DeepVAT: A Self-Supervised Technique for Cluster Assessment in Image Datasets

Alokendu Mazumder[1], Tirthajit Baruah[1], Akash Kumar Singh[2], Pagadala Krishna Murthy[1], Vishwajeet Pattanaik[1], Punit Rathore[1]

[1]Indian Institute of Science Bangalore, India
[2]Indian Institute of Science Education and Research Bhopal, India

Abstract

Estimating the number of clusters and cluster structures in unlabeled, complex, and high-dimensional datasets (like images) is challenging for traditional clustering algorithms. In recent years, a matrix reordering-based algorithm called Visual Assessment of Tendency (VAT), and its variants have attracted many researchers from various domains to estimate the number of clusters and inherent cluster structure present in the data. However, these algorithms face significant challenges when dealing with image data as they fail to effectively capture the crucial features inherent in images. To overcome these limitations, we propose a deep-learning-based framework that enables the assessment of cluster structure in complex image datasets. Our approach utilizes a self-supervised deep neural network to generate representative embeddings for the data. These embeddings are then reduced to 2-dimension using t-distributed Stochastic Neighbour Embedding (t-SNE) and inputted into VAT based algorithms to estimate the underlying cluster structure. Importantly, our framework does not rely on any prior knowledge of the number of clusters. Our proposed approach demonstrates superior performance compared to state-of-the-art VAT family algorithms and two other deep clustering algorithms on four benchmark image datasets, namely MNIST, FMNIST, CIFAR-10, and INTEL.

1. Introduction

Data clustering is a widely used unsupervised learning technique that involves dividing a collection of unlabeled objects into k groups of similar objects. Various clustering algorithms are available in the literature, such as hierarchical clustering, centroid-based approaches, density-based algorithms, and distribution-based clustering. Most clustering algorithms require k, the number of clusters to seek, as an input, which is the clustering tendency assessment problem. One common method to determine the number of clusters and their underlying structure is to visualize the data points using a 2D or 3D plot. However, this approach is only feasible for two- or three-dimensional datasets. For high-dimensional datasets such as images, time-series, visualizing and interpreting cluster structures using 2D or 3D visualization is not practical. Although various dimensionality reduction techniques, such as principal component analysis (PCA) and linear discriminant analysis (LDA), exist in the literature, these techniques often result in a low-dimensional representation of complex, high-dimensional datasets that may not fully reflect the inherent cluster structure due to information loss.

There are various formal (based on statistics) and informal (other approaches) techniques [7, 12] available in the literature for cluster structure assessment, but they are not completely effective. In contrast, visual approaches [6] have been in use for many years and serve as the foundation for many visual data analysis methods. The *Visual Assessment of Clustering Tendency* (VAT) [3], a matrix reordering-based visual-analytical method, is one of such algorithm which provides a visual way to assess the clustering tendency of various datasets. There are several variants of VAT available for different types of data, which are collectively known as the VAT family of algorithms. The VAT family of algorithms has become an acceptable and widely used tool in several domains like biomedical applications, speech processing, image segmentation, transportation applications, and *etc* for exploratory data analysis.

VAT algorithm employs a variant of Prim's minimum spanning tree algorithm [18] to perform matrix reordering of the pairwise dissimilarity matrix to generate a reordered dissimilarity matrix. The reordered dissimilarity matrix can be viewed as a monochrome image called a *Reordered Dissimilarity Image* (RDI) or cluster heat map. The RDI dis-

plays a possible cluster structure of the data set by showing dark blocks (data points of low dissimilarity values) along the diagonal. One method to obtain an accurate estimate of the number of clusters (k) from the RDI in the data is to count the number of dark blocks along the diagonal of the RDI. That means VAT not only can be used for cluster tendency assessment but also can be used for subsequent clustering of the input datasets, without needing the number of clusters.

This method is particularly effective for datasets with well-separated, compact clusters since the dark blocks along the diagonal are easily identifiable. However, for complex datasets (e.g., images, time series) having overlapping cluster structures (which is the case for most real-life datasets), existing VAT approaches perform poorly as the RDI quality degrades and the contrast between dark blocks along the diagonal and the rest of the image decrease. This makes it difficult to count the dark blocks along the diagonal.

There have been some efforts [10, 19, 30] to improve the quality of VAT generated RDI to accurately estimate the number of clusters for various complex geometry datasets. The VAT family algorithms, commonly used for analyzing cluster structures, exhibit poor performance when applied to image datasets, especially those with overlapping clusters. In the typical workflow, images are flattened before employing the VAT algorithms, resulting in the loss of their crucial spatial features. Consequently, the pixel-wise Euclidean distance becomes less effective in accurately capturing similarities or dissimilarities between images due to the feature loss incurred during flattening and curse of dimensionality. Fig. 1 shows an *improved Visual Assessment of Tendency* (iVAT) [10] RDI for a synthetic, high-dimensional dataset (number of samples = 1000, dimensions= 100) having three well-separated Gaussian mixtures (so k=3) in View (a), and RDI for a sample of popular MNIST dataset (number of samples = 1000 dimensions= 784, $k = 10$ classes) in View (b). It is evident from the figure that iVAT performs well when the data has inherently well separated clusters as we can clearly see three dark blocks along the diagonal in its RDI representing three clusters. However, when it comes to image datasets like MNIST, it struggles to provide meaningful results, as the resulting RDI does not exhibit clear dark blocks along the diagonal. This limitation highlights the need for a VAT variant that can effectively preserve the essential features of images, enabling more accurate assessments of cluster structures in image datasets.

As unsupervised deep learning methods (like autoencoders [23] and contrastive learning [4]) excel at extracting robust features from complex images, it makes them well-suited for developing a dedicated VAT algorithm for image datasets.

To address the above concerns, we propose a novel visual-analytical framework called DeepVAT. DeepVAT utilizes deep learning techniques to extract meaningful deep features from images, enabling more effective assessment of cluster structures. Unlike traditional VAT approaches, DeepVAT can uncover hidden cluster structures within image data, even in situations where ground truth labels or information about the number of classes are unavailable. Our major contributions are as follows:

(a) (b)

Figure 1: (a) iVAT image of 100-dimensional Compact and Separated (CS) Gaussian mixture data (3 Gaussians); (b) iVAT image of flattened MNIST data (784 dimensional)

1. We proposed a deep, self-supervised learning framework, DeepVAT, that can provide visual evidence of the number of clusters present in complex image datasets.

2. In our method, we did not incorporate any prior knowledge about the ground truth number of clusters of data.

3. We performed experiments on four real-world, publicly available, large image datasets to show the superiority of DeepVAT over other state-of-the-art VAT family algorithms (proposed for high-dimensional data) in terms of quality of RDI, clustering accuracy, and normalized mutual information (NMI) score.

To the best of our knowledge, our work represents the first investigation in the literature exploring the utilization of deep features from images in the context of VAT methods. This contribution highlights the importance of incorporating deep learning techniques in the development of VAT models for accurate and insightful analysis of image datasets.

Here is an outline of the rest of this article. Section 2 presents the preliminaries for the VAT/iVAT algorithm and reviews related work. The proposed algorithm, DeepVAT, is discussed in Section 3. Section 4 discusses the experiments and results, followed by conclusions in Section 5.

2. Preliminaries and Related work

2.1. VAT and iVAT

Consider we have a set of N objects, denoted as $O = o_1, o_2, \ldots, o_N$, where each object in O is described by a p-dimensional feature vector ($\in \mathbb{R}^p$). Alternatively, the data can be represented as a dissimilarity matrix, denoted

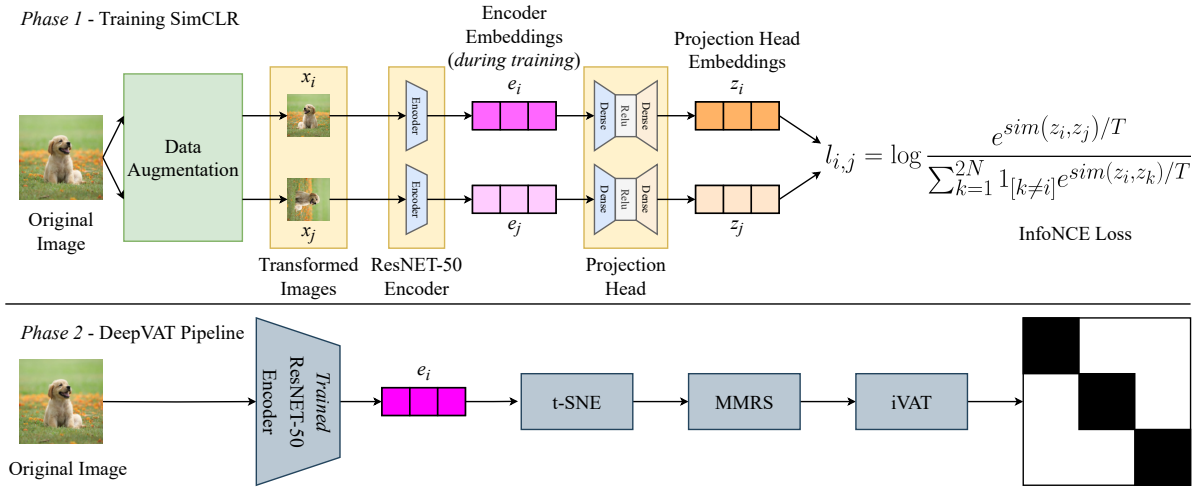


Figure 2: The proposed architecture of DeepVAT

as $D_N = [d_{ij}]$, where d_{ij} indicates the dissimilarity between object o_i and object o_j computed using a suitable distance measure. The VAT algorithm considers the dissimilarity matrix, D_N as input and reorders (by shuffling the rows and columns) using a modified Prim's algorithm. The image $I(D_N)$ of the reordered distance matrix D_N displays each pixel's intensity to indicate the dissimilarity between the corresponding row and column objects. When dark blocks appear along the diagonal, they might represent distinct clusters, ideally k (the original number of cluster in data) clusters. As single-linkage clusters are always diagonally aligned in VAT ordered images [11], k_p aligned clusters can be obtained by cutting the largest $(k_p - 1)$ edges (given by the MST cut magnitude order d) from the MST. Here, k_p is the estimated number of clusters from VAT/iVAT RDI.

The improved-VAT (iVAT) [10] enhances the quality of VAT [3] RDI by using path-based distance transformation. The iVAT transformed matrix $D'_N = [d'_{ij}]$ is generated using a path-based minimax distance [18]:

$$d'_{ij} = \min_{p \in P_{ij}} \max_{1<h<|p|} \mathbf{D}_{p[h]p[h+1]} \tag{1}$$

where $p \in P_{ij}$ is an acyclic path in the set of all acyclic paths between objects (o_i) and (o_j) (vertices i and j) in O.

2.2. VAT Variants for Large Volumes of High-Dimensional Data

Although the VAT tool, discussed above, finds its usefulness in many applications, it can be computationally expensive as the size of the data set grows due to its $\mathcal{O}(N^2)$ complexity. To understand the clustering structure for large volume datasets, a scalable version of VAT called *scalable VAT* (sVAT) was developed by Hathaway *et al.* [9], which utilizes a smart sampling based approach. To begin, sVAT extracts a smart sample of size n (where $n << N$) from the large data set X using *Maximin Random Sampling* (MMRS) [13]. The extracted sample is then used to compute the distance matrix D_n, which is input into VAT.

To handle large volumes of high-dimensional datasets, Rathore *et. al* in [19] proposed FensiVAT, an ensemble-based, hybrid clustering framework that combines fast data-space reduction using random projection with an intelligent sampling strategy to assess the clustering tendency of high-dimensional data. Recently, Zhang *et al.* [30] proposed another method that leverages a kernel-based dissimilarity matrix to refine the RDI further, called kernel-based iVAT (KernelVAT). They use a Gaussian kernel and Isolation kernel (data-dependent) to transform the RDI.

The SpecVAT [24] algorithm is another approach that improves the quality of the RDI produced by VAT. It utilizes spectral graph theory to transform the raw distance matrix into a graph embedding space using graph Laplacian. It then creates an alternative feature representation of the data by selecting the r most significant eigenvectors that correspond to the highest eigenvalues. VAT is then applied to this transformed representation, resulting in a much-improved RDI.

To our knowledge, none of the existing VAT family of algorithms, including those reviewed in this section, have been investigated thoroughly on image datasets. Moreover, they have been shown to perform poorly on image datasets in their numerical experiments. Below, we discuss our proposed framework, DeepVAT.

3. Proposed Framework: DeepVAT

In this paper, we propose a deep learning-based framework, DeepVAT, to advance the VAT family of algorithms for cluster structure assessment in complex image

datasets. Fig. 2 presents each step of our proposed framework. Below, we briefly explain each step of DeepVAT keyed to the blocks shown in Fig. 2.

3.1. Generating Image Embeddings

The first step in our framework is representation learning by employing deep learning architectures. The objective of this step is to attain a *cluster-friendly* representation of images, which involves bringing similar data points closer to each other and pushing dissimilar points further away. While deep learning architectures like autoencoders can be explored for this purpose, they may lack the inherent capability to produce a truly *cluster-friendly* representation.

Recently, a wide range of self-supervised approaches such as contrastive learning based models has been proposed that can provide *cluster-friendly* representations for images using deep neural networks, without the need for ground truth information. These models include *Simple Contrastive Learning of Representations* (SimCLR) [4], Barlow Twins [29], *Decoupled Contrastive Learning* (DCL) [1], SimSiam [5], *Bootstrap Your Own Latent* (BYOL) [8], and many others.

We observed that incorporating SimCLR as the feature extractor in the DeepVAT pipeline led to significantly superior iVAT images compared to when an autoencoder was used as the feature extractor. The significant difference observed can be attributed to the inherent capabilities of SimCLR compared to basic autoencoders. SimCLR has the ability to effectively group similar points together and push dissimilar points apart, thanks to the InfoNCE loss it minimizes [21]. In contrast, basic autoencoders lack this inherent capability. Additionally, a recent study [17] suggests that the InfoNCE loss aids in learning cluster-preserving representations of images, further highlighting the suitability of SimCLR for DeepVAT. Hence, we chose SimCLR as our primary model for creating embeddings in our proposed framework.

In SimCLR, the first stage includes performing auxiliary tasks or a given batch of images, such as corrupting the data, adding noise, and creating augmented views of the same data. These transformations generate fresh views of the same images, effectively enlarging the training set. Grayscale images cannot undergo certain transformations such as color jitters. Instead, an affine stretch is utilized along with rotation, resizing, and blurring. Through these tasks, the models can acquire a rich and beneficial representation of the data.

SimCLR consists of an encoder network and a non-linear projection head. The augmented images are fed into the encoder to extract high-level features. The encoder consists of several convolutional and fully connected layers and is trained using a contrastive loss function. The SimCLR framework utilizes an InfoNCE loss function [21] to measure the similarity between different views of an image. The model aims to maximize the similarity between the two views of the same image and minimize the similarity between views of different images. By doing so, SimCLR learns to extract valuable features robust to variations in the input data, which is helpful for generalization in real-world scenarios. The encoder projects the images into (say) d-dimensional space.

Then, the projection head, a small neural network, further maps the encoded features (d-dimensional) to a (lower) m-dimensional set of embeddings, and then back to a lower-manifold of d-dimensional space, resulting in a rank-deficient weight matrix. This projection head is trained alongside the encoder during training. After training successfully, the projection head is discarded, and the data is passed through the trained encoder to generate embeddings. The projection head serves as an additional non-linear transformation that helps to increase the quality of the learned features.

3.2. Dimensional Reduction using t-SNE

Despite the fact that SimCLR embeddings (shown with pink bar in Fig. 2) can be used to compute the dissimilarity matrix for VAT/iVAT, the high dimensionality of the SimCLR embeddings can lead to the curse of dimensionality problem, which can affect the quality of the resulting visualization. In our experiments, as discussed in Section 4, we observed that using SimCLR embeddings to compute the dissimilarity matrix did not result in a significant improvement in the quality of the resulting RDI for complex image datasets (CIFAR-10 [14] and INTEL [2]).

One way to tackle this issue is to apply t-SNE on a data representation obtained from SimCLR. Compared to the original flattened image data, t-SNE works better on SimCLR embeddings because SimCLR is a deep-layer architecture that can more efficiently represent the highly varying data manifold in multiple nonlinear layers [4, 22]. The projections generated by SimCLR's projection head can identify highly varying manifolds better than a local method like t-SNE, resulting in a higher quality visualization compared to using t-SNE on the original high-dimensional data [22]. However, it is important to acknowledge that representing the complete structure of intrinsically high-dimensional data in just two or three dimensions is fundamentally impossible, highlighting a fundamental limitation.

3.3. Smart Sampling: Maximin Random Sampling (MMRS)

Computing and analyzing VAT RDI using t-SNE embeddings (shown with a pink bar in Fig. 2), generated in the last step, may be infeasible for image datasets with large samples (N) due to $\mathcal{O}(N^2)$ complexity of VAT. To deal with large image datasets, we exploit a smart sampling approach

called *Maximin and Random Sampling* (MMRS).

Let $\mathbf{X} = \{x_i\}_{i=1}^{N}$ represent the set of t-SNE reduced embeddings obtained from the trained encoder, where $x_i \in \mathbb{R}^2$. The MMRS technique is an intelligent way to obtain samples in large batch data sets by combining MaxiMin (MM) and Random Sampling (RS). The MM sampling process starts by identifying a set of $k^{'}$ (an overestimate of k) distinguished objects, which are the farthest from each other in the input data $\mathbf{X}$. Then each point in the set $\mathbf{X}$ is grouped with its nearest distinguished object using the nearest prototype rule (NPR) (mentioned in [19]), which divides the entire dataset into k' groups $\{G_i\}_{i=1}^{k^{'}}$ where $G_i \subseteq \mathbf{X}$, $\forall i \in \{1, 2, \ldots, k^{'}\}$ by associating $|G_i|$ points to i^{th} *MM sample*, which represents each of the $k^{'}$ group. Finally, the sample $\mathcal{S}$ of size $n << N$ is formed by selecting random data-points from each of the $k^{'}$ groups $\{G_i\}_{i=1}^{k^{'}}$. The number of points n_j extracted from group G_j is proportional to the cardinality of G_j, i.e $n_j \propto |G_j|$. To be precise, $n_j = \lceil n \times |G_j|/N \rceil$, where $\lceil . \rceil$ is the ceiling function. This step gives us a smart sample of size $n << N$ in lower dimensional space. Rather than feeding a large number of embeddings directly into iVAT for visualization, we feed a *smart sample* of size n, obtained using MMRS.

3.4. Dissimilarity Matrix Computation for VAT/iVAT

The reduced-dimension, smart samples are used to compute dissimilarity matrix D_n which is fed to the VAT/iVAT algorithm to obtain reordered dissimilarity matrix $D_n^{'}$. The visualization of $I(D_n^{'})$ suggests the number of clusters k present in the dataset.

4. Experiments

We performed experiments on four publicly available, real, image datasets. We evaluated the ability of DeepVAT to suggest the number of clusters in image datasets and compared its performance with other VAT family methods that are claimed to work better with high-dimensional data. We also compare DeepVAT with two well known deep-clustering based methods. The experiments were conducted on a regular PC with the following configuration: OS: Ubuntu 22.04.2 LTS (64 bit); processor: Intel(R) Xeon(R) Gold 5220R CPU @ 2.20GHz; RAM: 62 GB; GPU: Nvidia Quadro RTX 6000, 24 GB.

4.1. Datasets

We performed our experiments on the following datasets:

1. **MNIST** [15]: It has a total of $60,000$ grayscale training images of digits with a dimension of $28{*}28$ ranging from 0 to 9, i.e., total 10 classes, with each class having 6,000 images. The full training set is used in all experiments (60,000 images).

2. **FMNIST** [26]: It has a total of 60,000 grayscale training images of fashion apparel with a dimension of $28 * 28$, i.e., it has a total of 10 classes, with each class having 6,000 images. The full training set is used in all experiments (60,000 images).

3. **CIFAR10** [14]: It has a total of $50,000$ natural RGB training images with a dimension of $32 * 32 * 3$. It has a total of 10 classes, with each class having $5,000$ images. The full training set is used in all experiments (50,000 images).

4. **Intel Image Dataset** [2]: It has $14,034$ natural RGB training images and 3,000 testing images with 6 classes. We clubbed both sets and used the final count of $17,000$ images to perform various experiments. Each image has a dimension of $32 * 32 * 3$.

4.2. Evaluation Criteria

We show all (best) iVAT images with an estimated number of clusters (k_p) for all the compared algorithms in Table 1. To estimate k_p, we used the algorithm presented in [25]. As mentioned in section 2.1, k_p clusters can be obtained by cutting *(k_p-1)* edges in MST provided by VAT/iVAT algorithm. We used the predicted labels and ground truth information of each dataset to compute the partition accuracy (PA) for the estimated value of k (from iVAT image) and normalized mutual information (NMI). The PA of a clustering algorithm is the percentage (%) ratio of the number of samples with matching ground truth and algorithmic labels to the total number of samples in the dataset. To ensure consistent label mapping between the predicted and true labels, the Kuhn-Munkres algorithm [16] is employed to find the best mapping between the predicted and ground truth labels. A higher value of PA and NMI implies a better match to the ground truth partition.

4.3. Comparison of DeepVAT with other Models

In this section, we make a qualitative and quantitative comparison of DeepVAT with existing state-of-the-art VAT family methods that claim to work with high-dimensional and complex data (images when flattened can be seen as high-dimensional data). Specifically, we compare DeepVAT with the following methods:

1. **VAT family methods**

   (a) **FensiVAT**: FensiiVAT [19] is applied on a small MMRS subset of the embeddings extracted from the trained encoder of SimCLR.

(b) **KernelVAT**: KernelVAT [30]. is applied on a small MMRS subset of the embeddings extracted from the trained encoder of SimCLR.

(c) **SpecVAT**: SpecVAT [24] is applied on a small MMRS subset of the embeddings extracted from the trained encoder of SimCLR.

2. **Deep-Clustering methods**

(a) **DEC** [27]: iVAT is applied to the t-SNE reduced embeddings, extracted from the trained encoder of DEC. Specifically, iVAT is applied to a smaller MMRS subset.

(b) **LSD-C** [20]: The t-SNE reduced embeddings, extracted from the trained encoder of DEC, are utilized for applying iVAT. More specifically, iVAT is applied to a smaller MMRS subset.

(c) **Autoencoder + iVAT**: We trained a vanilla autoencoder and obtained embeddings from the trained encoder network. Subsequently, t-SNE is applied to these embeddings, and iVAT is then applied specifically to a smaller MMRS subset of the reduced embeddings.

4.3.1 Parameter Settings

In DeepVAT, the SimCLR model was trained using the LARS optimizer [28] for each dataset, with $1,000$ epochs. The output dimension of the encoder network was set to $d = 2,048$, and the projection head network was chosen to have $m = 128$. We performed each experiment five times on each dataset and reported the average results. We use a batch size of 700 for MNIST and FMNIST and 256 for CIFAR10 and Intel Image Dataset. The parameters for MMRS sampling are k' = 15 for MNIST, FMNIST, and CIFAR10, and 10 for INTEL, number of samples, n: 4,000 for all datasets.

Euclidean distance is utilized as the metric to generate the RDI for the t-SNE reduced embeddings of the MNIST dataset. Likewise, in the ablation study discussed in Section 4.4, Euclidean distance is employed to generate the RDI for the t-SNE reduced embeddings of the raw-flattened MNIST data. For all other experiments, cosine dissimilarity is utilized as the dissimilarity measure to generate the RDI.

The input to all three VAT based methods is 2048-dimensional SimCLR embeddings as these methods transform the original data into a suitable embedding space/low dimensional space by virtue of their design. In KernelVAT, radial basis function (RBF) kernel is used, with the precision parameter (γ) set to 0.05. In FensiVAT, the down-space (reduced) dimension for random projection is chosen 100 when FensiVAT is applied to a 2048-dimensional SimCLR embedding. In SpecVAT, we performed iterations over the parameter *number of eigen-values* (r) ranging from 1 to 10 and noted the best result.

DEC [27] and LSD-C [20] heavily rely on prior information about the number of clusters in a dataset, while DeepVAT do not require this specific information. As stated in Section 4.3.1, we deliberately choose an overestimate for the number of clusters in all our experiments involving VAT algorithms. Consequently, for a fair comparison, we adopt the same overestimate ($k^{'}$ = 15 for MNIST, FMNIST, and CIFAR-10, and $k^{'}$ = 10 for INTEL) for DEC (which requires the value of k for performing k-means) and LSD-C (where the linear layer after the encoder has the same number of neurons as the number of classes).

To ensure fairness in the assessment, just like DeepVAT utilized t-SNE reduced embeddings from the SimCLR encoder, we also apply t-SNE to reduce the embeddings generated from the encoders of DEC and LSD-C to 2 dimensions before generating the RDI. We keep all the parameter settings the same unless stated otherwise.

4.4. Ablation Study

4.4.1 Results and Discussions

Table 1 shows the comparison of all six models based on the RDI quality and their ability to estimate the underlying clusters (k_p) accurately. Table 2 shows the comparison of DeepVAT with all the six methods mentioned above based on the PA and NMI.

We can see that the DeepVAT method generates much clearer and sharper dark blocks compared to the SpecVAT, KernelVAT, and FensiVAT models. Consequently, the number of dark blocks generated by DeepVAT (k_p) is close to the original number of classes (k) in the dataset, making it the most accurate in estimating the potential number of clusters compared to other algorithms. When applying FensiVAT and KernelVAT directly to the high-dimensional embedding, we observed that they produced blurry RDI (the cluster count is good but the quality of RDI is poor) and only achieved moderate quantitative results in terms of PA and NMI (refer Table 2) for simple datasets such as MNIST and FMNIST. However, when dealing with complex datasets like CIFAR-10 and INTEL, both algorithms failed to generate high-quality RDI and quantitative results (refer Table 2). Note that clustering algorithms face significant challenges when dealing with these datasets, as the ground truth labels may not accurately reflect distinct clusters within the feature vector representation of the data points. These results suggest that our approach produces more visually appealing and informative representations of the data.

Based on the results presented in Table 2, DeepVAT demonstrates a significant performance advantage over state-of-the-art VAT family methods in terms of both PA and NMI metrics. DeepVAT demonstrates its superiority over

Table 1: **Qualitative comparison with other methods.** When considering the visual quality of the RDI, our method stands out by providing the most accurate estimate of the number of clusters. **Note: Columns 2, 3, and 4:** FensiVAT, KernelVAT, and SpecVAT are applied to 2048-dimensional embeddings from SimCLR. **Columns 5 and 6:** iVAT is applied to the t-SNE reduced embeddings from the encoder of DEC and LSD-C, respectively. **Column 7:** iVAT is applied to the t-SNE reduced embeddings from the encoder of the autoencoder.

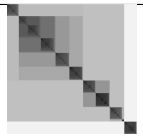
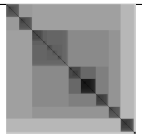
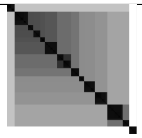
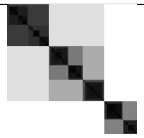
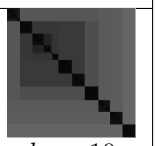
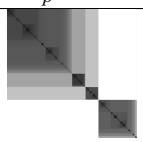
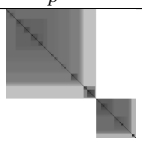
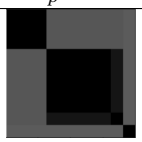
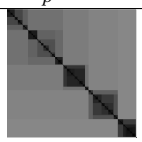
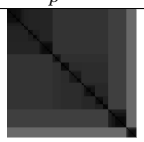
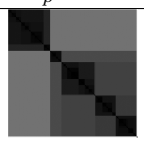
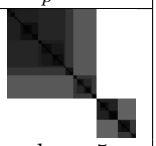
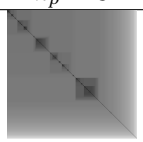
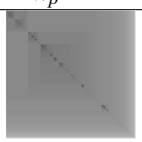
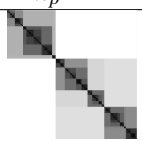
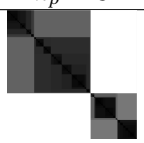
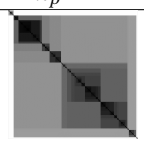
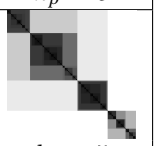
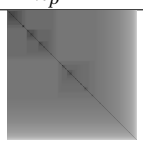
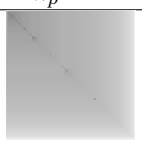
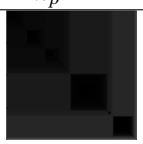
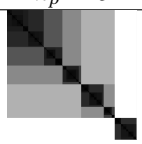
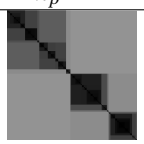
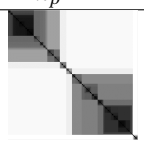
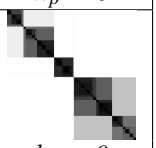
Dataset	*FensiVAT*	*KernelVAT*	*SpecVAT*	*DEC*	*LSD-C*	*Autoencoder*	**Ours**
MNIST							
$k = 10$	$k_p = 10$	$k_p = 8$	$k_p = 3$	$k_p = 15$	$k_p = 7$	$k_p = 10$	$k_p = 10$
FMNIST							
$k = 10$	$k_p = 3$	$k_p = 2$	$k_p = 3$	$k_p = 4$	$k_p = 3$	$k_p = 4$	$k_p = 5$
CIFAR-10							
$k = 10$	$k_p = 2$	$k_p = 1$	$k_p = 2$	$k_p = 5$	$k_p = 4$	$k_p = 4$	$k_p = 5$
INTEL							
$k = 6$	$k_p = 2$	$k_p = 1$	$k_p = 3$	$k_p = 5$	$k_p = 4$	$k_p = 3$	$k_p = 6$

Table 2: **Quantitative comparison with other methods.** Our method almost constantly reaches state-of-the-art performances by a good margin on all four datasets.

DATASETS / *MODELS*	*MNIST*		*FMNIST*		*CIFAR-10*		*INTEL*	
	ACC (%)	NMI	ACC (%)	NMI	ACC (%)	NMI	ACC (%)	NMI
FensiVAT †*on* Φ	51.78	0.62	29.5	0.5	10.2	0.01	17.8	0.01
KernelVAT †*on* Φ	40.01	0.58	29.4	0.49	10.13	0.01	17.55	0.01
SpecVAT †*on* Φ	11.37	0.004	10.27	0.006	10.19	0.004	17.6	0.003
DEC ‡*+ iVAT*	51.20	0.62	27.69	0.22	12.8	0.01	31.93	0.13
LSD-C ‡*+ iVAT*	51.94	0.58	41.47	0.46	38.17	0.41	39.64	0.31
Autoencoder + iVAT	41.32	0.61	40.80	0.53	17.80	0.075	30.05	0.14
Ours	**82.02**	**0.89**	**43.76**	**0.61**	**51.26**	**0.47**	**56.84**	**0.46**

Φ: 2048 dimensional SimCLR embedding; †: Models from *VAT/iVAT* family
‡: Deep clustering models.

deep-clustering algorithms by achieving a 35% improvement in PA and 30% improvement in NMI. Furthermore, it outperforms simple autoencoders by an impressive 95% on PA and 203% on NMI metrics, clearly highlighting its remarkable performance. As a result, DeepVAT surpasses all six competitive models in both PA and NMI measures.

The success of DeepVAT can be attributed to the use of SimCLR and t-SNE. SimCLR is effective at generating a robust representation of the dataset by leveraging non-linear functions, such as deep CNN encoders and projection heads, to approximate its intrinsic dimensionality. By applying t-SNE on the representation produced by SimCLR,

Table 3: **Ablation study.** We analyze the effects of removing different blocks from the DeepVAT pipeline on PA and NMI.

DATASETS / *MODELS*	*MNIST*		*FMNIST*		*CIFAR-10*		*INTEL*	
	ACC (%)	NMI	ACC (%)	NMI	ACC (%)	NMI	ACC (%)	NMI
Full DeepVAT	**82.02**	**0.89**	**43.76**	**0.61**	**51.26**	**0.47**	**56.84**	**0.46**
DeepVAT minus *SimCLR*	37.27	0.51	29.87	0.51	18.25	0.07	31.27	0.15
DeepVAT minus *t-SNE*	40.73	0.60	30.61	0.52	10.14	0.01	17.52	0.007
DeepVAT minus *MMRS*	–	–	–	–	–	–	–	–
DeepVAT minus *SimCLR* minus *t-SNE*	11.27	0.009	10.01	0.007	10.12	0.06	15.12	0.005

we obtain a better low-dimensional embedding, as SimCLR is better equipped to detect highly varying manifolds than t-SNE alone.

To examine the impact of various components in our model, we perform a three-part ablation study on all four datasets. We systematically eliminate one component at a time from the DeepVAT pipeline (Fig. 2) to assess its reliance within the complete pipeline. Our model is summarised as **DeepVAT = SimCLR + t-SNE + MMRS + iVAT**.

1. **DeepVAT *minus* SimCLR**: We flatten each image in the dataset and apply t-SNE on top of them. Then we sample using MMRS and compute the final iVAT image for the samples. This will show that our model not only benefits from the t-SNE block.

2. **DeepVAT *minus* t-SNE**: Images are passed through a trained SimCLR encoder, and we sample the learned high-dimensional embeddings using MMRS. The final iVAT image is computed on the sampled embeddings.

3. **DeepVAT *minus* MMRS**: iVAT image is computed on full set of embeddings. However, as iVAT/VAT family algorithms require computation of dissimilarity matrix, which has a time complexity of order $\mathcal{O}(N^2)$, it will take hours to get the results. Hence, due to such large time complexity and resource constraint, we are not reporting the results of this ablation.

4. **DeepVAT *minus* tSNE *minus* SimCLR**: We apply iVAT directly on the MMRS sub-set of raw flattened images.

The findings of the ablation study (1) (Table 3) suggest that the generation of RDI by DeepVAT is not solely reliant on t-SNE. Although t-SNE applied directly to raw flattened images produces reasonably good results, it is not as accurate as DeepVAT. However, when dealing with complex datasets like CIFAR-10, utilizing t-SNE on raw flattened images fails to provide meaningful information about the cluster structure. Additionally, the role of the SimCLR module in DeepVAT is investigated in the study (2). The results in Table 3 indicate that SimCLR alone does not yield satisfactory outcomes, although it still demonstrates limited interpretability for simple datasets like MNIST and FMNIST. Nevertheless, when iVAT is applied to SimCLR embeddings for complex datasets, it fails to convey meaningful results. This limitation may be attributed to the high dimensionality of the SimCLR embeddings (2048), which hinders the accurate inference of cluster presence by iVAT.

5. Conclusions and Future Work

This article proposes a deep, self-supervised learning based VAT framework, DeepVAT, for cluster structure assessment in image data. The self-supervised learning method SimCLR significantly improved the performance of iVAT both qualitatively and quantitatively. Our experimental results suggest that when t-SNE is used as dimensionality reduction on top of SimCLR embeddings, the iVAT yields a much sharper RDI, thus a more accurate estimate of the number of clusters. This is because SimCLR can capture the intrinsic dimensionality of image datasets which helped t-SNE in generating a good low dimensional representation. Based on our numerical experiments on four image datasets, we have also shown that DeepVAT significantly outperformed other VAT family methods (FensiVAT, KernelVAT and SpecVAT) and two deep clustering methods (DEC and LSD-C) based on clustering partition accuracy (PA) and NMI. We believe that deploying more deep learning based models like deep metric learning and semi-supervised, which have partial access to labels can further improve the iVAT image for complex datasets.

Another future research could explore different contrastive and generative learning architectures for VAT/iVAT, aiming to discover novel approaches and enhance the performance of existing methods.

Acknowledgement

This research is funded by a grant from the Science and Engineering Research Board (SERB), Government of India (Grant No. SRG/2023/001539).

References

[1] Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors. *Computer Vision – ECCV 2022*. Springer Nature Switzerland, 2022. 4

[2] Puneet Bansal. Intel image classification, 2016. 4, 5

[3] J.C. Bezdek and R.J. Hathaway. Vat: a tool for visual assessment of (cluster) tendency. In *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No.02CH37290)*, volume 3, pages 2225–2230 vol.3, 2002. 1, 3

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020. 2, 4

[5] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15745–15753, 2021. 4

[6] William S Cleveland. *Visualizing Data*. Hobart Press, 1993. 1

[7] Brian S Everitt. *Graphical techniques for multivariate data*. North-Holland, 1978. 1

[8] 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, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent - a new approach to self-supervised learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc. 4

[9] Richard J. Hathaway, James C. Bezdek, and Jacalyn M. Huband. Scalable visual assessment of cluster tendency for large data sets. *Pattern Recognition*, 39(7):1315–1324, 2006. 3

[10] Timothy C. Havens and James C. Bezdek. An efficient formulation of the improved visual assessment of cluster tendency (ivat) algorithm. *IEEE Transactions on Knowledge and Data Engineering*, 24(5):813–822, 2012. 2, 3

[11] Timothy C. Havens, James C. Bezdek, and Marimuthu Palaniswami. Scalable single linkage hierarchical clustering for big data. In *2013 IEEE Eighth International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pages 396–401, 2013. 3

[12] Anil K. Jain and Richard C. Dubes. *Algorithms for Clustering Data*. Prentice-Hall, Inc., USA, 1988. 1

[13] M.E. Johnson, L.M. Moore, and D. Ylvisaker. Minimax and maximin distance designs. *Journal of Statistical Planning and Inference*, 26(2):131–148, 1990. 3

[14] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research), 2009. 4, 5

[15] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. http://yann.lecun.com/exdb/mnist/, 2010. 5

[16] L Lovász and MD Plummer. Matching theory, akad. *Kiadó, Budapest*, 1986. 5

[17] Advait Parulekar, Liam Collins, Karthikeyan Shanmugam, Aryan Mokhtari, and Sanjay Shakkottai. Infonce loss provably learns cluster-preserving representations. *arXiv preprint arXiv:2302.07920*, 2023. 4

[18] R. C. Prim. Shortest connection networks and some generalizations. *The Bell System Technical Journal*, 36(6):1389–1401, 1957. 1, 3

[19] Punit Rathore, Dheeraj Kumar, James C. Bezdek, Sutharshan Rajasegarar, and Marimuthu Palaniswami. A rapid hybrid clustering algorithm for large volumes of high dimensional data. *IEEE Transactions on Knowledge and Data Engineering*, 31(4):641–654, 2019. 2, 3, 5

[20] S. Rebuffi, S. Ehrhardt, K. Han, A. Vedaldi, and A. Zisserman. Lsd-c: Linearly separable deep clusters. In *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 1038–1046, Los Alamitos, CA, USA, oct 2021. IEEE Computer Society. 6

[21] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2019. 4

[22] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. 4

[23] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, Pierre-Antoine Manzagol, and Léon Bottou. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(12), 2010. 2

[24] Liang Wang, Xin Geng, James Bezdek, Christopher Leckie, and Ramamohanarao Kotagiri. Specvat: Enhanced visual cluster analysis. In *2008 Eighth IEEE International Conference on Data Mining*, pages 638–647, 2008. 3, 6

[25] Liang Wang, Christopher Leckie, Kotagiri Ramamohanarao, and James Bezdek. Automatically determining the number of clusters in unlabeled data sets. *IEEE Transactions on knowledge and Data Engineering*, 21(3):335–350, 2009. 5

[26] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017. 5

[27] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, page 478–487. JMLR.org, 2016. 6

[28] Yang You, Igor Gitman, and Boris Ginsburg. Large batch training of convolutional networks. 2017. 6

[29] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stephane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 12310–12320. PMLR, 18–24 Jul 2021. 4

[30] Baojie Zhang, Ye Zhu, Sutharshan Rajasegarar, Gang Li, and Gang Liu. Kernel-based ivat with adaptive cluster extraction. 2022. 2, 3, 6