# Vision Transformer for Image Clustering

**Ryan Low**
Department of Computer Science
University of Maryland, College Park
rwlow@umd.edu

## Abstract

The Vision Transformer (ViT) is a relatively new development in the field of computer vision which has proven to be capable of significant performance improvements over the traditional convolutional neural network (CNN) for tasks such as image classification and semantic segmentation. One task which has not seen as much research focus as others is unsupervised image clustering, which attempts to group images into semantically meaningful clusters in the absence of ground-truth labels. Most methods for image clustering have utilized CNNs, autoencoders, and generative models to learn features representative of the data. In this work, we leverage the performance of ViT and an end-to-end contrastive learning method to learn feature representations and cluster assignments simultaneously. Specifically, two randomly transformed versions of each image in a batch are passed through a shared ViT backbone, then two independent projection heads take the ViT encodings of these images and perform both instance-level contrastive learning and global cluster learning. This paper demonstrates that our approach, which we call Vision Transformer for Image Clustering (ViT-IC), is able to achieve comparable results to state-of-the-art methods on several image datasets based on metrics commonly used to evaluate image clusterings.

## 1 Introduction

The clustering task is one of the most important in the unsupervised learning domain. Without labels for objects in a dataset, organizing the data into distinct groups of similar objects can provide valuable insights for data analysis. While many of the methods which already exist for clustering data demonstrate acceptable practical results [1], [2], most of these algorithms struggle on high-dimensional data, such as image datasets, because they fail to find sufficient representations of the data. To address this issue, deep clustering [3] utilizes neural networks to extract representative information from data to facilitate downstream clustering. Furthermore, interest has shifted to deep clustering in an end-to-end fashion, for methods which alternate between separate stages of representation learning and clustering tend to suffer from error accumulation and suboptimal clustering performance [4].

Coincidentally, the Vision Transformer (ViT) [5] has become a promising alternative to the convolutional neural network (CNN) [6] in many computer vision tasks. While the CNN is very effective in capturing feature locality, ViT is able to capture global dependencies through a multi-head self-attention mechanism [7]. In ViT, an image is split into a sequence of small patches and then fed through a Transformer encoder [7] for representation learning. As a result of its demonstrated success in the field, Transformer-based architectures have become predominant for a variety of vision tasks like image classification [5] and semantic segmentation [8].

While some research works have recently looked into the use of ViT for self-supervised learning, none have thoroughly investigated ViT for the image clustering task. We propose Vision Transformer for Image Clustering (ViT-IC), a novel deep clustering method for performing image clustering with ViT. By drawing on recent developments in contrastive learning [9], ViT-IC is able to effectively learn instance and cluster feature representations on image datasets end-to-end.

## 2 Related work

Deep clustering methods expand on traditional machine learning clustering approaches, like $K$-means [1] and spectral clustering (SC) [2], which only focus on local pixel-level information but ignore high-level and semantic information [4]. DeepCluster [10] proposes an iterative process to cluster CNN deep features and use $K$-means cluster assignments as pseudo-labels to learn the CNN parameters. Deep Embedded Clustering (DEC) [11] utilizes a stacked autoencoder which drops the decoder after training and uses the features extracted by the encoder to serve as input for a clustering module. Deep convolutional generative adversarial network (DCGAN) [12] introduces a class of CNNs which can be trained in a manner similar to generative models and learn a hierarchy of representations. SimCLR [13] demonstrates the importance of multiple data augmentations for constructing the pairs used in contrastive learning.

ViT is also beginning to emerge in self-supervised contrastive learning works. Notably, DINO [14] describes an approach called self-distillation with no labels that combines self-supervision and knowledge distillation, while also explores the properties of self-supervised ViT compared to supervised ViT. Momentum Contrast (MoCo) v3 [15] demonstrates using ViT in the MoCo framework, where momentum contrastive learning with encoders and a queue for saving negative samples are employed.

## 3 ViT-IC

In this section, we describe the details of the proposed framework as shown in Figure 1. Section 3.1 first explains the ViT backbone and the projection heads we use. Then, Section 3.2 briefly provides the procedures used to augment the data for training and evaluating ViT-IC. Lastly, Section 3.3 describes the contrastive loss functions used to train ViT-IC.

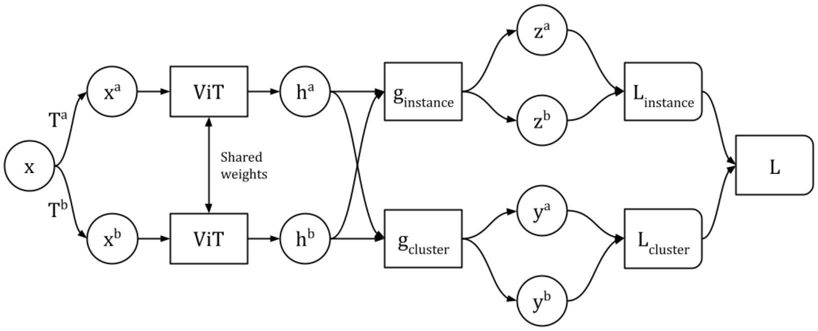


Figure 1: Vision Transformer for Image Clustering (ViT-IC).

### 3.1 Vision Transformer

ViT-IC implements a vanilla version of ViT [5] with a convolutional stem [16] as the backbone for feature extraction. While a patchify stem performing $16 \times 16$ convolutions with a stride of 16 followed by a flattening of the patches is traditionally used to create the input sequence of feature embeddings in ViT, replacing the patchify stem with a convolutional stem can help make optimization more stable to choices of optimizer, dataset, and other hyperparameters while improving overall performance [16]. The convolutional stem consists of a small series of stride-2 $3 \times 3$ convolutions and batch normalization layers followed by a $1 \times 1$ convolution to match the Transformer encoder dimension. The use of small and overlapping convolutions helps the model capture fine-grained local features and associations compared to the patchify stem. In addition, because ViT models global relations in the input sequence of feature embeddings but is permutation-invariant, sine-cosine positional encodings [7] are added to the input sequence of the self-attention layers to incorporate information regarding the relative positions of the embeddings.

Directly after ViT, the ViT encodings get passed to instance-level and cluster-level projection heads [13]. Projection heads are necessary to avoid information loss induced by the contrastive loss when using the feature representations from ViT directly [13]. The projection heads we use are small neural networks which map the feature representations to a space where contrastive loss is applied.

## 3.2 Data transformations

For each training image, two data transformations composed of stochastic augmentations ResizedCrop, ColorJitter, Grayscale, GaussianBlur, HorizontalFlip, and Solarize are used to produce two correlated images similar to the setting in SimCLR [13]. That is, for image $x$, we obtain images $x^a = T^a(x)$ and $x^b = T^b(x)$ where $T^a, T^b \sim T$. These transformed images are what we pass to our model during training as shown in Figure 1. The purpose of performing these transformations is so we can employ a contrastive learning strategy that creates positive and negative pairs for the model to discriminate between [17]. During evaluation, the image is only resized to the model input size.

## 3.3 Loss functions

ViT-IC performs both instance-level and cluster-level contrastive learning through the two independent projection heads. Contrastive learning aims to learn a low-dimensional representation of the data which maximizes the similarities of positive pairs and minimizes the similarities of negative pairs. We adopt the same loss functions used by Contrastive Clustering (CC) [9] and discuss them in the following sections.

### 3.3.1 Instance-level contrastive loss

To measure pairwise similarity of two equally-sized vectors, we use cosine similarity. The cosine similarity is given by the equation

$$S(\alpha, \beta) = \frac{\alpha^\top \beta}{||\alpha|| \cdot ||\beta||} \tag{1}$$

Suppose we have a mini-batch of $N$ images resulting in a total of $2N$ augmented images, that is $I = \{x_1^a, ..., x_N^a, x_1^b, ..., x_N^b\}$. For a single image $x_i^a$, this set contains one positive pair (with $x_i^b$) and $2N - 2$ negative pairs. Let $z_i^a$ and $z_i^b$ be the instance-level feature representations produced by the instance projection head for image augmentations $x_i^a$ and $x_i^b$ respectively. To distinguish $x_i^a$ from other images that would form a negative pairing, we use InfoNCE loss [18] and get the instance loss of $x_i^a$:

$$l_i^a = -\log \frac{\exp\left(S(z_i^a, z_i^b)/\tau_I\right)}{\sum_{j=1}^{N}[\exp\left(S(z_i^a, z_j^a)/\tau_I\right) + \exp\left(S(z_i^a, z_j^b)/\tau_I\right)]} \tag{2}$$

where $\tau_I$ is the instance-level temperature parameter, and $l_i^b$ defined similarly. The instance-level contrastive loss for the entire mini-batch is then defined as the average loss of all augmented images in the batch, that is

$$L_{instance} = \frac{1}{2N} \sum_{i=1}^{N} (l_i^a + l_i^b) \tag{3}$$

### 3.3.2 Cluster-level contrastive loss

The $C$-dimensional feature vector that the cluster projection head outputs for a sample can be thought as a soft label representing the probabilities of it belonging to each of the $C$ clusters, *i.e.*, the $i$-th entry of the vector corresponds to the probability that the sample belongs to the $i$-th cluster. In this sense, we can define $Y^a, Y^b \in \mathbb{R}^{N \times C}$ to be the outputs of the cluster projection head for a mini-batch of size $N$ under $T^a$ and $T^b$ respectively. Furthermore, let $y_i^a$ and $y_i^b$ be the $i$-th column of $Y^a$ and $Y^b$. These columns form representations of the $C$ clusters in the mini-batch and can be used to form positive and negative pairs. With the set of cluster representations $U = \{y_1^a, ..., y_C^a, y_1^b, ..., y_C^b\}$, we form for representation $y_i^a$ a positive pair with $y_i^b$ and make the remaining $2C - 2$ pairs negative. We again use InfoNCE loss to get the cluster loss of cluster $y_i^a$:

$$\hat{l}_i^a = -\log \frac{\exp\left(S(y_i^a, y_i^b)/\tau_C\right)}{\sum_{j=1}^{C}[\exp\left(S(y_i^a, y_j^a)/\tau_C\right) + \exp\left(S(y_i^a, y_j^b)/\tau_C\right)]} \tag{4}$$

where $\tau_C$ is the cluster-level temperature parameter, and $\hat{l}_i^b$ similar. The cluster-level contrastive loss of the entire mini-batch for the $C$ clusters is then given by

$$L_{cluster} = \frac{1}{2C} \sum_{i=1}^{C} (\hat{l}_i^a + \hat{l}_i^b) - H(Y) \tag{5}$$

where $H(Y) = \sum_{i=1}^{C} [P(y_i^a) \log P(y_i^a) + P(y_i^b) \log P(y_i^b)]$ is the entropy of the cluster assignment probabilities $P(y_i^k) = \sum_{j=1}^{N} Y_{ji}^k \,/\, ||Y^k||_1, k \in \{a, b\}$ in the mini-batch. This entropy term is needed to avoid the trivial solution where most samples are assigned to the same cluster.

### 3.3.3 Overall loss

To jointly optimize the instance-level and cluster-level contrastive loss functions, we take the simple addition of the loss functions and obtain an overall multi-task loss function, *i.e.*,

$$L = L_{instance} + L_{cluster} \tag{6}$$

## 4 Experiments

We evaluate ViT-IC on three image datasets and compare its performance to other state-of-the-art clustering approaches found in the literature. We first introduce the configuration that we use for ViT-IC and the benchmarks that we use to assess its performance before presenting our results.

### 4.1 Implementation details

Our ViT model has the same configuration as ViT-Small [19] with a few modifications. Because of resource constraints, we simplify the model to have fewer parameters. There are instead 8 Transformer blocks, each with an embedding dimension of 288 and 12 attention heads. Our ViT also uses $128 \times 128$ inputs compared to the standard $224 \times 224$. The convolutional stem includes four $3 \times 3$ convolutions of stride 2 followed by a $1 \times 1$ convolution. Both instance and cluster projection heads contain three fully-connected layers with 1344 hidden units each, and the output feature dimension of the instance projection head is 128. The instance-level temperature parameter $\tau_I$ is set to 0.5, and the cluster-level temperature parameter $\tau_C$ is set to 1.0. We use the Adam optimizer with a constant learning rate of 0.001. We train ViT-IC using this configuration with a batch size of 256 for 1000 epochs.

### 4.2 Datasets

The datasets we use in our experiments are CIFAR-10 [20], CIFAR-100 [20], and STL-10 [21]. CIFAR-10 and CIFAR-100 are image datasets prevalent in the computer vision community. In the case of CIFAR-100, we use the 20 superclasses as our labels rather than the 100 fine labels. The STL-10 dataset contains 100,000 unlabeled images in addition to 13,000 labeled images of 10 classes from ImageNet [22] and is often used in unsupervised learning. We train ViT-IC on these datasets with both labeled and unlabeled images but evaluate only with the labeled images.

### 4.3 State-of-the-art methods

The clustering methods we compare ViT-IC to include four traditional clustering methods and six deep clustering methods that are representative of the state-of-the-art for image clustering. The traditional clustering methods are $K$-means [1], spectral clustering (SC) [2], agglomerative clustering (AC) [23], and non-negative matrix factorization (NMF) [24]. The deep clustering methods are deep convolutional generative adversarial network (DCGAN) [12], variational autoencoder (VAE) [25], Deep Embedded Clustering (DEC) [11], PartItion Confidence mAximisation (PICA) [26], Semantic Clustering by Adopting Nearest neighbors (SCAN) [27], and Contrastive Clustering (CC) [9]. The results for SC, NMF, DCGAN, and VAE are obtained by performing $K$-means clustering on the extracted features. These methods were evaluated using inputs of size $224 \times 224$ compared to ours which only takes $128 \times 128$ images.

Table 1: Comparison of ViT-IC to other clustering methods. Higher metric values indicate better clustering performance.

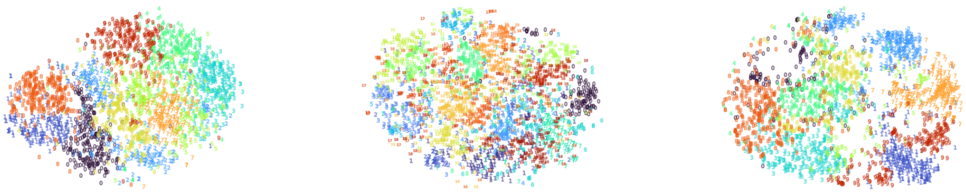
| Dataset | CIFAR-10 | | | CIFAR-100 | | | STL-10 | | |
|---|---|---|---|---|---|---|---|---|---|
| Metric | NMI | ARI | ACC | NMI | ARI | ACC | NMI | ARI | ACC |
| K-means | 0.087 | 0.049 | 0.229 | 0.084 | 0.028 | 0.130 | 0.125 | 0.061 | 0.192 |
| SC | 0.103 | 0.085 | 0.247 | 0.090 | 0.022 | 0.136 | 0.098 | 0.048 | 0.159 |
| AC | 0.105 | 0.065 | 0.228 | 0.098 | 0.034 | 0.138 | 0.239 | 0.140 | 0.332 |
| NMF | 0.081 | 0.034 | 0.190 | 0.079 | 0.026 | 0.118 | 0.096 | 0.046 | 0.180 |
| DCGAN | 0.265 | 0.176 | 0.315 | 0.120 | 0.045 | 0.151 | 0.210 | 0.139 | 0.298 |
| VAE | 0.245 | 0.167 | 0.291 | 0.108 | 0.040 | 0.152 | 0.200 | 0.146 | 0.282 |
| DEC | 0.257 | 0.161 | 0.301 | 0.136 | 0.050 | 0.185 | 0.276 | 0.186 | 0.359 |
| PICA | 0.591 | 0.512 | 0.696 | 0.310 | 0.171 | 0.337 | 0.611 | 0.531 | 0.713 |
| SCAN | 0.712 | 0.665 | 0.818 | 0.441 | 0.267 | 0.422 | 0.654 | 0.590 | 0.755 |
| CC | 0.705 | 0.637 | 0.790 | 0.431 | 0.266 | 0.429 | 0.764 | 0.726 | 0.850 |
| ViT-IC | 0.495 | 0.363 | 0.562 | 0.362 | 0.212 | 0.357 | 0.510 | 0.400 | 0.567 |

### 4.4 Metrics

To compare the clustering performance of ViT-IC against these algorithms, we utilize in our experiments the normalized mutual information (NMI) [28], the adjusted Rand index (ARI) [29], and the cluster accuracy (ACC) [30] metrics. These metrics are standard and widely used for the image clustering task. All metric scores fall between 0.0 and 1.0, where a score close to 0.0 implies the clusterings largely disagree and a score of 1.0 implies identical clusterings. We treat the output from each clustering approach as one set of clusterings and the ground-truth labels as another set of clusterings for calculating the scores. Note that the metric scores are independent of the actual cluster label values assigned by the methods.

### 4.5 Results

Table 1 summarizes the performance of the clustering methods on the benchmark datasets with the three metrics. ViT-IC is able to outperform seven of the ten baselines by a large margin on all three datasets. Particularly, the metric scores that ViT-IC achieve exceed that of the best-performing among these baselines, DCGAN on CIFAR-10 and DEC on CIFAR-100 and STL-10, by more than or nearly twofold. However, ViT-IC does not perform better than SCAN or CC on any benchmark, and performs better than PICA on CIFAR-100 only. On CIFAR-10, ViT-IC obtains an NMI of 0.495 while the method that performs directly better, PICA, sees a relative gain of 19.4% with an NMI of 0.591. On CIFAR-100, the directly better method to ViT-IC, CC, achieves a 19.1% relative gain over ViT-IC with an NMI of 0.431 compared to ViT-IC with an NMI of 0.362. On STL-10, ViT-IC scores an NMI of 0.510 and PICA shows a 19.8% relative gain over ViT-IC with an NMI of 0.611. Similar observations in terms of ARI and ACC can be seen. However, because of the trade-off we make which leaves ViT-IC with fewer parameters and a smaller input size, ViT-IC can likely achieve performance that matches or even exceeds these methods if the model was configured to be larger.



(a) CIFAR-10

(b) CIFAR-100

(c) STL-10

Figure 2: The t-SNE visualizations of ViT-IC instance feature representations on the three datasets.

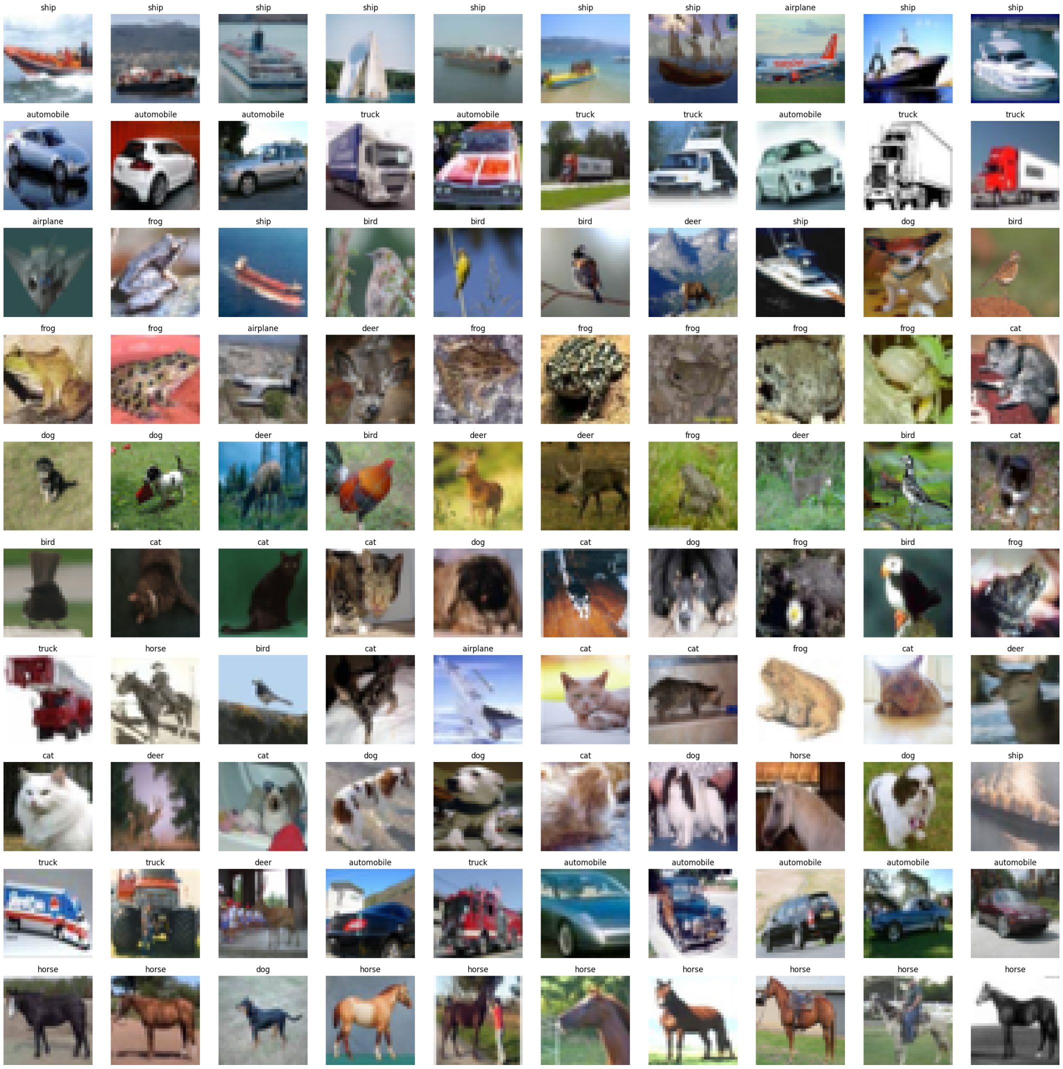


Figure 3: Image clusters produced by ViT-IC on CIFAR-10. Each row represents a different cluster.

By performing contrastive learning on both instance-level and cluster-level simultaneously, ViT-IC should learn discriminative features which form separate clusters. The t-SNE [31] visualizations of the instance features and cluster assignments after training on each dataset are shown in Figure 2, where different colors represent different clusters assigned by the cluster projection head. Particularly in the cases of CIFAR-10 and STL-10, some cluster separation can be observed. We can also see for all datasets that clusters have roughly the same number of features assigned to them as a result of the entropy term in the cluster-level contrastive loss.

Figure 3 shows a visualization of the clusters computed by ViT-IC on CIFAR-10. For each cluster, a random sample of 10 images is taken and displayed in a row with the ground-truth labels above them. We can see that images in the same cluster appear to contain similar visual features, and that the ground-truth labels are semantically close. Especially for the top and bottom rows in the figure, the labels are almost homogeneous. The visualizations on CIFAR-100 and STL-10 can be found in Appendix A.

## 5 Conclusion

In this work, we describe an unsupervised image clustering method with a ViT backbone that can group images into clusters that appear semantically similar. Vision Transformer for Image Clustering

(ViT-IC) performs contrastive learning at both the instance-level and cluster-level simultaneously. By passing two stochastically transformed versions of images through a shared ViT backbone, then passing the ViT encodings to instance and cluster projection heads, ViT-IC optimizes a multi-task contrastive loss to learn discriminative feature representations and cluster assignments. We show that ViT-IC is able to deliver results that are comparable with state-of-the-art approaches on CIFAR-10, CIFAR-100, and STL-10 datasets based on metrics that are standard for evaluating clustering methods.

## References

[1] J. B. MacQueen. “Some Methods for Classification and Analysis of MultiVariate Observations”. In: *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*. Ed. by L. M. Le Cam and J. Neyman. Vol. 1. University of California Press, 1967, pp. 281–297.

[2] Lihi Zelnik-manor and Pietro Perona. “Self-Tuning Spectral Clustering”. In: *Advances in Neural Information Processing Systems*. Ed. by L. Saul, Y. Weiss, and L. Bottou. Vol. 17. MIT Press, 2004.

[3] Erxue Min et al. “A Survey of Clustering With Deep Learning: From the Perspective of Network Architecture”. In: *IEEE Access* 6 (2018), pp. 39501–39514. DOI: 10.1109/ACCESS.2018.2855437.

[4] Zhiyuan Dang et al. “Doubly Contrastive Deep Clustering”. In: *CoRR* abs/2103.05484 (2021). arXiv: 2103.05484.

[5] Alexey Dosovitskiy et al. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. 2020. DOI: 10.48550/ARXIV.2010.11929.

[6] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems*. Ed. by F. Pereira et al. Vol. 25. Curran Associates, Inc., 2012.

[7] Ashish Vaswani et al. “Attention Is All You Need”. In: *CoRR* abs/1706.03762 (2017). arXiv: 1706.03762.

[8] Zhe Chen et al. “Vision Transformer Adapter for Dense Predictions”. In: *arXiv preprint arXiv:2205.08534* (2022).

[9] Yunfan Li et al. “Contrastive Clustering”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 10. 2021, pp. 8547–8555.

[10] Mathilde Caron et al. *Deep Clustering for Unsupervised Learning of Visual Features*. 2018. DOI: 10.48550/ARXIV.1807.05520.

[11] Junyuan Xie, Ross Girshick, and Ali Farhadi. *Unsupervised Deep Embedding for Clustering Analysis*. 2015. DOI: 10.48550/ARXIV.1511.06335.

[12] Alec Radford, Luke Metz, and Soumith Chintala. *Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks*. 2015. DOI: 10.48550/ARXIV.1511.06434.

[13] Ting Chen et al. “A Simple Framework for Contrastive Learning of Visual Representations”. In: *CoRR* abs/2002.05709 (2020). arXiv: 2002.05709.

[14] Mathilde Caron et al. *Emerging Properties in Self-Supervised Vision Transformers*. 2021. DOI: 10.48550/ARXIV.2104.14294.

[15] Xinlei Chen, Saining Xie, and Kaiming He. *An Empirical Study of Training Self-Supervised Vision Transformers*. 2021. DOI: 10.48550/ARXIV.2104.02057.

[16] Tete Xiao et al. “Early Convolutions Help Transformers See Better”. In: *CoRR* abs/2106.14881 (2021). arXiv: 2106.14881.

[17] Alexey Dosovitskiy et al. “Discriminative Unsupervised Feature Learning with Convolutional Neural Networks”. In: *CoRR* abs/1406.6909 (2014). arXiv: 1406.6909.

[18] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. “Representation Learning with Contrastive Predictive Coding”. In: *CoRR* abs/1807.03748 (2018). arXiv: 1807.03748.

[19] Hugo Touvron et al. “Training data-efficient image transformers & distillation through attention”. In: *CoRR* abs/2012.12877 (2020). arXiv: 2012.12877.

[20] Alex Krizhevsky and Geoffrey Hinton. *Learning multiple layers of features from tiny images*. Tech. rep. 0. Toronto, Ontario: University of Toronto, 2009.

[21] Adam Coates, Andrew Ng, and Honglak Lee. “An Analysis of Single-Layer Networks in Unsupervised Feature Learning”. In: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. Ed. by Geoffrey Gordon, David Dunson, and Miroslav Dudík. Vol. 15. Proceedings of Machine Learning Research. Fort Lauderdale, FL, USA: PMLR, 2011, pp. 215–223.

[22] Jia Deng et al. “ImageNet: A large-scale hierarchical image database”. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.

[23] K. Chidananda Gowda and G. Krishna. “Agglomerative clustering using the concept of mutual nearest neighbourhood”. In: *Pattern Recognition* 10.2 (1978), pp. 105–112. DOI: https://doi.org/10.1016/0031-3203(78)90018-3.

[24] Deng Cai et al. “Locality Preserving Nonnegative Matrix Factorization”. In: *Proceedings of the 21st International Joint Conference on Artificial Intelligence*. IJCAI’09. Pasadena, California, USA: Morgan Kaufmann Publishers Inc., 2009, pp. 1010–1015.

[25] Diederik P Kingma and Max Welling. *Auto-Encoding Variational Bayes*. 2013. DOI: 10.48550/ARXIV.1312.6114.

[26] Jiabo Huang, Shaogang Gong, and Xiatian Zhu. “Deep Semantic Clustering by Partition Confidence Maximisation”. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 8846–8855. DOI: 10.1109/CVPR42600.2020.00887.

[27] Wouter Van Gansbeke et al. “Learning To Classify Images Without Labels”. In: *CoRR* abs/2005.12320 (2020). arXiv: 2005.12320.

[28] Alexander Strehl and Joydeep Ghosh. “Cluster Ensembles — a Knowledge Reuse Framework for Combining Multiple Partitions”. In: *J. Mach. Learn. Res.* 3.null (2003), pp. 583–617. DOI: 10.1162/153244303321897735.

[29] Nguyen Xuan Vinh, Julien Epps, and James Bailey. “Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance”. In: *Journal of Machine Learning Research* 11.95 (2010), pp. 2837–2854.

[30] Nam Nguyen and Rich Caruana. “Consensus Clusterings”. In: Nov. 2007, pp. 607–612. DOI: 10.1109/ICDM.2007.73.

[31] Laurens van der Maaten and Geoffrey Hinton. “Visualizing Data using t-SNE”. In: *Journal of Machine Learning Research* 9.86 (2008), pp. 2579–2605.

# A Appendix

As supplementary material, we include the visualizations of the image clusters produced by ViT-IC on CIFAR-100 [20] and STL-10 [21].

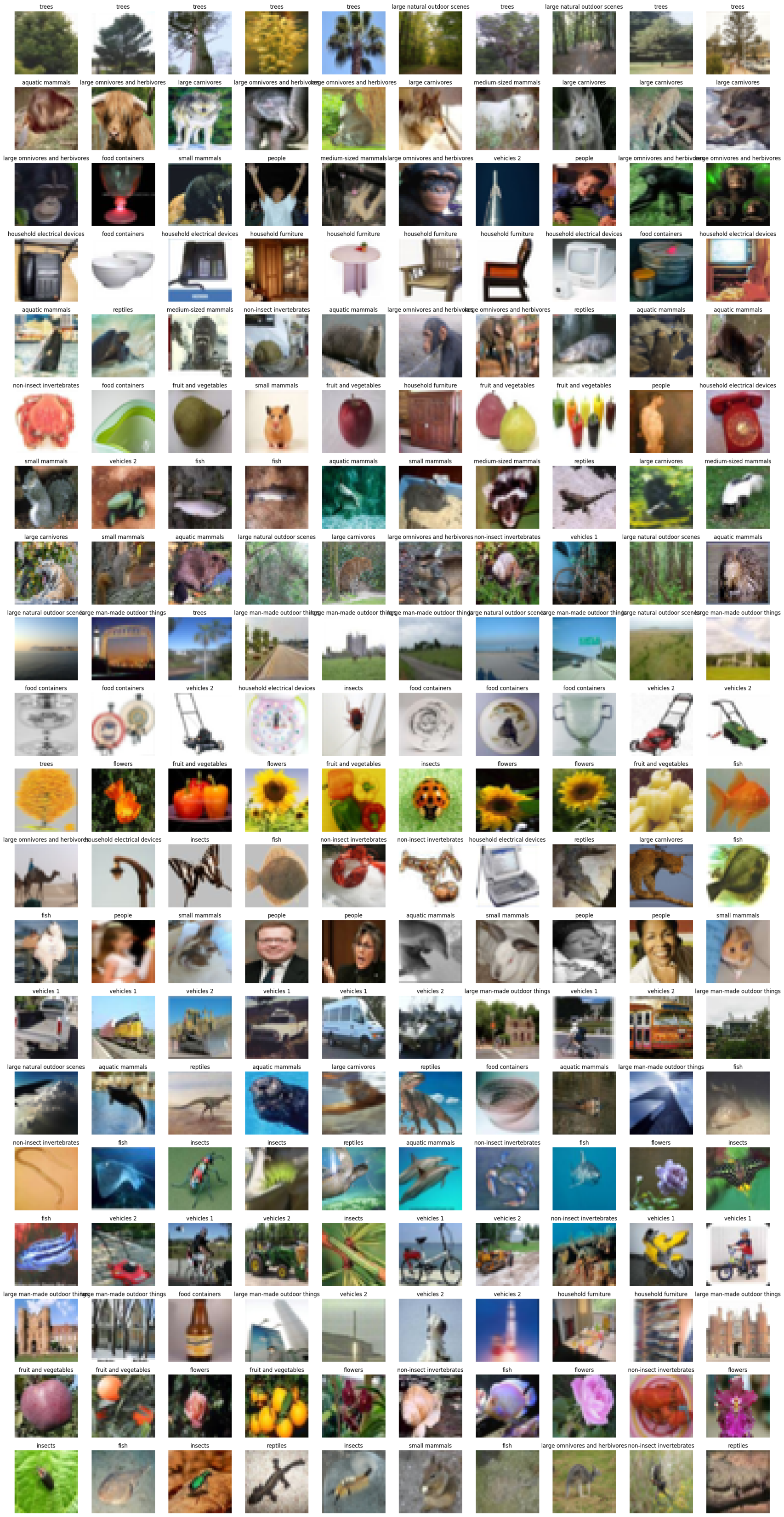


Figure 4: Image clusters produced by ViT-IC on CIFAR-100.

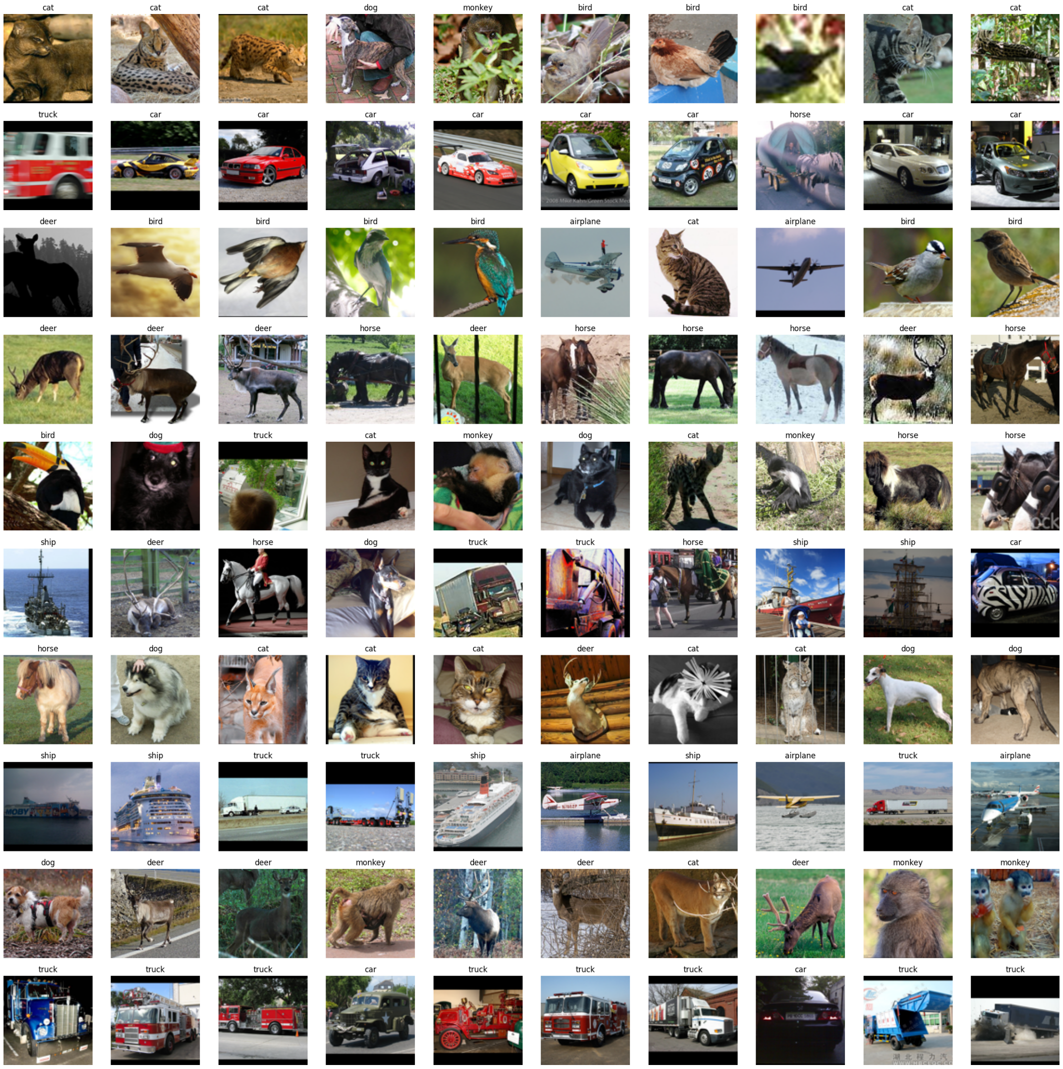


Figure 5: Image clusters produced by ViT-IC on STL-10.