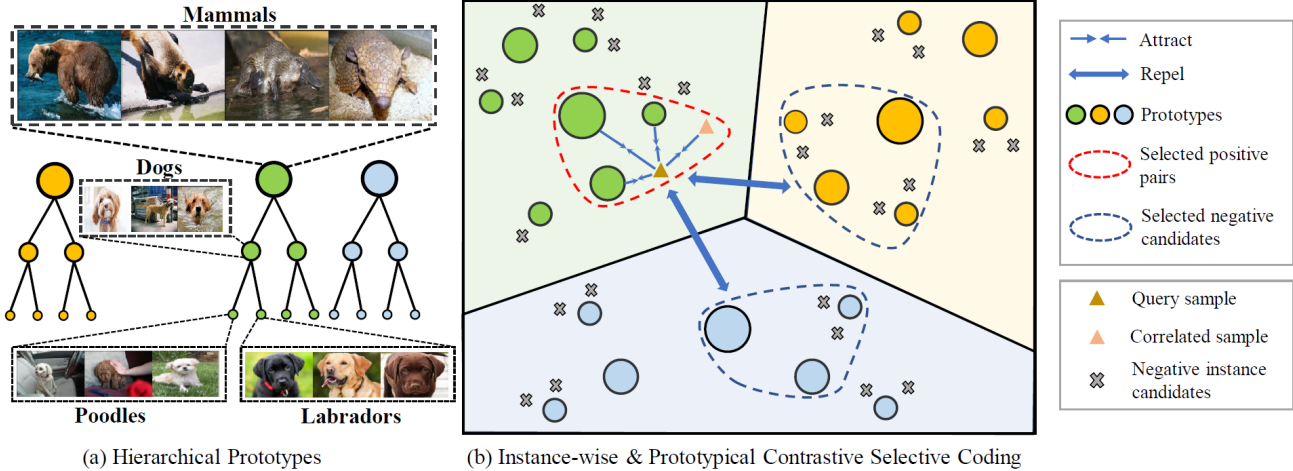
Figure 2. **Illustration of HCSC framework.** (a) In the latent space, a set of *hierarchical prototypes* are used to represent the hierarchical semantic structures underlying an image dataset. (b) *Instance-wise and prototypical contrastive selective coding* select semantically correct positive and negative pairs for contrastive learning, which is guided by the semantic information from hierarchical prototypes.

the feature representations of all images in the dataset are extracted by the image encoder, and K-means clustering is applied upon these image representations to obtain the prototypes of the first hierarchy. After that, the prototypes of each higher hierarchy are derived by iteratively applying K-means clustering to the prototypes of hierarchy below. To construct the hierarchical semantic structure, we further connect each prototype c_i^{l-1} with its parent prototype c_j^l at the higher hierarchy, in which c_i^{l-1} is assigned to c_j^l during K-means clustering. All such connections form an undirected edge set E. In this way, the hierarchical prototypes are structured as a set of trees (Fig. 2(a)). In this algorithm, the number L of semantic hierarchies and the number M_l of prototypes at the l-th hierarchy are specified in Sec. 5.1, and their sensitivities are studied in Sec. 6.2.

Since image representations are updated along the training process, a maintenance scheme on hierarchical prototypes is also required to ensure that they are representative embeddings in the latent space. In our implementation, to balance between precision and efficiency, we conduct the hierarchical K-means algorithm before the start of each epoch to update hierarchical prototypes according to the current image representations. Upon such hierarchical prototypes, we seek to promote instance-wise and prototypical contrastive learning.

4.3. Instance-wise Contrastive Selective Coding

The gist of instance-wise contrastive learning is to embed similar instances nearby in the latent space while embed those dissimilar ones far apart. It is easy to obtain a similar (*i.e.* positive) instance pair via data augmentation, while the definition of dissimilar (*i.e.* negative) instance pair is nontrivial. Previous methods derived negative samples by sampling uniformly over the dataset [7,8,19] or sampling from a

debiased data distribution [10]. However, they cannot guarantee that the produced negative samples own exactly distinct semantics relative to the query sample. Such a defect hampers instance-wise contrastive learning, in which those semantically relevant positive candidates could be wrongly expelled from the query sample in the latent space, and the semantic structure is thus broken to some extent. To overcome this drawback, we aim to select more precise negative samples that own truly irrelevant semantics with the query.

For a specific query image x, instead of contrasting it indiscriminately with all negative candidates $\mathcal{N}$ in a queue [8,19], we select truly negative samples for contrasting by performing Bernoulli sampling on each negative candidate. Intuitively, in such a sampling process, we would like to eliminate those candidates sharing highly similar semantics with the query, while keep the ones that are less semantically relevant to the query. To achieve this goal, we first define a similarity measure between an image and a semantic cluster. Following PCL [26], for a semantic cluster represented by prototype $c \in C$, we define the semantic similarity between image representation z and this cluster using a cluster-specific dot product:

$$s(z,c) = \frac{z \cdot c}{\tau_c}, \quad \tau_c = \frac{\sum_{z_i \in Z_c} ||z_i - c||_2}{|Z_c| \log(|Z_c| + \epsilon)}, \quad (3)$$

where Z_c consists of the representations of the images assigned to cluster c (details of constructing Z_c are stated in Sec. 5.1), and ϵ is a smooth parameter balancing the scale of temperature τ_c among different clusters.

On such basis, we conduct negative sample selection at each semantic hierarchy. At the l-th hierarchy, we denote the cluster that owns highest semantic similarity with the query image as $c^l(z) = \arg\max_{c \in \{c_i^l\}_{i=1}^{M_l}} s(z, c)$ (z is the representation of query). For a negative candidate $z_j \in \mathcal{N}$,

we are more likely to select it if its similarity with $c^l(z)$ is less prominent compared its similarities with other clusters at the same hierarchy. Based on such an intuition, we define the selected probability of z_j with the following formula:

$$p^l_{\text{select}}(z_j; z) = 1 - \frac{\exp\left[s(z_j, c^l(z))\right]}{\sum_{i=1}^{M_l} \exp\left[s(z_j, c^l_i)\right]}. \quad (4)$$

After that, a Bernoulli sampling is performed on each negative candidate to derive more precise negative samples for the specific query:

$$\mathcal{N}^l_{\text{select}}(z) = \left\{\mathcal{B}\left(z_j; p^l_{\text{select}}(z_j; z)\right) | z_j \in \mathcal{N}\right\}, \quad (5)$$

where $\mathcal{B}(z; p)$ denotes a Bernoulli trial of accepting z with probability p. Such a selection scheme is performed on all L semantic hierarchies, which diversifies the composition of negative samples. In this way, L negative sample sets, i.e. $\{\mathcal{N}^l_{\text{select}}(z)\}_{l=1}^L$, are produced for the query.

By using these refined negative samples, we define the objective function of instance-wise contrastive selective coding (ICSC) as below:

$$\mathcal{L}_{\text{ICSC}} = \mathbb{E}_{x \sim p_d}\left[\frac{1}{L}\sum_{l=1}^L \mathcal{L}_{\text{InfoNCE}}(z, z', \mathcal{N}^l_{\text{select}}(z), \tau)\right], \quad (6)$$

where p_d denotes the data distribution, z and z' are the representations of x and an augmented view of x, respectively. We fix the temperature parameter τ as 0.2 for instance-wise contrast following [8].

4.4. Prototypical Contrastive Selective Coding

Prototypical contrastive learning aims to derive compact image representations in the latent space, in which each image is closely embedded surrounding its associated cluster center. Given a query sample, previous works [6, 26] compare it with a single pool of prototypes, and they regard its most similar prototype as its positive companion and all other prototypes as negative samples. However, this scheme neglects the semantic correlation among different clusters, and it could excessively penalize on some semantically relevant clusters. For example, given an image of Labradors, it should not be strongly expelled from the image clusters of other kinds of dogs, considering the semantic correlation among these clusters. To mitigate such over-penalization, we seek to select more exact negative clusters that are semantically distant from the query.

Given a query image represented by embedding vector z, we first retrieve its most similar prototype at the l-th hierarchy, i.e. $c^l(z) = \arg\max_{c \in \{c^l_i\}_{i=1}^{M_l}} s(z, c)$, using the same similarity measure as in instance-wise contrast (Eq. (3)). We regard $(z, c^l(z))$ as a positive pair, and the rest prototypes at that hierarchy serve as the candidates of negative clusters, denoted as $\mathcal{N}^l$. Since some of these negative candidates might own similar semantics as the positive cluster

$c^l(z)$, it is unreasonable to embed the query distantly from these semantically relevant clusters. Therefore, we aim to perform prototypical contrast with the candidates that are more semantically distant from $c^l(z)$. To attain this goal, we take advantage of the semantic structures captured by hierarchical prototypes, and a Bernoulli sampling is performed on each negative candidate under such guidance.

Concretely, according to the edges of hierarchical prototypes, we first identify the parent node of $c^l(z)$ in its corresponding tree structure, denoted as $\text{Parent}(c^l(z))$. This parent cluster summarizes the common semantics shared by its child clusters at the l-th hierarchy, e.g. the general characters of different types of dogs. The negative clusters used for prototypical contrast are desired to share a semantic correlation with $\text{Parent}(c^l(z))$ as low as possible. Therefore, we are more likely to select a negative candidate $c_j \in \mathcal{N}^l$ if its similarity with $\text{Parent}(c^l(z))$ does not dominate its similarities with other clusters at the $(l+1)$-th hierarchy, which derives the selected probability of c_j as below:

$$p^l_{\text{select}}(c_j; c^l(z)) = 1 - \frac{\exp\left[s\left(c_j, \text{Parent}(c^l(z))\right)\right]}{\sum_{i=1}^{M_{l+1}} \exp\left[s(c_j, c^{l+1}_i)\right]}, \quad (7)$$

where the similarity measure $s(\cdot, \cdot)$ follows Eq. (3). We then conduct a Bernoulli sampling on each negative candidate to select exactly negative samples for prototypical contrast:

$$\mathcal{N}^l_{\text{select}}(c^l(z)) = \left\{\mathcal{B}\left(c_j; p^l_{\text{select}}(c_j; c^l(z))\right) | c_j \in \mathcal{N}^l\right\}. \quad (8)$$

Note that, we use all the negative candidates $\mathcal{N}^L$ at the top hierarchy without selection, since the semantic clusters at the top hierarchy share little semantic correlation and can thus be safely deemed as negative samples of each other.

By contrasting the positive pair $(z, c^l(z))$ with these selected negative samples, the objective function of prototypical contrastive selective coding (PCSC) is defined as below:

$$\mathcal{L}_{\text{PCSC}} = \mathbb{E}_{x \sim p_d}\left[\frac{1}{L}\sum_{l=1}^L \mathcal{L}_{\text{ProtoNCE}}(z, c^l(z), \mathcal{N}^l_{\text{select}}(c^l(z)), \{\tau_c\})\right], \quad (9)$$

where p_d denotes the data distribution, z is the representation of image x, and we have $\mathcal{N}^L_{\text{select}}(c^L(z)) = \mathcal{N}^L$ at the top hierarchy. We use a cluster-specific temperature parameter τ_c for each cluster, whose formulation follows Eq. (3).

4.5. Overall Objective

In general, instance-wise contrastive learning exploits the local instance-level structures, and prototypical contrastive learning constructs global semantic structures in the latent space. Our approach leverages the advantages of both worlds, and it further injects the semantic constraints from hierarchical prototypes into the learning objective by hierarchical pair selection. Therefore, the overall objective of Hierarchical Contrastive Selective Coding (HCSC) is to

learn an image encoder f_θ that minimizes the costs of both instance-wise and prototypical contrastive selective coding:

$$\min_{f_\theta} \mathcal{L}_{\text{ICSC}} + \mathcal{L}_{\text{PCSC}}. \qquad (10)$$

5. Experiments

5.1. Experimental Setups

Model details. The image encoder is with a ResNet-50 [20] backbone, and it maintains another momentum encoder following MoCo v2 [8]. The implementation of hierarchical K-means (Alg. 1) is based on faiss [23], an efficient clustering package. Following SwAV [6], we set the number of prototypes at the first hierarchy (*i.e.* M_1) as 3000, which fits the statistic patterns of ImageNet [11], *i.e.* our pre-training database. We recursively conduct another two rounds of K-means clustering to get two higher-level semantic hierarchies. The clustering hyperparameters are $L = 3$ and $(M_1, M_2, M_3) = (3000, 2000, 1000)$, and the sensitivity analysis for L and $\{M_l\}_{l=1}^L$ is in Sec. 6.2. Those clusters with less than 10 samples are discarded to avoid trivial solutions. We directly obtain the assignments Z_c of each cluster from hierarchical K-means results. The smooth parameter ϵ is set as 10 following PCL [26]. Note that, hierarchical prototypes and clustering assignments are fixed during each epoch training, which eases the optimization objective and benefits the convergence of image encoder.

Training details. We train with an SGD optimizer [2] (weight decay: 1×10^{-4}; momentum: 0.9; batch size: 256) and a cosine annealing scheduler [31] for 200 epochs. In the first 20 epochs, the model is warmed up with only the instance-wise loss $\mathcal{L}_{\text{ICSC}}$ (Eq. (6)). The same set of augmentation functions as in MoCo v2 [8] are used to generate positive pairs, and a queue of 16384 negative candidates are utilized for instance-wise contrastive selective coding (Sec. 4.3). On 8 Tesla-V100-32GB GPUs, 200 epochs training takes about 4 days *w/o* multi-crop and about 7.5 days *w/* multi-crop.

Performance comparisons. We compare the proposed HCSC approach with superior contrastive learning methods, including NPID [38], LocalAgg [48], MoCo [19], SimCLR [7], MoCo v2 [8], CPC v2 [34], PCL v2 [26], PIC [3], MoCHi [24], DetCo [39] and AdCo [21]. For fair comparison, in the main paper, we report the performance of these methods under 200 epochs training on ImageNet [11].

5.2. Experimental Results

5.2.1 Linear Classification and KNN Evaluation

Evaluation details. The standard linear classification protocol [38] is adopted, where a linear layer is learned upon the fixed encoder to classify ImageNet [11] images. Follow-

Table 1. Performance comparison on linear and KNN evaluation.

Method	Batch size	Top1-Acc	KNN-Top1-Acc
NPID [38]	256	58.5	46.8
LocalAgg [48]	128	58.8	-
MoCo [19]	256	60.8	45.0†
SimCLR [7]	256	61.9	57.4†
MoCo v2 [8]	256	67.5	55.8†
CPC v2 [34]	512	67.6	-
PCL v2 [26]	256	67.6	58.1†
PIC [3]	512	67.6	54.7†
MoCHi [24]	512	67.6	57.5†
DetCo [39]	256	68.6	58.9†
AdCo [21]	256	68.6	57.2†
HCSC	256	**69.2**	**60.7**
SwAV* [6]	256	72.7	62.4†
AdCo* [21]	256	73.2	66.3†
HCSC*	256	**73.3**	**66.6**

* With multi-crop augmentation.
† Evaluated by us with officially released model weights.

ing PCL [26], an SGD optimizer (weight decay: 0; momentum: 0.9; batch size: 256) optimizes model for 100 epochs.

In KNN evaluation [38], the label of each sample is predicted by aggregating the labels of its nearest neighbors. Following NPID [38], we report the highest accuracy of such a KNN classifier over $K \in \{10, 20, 100, 200\}$.

Results. The second column of Tab. 1 compares different approaches on linear classification. Under the two commonly-used settings with and without multi-crop augmentation [6], HCSC outperforms all baseline methods. Especially, HCSC surpasses PCL v2 [26] which also incorporates instance-wise and prototypical contrastive learning. This performance gain illustrates the effectiveness of our pair selection scheme on selecting high-quality positive and negative pairs for instance-wise and prototypical contrast.

For KNN evaluation (the third column of Tab. 1), when the multi-crop augmentation is not used, HCSC achieves a 3.5% performance increase compared to previous methods. After adding multi-crop augmentation, HCSC also outperforms the state-of-the-art approach, AdCo [21].

5.2.2 Semi-Supervised Learning

Evaluation details. The encoder and linear classifier are fine-tuned on 1% or 10% labeled data in ImageNet. As in NPID [38], an SGD optimizer (weight decay: 0; momentum: 0.9; batch size: 256) optimizes model for 70 epochs.

Results. Tab. 2 presents the results of various methods on semi-supervised learning. Among all approaches, HCSC consistently achieves the best performance under different amount of labeled data. These results verify that the representations learned by HCSC possess decent global semantic structures that benefit the learning from insufficient data.

Table 2. Performance comparison on semi-supervised learning.

Labeled data	1%		10%	
Method	Top1-Acc	Top5-Acc	Top1-Acc	Top5-Acc
NPID [38]	-	39.2	-	77.4
MoCo v2 [8]†	36.7	64.4	60.7	83.4
MoCHi [24]†	38.2	65.4	61.1	83.5
SimCLR [7]†	46.8	74.2	63.6	**86.0**
PCL v2 [26]†	46.2	72.7	62.6	84.4
AdCo [21]†	43.6	71.6	61.8	84.2
HCSC	**48.0**	**75.6**	**64.3**	**86.0**
SwAV* [6]†	51.3	76.7	65.5	87.5
AdCo* [21]†	54.4	79.9	66.9	87.4
HCSC*	**55.5**	**80.9**	**68.7**	**88.6**

* With multi-crop augmentation.
† Evaluated by us with officially released model weights.

Table 3. Performance comparison on transfer learning.

Task	Object Classification		Object Detection	
Dataset	VOC07	Places205	VOC07+12	COCO
Method	mAP	Top1-Acc	AP_{50}	AP
NPID++ [38]	76.6	46.4	79.1	-
MoCo [19]	79.2	48.9	81.1	-
MoCo v2 [8]	84.0	50.1	82.4	40.6†
PCL v2 [26]	85.4	50.3	78.5	41.0†
AdCo [21]†	92.0	51.1	**82.6**	41.2
HCSC	**92.8**	**52.2**	82.5	**41.4**
SwAV* [6]	87.6	51.2	74.5†	38.6†
AdCo* [21]†	93.1	53.9	82.7	41.4
HCSC*	**93.3**	**55.0**	**83.2**	**41.6**

* With multi-crop augmentation.
† Evaluated by us with officially released model weights.

5.2.3 Transfer Learning

Evaluation details. We adopt two classification tasks on PASCAL VOC [15] and Places205 [47] and two detection tasks on PASCAL VOC [15] and COCO [29] to evaluate transfer learning. The fine-tuning paradigms on these two types of tasks completely follow those in MoCo [19].

Results. Tab. 3 presents the comparison among different approaches on transfer learning. Compared to the state-of-the-art method, AdCo [21], HCSC obtains better performance on seven of eight experimental settings. These results demonstrate that the image encoder learned by HCSC succeeds in capturing critical visual patterns shared across different image datasets.

5.2.4 Clustering Evaluation

Evaluation details. We employ two standard metrics for clustering evaluation, *i.e.* Normalized Mutual Information (NMI) [14] and Adjusted Mutual Information (AMI) [35]. Following PCL [26], the evaluation with 25,000 and 1,000 clusters are respectively performed.

Results. In Tab. 4, we report the results of clustering evaluation. PCL [26] included MoCo v2 [8] as a baseline by

Table 4. Performance comparison on clustering evaluation.

Method	# Clusters	NMI	AMI
DeepCluster [4]	25000	-	0.281
MoCo v2 [8]	25000	-	0.285
PCL v2 [26]	25000	0.616†	0.410
HCSC	25000	**0.629**	**0.462**
PCL v2 [26]†	1000	0.629	0.606
HCSC	1000	**0.638**	**0.616**

† Evaluated by us with officially released model weights.

Table 5. Ablation studies for different components of HCSC. All results are reported on linear classification.

HP	IL	PL	IS	PS	Top1-Acc
	✓	✓			67.6
✓	✓	✓			68.1
✓	✓	✓		✓	68.2
✓	✓	✓	✓		68.9
✓	✓	✓	✓	✓	**69.2**
✓		✓		✓	65.7
✓	✓		✓		68.2

HP: **H**ierarchical **P**rototypes; IL: **I**nstance-wise Contrastive **L**oss; PL: **P**rototypical Contrastive **L**oss; IS: **I**nstance-wise Pair **S**election; PS: **P**rototypical Pair **S**election.

conducting K-means on the image representations learned by MoCo v2, and we also adopt this baseline in our comparison. Under both the configurations with 25,000 and 1,000 clusters, HCSC clearly surpasses other baseline methods, which illustrates that the semantic hierarchies established in our approach can indeed improve the clustering quality.

6. Analysis

6.1. Ablation Study

Effect of hierarchical prototypes. Under the full model configuration, we fairly compare single prototype hierarchy with multiple prototype hierarchies by using the same number of prototypes for both cases. Comparing the second and fourth row of Tab. 6, we can observe a 1.1% performance gain after adding two more hierarchies of prototypes, which verifies the benefit of using hierarchical prototypes.

Effect of pair selection. In Tab. 5, a 0.8% performance gain is obtained after adding the negative pair selection scheme in instance-wise contrastive learning (fourth row *vs.* second row). Prototypical pair selection is less effective when applied individually (third row). However, by combining two pair selection schemes, the full model achieves the highest accuracy of 69.2% (fifth row), which demonstrates the complementarity of two pair selection schemes.

Effect of instance-wise and prototypical contrastive loss. From the last three rows of Tab. 5, we observe a 3.5% (1.0%) performance decay when the instance-wise (prototypical) contrastive loss is removed, which proves that these two kinds of contrastive losses are complementary.

Hierarchy - 1 | Hierarchy - 2 | Hierarchy - 3

Grizzly bears with fish in their mouth and rambling in water

Bears or dogs swimming or rambling in water

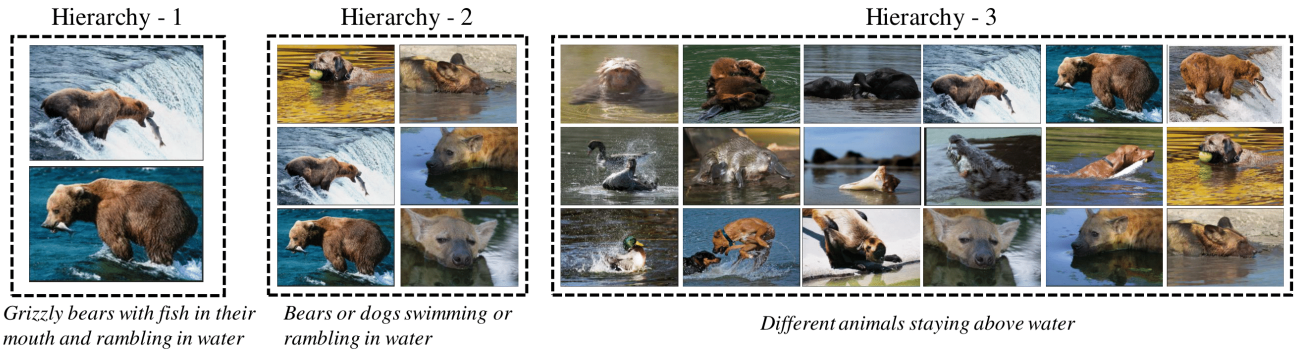*Different animals staying above water*

Figure 3. Visualize the images associated to a chain of prototypes from the bottom (*i.e.* first) hierarchy to the top (*i.e.* third) hierarchy.

Table 6. Sensitivity analysis on the number of hierarchies and the number of prototypes. All results are on linear classification.

Configuration of prototypes	Top1-Acc
3000	68.0
6000	68.1
3000-2000	69.0
3000-2000-1000	69.2
3000-2000-1000-500	69.2
1000-500-200	68.7
3000-2000-1000	69.2
10000-5000-1000	69.2
30000-10000-1000	69.3

6.2. Sensitivity Analysis

Sensitivity to the number of semantic hierarchies. In the first section of Tab. 6, we find that more semantic hierarchies (3 or 4 hierarchies) obviously benefit model's performance on linear classification. Under such settings, the hierarchical semantic structures underlying the pre-training database can be well captured by hierarchical prototypes.

Sensitivity to the number of prototypes. According to the second section of Tab. 6, the three configurations with sufficient prototypes (second to fourth rows) perform comparably well, while the configuration with insufficient prototypes (first row) performs worse. These results illustrate the importance of using abundant prototypes to fully capture the semantic clusters underlying the data.

6.3. Visualization

Visualization of hierarchical semantic structures. In Fig. 3, we visualize the images assigned to a chain of prototypes, which capture some interesting semantic hierarchies, *i.e.* "grizzly bears with fish in their mouth and rambling in water" → "bears or dogs swimming or rambling in water" → "different animals staying above water".

Visualization of pair selection. Fig. 4 shows the query sample and the negative candidates eliminated by our selection scheme. Most of these eliminated candidates own similar semantics with the query, which verifies the effectiveness of the proposed instance-wise pair selection.

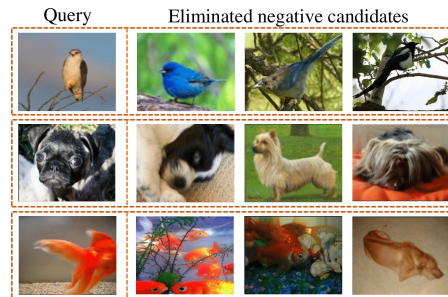Query | Eliminated negative candidates

Figure 4. Visualize the query sample and the negative candidates eliminated by our pair selection approach.

7. Conclusions and Future Work

This work proposes a novel contrastive learning framework, Hierarchical Contrastive Selective Coding (HCSC). In this framework, the hierarchical semantic structures underlying the data are captured by hierarchical prototypes. Upon these prototypes, a novel pair selection scheme is designed to better select positive and negative pairs for contrastive learning. Extensive experiments on various downstream tasks verify the superiority of our HCSC method.

The main limitation of the current HCSC method is that the hierarchical prototypes discovered during pre-training are discarded in downstream tasks. However, these prototypes contain rich semantic information, and they should benefit the semantic understanding of a downstream application in some way. Therefore, our future work will mainly focus on enhancing model's performance on downstream tasks by fully utilizing hierarchical prototypes.

References

[1] Yuki Markus Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. *arXiv preprint arXiv:1911.05371*, 2019. 2

[2] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*. Springer, 2010. 6

[3] Yue Cao, Zhenda Xie, Bin Liu, Yutong Lin, Zheng Zhang, and Han Hu. Parametric instance classification for unsupervised visual feature learning. *arXiv preprint arXiv:2006.14618*, 2020. 6

[4] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision*, 2018. 2, 7

[5] Mathilde Caron, Piotr Bojanowski, Julien Mairal, and Armand Joulin. Unsupervised pre-training of image features on non-curated data. In *IEEE International Conference on Computer Vision*, 2019. 2

[6] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Advances in Neural Information Processing Systems*, 2020. 1, 2, 3, 5, 6, 7

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, 2020. 1, 2, 3, 4, 6, 7

[8] Xinlei Chen, Haoqi Fan, Ross B. Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *CoRR*, abs/2003.04297, 2020. 1, 4, 5, 6, 7

[9] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *IEEE International Conference on Computer Vision*, 2021. 2

[10] Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. In *Advances in Neural Information Processing Systems*, 2020. 4

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009. 6

[12] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015. 1

[13] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. *arXiv preprint arXiv:2104.14548*, 2021. 2

[14] Pablo A Estévez, Michel Tesmer, Claudio A Perez, and Jacek M Zurada. Normalized mutual information feature selection. *IEEE Transactions on Neural Networks*, 20(2):189–201, 2009. 7

[15] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. 7

[16] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations*, 2018. 1, 2

[17] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020. 2

[18] Sungwon Han, Sungwon Park, Sungkyu Park, Sundong Kim, and Meeyoung Cha. Mitigating embedding and class assignment mismatch in unsupervised image classification. In *European Conference on Computer Vision*. Springer, 2020. 2

[19] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 1, 2, 3, 4, 6, 7

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE International Conference on Computer Vision*, 2016. 6

[21] Qianjiang Hu, Xiao Wang, Wei Hu, and Guo-Jun Qi. Adco: Adversarial contrast for efficient learning of unsupervised representations from self-trained negative adversaries. In *IEEE International Conference on Computer Vision*, 2021. 6, 7

[22] Xu Ji, Joao F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2

[23] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 2019. 6

[24] Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. *arXiv preprint arXiv:2010.01028*, 2020. 6, 7

[25] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. In *European Conference on Computer Vision*. Springer, 2016. 2

[26] Junnan Li, Pan Zhou, Caiming Xiong, Richard Socher, and Steven C. H. Hoi. Prototypical contrastive learning of unsupervised representations. In *International Conference on Learning Representations*, 2021. 1, 2, 3, 4, 5, 6, 7

[27] Yunfan Li, Peng Hu, Zitao Liu, Dezhong Peng, Joey Tianyi Zhou, and Xi Peng. Contrastive clustering. In *AAAI Conference on Artificial Intelligence*, 2021. 2

[28] Renjie Liao, Alexander Schwing, Richard S Zemel, and Raquel Urtasun. Learning deep parsimonious representations. In *Advances in Neural Information Processing Systems*, 2016. 2

[29] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence

Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*. Springer, 2014. 7

[30] Stuart Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982. 2

[31] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 6

[32] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*. Springer, 2016. 1, 2

[33] Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation learning by learning to count. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 1, 2

[34] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *CoRR*, abs/1807.03748, 2018. 1, 2, 3, 6

[35] Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *The Journal of Machine Learning Research*, 11:2837–2854, 2010. 7

[36] Xudong Wang, Ziwei Liu, and Stella X Yu. Unsupervised feature learning by cross-level instance-group discrimination. In *IEEE International Conference on Computer Vision*, 2021. 2

[37] Xiao Wang and Guo-Jun Qi. Contrastive learning with stronger augmentations. *arXiv preprint arXiv:2104.07713*, 2021. 2

[38] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *IEEE International Conference on Computer Vision*, 2018. 1, 2, 6, 7

[39] Enze Xie, Jian Ding, Wenhai Wang, Xiaohang Zhan, Hang Xu, Peize Sun, Zhenguo Li, and Ping Luo. Detco: Unsupervised contrastive learning for object detection. In *IEEE International Conference on Computer Vision*, 2021. 6

[40] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *International Conference on Machine Learning*. PMLR, 2016. 2

[41] Minghao Xu, Hang Wang, Bingbing Ni, Hongyu Guo, and Jian Tang. Self-supervised graph-level representation learning with local and global structure. *arXiv preprint arXiv:2106.04113*, 2021. 2

[42] Bo Yang, Xiao Fu, Nicholas D Sidiropoulos, and Mingyi Hong. Towards k-means-friendly spaces: Simultaneous deep learning and clustering. In *International Conference on Machine Learning*. PMLR, 2017. 2

[43] Jianwei Yang, Devi Parikh, and Dhruv Batra. Joint unsupervised learning of deep representations and image clusters. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 2

[44] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv preprint arXiv:2103.03230*, 2021. 2

[45] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *European Conference on Computer Vision*. Springer, 2016. 1, 2

[46] Wei Zhang, Xiaogang Wang, Deli Zhao, and Xiaoou Tang. Graph degree linkage: Agglomerative clustering on a directed graph. In *European Conference on Computer Vision*, 2012. 2

[47] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In *Advances in Neural Information Processing Systems*, 2014. 7

[48] Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *IEEE International Conference on Computer Vision*, 2019. 6