Grad_2VAE: An explainable variational autoencoder model based on online attentions preserving curvatures of representations*

Mohanad Abukmeil, Stefano Ferrari, Angelo Genovese, Vincenzo Piuri, and Fabio Scotti
Department of Computer Science, Università degli Studi di Milano, Italy
{*firstname.lastname*}@unimi.it

Abstract. Unsupervised learning (UL) is a class of machine learning (ML) that learns data, reduces dimensionality, and visualizes decisions without labels. Among UL models, a variational autoencoder (VAE) is considered a UL model that is regulated by variational inference to approximate the posterior distribution of large datasets. In this paper, we propose a novel explainable artificial intelligence (XAI) method to visually explain the VAE behavior based on the second-order derivative of the latent space concerning the encoding layers, which reflects the amount of acceleration required from encoding to decoding space. Our model is termed as Grad_2VAE and it is able to capture the local curvatures of the representations to build online attention that visually explains the model's behavior. Besides the VAE explanation, we employ our method for anomaly detection, where our model outperforms the recent UL deep models when generalizing it for large-scale anomaly data.

Keywords: Unsupervised Learning · VAE · XAI · Anomaly Detection.

1 Introduction

Explainable artificial intelligence (XAI) is an emerging field in artificial intelligence (AI) and machine learning (ML), and deals with explaining the decisions and behaviors of learned models. XAI models are also associated with unsupervised learning (UL) to learn and visualize the hidden structure of data with limited levels of prior assumptions. Autoencoder models (AEs) are a class of generative UL (UGL) methods, which can reduce dimensionality, visualize and generate data, and perform other ML tasks such as object recognition [1,2]. Many different types of AEs have been introduced recently and they are characterized by a regularization term; such term enforces the AEs to learn with an additional penalty to capture different representations for a better generalization [7,3].

Deep AEs encompass many different encoding and decoding stages, where at each stage diverse layers with associated parameters ($\theta = \{W, B\}$, W and B are weights and biases, respectively) are employed to perform a specific mapping (convolution, deconvolution, dense multiplication, etc.), by utilizing several

* We thank the NVIDIA Corporation for the GPU donated.

sets of representations to capture the neurons' responses [6]. Moreover, for each setting among parameters (after each learning iteration or epoch), the gradient is approximated between the input and output by using the first-order partial derivative to optimally fit the model to the data [4]. AEs comprise classic, denoising [27], contractive [23], sparse [20], variational-AE (VAE) [14], and they can also be integrated with other UL models, for example, when combining the generative adversarial networks (GANs) with VAE [17].

Among all AEs, the VAE is regularized by variational inference (VI) [31] to optimize the posterior distribution for large datasets, and it outperforms the others in terms of large-scale generalization (when testing data are larger than the training set). The VAE is utilized in many different fields including object detection, image reconstruction and recognition, compression sensing, and other deep learning tasks [14]. However, explaining VAEs did not receive an appropriate interest in the literature, where explaining such a UGL model is considered essential to understanding the behaviors of neurons when new data (normal or anomaly) is generated or reconstructed [9,24,25]. Thus, different works have been proposed for explaining models through supplementary inputs to carry out specific tasks. However, such works did not explain the behaviors of the models themselves [26,18,32].

The first explainable VAE has been introduced in [16], where it generates offline attention maps (after model learning) by reduplicating the last layer of the encoder, then scaling it up by the global average pooling of the gradient of the latent space concerning that layer. Such attention is seen similar to explaining discriminating models [30]; however, the proposed attention is scaled up by the gradient. The drawback of such an attention map lies in unfair scaling, i.e., related and unrelated features in the channels of the filters are scaled with the same factor.

To help to explain VAEs, we propose Grad_2VAE, a novel XAI model utilizing online mapping, i.e., after each epoch, visual attention can be produced. Moreover, the Grad_2VAE utilizes the second-order (2^{nd}) derivative between the latent and 1^{st} encoder's layers to obtain the 1^{st} derivative of the gradient, which captures the curvatures of neurons responses that are aggregated to show how the VAE learns data without additional scaling. Therefore, our contribution is twofold: *(i)* introducing a novel method to explain the VAEs employing the gradient derivation, and *(ii)* expanding our method to accelerate VAE learning (reduced epochs) and one-class anomaly detection. The rest of this paper is organized as follows. Section 2 highlights the VAE and the 2^{nd} derivative interpretation. Section 3 describes the Grad_2VAE. The experimental results are given in Section 4. The conclusion and future works are reported in Section 5.

2 VAE and 2^{nd} derivative interpretation

2.1 VAE model

Similarly to any AE model, the VAE contains two main modules: *(i)* the encoding module (inference side) that is employed to map data $X = \{x_i \,|\, x_i \in \mathbb{R}^D, i =$

$1, \ldots, N\}$, D is the original dimensionality, to a latent space $Z = f(X) = \{z_i = f(x_i) \in \mathbb{R}^d, | \, i = 1, \ldots, M\}$; such a module reduces dimensionality where $0 < d < D$, and it is used to infer the model likelihood $P(X|\theta)$; *(ii)* the decoding module (generation side) that is utilized to generate or reconstruct the original data $\tilde{X}$ from the latent space Z [12,11]. For a given data $X \in \mathbb{R}^D$, the encoding module creates a mapping $f : \mathbb{R}^D \to \mathbb{R}^d$, while the decoding module creates a mapping $g : \mathbb{R}^d \to \mathbb{R}^D$, which generates an approximation of the original data: $\tilde{X} = g(Z; \hat{\theta}_d)$ [2]. The AEs are regulated to find the parameters $(\hat{\theta}_e, \hat{\theta}_d)$ that achieve a better generalization [7], and to obtain the minimum loss $\mathbf{L}_{\text{REC}}$:

$$\mathbf{L}_{\text{REC}_{\{\hat{\theta}_e, \, \hat{\theta}_d\}}} = \min \|X - (f \circ g)X\|^2_{\text{Er}} \,, \tag{1}$$

where the reconstruction error E_r can be measured by different metrics including mean square error (MSE), Frobenius norm, reconstruction cross-entropy, or β-divergence [1,3].

Among all AE models, VAE is regulated by the VI, and it is optimized based on two different losses that are minimized simultaneously [14]. VI method is one of the Bayesian techniques, which can be utilized to estimate an intractable posterior over a big dataset using a simpler variational distribution to obtain the solution to an optimization problem [31], i.e., the VI approximates probability densities through optimization. By considering the encoder module output, the approximate posterior distribution $Q(Z|X)$ is estimated, which parameterizes the shape of the latent distribution according to the original input data X. Moreover, optimizing $Q(Z|X)$ characterizes the VAE, where it enforces the latent space distribution to follow a unit Gaussian distribution with a certain mean μ (which reflects the center of the Gaussian), and a standard deviation σ (which reflects the Gaussian shape).

Initially, the prior distribution of latent space $P(Z)$ is drawn (simply by copying the unit Gaussian distribution of the data manifold $P(X)$). Thereafter, the approximated distribution $Q(Z|X)$ and the prior $P(Z)$ are compared using the KL divergence [22]. The KL divergence is defined as $\text{KL}(P\|Q) = \Sigma_x P(x) \log \frac{P(x)}{Q(x)}$, which is always positive and tends to zero if and only if P and Q are almost equal. Moreover, appending noise to $Q(Z|X)$ throughout varying σ by a small value ϵ, and then enforcing the AE to reconstruct the data following the true (not varied one) Gaussian $P(Z)$ is called the reparameterization trick; such a trick generates several different distributions (similarly to duplicate the training data with fusion) that are optimized and compared with prior distribution by the KL divergence, thus the model can be better generalized for a large-scale testing stage [14]. Finally, the VAE is optimized to minimize the $\mathbf{L}_{\text{REC}}$ according to Eqn. (1), and it is also optimized to minimize the latent loss between $Q(Z|X)$ and $P(Z)$ using $\text{KL}(P\|Q)$, which measures to which extent the reparameterized latent distribution can follow a unit Gaussian:

$$\mathbf{L}_{\text{VAE}_{\{\hat{\theta}_e, \, \hat{\theta}_d, \, \hat{\mu}_X, \, \hat{\sigma}_X, \, \hat{\mu}_Z, \, \hat{\sigma}_Z\}}} = \min[\mathbf{L}_{\text{REC}} + \text{KL}(P\|Q)] \,. \tag{2}$$

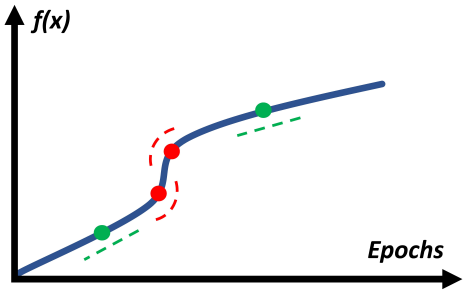


Fig. 1: The neuron activation and gradient over epochs.

2.2 The second-order (2^{nd}) derivative interpretation

The first-order (1^{st}) partial derivative between input and output neurons reflects the 1^{st} gradient, which measures the instantaneous rate of change (velocity or speed) ∂ [21] among model parameters θ that are employed to optimally fit ML model [13]. Moreover, if the gradient sign is negative, then it is decreasing (velocity is reduced), while if the gradient sign is positive then it is increasing (velocity is accelerated).

For a VAE with an encoding layer L_{e1} and a latent layer Z, the 1^{st} gradient of Z with respect to L_{e1} is computed by carrying out the partial derivative of each neuron z_i as $\frac{\partial z_i}{\partial L_{e1}}$; considering that if an additional layer L_{e2} acts between L_{e1} and Z, then the chain rule is introduced as $\frac{\partial z_i}{\partial L_{e1}} = \frac{\partial z_i}{\partial L_{e2}} \frac{\partial L_{e2}}{\partial L_{e1}}$ [5]. The final result of derivations gives all possible rates of changes, which are required to update θ laying between L_{e1} and Z. Because the rate of change (∂) of a neuron's response (activation) is changing during a period of time (over several epochs), thus capturing the variation in the rate of change is essential to obtain the acceleration required for a neuron from the velocity [29], and it is achieved by considering the derivative of the gradient, i.e., 2^{nd} derivative $\frac{\partial^2 z_i}{\partial L_{e1}^2}$ [8].

Graphically, a neuron response that is modeled by a non-linear ReLU function [19] is given according to Fig. 1: the 1^{st} gradient is the slope at a point in the graph (blue curve), whereas the 1^{st} derivative of gradient explains how the slope is changing over time (the red and green points). As it is noticed from Fig. 1, the gradient of a neuron response can be steady at a period of the learning time, i.e., the 2^{nd} derivative around the green points is ≈ 0; however, it can change at a different period, i.e., the 2^{nd} derivative around the red points is $>$ or < 0.

Accordingly, utilizing the 2^{nd} derivative that measures how the 1^{st} gradient of neurons responses is changing (as in deriving the acceleration from speed), the curvatures of representations and the temporal behavior of neurons when learning data can be captured. Moreover, such a strategy can be represented by a learnable attention map, which aggregates all 2^{nd} partial derivative to explain how the latent neurons of Z are activated to the local curves and edges.

Our Grad$_2$VAE employs the 2^{nd} gradient to visually explain the learned representations of the VAE in an online fashion by reconstructing attention maps, and it exploits such explanations in the application of one-class anomaly

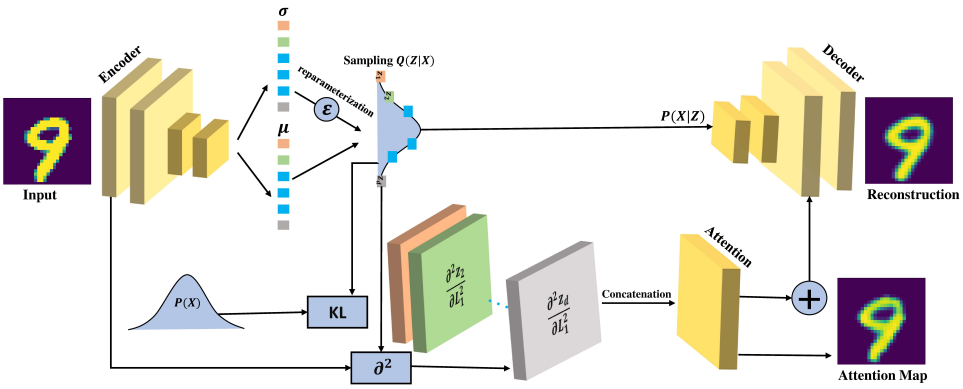


Fig. 2: The Grad$_2$VAE block diagram.

detection [24]. Moreover, we will show how such a strategy is able to accelerate the convergence among the learning parameters θ.

3 Grad$_2$VAE

Fig. 2 shows our proposed Grad$_2$VAE, where it comprises an encoder, a decoder, and an attention module. Both encoder and decoder contain one stage of down-sampling (convolution with a stride of 2) and up-sampling (de-convolution with a stride of 2), respectively. Moreover, the size of the first two layers of the encoder and the last two layers of the decoder are fixed to uniform the dimensionality. Thus, the obtained attention of the 2$^{\text{nd}}$ derivative can be fused with the $L_{d_{n-1}}$ layer (d_n is the total number of the decoder's layers); such a fusion is seen as a form of residual learning [10], which enforces the Grad$_2$VAE to learn the residual of mapping between the encoder and decoder by utilizing the gradient attention. Accordingly, besides the explainability of the Grad$_2$VAE, it also boosts the reconstruction of data by utilizing the curvatures of representations that are combined with the decoder. Therefore, the Grad$_2$VAE optimizes two losses by using Adam [13] as:

$$\mathbf{L}_{\text{Grad}_2\text{VAE}} = \min[\mathbf{L}_{\text{VAE}} + \|X - \theta_{grad}(Z, L_{e1})\|^2_{\text{Er}}] \,, \tag{3}$$

where the first loss is taken from the vanilla VAE [14] that is depicted at Eqn. (2), and the second loss is the reconstruction loss between the data and the aggregated attention that is obtained from the attention module (see Fig. 2). Moreover, θ_{grad} represents the 2$^{\text{nd}}$ derivative between each latent neuron z_i with respect to L_{e1}, i.e., for each z_i there is a corresponding tensor of size of the L_{e1} to allocate all partial derivatives. Additionally, the derivative of gradient of Z can be implemented with respect to all other encoder's layers (as in considering L_{e3}); however, considering more depth layers requires re-scaling the dimensionality which needs more computational time and leads to the loss of global representations.

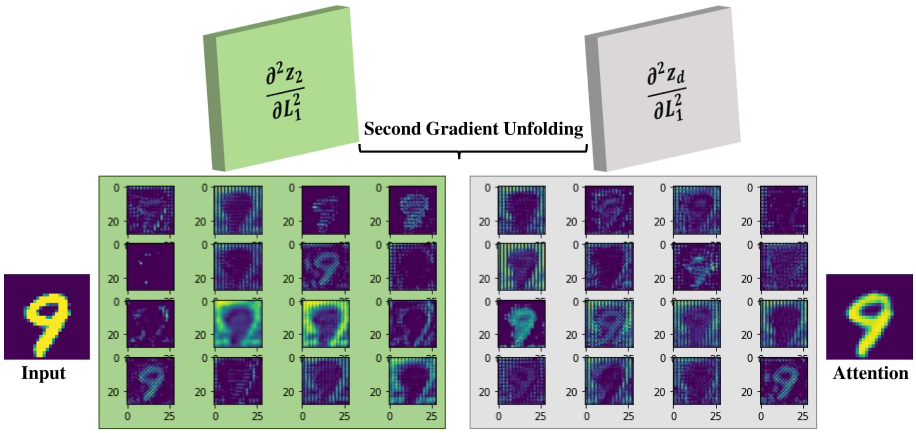
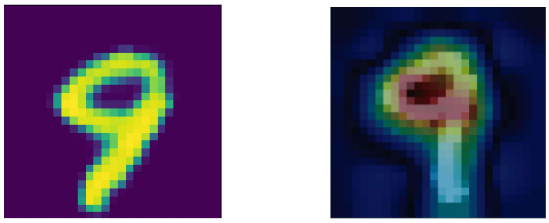


Fig. 3: The unfolding of the tensor that holds all second-order partial derivatives (Grade_2) of z_2 (green tensor) and z_{16} (gray tensor) concerning L_{e1}, respectively.



(a) Learned attention of Grad_2VAE.

(b) Offline attention [16].

Fig. 4: Grad_2VAE learnable attention vs. the offline attention proposed in [16] that depends on scaling of the penultimate encoding layer by the gradient of Z respecting that layer when learning the 10^{th} class from MNIST digits.

4 Experimental results

To show the performance of our Grad_2VAE, we employ both MNIST and fashion MNIST datasets [15,28]. Each comprises 60k images for training and 10k for testing, divided in 10 classes with image size of 28×28. Moreover, all quantitative analysis experiments are implemented with a batch size of 128, 100 epochs with a starting learning rate (η) of 0.001, where the η is reduced after 50 epochs by a factor of 10^{-2} to search and fine-tune the parameters θ.

4.1 Grad2VAE Explainability

The Grad_2VAE explainability lies in the attention module, which aggregates the derivatives of gradients and it reflects the curvatures among representations that are obtained at the neurons response level. For each z_i, the corresponding tensor of 2^{nd} partial derivative is produced and it represents the neuron attentions, where the number of tensors is a function of the latent space dimensions.

Thereafter, all tensors are aggregated by different matrix methods including the addition, mean, convolution, etc [16]. Fig. 3 shows the 2^{nd} partial derivative attentions of the second and last neurons of Z as a function of the depth of the convolutional filters, where the Grad$_2$VAE has been trained to show the explainability for the 9^{th} class of the MNIST, by considering 16 neurons for Z and a depth of 16 filters. Moreover, Fig. 4 shows a comparison between the aggregated attention of the Grad$_2$VAE (based on convolutional aggregation), and the attention proposed in [16] (based on the mean aggregation). As it can be noticed from the figure, considering the 2^{nd} partial derivative (a derivative of gradient) offers a better explainable visual attention that retains all possible curvatures of the representations.

4.2 Grad2VAE in one-class anomaly detection

Anomaly detection (AD) is a branch of ML that characterizes data samples that are misrepresented from what is normal or predicted [25]. One-class AD is referred to as a learning approach in which only normal data is considered at the training stage, where the ML model learns to classify or reconstruct the normal data only [24]. However, at the testing stage, all data samples that are falling out of the normal class distribution of the trained data are considered, and the ML model must be able to distinguish between normal and anomalies samples. The decoder of the Grad$_2$VAE is guided by the curvature of representations from the encoder, thus the reconstruction process is accelerated and optimized. Accordingly, the Grad$_2$VAE is directly applied to the AD due to its reconstruction ability. In the following, we employ the Grad$_2$VAE in the one-class AD, where we benchmark our model on the MNIST and fashion datasets. Moreover, the average area under the receiver operator characteristic curve (AUC-ROC) is considered as a metric to show the performance, where we report the qualitative and quantitative comparisons to the recent works.

Qualitative analysis comparison In this section, we visually compare our work with [16] based on the MNIST dataset. For this analysis, we consider 600 epochs for qualitative comparison. Moreover, we have followed [16] to train our model considering one class from the training set as a normal class, thereafter testing with all classes from the testing set. Hence, our model must be able to produce visual attentions for all testing classes avoiding bias to the previously learned normal class. Fig. 5 shows the attention maps comparison between our model and [16] when both are trained with the 2^{nd} and 4^{th} classes from the dataset separately, then considering data from other classes (out of the trained data distribution) as testing samples.

As it can be noticed from Fig. 5, our model produces visual attentions maps that completely retain the curvatures among representations of the input data regardless that they are normal (seen) or anomalies (unseen from other classes). Specifically, our model is able to visually explain the differences between normal and anomalies samples, avoiding further preprocessing such as scaling attentions maps by the gradient as in [16] to partially detect and explain the anomalies.

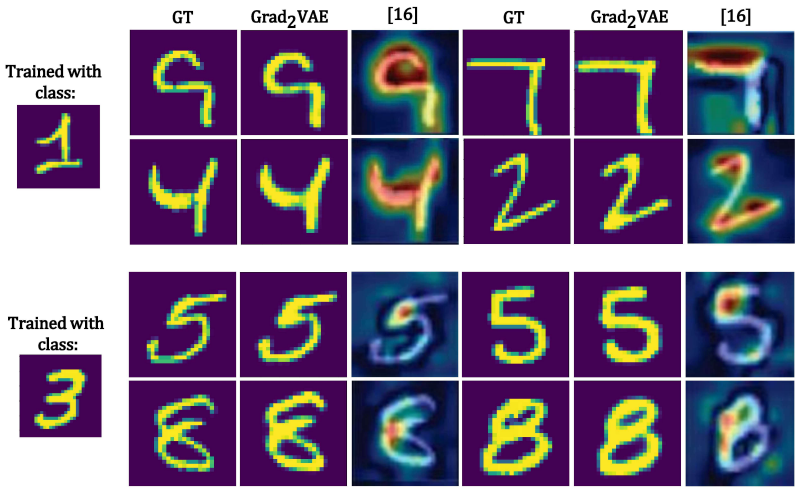


Fig. 5: The qualitative analysis comparison between our Grad_2VAE model that produces learnable attention maps (online attention) and the offline attention introduced in [16], where GT represents the ground truth samples. Moreover, the images that appear in the first two rows of the figure have been tested when only the second class of the MNIST digits has been trained, also in the bottom two rows of the figure, the model has been trained considering only the fourth class and tested with all other images from all classes.

Quantitative analysis comparison In this section, we compare our model with the recent deep UL models, where we consider the CAE OCSVM, Deep SVDD, and Inception-CAENN [24,25]. Moreover, we followed [25] to train 10 models for each normal class, thus reporting the average AUC-ROC over 10 Grad_2VAEs. Additionally, we used 16 and 32 neurons as the bottleneck size for the MNIST and fashion, respectively. Our results are reported in Table 1 considering only 100 epochs, due to the acceleration in learning ability that the Grad_2VAE possesses.

As it can be noticed from Table 1, the Grad_2VAE outperforms the other models under reduced epochs, where all other models have been trained considering 150 epochs [25]. Moreover, the Grad_2VAE does not show any bias to a class against all other classes, e.g., the Deep SVDD and Inception-CAENN learn the 2^{nd} class from the fashion dataset perfectly; however, they show minimum accuracies for the 6^{th} and 7^{th} classes, respectively. Finally, the Grad_2VAE shows a better mean standard deviation (mstd) among the averaged results (of 10 models), where our maximum mstd for both datasets did not exceed 0.117, whereas it reached 3.8, 3.9 for the Deep SVDD, and Inception-CAENN, respectively [25].

Furthermore, to show the learning complexity and performance convergence (acceleration) of our proposed Grad_2VAE model, we compare the accuracy convergence with the baseline convolution AE (Baseline CAE), vanilla VAE, and Inception CAENN models [3,25]. Moreover, under the same experimental setup reported in Section 4, the Baseline CAE learns 0.266 M, vanilla VAE learns

Table 1: The AUC-ROC metric comparison with recent deep learning models, where the first column shows the utilized datasets, the second column gives the data classes related to each dataset, and all other columns contain the results of the employed models to evaluate the performance. Moreover, each row in the table contains the results when training the model with the class of data that is labeled by the value of the normal class column and lies in that row, subsequently testing the model with the testing data from all classes.

Dataset	**Normal-class**	**CAE OCSVM**	**Deep SVDD**	**Inception CAENN**	**Grad$_2$VAE**
MNIST	0	95.40	99.10	98.70	97.62
	1	97.40	99.70	99.70	96.81
	2	77.60	95.40	96.70	98.84
	3	88.60	95.10	95.20	98.53
	4	83.60	95.90	95.00	97.98
	5	71.30	92.10	95.20	98.70
	6	90.10	98.50	98.30	98.54
	7	87.20	96.20	97.00	98.59
	8	86.50	95.70	96.20	98.09
	9	87.30	97.70	97.00	98.62
	Average	86.50	96.60	96.90	**98.23**
FASHION	T-shirt	88.00	98.80	92.40	96.54
	Trouser	97.30	99.77	98.80	95.88
	Pullover	85.50	93.50	90.00	96.59
	Dress	90.00	94.90	95.00	96.29
	Coat	88.50	95.10	92.00	96.49
	Sandal	87.20	90.40	93.40	96.39
	Shirt	78.80	98.00	85.50	96.65
	Sneaker	97.70	96.00	98.60	96.05
	Bag	85.80	95.40	95.10	96.58
	Boot	98.00	97.60	97.70	96.42
	Average	89.70	95.90	93.90	**96.37**

3.25 M, and Inception CAENN learns 0.335 M parameters to complete one training epoch. Additionally, our proposed Grad$_2$VAE learns 3.38 M parameters as a consequence of the online attention module and the 2$^{\text{nd}}$ order partial derivative parameters.

Fig. 6 depicts the learning acceleration ability over the learning epochs of our proposed Grad$_2$VAE model considering the related models in the literature. As it is noticed from the figure, our proposed Grad$_2$VAE shows the highest learning convergence ability at an early stage of the learning period, i.e., our proposed model is able to search and find the suitable learning parameters, ($\theta = \{W, B\}$), at an initial interval of the learning epochs, which accelerate the model learning and prevent the model from overfitting. That is by considering the XAI attention maps, which are fused with the penultimate layer to compensate for the loss

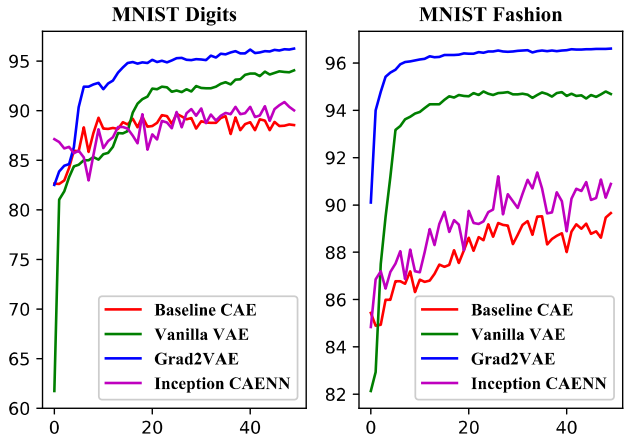


Fig. 6: AUC metric over the first 50 epochs for both MNIST datasets.

caused by encoding and decoding operations. Finally, our model learns a number of parameters that are equal to 12.7 x, 1.03 x, 10.10 x of the Baseline CAE, vanilla VAE, and Inception CAENN models, respectively. However, it converges to an accuracy of greater than 95% for both datasets considering the half number of the learning epochs required to learn the vanilla VAE, and the other remaining models did not reach such accuracy in the first 50 epochs. Accordingly, our proposed model converges to the optimal set of learning parameters under a limited number of learning epochs, and it also outperforms all related models in the literature considering both datasets.

5 Conclusions

We proposed an explainable VAE model termed (Grad$_2$VAE) to be utilized for XAI, image reconstruction, generation, object detection, and anomaly detection applications. We used the 2$^{\text{nd}}$ partial derivative of the neuron activation (or responses) between the latent space, Z, and the 1$^{\text{st}}$ encoding layer to capture the curvatures of the representations at an early stage by reconstructing visual attention maps. Our proposed model can be expanded for different data types and scales, it also accelerates the learning process by boosting the whole reconstruction process through the residual fusion. Moreover, we employed our proposed model to explain the learned representations through a learnable (online) visual attention mapping, where it shows a better visual explainability than the related works based on offline attention mapping. Furthermore, we generalized our proposed model in the application of one-class anomaly detection. Our model outperforms all related deep models in both qualitative and quantitative analysis. In future works, we plan to investigate our proposed method for other UGL models such as GANs and other ML applications.

References

1. Abukmeil, M., Ferrari, S., Genovese, A., Piuri, V., Scotti, F.: On approximating the non-negative rank: Applications to image reduction. In: Proc. of CIVEMSA (2020)
2. Abukmeil, M., Ferrari, S., Genovese, A., Piuri, V., Scotti, F.: Unsupervised learning from limited available data by $\beta-$NMF and dual autoencoder. In: Proc. of ICIP (2020)
3. Abukmeil, M., Ferrari, S., Genovese, A., Piuri, V., Scotti, F.: A survey of unsupervised generative models for exploratory data analysis and representation learning. Acm computing surveys (csur) **54**(5), 1–40 (2021)
4. Abukmeil, M., Genovese, A., Piuri, V., Rundo, F., Scotti, F.: Towards explainable semantic segmentation for autonomous driving systems by multi-scale variational attention. In: Proc. of ICAS (2021)
5. Ames, W.F.: Numerical methods for partial differential equations. Academic press (2014)
6. Baldi, P.: Autoencoders, unsupervised learning, and deep architectures. In: Proc. of ICML (2012)
7. Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. IEEE Trans. on Pattern Analysis and Machine Intelligence **35**(8), 1798–1828 (2013)
8. Fan, K., Wang, Z., Beck, J., Kwok, J., Heller, K.A.: Fast second order stochastic backpropagation for variational inference. In: Proc. of NIPS (2015)
9. Genovese, A., Piuri, V., Scotti, F.: Towards explainable face aging with generative adversarial networks. In: Proc. of ICIP (2019)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proc. of CVPR (2016)
11. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science **313**(5786), 504–507 (2006)
12. Hinton, G.E., Zemel, R.S.: Autoencoders, minimum description length and helmholtz free energy. In: Proc. of NIPS (1994)
13. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: Proc. of ICML (2014)
14. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: Proc. of ICLR (2014)
15. LeCun, Y., Cortes, C., Burges, C.J.: Mnist handwritten digit database. 2010. URL http://yann. lecun. com/exdb/mnist **7**(23), 6 (2010)
16. Liu, W., Li, R., Zheng, M., Karanam, S., Wu, Z., Bhanu, B., Radke, R.J., Camps, O.: Towards visually explaining variational autoencoders. In: Proc. of CVPR (2020)
17. Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., Frey, B.: Adversarial autoencoders. arXiv preprint arXiv:1511.05644 (2015)
18. Mejjati, Y.A., Richardt, C., Tompkin, J., Cosker, D., Kim, K.I.: Unsupervised attention-guided image-to-image translation. In: Proc. of NIPS (2018)
19. Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. In: Proc. of ICML (2010)
20. Ng, A.: Sparse autoencoder. CS294A Lecture notes **72**(2011), 1–19 (2011)
21. Pospiech, G., Michelini, M., Eylon, B.S.: Mathematics in physics education. Springer (2019)
22. Rezende, D.J., Mohamed, S., Wierstra, D.: Stochastic backpropagation and approximate inference in deep generative models. In: Proc. of ICML (2014)

23. Rifai, S., Vincent, P., Muller, X., Glorot, X., Bengio, Y.: Contractive auto-encoders: Explicit invariance during feature extraction. In: Proc. of ICML (2011)
24. Ruff, L., Vandermeulen, R., Goernitz, N., Deecke, L., Siddiqui, S.A., Binder, A., Müller, E., Kloft, M.: Deep one-class classification. In: Proc. of ICML (2018)
25. Sarafijanovic-Djukic, N., Davis, J.: Fast distance-based anomaly detection in images using an inception-like autoencoder. In: Proc. of DS (2019)
26. Tang, C., Srivastava, N., Salakhutdinov, R.R.: Learning generative models with visual attention. In: Proc. of NIPS (2014)
27. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A.: Extracting and composing robust features with denoising autoencoders. In: Proc. of ICML (2008)
28. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747 (2017)
29. Zeiler, M.D.: Adadelta: an adaptive learning rate method. arXiv preprint arXiv:1212.5701 (2012)
30. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional nets. In: Proc. of ECCV (2014)
31. Zhang, C., Butepage, J., Kjellstrom, H., Mandt, S.: Advances in variational inference. IEEE Trans. on Pattern Analysis & Machine Intelligence **41**(08), 2008–2026 (2019)
32. Zhang, J., Bargal, S.A., Lin, Z., Brandt, J., Shen, X., Sclaroff, S.: Top-down neural attention by excitation backprop. Int. Journal of Computer Vision **126**(10), 1084–1102 (2018)