

Deep Variational Clustering Framework for Self-labeling of Large-scale Medical Images

Farzin Soleymani¹, Mohammad Eslami², Tobias Elze², Bernd Bischl³, Mina Rezaei³

¹Department of Electrical and Computer Engineering, Technical University of Munich, Germany

²Massachusetts Eye and Ear Hospital, Harvard Medical School, Boston, MA, USA

³Department of Statistics, LMU Munich, Germany

mina.rezaei@stat.uni-muenchen.de

Abstract—

One of the most promising approaches for unsupervised learning is combining deep representation learning and deep clustering. Recent studies propose to simultaneously learn representation using deep neural networks and perform clustering by defining a clustering loss on top of embedded features. Unsupervised image clustering naturally requires good feature representations to capture the distribution of the data and subsequently differentiate data points from one another. Among existing deep learning models, the generative variational autoencoder explicitly learns data generating distribution in a latent space. We propose a Deep Variational Clustering (DVC) framework for unsupervised representation learning and clustering of large-scale medical images. DVC simultaneously learns the multivariate Gaussian posterior through the probabilistic convolutional encoder, and the likelihood distribution with the probabilistic convolutional decoder; and optimizes cluster labels assignment. Here, the learned multivariate Gaussian posterior captures the latent distribution of a large set of unlabeled images. Then, we perform unsupervised clustering on top of the variational latent space using a clustering loss. In this approach, the probabilistic decoder helps to prevent the distortion of data points in the latent space, and to preserve local structure of data generating distribution. The training process can be considered as a self-training process to refine the latent space and simultaneously optimizing cluster assignments iteratively. We evaluated our proposed framework on three public datasets that represented different medical imaging modalities. Our experimental results show that our proposed framework generalizes better across different datasets. It achieves compelling results on several medical imaging benchmarks. Thus, our approach offers potential advantages over conventional deep unsupervised learning in real-world applications. The source code of the method and of all the experiments are available publicly at: <https://github.com/csfarzin/DVC>

Clustering is a fundamental and challenging task of unsupervised learning that aims to group similar data points together without supervision. Unsupervised cluster algorithms were researched widely in terms of density-based modeling, centroid-based modeling, self-organization maps, and grouping algorithms. In recent years, several approaches have performed image clustering on top of features extracted by a deep neural network (DNN) [1]–[3]. Learning deep representation from data helps to improve cluster analysis compared to traditional centroid-based clustering such as K-means [4]. Deep embedded clustering (DEC) methods train an autoencoder to map a high-dimensional data space into a lower-dimensional feature space and define centroid-based clustering loss such as K-means and K-median [5] on top of the embedded layer [3], [6], [7]. However, DEC is not able to model the generative process of data.

We propose a novel deep generative variational autoencoder framework for simultaneously learning unsupervised representation and perform image clustering. Our proposed DVC is able to model data generative process by multivariate Gaussian model and deep convolutional neural network. In order to perform deep embedded clustering on large-scale medical images, we develop the pipeline with a deep convolutional neural networks which results in a better quality of feature maps, reducing the number of parameters, and preserving locality since weights are shared among all locations in the input.

In summary, the convolutional autoencoder is utilized to learn representations in an unsupervised way where the learned features can preserve essential local structure in data. Our probabilistic encoder utilizes the multivariate Gaussian and captures the latent distribution of a large set of unlabeled images. The clustering loss is applied on top of encoder features and helps to scatter embedded points. However, training with only clustering loss causes the corruption of latent space and results in inaccurate performance. Therefore, we propose a probabilistic decoder to prevent the corruption of data points in the latent space. The proposed decoder network modifies embedded space and helps to separate the clusters accurately.

I. INTRODUCTION

Deep learning algorithms have made outstanding results in many domains such as computer vision, natural language processing, recommendation systems, and medical image analysis. However, the outcome of current methods depends on a huge amount of training labeled data, and in many real-world problems such as medical image analysis and autonomous driving, it is not possible to create such an amount of training data. Learning from unlabeled data can reduce the deployment cost of deep learning algorithms where it requires annotations from experts such as medical professionals and doctors.

II. RELATED WORKS AND BACKGROUND

Deep Clustering In recent years, several approaches perform clustering on top of features extracted by deep neural network (DNN) [3], [8], [9]. Tian et al. [10] proposed a two-stage framework that runs K-means clustering on the feature space extracted by a DNN in the first stage. The proposed framework by Peng et al. [11] includes a sparse autoencoder that learns representations in nonlinear latent space, followed by conventional clustering algorithms to fulfill label assignment.

Deep embedded clustering [6] trains an autoencoder with a reconstruction loss paired with a cluster assignment loss. It then defines a soft cluster assignment distribution by using k-means on top of the learned latent representations. The algorithm was later improved by an additional reconstruction loss [7](to preserve local structure), an adversarial loss [12], and data augmentation [13]. JULE [14] is an end-to-end deep clustering framework that jointly learns Convnet features and clusters within a recurrent framework. Bise et al. [8] proposed a soft-constrained clustering method on top of CNN’s features and applied it for clustering of endoscopy images.

Deep Generative Clustering Deep generative models are the powerful class of machine learning which are able to capture the data distribution of the training data and generate artificial samples. This makes it suitable for bioinformatics use cases with limited labeled or unlabeled samples [15]–[17]. Generative adversarial networks (GANs) and variational autoencoders (VAEs) are the most popular and efficient approaches among generative models. In the context of image clustering, ClusterGAN [18] performs clustering with GAN framework and additional deep clustering network which trained with three players in an adversarial fashion.

Jiang et al. [19] introduce variational deep embedding (VaDE) and use Gaussian Mixture and VAE together for building the inference model. Similarly, DGG [20] and GM-VAE [21] exploit VAE and GMM to minimize the graph distances between embedding data points. The mentioned methods and also other previous studies such as [18]–[20] do not address problems arising due to local preservation. Here, DVC can simultaneously optimize cluster labels assignment and learn features that are suitable for clustering with local structure preservation by combining the clustering loss, KL loss, and autoencoder’s reconstruction loss.

Variational Autoencoder VAEs are a probabilistic twist of autoencoders that approximate data distributions by optimizing evidence lower bound loss (ELBO) [22]:

$$\mathcal{L}_n(\theta, \phi; x, z) = -\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] + \mathcal{D}_{KL}(q_\phi(z|x)||p(z)) \quad (1)$$

where the first term is a reconstruction loss and the second is a regularization term, ϕ and θ are the parameters of encoder and decoder network respectively. The $p(z)$ is a fixed prior distribution on latent distribution with a common choice of normal Gaussian:

$$p(z) = \mathcal{N}(\mu = 0, \sigma^2 = I) \quad (2)$$

The $q_\phi(z|x)$ and $p_\theta(x|z)$ in Eq.(1) are diagonal normal distributions parameterized by neural networks can be computed from:

$$q_\phi(z|x) = \mathcal{N}(z; \mu, \sigma^2 * I) \quad (3)$$

$$p_\theta(x|z) = \mathcal{N}(x; \mu, \sigma^2 * I) \quad \text{or} \quad \text{Ber}(x; p_\theta(z)) \quad (4)$$

However, VAEs performance influenced by the design of network architecture and choosing hyperparameters such as the size of latent variables, input and output dimension, and standard deviation of $p(x|z)$. Considering a pretrained model, sufficiently powerful neural networks, a large enough latent space, VAEs with Gaussian encoders and decoders can approximate the true data distribution. Therefore, after optimization, \mathcal{L}_{ELBO} is often used as a proxy for the likelihood of a data sample.

III. METHOD

Our goal is to cluster N samples $\{x_i\}_{i=1}^N$ from the input space $\mathcal{X} = \mathbb{R}^{d_x}$, $d_x \in \mathbb{N}$ into K clusters, represented by centroids $m_1, \dots, m_K \in \mathbb{R}^{d_x}$. The proposed framework is composed of two networks (see Fig 1). The encoder network q with parameters of ϕ computes $q_\phi(z|x) : x_i \rightarrow z_i$. The encoder maps an input image $x_i \in \mathcal{X}$ to its latent embedding $z_i \in \mathcal{Z}$ in a lower dimensionality space compared to the input space \mathcal{X} . The decoder network p parametrizes by θ , $p_\theta : z_i \rightarrow x'_i$, and reconstructs x_i from its latent embedding z_i . The autoencoder network is first pre-trained, with the network loss $\mathcal{L}_n = \mathcal{L}_r + \mathcal{D}_{KL}(q_\phi(z|x)||p(z))$, where $\mathcal{L}_r = \mathcal{L}_{CE}(x, x')$, to initialize latent variable z_i . After convergence, the clustering loss $\mathcal{L}_c = \mathcal{D}_{KL}(P(q)||Q(z, m))$ is introduced into the objective function, and the network’s training is continued to jointly perform clustering in the embedded space while preserving reconstruction capabilities. This is done by using a convex combination of the two losses. The relative weight of each of the two losses is indicated by $\lambda \in (0, 1)$, which controls the degree of distortion introduced in the embedded space:

$$\mathcal{L} = \lambda \mathcal{L}_c + (1 - \lambda) \mathcal{L}_r. \quad (5)$$

In other word, the training procedure of the proposed DVC is end-to-end in two-phase; In the first phase, we initialize the VAE parameters with a multivariate Gaussian samples. At the second phase, the parameters are optimized by training simultaneously deep variational embedded clustering and deep probabilistic reconstruction network. Therefore, we iterate between: computing an auxiliary target distribution and minimizing KL divergence to the computed auxiliary target distribution.

A. Deep Variational Clustering and Parameter Initialization

After convergence of the first training step of our network, yielding a good embedding representation of each training sample in the first step, our method performs clustering in the

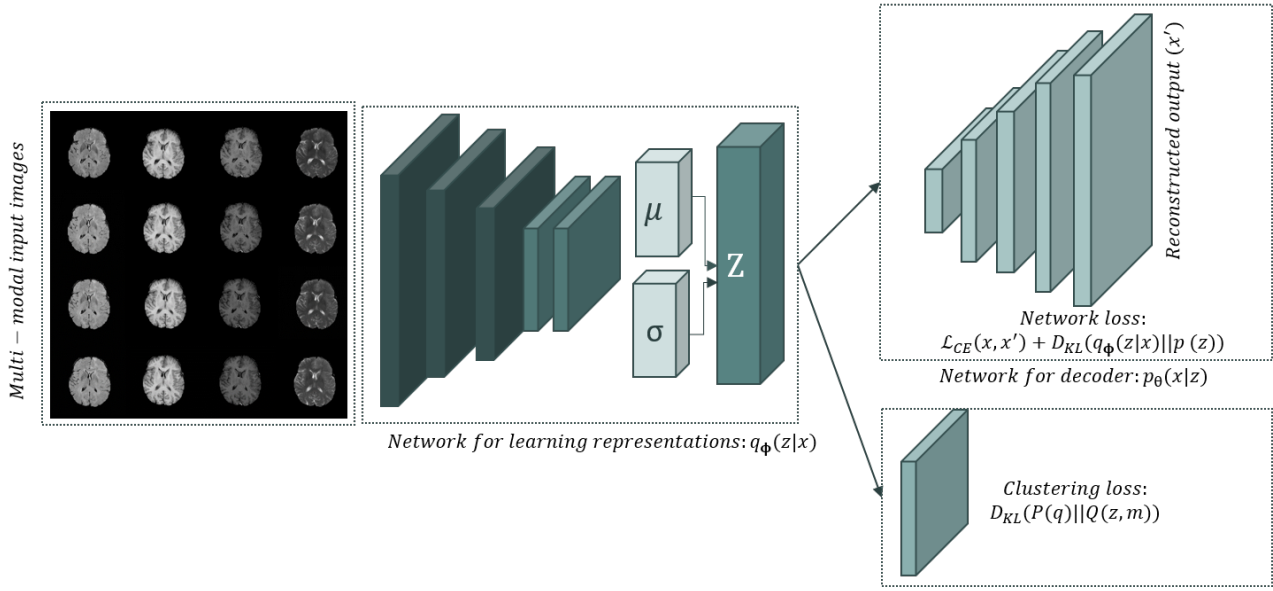


Fig. 1: Illustration of the proposed deep variation clustering network. The large-scale clustering is performed on top of deep features extracted by the convolutional neural network, variational embedded layer, and by minimizing the KL divergence loss between a prior distribution and extracted features by the encoder network.

latent space \mathcal{Z} . We use the Kullback-Leibler (KL) divergence as clustering loss:

$$\mathcal{L}_c = \text{KL}(P||Q) = \sum_i \sum_j p_{ij} \ln \frac{p_{ij}}{q_{ij}} \quad (6)$$

where Q is a soft labeling distribution with elements q_{ij} . P is an auxiliary target distribution derived from Q with elements p_{ij} . More specifically, p_{ij} are the elements of the target distribution while q_{ij} is the distance between the embedded z_i and the center m_j of the j -th cluster. This distance is measured by a Student's t -distribution (cf. [23]):

$$q_{ij} = \frac{(1 + \|z_i - m_j\|^2 / \alpha)^{-\frac{\alpha+1}{2}}}{\sum_k (1 + \|z_i - m_k\|^2 / \alpha)^{-\frac{\alpha+1}{2}}} \quad (7)$$

where α is the degrees of freedom of the Student's t -distribution (we here only consider $\alpha = 1$).

In order to address problems associated with small disjuncts, we modify the target distribution P by pushing data points that are similar in the original space closer together in the latent space. Thereby samples from less-frequent classes can be identified as a cluster. Then, p_{ij} is computed as follows:

$$p_{ij} = \frac{q_{ij}^2 / (u_j + v_j)}{\sum_k q_{ik}^2 / v_k} \quad (8)$$

where $u_j = \sum_i q_{ij}$ are soft cluster frequencies while v_j normalize the frequency of samples per cluster:

$$v_j = - \sum_i \sum_j \sqrt{\frac{\sum_k N_k}{N_j} (1 - q_{ij})^\gamma \log(q_{ij})}. \quad (9)$$

Here, N_j is the estimated cardinality of cluster j , γ is a relaxation parameter in laymen's terms and set to 2. Note that, to prevent instability in the training procedure, we do not update P at every iteration. P is only updated if changes in the label assignments between two consecutive updates of the target distribution are less than a threshold δ . This tolerance threshold and its empirical property are discussed in more detail in Section IV.

For parameter initialization, we follow the standard procedure by [6] and [7]: the autoencoder is pre-trained separately, and the centroids m_1, \dots, m_K are initialized by performing standard K -means clustering on the latent embeddings of the training samples.

Both probabilistic encoder and decoder build by convolutional neural network architecture which brings the following advantages: 1) suitable for large-scale medical images, 2) better quality of feature maps in which results in better representation 3) fewer parameters and hyperparameter needed to be tuned.

Self-labeling and Optimization We perform multi-objective optimization to jointly optimize the cluster loss and the reconstruction loss (ELBO) using mini-batch stochastic gradient variational Bayes (SGVB). In each iteration, the probabilistic encoder network's weights ϕ , cluster centers, probabilistic decoder's weights θ , and target distribution P are updated and optimized on the basis of (5). The target distribution P plays as the ground-truth of the soft label. By iterating these updates, the label assigned of x is obtained using:

$$y_i = \underset{j}{\operatorname{argmax}} q_{ij}, \quad (10)$$

The overall training process is repeated until a convergence

Algorithm 1: Deep Variational Clustering Algorithm

Input : input data $\mathcal{X} = \mathbb{R}^{d_x}$, initial number of clusters K , convergence threshold δ

Output: cluster label y_i of $x_i \in \mathcal{X}$

```
1 initialize  $\theta, \phi$  as described in III;
2 initialize  $m_1, \dots, m_K$  using K-means;
3 for  $itr \in \{0, 1, \dots, Itr_{max}\}$  do
4   if not converged then
5     if  $itr \% update\_interval == 0$  then
6       calculate all embedded points
7          $\{z_i = f_\theta(x_i)\}_{i=1}^n$ ;
8       calculate student's t-distribution
9          $Q, q_{ik} = \frac{(1 + \|z_i - m_k\|^2)^{-1}}{\sum_k (1 + \|z_i - m_k\|^2)^{-1}}$  (Eq. 7);
10      update target distribution
11         $P, p_{ij} = \frac{q_{ij}^2 / (u_j + v_j)}{\sum_k q_{ik}^2 / v_k}$  (Eq. 8);
12      do soft label assignment (Eq. 10) ;
13    else
14      select a mini-batch of samples ;
15      calculate  $z_i$  and  $q_i$  for each mini-batch;
16      calculate  $x'_i = f_\phi(z_i)$  ;
17      compute  $\mathcal{L}_r$  and  $L_c$ ;
18      update  $m_1, \dots, m_K, \theta, \phi$ ;
19    end
20  else
21    Stop training.
22  end
23 end
```

criterion based on the KL loss is met. Algorithm 1 summarizes the training procedure.

IV. EXPERIMENTS

In this section, we conduct experiments to examine our proposed framework. First, we compare the results of our method with several of the related state-of-the-art methods on various clustering task.

Datasets The proposed method is evaluated on MNIST [24], BRATS 2018 [25]–[28] and REFUGE-2 [29] image datasets. *MNIST* consists of 60,000 images for training and 10,000 for testing, each image has a size of 28×28 pixels and is from one of 10 classes. We train on the full training set and report as well as compare the results to other methods on the test set. *BRATS 2018* [25]–[28] consists of two different brain diseases; high and low-grade glioma (HGG/LGG) brain tumour(s). All brains in the dataset have the same orientation. The dataset composed of 1156 magnetic resonance images in four modalities T2, Flair, T1, and T1c from 724 HGG patients and 432 LGG patients.

REFUGE-2 [29] is an active challenge on classification of clinical Glaucoma and part of the MICCAI-2020 conference. The organizers released 1,200 microscopy retinal scans with size of 2124×2056 pixels from two different machines and

scanned by two clinics. The dataset is imbalanced with a ratio of 1:30.

Evaluation Metrics As unsupervised evaluation metric, we use the clustering Accuracy (ACC), Normalized Mutual Information (NMI) and Adjusted Rand Index (ARI) for evaluations. These measures range in $[0, 1]$, higher scores show more accurate clustering results.

Compared Method We compare and show the efficacy of our proposed algorithm by comparing with unsupervised DEC [6], IDEC [7], DEPICT [30], VaDE [19], variational clustering [31], and VDEC [32], which can be viewed as a variant of our method when the reconstruction loss and network architecture are different. Note that the reported results for DEC and IDEC is based on our implementation code in pytorch.

Experimental Setting We evaluate two architectural choices for our proposed framework by modifying different types of auto-encoders. In the first experiment (*DVC-1*), we study the impact of convolutional neural network architecture and combination of *KL*-loss and binary cross-entropy loss for large scale image clustering. Both probabilistic encoder and decoder build by convolutional neural network architecture. We use convolutional layers with kernel size 5×5 and stride 2 in the encoder part. In the decoder part, we perform up-sampling by image re-size layers with a factor of 2 and a convolutional layer with kernel size 3×3 and stride 1. The input size for images from REFUGE and BRATS is 128×128 . Using convolutional encoder and decoder are beneficial for medical images which are usually large-scale and high-dimensional, and also can bring feature maps and representations. Moreover, with convolutional layers, fewer parameters and hyperparameter are required to be considered.

The second experiment *DVC-2*, includes a fully-connected multilayer perceptron (MLP) with dimensions d_x -500-500-1000-10 as encoder for all datasets. Here, d_x is the dimension of the input data. The decoder network is also a fully-connected MLP with dimensions 10-1000-500-500- d_x . Each layer is pre-trained for 100,000 iterations with dropout. The entire deep autoencoder is further fine-tuned for 200,000 iterations without dropout for both layer-wise pre-training and end-to-end tuning. The mini-batch size is set to 256 for MNIST, 64 for BRATS, and 8 for REFUGE. We use a learning rate of 0.01 which is divided by 10 every 20,000 iterations and set weight decay to zero. After pretraining, the coefficient λ of clustering loss is set to 0.1. The convergence threshold δ is set to 0.001 while the update intervals for target distribution are 70, 80, 100 iterations for REFUGE, BRATS, and MNIST respectively.

Image Clustering We include the quantitative clustering results of these clustering methods in Table I. Based on columns 4-9, our proposed DVC-1 outperforms the other methods with significant margins on all three clustering quality measures. Other points that can be also observed

from Table I are: (1) The performance of convolutional-based methods (DVC-1) is much better than those fully connected MLP methods (DVC-2 and VAE [22]). The reason is the quality of the feature map and fewer parameters that need to be tuned. (2) The performance of representation-based clustering algorithms (DAC [33]) is much better than the conventional clustering algorithms (i.e. K-means [34]). It shows that representation learning acts a crucial role in image clustering. The qualitative results are shown in Figure 2 includes the cluster distributions and several generated image samples. Figure 2 shows promising abilities of the proposed method.

V. CONCLUSION

We proposed end-to-end CNN-based VAEs with multivariate Gaussian priors to perform unsupervised image clustering. We performed clustering on top of strong latent representation made with both prior and posterior distribution. In comparison with the existing approaches, the proposed method achieves superior performance on two real patient medical imaging datasets and competitive results on the MNIST and CIFAR-10 datasets. It shows that our method can deal with large-scale medical images in different image modalities and is not limited to simple image datasets. We presented the application of our proposed method for the task of unsupervised clustering. However, the synthesized samples look realistic with high-resolution therefore they can tackle some DL challenges such as class imbalance and data augmentation.

VI. ACKNOWLEDGMENT

This work has been funded in part by the German Federal Ministry of Education and Research (BMBF) under Grant No. 01IS18036A, Munich Center for Machine Learning (MCML). F.S, M.R, and B.B were supported by the German Federal Ministry of Education and Research (BMBF) under Grant No. 01IS18036A. The authors of this work take full responsibilities for its content.

REFERENCES

- [1] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 132–149.
- [2] A. Banerjee, S. Merugu, I. S. Dhillon, and J. Ghosh, "Clustering with bregman divergences," *Journal of machine learning research*, vol. 6, no. Oct, pp. 1705–1749, 2005.
- [3] M. Rezaei, E. Dorigatti, D. Ruegamer, and B. Bischl, "Learning statistical representation with joint deep embedded clustering," *arXiv preprint arXiv:2109.05232*, 2021.
- [4] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," Stanford, Tech. Rep., 2006.
- [5] S. Dasgupta, N. Frost, M. Moshkovitz, and C. Rashtchian, "Explainable k-means and k-medians clustering," *ICML*, vol. 86, 2020.
- [6] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *International conference on machine learning*, 2016, pp. 478–487.
- [7] X. Guo, L. Gao, X. Liu, and J. Yin, "Improved deep embedded clustering with local structure preservation," in *IJCAI*, 2017, pp. 1753–1759.
- [8] R. Bise, K. Abe, H. Hayashi, K. Tanaka, and S. Uchida, "Efficient soft-constrained clustering for group-based labeling," in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, D. Shen, T. Liu, T. M. Peters, L. H. Staib, C. Essert, S. Zhou, P.-T. Yap, and A. Khan, Eds. Springer International Publishing, 2019, pp. 421–430.
- [9] K. Cao, C. Wei, A. Gaidon, N. Arechiga, and T. Ma, "Learning imbalanced datasets with label-distribution-aware margin loss," in *Advances in Neural Information Processing Systems*, 2019, pp. 1567–1578.
- [10] F. Tian, B. Gao, Q. Cui, E. Chen, and T.-Y. Liu, "Learning deep representations for graph clustering," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 28, no. 1, 2014.
- [11] X. Peng, S. Xiao, J. Feng, W.-Y. Yau, and Z. Yi, "Deep subspace clustering with sparsity prior," in *IJCAI*, 2016, pp. 1925–1931.
- [12] N. Mrabah, M. Bouguessa, and R. Ksantini, "Adversarial deep embedded clustering: on a better trade-off between feature randomness and feature drift," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, 2020.
- [13] X. Guo, E. Zhu, X. Liu, and J. Yin, "Deep embedded clustering with data augmentation," in *Asian conference on machine learning*, 2018, pp. 550–565.
- [14] J. Yang, D. Parikh, and D. Batra, "Joint unsupervised learning of deep representations and image clusters," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5147–5156.
- [15] M. Rezaei, T. Uemura, J. Näppi, H. Yoshida, C. Lippert, and C. Meinel, "Generative synthetic adversarial network for internal bias correction and handling class imbalance problem in medical image diagnosis," in *Medical Imaging 2020: Computer-Aided Diagnosis*, vol. 11314. International Society for Optics and Photonics, 2020, p. 113140E.
- [16] M. Rezaei, J. J. Näppi, C. Lippert, C. Meinel, and H. Yoshida, "Generative multi-adversarial network for striking the right balance in abdominal image segmentation," *International Journal of Computer Assisted Radiology and Surgery*, vol. 15, no. 11, pp. 1847–1858, 2020.
- [17] M. Rezaei, "Generative adversarial network for cardiovascular imaging," *Machine Learning in Cardiovascular Medicine*, p. 95, 2020.
- [18] S. Mukherjee, H. Asnani, E. Lin, and S. Kannan, "Clustergan: Latent space clustering in generative adversarial networks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 4610–4617.
- [19] Z. Jiang, Y. Zheng, H. Tan, B. Tang, and H. Zhou, "Variational deep embedding: An unsupervised and generative approach to clustering," *arXiv preprint arXiv:1611.05148*, vol. 86, 2016.
- [20] L. Yang, N.-M. Cheung, J. Li, and J. Fang, "Deep clustering by gaussian mixture variational autoencoders with graph embedding," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 6440–6449.
- [21] N. Dilokthanakul, P. A. Mediano, M. Garnelo, M. C. Lee, H. Salimbeni, K. Arulkumaran, and M. Shanahan, "Deep unsupervised clustering with gaussian mixture variational autoencoders," *arXiv preprint arXiv:1611.02648*, vol. 33, 2016.
- [22] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," vol. 86, 2014.
- [23] L. Van Der Maaten, "Learning a parametric embedding by preserving local structure," in *Artificial Intelligence and Statistics*, 2009, pp. 384–391.
- [24] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [25] B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest *et al.*, "The multimodal brain tumor image segmentation benchmark (brats)," *IEEE transactions on medical imaging*, vol. 34, no. 10, pp. 1993–2024, 2015.
- [26] S. Bakas, H. Akbari, A. Sotiras, M. Bilello, M. Rozycki, J. Kirby, J. Freymann, K. Farahani, and C. Davatzikos, "Advancing The Cancer Genome Atlas glioma MRI collections with expert segmentation labels and radiomic features," *Nature Scientific Data*, DOI:10.1038/sdata.2017.117, vol. 86, 2017.
- [27] S. Bakas and H. Akbari, "Segmentation labels and radiomic features for the pre-operative scans of the tcga-gbm collection," *The Cancer Imaging Archive*, DOI:10.7937/K9/TCIA.2017.KLXWJ1JQ, vol. 286, 2017.
- [28] S. Bakas, H. Akbari, A. Sotiras, M. Bilello, M. Rozycki, J. Kirby, J. Freymann, K. Farahani, and C. Davatzikos, "Segmentation labels and radiomic features for the pre-operative scans of the tcga-lgg collection," *The Cancer Imaging Archive*, DOI:10.7937/K9/TCIA.2017.GJQ7R0EF, vol. 286, 2017.
- [29] J. I. Orlando, H. Fu, J. B. Breda, K. van Keer, D. R. Bathula, A. Diaz-Pinto, R. Fang, P.-A. Heng, J. Kim, J. Lee *et al.*, "Refuge challenge: A unified framework for evaluating automated methods for glaucoma

TABLE I: Comparison results of our achieved accuracy on MNIST, BRATS and REFUGE datasets. Results of (*) methods are taken from reference [33].

Model	MNIST [24]			REFUGE [35]			BRATS [25]–[28]		
	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC
DVC-1	0.798	0.777	0.888	0.712	0.697	0.751	0.943	0.951	0.963
DVC-2	0.767	0.786	0.876	0.653	0.583	0.716	0.889	0.892	0.901
*K-means [34]	0.499	0.365	0.572	0.083	0.028	0.129	0.556	0.583	0.612
AE [36]	0.725	0.613	0.812	0.100	0.047	0.164	0.582	0.607	0.679
VAE [22]	0.736	0.712	0.831	0.407	0.440	0.521	0.604	0.631	0.682
*JULE [14]	0.913	0.927	0.964	0.602	0.632	0.662	-	-	-
IDEC [7]	0.816	0.868	0.880	0.627	0.642	0.681	0.837	0.844	0.892
DEC [6]	0.771	0.741	0.843	0.513	0.549	0.585	0.805	0.842	0.853
VDEC [32]	0.836	0.748	0.842	0.612	0.645	0.679	0.833	0.874	0.901
*DAC [33]	0.935	0.948	0.975	0.703	0.717	0.734	-	-	-

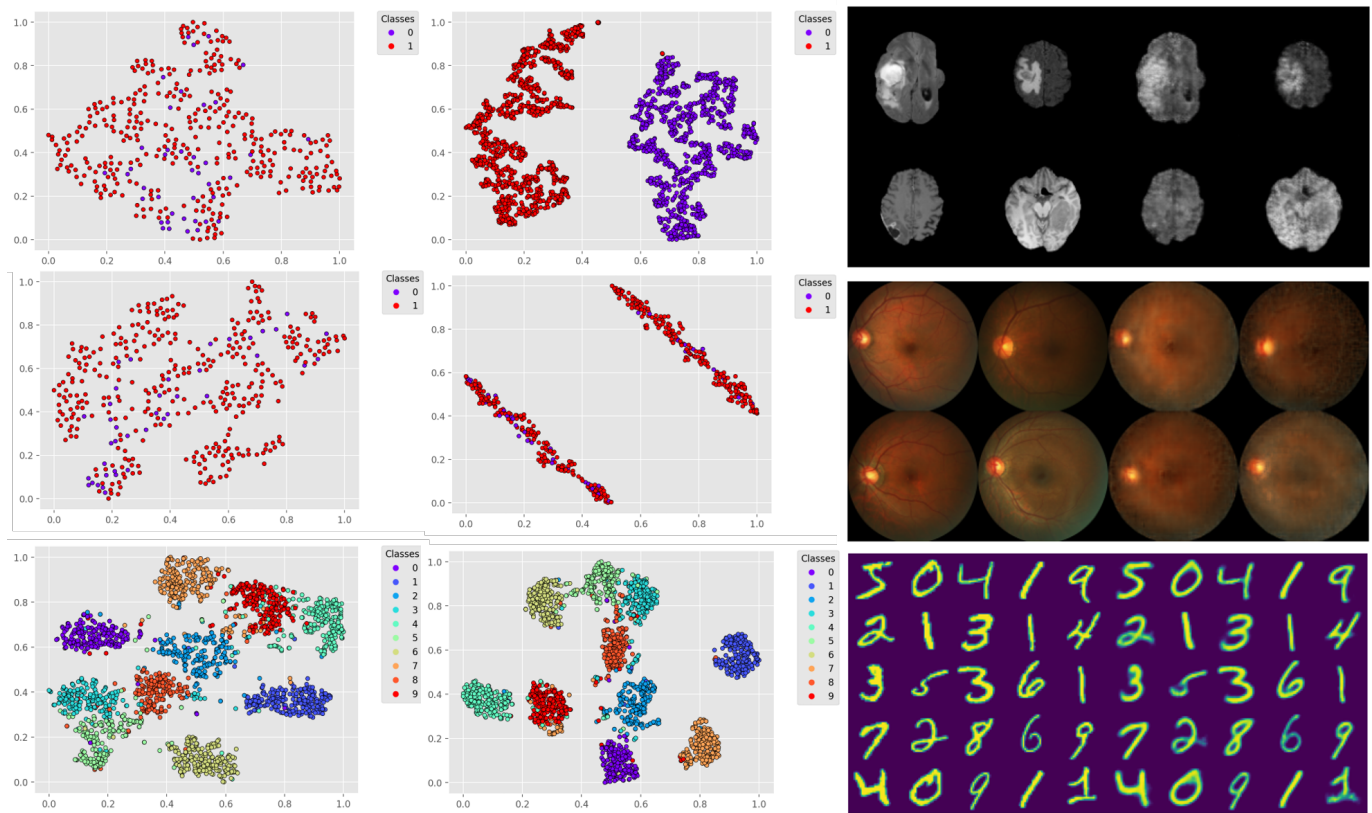


Fig. 2: Illustration of the qualitative results achieved by our proposed method. Left) the clusters in the first epoch, Middle) computed clusters in the final epoch, Right) Several samples generated by our trained model. Top to bottom is BRATS, REFUGE, and MNIST dataset. The results show the promising functionality of the proposed method.

assessment from fundus photographs,” *Medical image analysis*, vol. 59, p. 101570, 2020.

- [30] K. Ghasedi Dizaji, A. Herandi, C. Deng, W. Cai, and H. Huang, “Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5736–5745.
- [31] V. Prasad, D. Das, and B. Bhowmick, “Variational clustering: Leveraging variational autoencoders for image clustering,” in *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020, pp. 1–10.
- [32] P. Ghosh, M. S. Sajjadi, A. Vergari, M. Black, and B. Schölkopf, “From variational to deterministic autoencoders,” vol. 86, 2020.
- [33] J. Chang, L. Wang, G. Meng, S. Xiang, and C. Pan, “Deep adaptive image clustering,” in *Proceedings of the IEEE international conference*

on computer vision, 2017, pp. 5879–5887.

- [34] J. Wang, J. Wang, J. Song, X.-S. Xu, H. T. Shen, and S. Li, “Optimized cartesian k-means,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 1, pp. 180–192, 2014.
- [35] A. Krizhevsky and G. Hinton, “Learning multiple layers of features from tiny images,” *University of Toronto*, vol. 86, p. 60, 05 2009.
- [36] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, “Greedy layer-wise training of deep networks,” *Advances in neural information processing systems*, vol. 19, pp. 153–160, 2006.