# Classification et détection d'anomalie conjointes par structuration de l'espace latent d'un auto-encodeur variationnel

Maxime OSSONCE[1,2], Florence ALBERGE[2], Pierre DUHAMEL[2]

[1]ESME Sudria
38 rue Molière
94200 Ivry-sur-Seine

[2]Université Paris-Saclay, CNRS, CentraleSupélec
Laboratoire des signaux et systèmes
91190 Gif-sur-Yvette

maxime.ossonce@esme.fr
florence.alberge@centralesupelec.fr, pierre.duhamel@centralesupelec.fr

**Résumé –** Les réseaux de neurones profonds ont prouvé leur efficacité dans de multiples tâches, comme la classification d'image. Ils on toutefois tendance à surestimer la fiabilité de leurs décisions, notamment lorsque l'entrée est différente des exemples fournis lors de l'entraînement du modèle. Une telle entrée est qualifiée d'hors-distribution (OoD) et sa détection est un élément essentiel de la sécurité de toute chaîne de traitement basée sur l'apprentissage machine. Nous proposons ici un modèle d'auto-encodeur capable de réaliser conjointement la tâche de classification et de détection d'OoD en nous appuyant sur le paradigme du classifieur génératif : nous dotons la variable latente de l'auto-encodeur variationnel d'un a priori $p(z|y)$ conditionné à la classe. Le modèle donnera alors, pour une image $x$, une estimation de $p(x|y)$ permettant d'effectuer la classification et la détection d'OoD. La méthode proposée a de meilleures performances que les méthodes classiques que ce soit pour la détection d'OoD lointains ou proches tout en présentant des performances identiques en terme de classification.

**Abstract –** Deep Neural networks are very efficient at object recognition but have sometimes an overestimated reliability for erroneous decisions. It is therefore very important to check if the distribution of images being processed matches that of the training set. This has been studied as OoD detection. This paper proposes an algorithm allowing classification as well as OoD detection, based on the generative classifier paradigm. While a discriminative classifier learns $p(y|x)$ to achieve classification and a generative network learns $p(x)$ to achieve OoD detection, the generative classifier learns $p(x|y)$ to achieve classification *as well as* OoD detection. The proposed model is shown to outperform classical approaches based on discriminative classifiers for far OoD and GAN based models for near OoD without losing any performance for the classification task.

## 1 Introduction

La robustesse des réseaux de neurones profonds (DNN) utilisés pour la classification d'images est un prérequis à leur déploiement. Les réseaux de neurones sont en effet entraînés à partir de données aux caractéristiques communes et constituant des échantillons issus d'une certaine distribution (InD) pour laquelle le réseau est garanti de fonctionner correctement. Dans un monde réel, ouvert, le réseau de neurones peut être utilisé sur des données qui ne sont pas compatibles avec les données pour lesquelles il a été entraîné. On parle alors de données hors-distribution (OoD). Cette situation pathologique peut être due à un environnement non-stationnaire, à un acte malveillant ou à une erreur de l'utilisateur. Sans attention particulière, cette situation est indétectable puisque le réseau de neurones fournit toujours un résultat (de classification par exemple) parfois avec l'illusion d'une très grande fiabilité même lorsque les données sont aberrantes [5]. Il est donc important de détecter ces situations, en particulier pour des applications sensibles comme celles relevant du domaine médical.

Les méthodes de détection d'OoD peuvent être cataloguées selon l'emploi (ou non) d'exemples OoD pendant la phase d'entraînement. On parle alors de méthodes supervisées, semi-supervisées ou non supervisées. Dans cet article, afin de nous placer dans des hypothèses réalistes, nous considérerons uniquement des techniques non supervisées c'est à dire ne requérant aucune information a priori sur les OoD. Il faut noter que certaines méthodes dites non supervisées utilisent toutefois un jeu de validation OoD pour le réglage de certains hyper-paramètres, ce qui n'est pas le cas de la méthode proposée ici.

Les premiers travaux sur la détection d'OoD reposent sur un score de confiance calculé à partir des probabilités a posteriori fournies en sortie du classifieur. La méthode basique proposée dans [6] a été améliorée (ODIN, [10]) en recalibrant les a posteriori par un facteur d'échelle sur les logits et en ajoutant une perturbation sur l'entrée dans la direction du gradient afin d'améliorer la performance du détecteur. Une distance de Mahalanobis entre les sorties de couches intermédiaires et les espérances

conditionnelles à la classe de ces dernières peut être utilisée [8] pour la détection d'OoD. Bien que les méthodes [10, 8] soient considérées comme étant non supervisées, le réglage de certains paramètres se fait à l'aide d'un jeu de validation OoD ; elles ne rentrent donc pas dans la champ de cet article. Néanmoins, ODIN étant une référence dans la détection d'OoD, nous nous comparerons à elle même si notre méthode a l'avantage de n'utiliser absolument aucune information sur les OoD.

Les modèles génératifs peuvent aussi être utilisés pour la détection d'OoD. Ils sont entraînés afin que seuls les éléments de la distribution (InD) puissent être générés correctement. Dans les réseaux antagonistes génératifs (GAN), les exemples adversariaux générés servent de modèle d'OoD lors de l'entraînement du discriminateur (*e.g.* GANomaly [1]). Nous montrons dans les simulations que cette méthode échoue à détecter des OoD lointains. Les modèles génératifs basés sur les auto-encodeurs variationnels (VAE) [7] et utilisés pour la détection d'OoD se basent sur le fait qu'ils ne seront capables de générer correctement que les images InD.

Lorsque la tâche principale du modèle est la classification, il est possible d'utiliser un classifieur génératif (GC) : quand un classifieur discriminant (DC) apprend la distribution $p(y|x)$ pour inférer la classe $y$ à laquelle appartient l'image $x$, le GC apprend une estimation $p(x|y)$ permettant d'effectuer la classification *et* la détection d'OoD. Par ailleurs un conditionnement à la classe offre la possibilité d'un ajustement plus fin du modèle. Les GC sont réputés [9] plus robustes que les DC mais ont de moins bonnes performances de classification. Un modèle de GC construit sur un VAE et doté d'un modèle à mélange gaussien (GMM) pour la variable latente est proposé par [4]. Un réseau de neurones inversible avec un GMM est proposé par [11] pour résister aux attaques adversariales.

Nous montrons ici qu'un modèle de VAE dont l'a priori sur la variable latente est un GMM permet d'effectuer la classification conjointement à la détection d'OoD. Notre méthode généralise [4] en incluant les paramètres critiques de l'espace latent dans la boucle d'apprentissage. Cela permet de pouvoir considérer des tâches de classification plus complexes que celles considérées dans [4] et d'obtenir de bien meilleurs résultats en terme de détection d'OoD. Par ailleurs, notre méthode se compare très favorablement à l'état de l'art pour la détection d'OoD lointains comme d'OoD proches sans perte de performance sur la classification.

# 2 Modèle proposé

Le modèle proposé ici est basé sur le VAE [7] décrit section 2.1. Alors qu'un VAE apprend une approximation de $p(x)$, l'auto-encodeur variationnel conditionnel à la classe (CCVAE) décrit en section 2.2 apprend $p(x|y)$. La variable $y$ représente la classe à laquelle $x$ est susceptible d'appartenir. L'estimation de $p(x|y)$ permettra de déterminer la classe d'appartenance de $x$ selon $\hat{y} = \arg\max_y p(x|y)$. Elle permettra également de déterminer si $x$ est OoD en comparant $\max_y p(x|y)$ à un seuil.

## 2.1 Auto-encodage variationnel

Le VAE est un modèle génératif ayant une structure d'auto-encodeur qui suppose l'existence d'une variable latente $z \sim \mathcal{N}(0, I_\kappa)$ et du processus de génération des échantillons $x \in \mathcal{X} \subset \mathbb{R}^d : x \sim p_\theta(x|z)$. Le décodeur $p_\theta(x|z)$ appartient à une famille paramétrique $p_\theta(x|z) = \mathcal{N}\left(x|f_\theta(z), \sigma^2\right)$. L'a posteriori $p_\theta(z|x)$ n'étant pas calculable simplement, il sera remplacé par une approximation variationnelle $q_\phi(z|x)$ appartenant aussi à une famille paramétrique. La fonction objectif optimisée à l'entraînement est alors une borne inférieure de l'évidence (ELBO) sur $\log p(x)$ :

$$\begin{aligned}\mathsf{elbo}(x) &\coloneqq \log p_\theta(x) - \mathrm{KL}[q_\phi(z|x) \,\|\, p_\theta(z|x)] \\ &= \mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\theta(x|z)] - \mathrm{KL}[q_\phi(z|x) \,\|\, p(z)].\end{aligned}$$

Au moment du test, une approximation plus fine de $p(x)$ peut être obtenue par échantillonnage préférentiel (IWS).

## 2.2 ELBO modifiée

De manière similaire à [4], une borne sur $p(x|y)$ est calculée par le modèle proposé ici :

$$\begin{aligned}\mathsf{elbo}(x|y) &\coloneqq \log p_\theta(x|y) - \mathrm{KL}[q_\phi(z|x) \,\|\, p_\theta(z|x,y)] \\ &= \mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\theta(x|z)] \\ &\quad - \mathrm{KL}[q_\phi(z|x) \,\|\, p_\phi(z|y)].\end{aligned} \tag{1}$$

Cette ELBO fait apparaître un nouvel a priori $p_\phi(z|y)$ conditionnel à la classe $y$ sur la variable latente $z$ dont on supposera qu'il suit la loi $\mathcal{N}(m_\phi^y, I_\kappa)$ où $m_\phi^y$ représente la classe dans l'espace latent. Lorsque le décodeur $p_\theta(x|z)$ est supposé gaussien $p_\theta(x|z) = \mathcal{N}\left(x|f_\theta(z), \sigma^2 I_d\right)$, la première partie de (1) fait apparaître l'erreur quadratique moyenne (MSE) entre l'entrée et sa reconstruction $f_\theta(z)$ :

$$\log p_\theta(x|z) = -\frac{d}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\|x - f_\theta(z)\|^2. \tag{2}$$

De plus, le codeur étant restreint à une famille paramétrique $q_\phi(z|x) = \mathcal{N}\left(z|\mu_\phi(x), \Lambda_\phi(x)\right)$, la divergence de Kullback-Leibler (DKL) de (1) est tractable :

$$\begin{aligned}\mathrm{KL}[q_\phi(z|x) \,\|\, p_\phi(z|y)] &= \frac{1}{2}\|\mu_\phi(x) - m_\phi^y\|^2 \\ &\quad + \frac{1}{2}\left(\mathrm{tr}\left(\Lambda_\phi(x)\right) - \kappa - \log\left|\Lambda_\phi(x)\right|\right).\end{aligned} \tag{3}$$

où $\mu_\phi(x)$ et $\Lambda_\phi(x)$ (diagonale de déterminant $\left|\Lambda_\phi(x)\right|$) sont des sorties d'un DNN de paramètres $\phi$. Le premier terme de (3) favorise la classification : à l'entraînement $\mu_\phi(x)$ est poussé vers $m_\phi^y$, permettant une classification par plus proche voisin. Le deuxième terme, permet la réduction de dimension propre au VAE. Cette restriction de l'espace de reconstruction aux images InD est essentiel pour la détection d'OoD par les VAE.

## 2.3 Entraînement du modèle

La fonction de coût minimisée à l'entraînement est l'opposée de l'ELBO (1) moyennée sur des mini-batchs :

$$\mathscr{L}_{\mathrm{CCVAE}}(\theta,\phi) = \mathscr{L}_{\mathrm{MSE}}(\theta,\phi) + \mathscr{L}_{\mathrm{KL}}(\theta,\phi).$$

La variance $\sigma^2$ (2) rend compte de la capacité de reconstruction du modèle. Elle peut être fixée a priori. Les résultats expérimentaux montrent qu'elle permet de créer un compromis entre les capacités de classification et de détection. La valeur de $\sigma^2$ minimisant la fonction de coût est la MSE atteinte en moyenne sur le jeu d'entraînement. Il est ainsi possible d'inclure $\sigma^2$ dans la boucle d'apprentissage en le considérant comme une sortie du décodeur dépendant de l'entrée $x$, l'association $x \mapsto \sigma_\phi(x)$ est alors apprise et constitue un prédicteur de la MSE.

Dans [4] les moyennes $\{m^y\}$ du GMM $p_\phi(z|y)$ ont été fixées. Leurs valeurs ont donc un impact arbitraire sur les performances du modèle. Si l'on met les moyennes $\{m^y_\phi\}$ directement dans la boucle d'apprentissage, on observe qu'elles convergent vers $0_\kappa$ : le CCVAE se simplifie alors en VAE. Il s'agit pourtant de paramètres structurant l'espace latent dont l'ajustement devrait permettre une amélioration des performances du modèle. Dans le modèle que nous proposons, nous assurons la séparation des classes dans l'espace latent : une entropie croisée (CE) entre la classe $y$ et la sortie *softmax* d'un classifieur linéaire entraîné sur la variable latente $z$ est ajoutée à la fonction de coût. Ceci fait alors apparaître un hyper-paramètre $\gamma$ pondérant la CE :

$$\mathscr{L}^\gamma(\theta,\phi) = \mathscr{L}_{\text{MSE}}(\theta,\phi) + \mathscr{L}_{\text{KL}}(\theta,\phi) + \gamma\mathscr{L}_{\text{CE}}(\theta,\phi).$$

Les résultats expérimentaux montrent qu'une bonne classification sur des jeux d'images complexes (tels que CIFAR10) n'est obtenue qu'avec des moyennes $\{m^y_\phi\}$ apprises et une variance $\sigma^2 = \sigma^2_\phi(x)$. Par ailleurs les résultats sont peu sensibles à la valeur de $\gamma$ qui est identique pour tous les jeux testés.

### 2.4 Classification et détection d'OoD

La classification se fait par recherche parmi les moyennes $\{m^y_\phi\}$ du plus proche voisin de $\mu_\phi(x)$. Le modèle permet d'obtenir au moment du test, pour une image $x$ et une classe $y$ quelconques une estimation $\hat{p}(x|y)$, par IWS, de $p(x|y)$. Lorsque $x$ est OoD, $\hat{p}(x|y)$ sera faible pour toute classe $y$ considérée. La statistique $t(x) = \max_y \hat{p}(x|y)$ sera alors seuillée pour déterminer si $x$ est un OoD. Les performances de la détection d'OoD seront mesurées par la courbe fonction d'efficacité du récepteur (ROC) dont on présentera l'aire sur la courbe ROC (AUROC) ainsi que la valeur du taux de faux positifs (FPR) pour un taux de vrais positifs (TPR) donné [6] (ici 95 %).

Il a été observé [13] que dans certaines configurations (*e.g.* en considérant le jeu SVHN comme OoD sur un modèle entraîné sur CIFAR10) les VAE peuvent assigner une plus grande valeur de densité $p(x)$ aux données OoD qu'aux données InD. Pour contourner ce problème il est possible [14] d'utiliser un test bilatéral sur $p(x)$ ($t(x)$ dans notre cas) dont les fondements sont basés sur la typicalité [3] d'une séquence i.i.d. Cette technique a été utilisée avec succès pour la détection d'OoD d'un seul échantillon [12, 11] et permet d'obtenir de bonnes performances de détection avec le modèle proposé ici. Un test asymétrique a été retenu, pour lequel un TPR de 95 % est obtenu en réglant les seuils (grâce à un jeu de validation InD) de sorte à répartir les 5 % d'erreur en 4 % à gauche et 1 % à droite.

| dataset | arch. | VIB | CCVAE |
|---|---|---|---|
| MNIST | VGG11 | 99,3$_{\pm 0.2}$ | **99,4**$_{\pm 0.2}$ |
| CIFAR10 | VGG19 | 91,4$_{\pm 0.6}$ | **91,5**$_{\pm 0.6}$ |

TABLE 1 – Résultats de classification et architectures utilisées.

| | FPR@95 | AUROC |
|---|---|---|
| OoD | baseline/ODIN/FV/IWS-A-4-1 | |
| FASHION | 22,6/20,5/12,9/ **0,0** | 89,3/91,8/96,9/**99,7** |
| EMNIST | 44,3/41,8/21,3/ **0,3** | 73,9/78,5/91,6/**99,5** |
| average | 33,5/31,3/17,1/ **0,2** | 81,6/85,2/94,2/**99,6** |

TABLE 2 – Détection d'OoD - Modèle entraîné sur MNIST. Sur les FPR indiqués en gras les IC sont au plus de $\pm 0{,}1$.

## 3 Résultats

Le modèle proposé est appliqué à la classification d'images. L'architecture VGG [15] basée sur un nombre élevé de couches convolutionnelles [16] est utilisée pour le codeur du CCVAE. Le décodeur est construit avec des couches dites de déconvolution [16]. Pour la comparaison avec les méthodes basées sur des DC, (ODIN, baseline) une architecture identique est appliquée à un VIB [2]. Le VIB est similaire au CCVAE pour ce qui est de la classification (le $\beta$ du VIB est équivalent à $\gamma^{-1}$ dans le CCVAE).

Les résultats de classification obtenus avec un CCVAE sont présentés table 1 et comparés avec un VIB qui partage la même architecture d'encodeur. On constate que la capacité de détection d'OoD du CCVAE ne se fait pas au détriment de la classification puisque les performances atteintes sont légèrement supérieures à celles du VIB (91,5 % sur CIFAR10). Les résultats de détection d'OoD *lointains* sont présentés pour des modèles entraînés sur MNIST (table 2) et CIFAR10 (table 3). Pour la méthode ODIN [10], la température $T$ et l'amplitude $\varepsilon$ sont choisies de sorte à maximiser le FPR sur le jeu OoD considéré. La méthode notée FV est un CCVAE tel qu'utilisé dans [4] : les moyennes du mélange gaussien modélisant l'a priori sur $z$ sont des encodage *one-hot* des classes ; la variance de $z$ est fixée, $\Lambda(x) = 3{,}5 \times 10^{-4}$ ; $\sigma^2$ est choisi de sorte à obtenir la même précision de classification que notre modèle. Ce réglage n'est adapté [4] qu'à des jeux de données simples tels que MNIST. Les résultats sur MNIST (table 2) montrent que l'ajustement des moyennes du GMM et la variance de reconstruction $\sigma^2$ adaptée à l'entrée permettent une amélioration drastique des performances de détection d'OoD. Les résultats sur CIFAR10 (table 3). montrent que la paire (InD,OoD) la plus difficile est (CIFAR10, LSUN(r)) pour laquelle le FPR est de 14,4 % avec notre méthode contre 57,1 % avec ODIN. Les performances de détection d'OoD *lointains* atteintes par GANomaly ne sont pas bonnes. Ce dernier est plus adapté à la détection d'OoD *proches*, étudiées plus bas. Par ailleurs les résultats montrent que notre méthode permet d'obtenir des FPR plus petits que la méthode ODIN sur toutes les paires (InD,OoD) étudiées.

Pour l'étude de la détection d'OoD *proches*, nous utilisons la même procédure que [1] : le modèle est entraîné avec un jeu

| | FPR@95 | AUROC |
|---|---|---|
| OoD | baseline/ODIN/GANomaly/IWS-A-4-1 | |
| SVHN | 69,4/27,5/99,5/ **2,4** | 89,6/91,3/30,9/**98,9** |
| LSUN(r) | 71,0/57,1/86,2/**14,4** | 85,4/84,6/56,3/**97,1** |
| LSUN(c) | 56,9/30,9/98,7/ **3,4** | 90,8/91,6/48,8/**98,4** |
| average | 65,8/40,3/94,8/ **6,7** | 88,6/88,2/45,3/**98,2** |

TABLE 3 – Detection d'OoD - Modèle entraîné sur CIFAR10. Sur les FPR indiqués en gras les IC sont au plus de $\pm$0,7.

| | FPR@95 | AUROC |
|---|---|---|
| OoD | baseline/ODIN/GANomaly/IWS-A-4-1 | |
| 0 | 82,6/61,6/**10,9**/60,4 | 76,5/79,8/**96,7**/76,0 |
| 1 | 95,3/92,6/68,6/ **0,1** | 49,1/42,6/67,4/**99,6** |
| 2 | 74,8/68,6/94,3/**54,8** | 77,9/**82,4**/55,1/79,2 |
| 3 | 78,2/74,0/87,3/**67,4** | 77,2/77,4/61,0/**87,0** |
| 4 | 72,3/63,5/70,2/**26,8** | 86,7/86,5/75,7/**91,1** |
| 5 | 82,8/**57,5**/69,9/60,0 | 81,7/88,3/72,9/**89,8** |
| 6 | 77,2/72,3/**36,1**/96,3 | 82,1/66,8/**88,7**/44,9 |
| 7 | 79,1/78,9/81,8/**17,4** | 76,4/75,7/63,8/**96,6** |
| 8 | 86,6/85,7/**34,1**/74,3 | 70,7/67,9/**88,4**/66,6 |
| 9 | 86,0/82,3/61,2/ **0,1** | 69,0/59,7/70,7/**99,0** |
| avg | 81,5/73,7/61,4/**45,8** | 74,7/72,7/74,0/**83,0** |

TABLE 4 – Detection d'OoD proches - Modèle entraîné sur CIFAR10 (une classe enlevée). Les IC sont au plus de $\pm$3,2 sur les FPR.

de données dont une des classes est enlevée et considérée au moment du test comme OoD. Les résultats, présentés table 4, montrent que la méthode ODIN n'est pas adaptée à la détection d'OoD *proches* (avec un FPR moyen de 73,7 %) et que le CCVAE est plus performant que GANomaly dans sept cas sur dix ainsi qu'en moyenne.

# 4 Conclusion

Les méthodes classiques de détection d'OoD montrent de bonnes performances soit sur les OoD lointains, soit sur les OoD proches. Nous montrons qu'un GC conçu conjointement pour la classification et la détection d'OoD permet une prise en compte plus globale des OoD sans perte de performance de classification.

# Références


[1] S. Akcay, A. Atapour-Abarghouei, and T. P. Breckon. Ganomaly : Semi-supervised anomaly detection via adversarial training. In *Asian Conference on Computer Vision*, pages 622–637. Springer, 2018.

[2] A. A. Alemi, I. Fischer, J. V. Dillon, and K. Murphy. Deep variational information bottleneck. In *5th International Conference on Learning Representations (ICLR)*, 2017.

[3] T. Cover and J. Thomas. *Elements of information theory*. Wiley-Interscience, 2006.

[4] P. Ghosh, A. Losalka, and M. J. Black. Resisting adversarial attacks using gaussian mixture variational autoencoders. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI*, pages 541–548, 2019.

[5] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. In *3rd International Conference on Learning Representations, ICLR, San Diego, CA,*, 2015.

[6] D. Hendrycks and K. Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *5th International Conference on Learning Representations, ICLR Toulon, France,*, 2017.

[7] D. P. Kingma and M. Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR, Banff, AB, Canada*, 2014.

[8] K. Lee, K. Lee, H. Lee, and J. Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems 31*, pages 7167–7177. 2018.

[9] Y. Li and Y. Sharma. Are generative classifiers more robust to adversarial attacks ? In *ICML*, 2019.

[10] S. Liang, Y. Li, and R. Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *6th International Conference on Learning Representations, ICLR, Vancouver, BC, Canada.*, 2018.

[11] R. Mackowiak, L. Ardizzone, U. Köthe, and C. Rother. Generative classifiers as a basis for trustworthy image classification. In *Proc. of the IEEE Conf. on Comput. Vision and Pattern Recogn. (CVPR)*, pages 2971–81, 2021.

[12] W. R. Morningstar, C. Ham, A. G. Gallagher, B. Lakshminarayanan, A. A. Alemi, and J. V. Dillon. Density of states estimation for out-of-distribution detection. In *Proc. of the 24th International Conference on Artifi-cial Intelligence and Statistics (AISTATS)*, pages 3232–3240, 2021.

[13] E. Nalisnick, A. Matsukawa, Y. W. Teh, D. Gorur, and B. Lakshminarayanan. Do deep generative models know what they don't know ? In *7th International Conference on Learning Representations (ICLR)*, 2019.

[14] E. Nalisnick, A. Matsukawa, Y. W. Teh, and B. Lakshminarayanan. Detecting out-of-distribution inputs to deep generative models using typicality. http ://arxiv.org/abs/1906.02994, 2019.

[15] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, (ICLR), San Diego, CA*, 2015.

[16] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *ECCV*, pages 818–833, 2014.