Out-of-Distribution Detection of Melanoma using Normalizing Flows

OUT-OF-DISTRIBUTION DETECTION OF MELANOMA USING NORMALIZING FLOWS

M.M.A. Valiuddin, C.G.A Viviers

Department of Electrical Engineering, Eindhoven University of Technology

ABSTRACT

Generative modelling has been a topic at the forefront of machine learning research for a substantial amount of time. With the recent success in the field of machine learning, especially in deep learning, there has been an increased interest in explainable and interpretable machine learning. The ability to model distributions and provide insight in the density estimation and exact data likelihood is an example of such a feature. Normalizing Flows (NFs), a relatively new research field of generative modelling, has received substantial attention since it is able to do exactly this at a relatively low cost whilst enabling competitive generative results. While the generative abilities of NFs are typically explored, we focus on exploring the data distribution modelling for Out-of-Distribution (OOD) detection. Using one of the state-of-the-art NF models, GLOW, we attempt to detect OOD examples in the ISIC dataset. We notice that this model under performs in conform related research. To improve the OOD detection, we explore the masking methods to inhibit co-adaptation of the coupling layers however find no substantial improvement. Furthermore, we utilize Wavelet Flow which uses wavelets that can filter particular frequency components, thus simplifying the modeling process to data-driven conditional wavelet coefficients instead of complete images. This enables us to efficiently model larger resolution images in the hopes that it would capture more relevant features for OOD. The paper that introduced Wavelet Flow mainly focuses on its ability of sampling high resolution images and did not treat OOD detection. We present the results and propose several ideas for improvement such as controlling frequency components, using different wavelets and using other state-of-the-art NF architectures.

***Index Terms*—** Normalizing Flows, Generative Modelling, Deep Learning, Out-of-Distribution Detection, Wavelets

1. INTRODUCTION

Research in probabilistic generative modeling is receiving significant attention. This, because rapid advances in computational power in combination with an increased availability of data makes modeling complex data distributions more feasible. Recently developed techniques show the feasibility of modelling the probability distribution p_X from observed data $\{\mathbf{x}\}_{i=1}^{N}$. The applications of the generative models include density estimation, outlier detection, sampling new data and much more. Models such as Variational Auto-encoders (VAEs) [1] and Auto-Regressive Generative Neural Networks (such as pixelRNN [2]) have shown impressive attempts at learning the data distribution. Each of these implementations have shown a unique generative ability, but they also have their own set of drawbacks. For instance, in the case of VAEs, we approximate the density of the data through minimizing the evidence lower bound (ELBO). This design choice will rarely make the estimated distribution closely match the actual data distribution and sacrifices tractability during sampling and inference. On the other hand, Generative Adversarial Networks (GANs) do not model the density of the data explicitly but rather focus on the ability to generate realistic images through a mini-max game between a discriminator a generative network. The discriminator can then try to distinguish between in and out of distribution data with no tractable density. This makes qualitative evaluation of the (implicit) distribution not possible. Another downside that comes with it is difficulty during training. Mode collapse, posterior collapse, vanishing gradients and training instability [3, 4] make them a less reliable solution. Auto-Regressive Generative models in fact, do explicitly model the density. The generated samples are relatively good yet the process very slow due to its sequential data generation.

In the research community there has been a trend towards explainable low cost models with tractable probabilities. Normalizing Flows (NF) are capable of sampling as well as do density estimation with tractable probabilities and is gaining significant interest. An impressive characteristic of this architecture is the fact that NFs attempt to model any complex probability distribution [5–7] through likelihood maximization. Despite the NFs being a fairly recent research area, it has produced competitive results on public benchmark datasets. With the recent developments in the NFs architectures [8–10], the qualitative and quantitative results as well as computational requirements are extremely competitive. The applications include but are not limited to image generation [8, 11], noise modelling [12], video generation [13], audio generation [14–16], graph generation [17], reinforcement learning [18–20], computer graphics [21] and physics [22–26].

In this research project the Out-of-Distribution (OOD) detection using state-of-the-art models GLOW and Wavelet Flow is investigated. This is done on public datasets as well as The International Skin Imaging Collaboration (ISIC) dataset [27]. The dataset contains images of benign as well as malignant melanoma. To the best of our knowledge, the ability of OOD detection with NFs on medical imaging has not been investigated. Since in most practical scenarios anomaly data is scarce, we choose to model the benign images and consider the malignant images anomalous. We find that OOD detection using the GLOW architecture performs poorly due to the exploitation of local pixel correlations and the co-adaptation of the coupling layers. Supported by recent research by Kirichenko *et al.* [28], the cycle masking technique in the coupling layers are tested an an attempt to inhibit co-adaptation of coupling layers but provide limited improvement.

Recently, a model named Wavelet Flow [10] has been published. Where instead of the data distribution, the detail coefficients are modeled from the low frequency Haar wavelet component. It is shown that wavelets in combination NFs provide numerous benefits including likelihood estimation and sampling at a higher resolutions. The authors showcase this by sampling at a 8x times higher resolution than that of the training images. Furthermore, the proposed model delivers competitive results on public benchmark datasets in bits per dimension with up to 15x faster training. We see wavelets as an opportunity to control the components of the images being modeled by the NFs and leverage it for OOD detection at higher resolution. OOD detection using Wavelet Flow has not been investigated, even the original published work. Thus, we decided to investigate this by applying it to the ISIC dataset and report our findings.

2. RELATED WORK

Kirichenko *et al.* [28] investigate OOD detection and *Why Normalizing Flows Fail to Detect Out-of-Distribution Data*. The paper argues that NFs mainly learn local pixel correlations and generic image-to-latent-space transformations instead of capturing semantic information of the images. Furthermore, the authors argue that coupling NFs suffer from co-adaptation, where it can equally likely predict OOD data due to dependency of the coupling layers hence assigning high probabilities to OOD data. One of the proposed solutions cycle masking, has proven to tackle this issue. However, the effectiveness of this method shows to be dependent on the datasets being used. We implement cycle masking in our research in order to determine if it can improve OOD detection for the ISIC dataset.

Ren *et al.* [29] observe that the likelihood score in OOD detection from deep generative models are often heavily affected by population level background statistics. They propose a likelihood ratio method to effectively correct this. We draw our inspiration to use wavelets to control what components of the image NFs learn.

3. METHODOLOGY

This internship research aims to answer the following questions:

- Can we reproduce some of the research of Out-of-Distribution detection with Normalizing Flows and apply it to the ISIC melanoma dataset?
- Which techniques can we employ to improve Out-of-Distribution detection on the ISIC melanoma dataset?
- Does Wavelet Flow outperform GLOW in Out-of-Distribution detection on the ISIC melanoma dataset?

In order to answer these questions, we investigated the theory behind NFs. We apply our newly acquired knowledge of Normalizing Flows to Out-of-Distribution on the ISIC dataset. We trained the chosen models on the benign images of the dataset and evaluate it against the malignant classes.

4. NORMALIZING FLOWS

We call a sequence of bijective transformations a Normalizing Flow. NFs map a complex distribution $p_{\mathbf{x}}(\mathrm{X})$ to a more simple distribution $p_{\mathbf{z}}(\mathrm{Z})$ (normalizing direction) through a chain of transformations. By constraining these transformations to be bijective, an inverse transformations exist that transforms the base distribution $p_0(\mathbf{z}_0)$ to the complex distribution $p_k(\mathbf{z}_k)$ (generative direction) and thus enabling sample generation. Since we want to map $p_{\mathbf{x}}(X)$ from $p_{\mathbf{z}}(Z)$ to enable data generation, it is required that these transformations are invertible. This idea is visualized in Figure 1, where k equals the final complex transformed distribution while i denotes the ith intermediate distribution of the chain. In Section 4.1 a mathematical formulation on NFs are provided. This is followed by an overview of different types of flows in Section 4.2. The NF architectures implemented during this internship project are presented in Section 4.3 and Section 4.4.

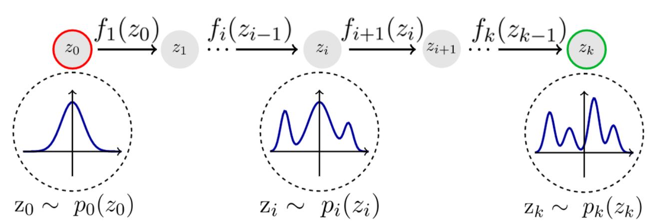


Fig. 1: Series of consecutive transformations transforming a Gaussian base to a complex non-Gaussian target distribution. Image taken from *Janosh Dev.* [30]

4.1. Mathematical background

A bijection between distributions should adhere to the law of total probability as

$$\int p(x)dx = \int p(y)dy. \tag{1}$$

We are only concerned in the absolute change of volume, hence the sign of it is disregarded. We can represent a sample in Y in terms of a sample in X. As shown below in the change of variables formula (Eqn. 2), we can evaluate $p_{\mathbf{y}}(Y)$ and maintain the total density through any transformation.

$$p(y) = p(x)\left|\frac{dx}{dy}\right|. \tag{2}$$

This local-linear change approximation is in fact the Jacobian determinant. Given that $y = f(x)$ and thus $x = f^{-1}(y)$ we can write

$$p_Y(y) = p_X(f^{-1}(y))\left|\det\frac{\partial f^{-1}}{\partial y}\right|. \tag{3}$$

We denote a sample and its transformation as

$$\begin{aligned} \mathbf{z}_{i-1} &\sim p_{i-1}\left(\mathbf{z}_{i-1}\right) \\ \mathbf{z}_i &= f_i\left(\mathbf{z}_{i-1}\right), \mathbf{z}_{i-1} = f_i^{-1}\left(\mathbf{z}_i\right). \end{aligned} \tag{4}$$

Given the chain of the transformations

$$\mathbf{x} = \mathbf{z}_K = f_K \circ f_{K-1} \circ \cdots f_1\left(\mathbf{z}_0\right) \tag{5}$$

the log-likelihood of the observed data $\mathbf{x}$ is concisely written as

$$\log p(\mathbf{x}) = \log p_0\left(\mathbf{z}_0\right) - \sum_{i=1}^{K}\log\left|\det\frac{df_i}{d\mathbf{z}_{i-1}}\right| \tag{6}$$

See appendix A for the complete derivation. Notice how the log-likelihood of $p(\mathbf{x})$ only requires inference on the base distribution from the latent vector $\mathbf{z_0}$. All other contributions to the log-likelihood are from the absolute value of Jacobian determinant of the bijections. An ideal bijection in NFs should be:

- invertible in all cases
- cheap to invert
- have an easy to compute Jacobian determinant
- expressive enough to acquire the target distribution.

For evaluation, bits per dimension (BPD) is used. BPD is defined as

$$bpd(\mathbf{x}) = \frac{\log p(\mathbf{x})}{W \times H \times C \times \log(2)} \tag{7}$$

where H, W and C stand for height, width and channels of the image.

4.2. Architectures

Several families of bijections exist with each having their own (dis-) advantages. Kobyzef *et al.* [5] provides extensive overview of NFs and their performance on the various benchmarks. In the following section we introduce basic flows and their limitations. We define coupling flows in more detail as they significantly improve on their predecessors.

4.2.1. Simple flows

Elementwise bijections are the most simple form of transformations in NFs. Let $h : \mathbb{R} \mapsto \mathbb{R}$ be a bijection of scalar value. Then given vector $\mathbf{x} \in \mathbb{R}$ we define a flow as

$$g(\mathbf{x}) = (h(x_1), h(x_2), ..., h(x_D))^T. \tag{8}$$

Here the Jacobian determinant is just the product of the absolute values of the derivative of h. A logical extension towards element-wise flows are linear mappings as

$$g(\mathbf{x}) = \underline{\mathbf{A}}\mathbf{x} + \mathbf{b}, \tag{9}$$

where the observation matrix and bias are $\underline{\mathbf{A}} \in \mathbb{R}^{DxD}$ and $\mathbf{b} \in \mathbb{R}^D$, respectively. Since the Jacobian determinant is $\det(\underline{\mathbf{A}})$, several restrictions can be placed on the observation matrix to make the computation of the Jacobian determinant more efficient, such as forcing it to be diagonal or triangular. Furthermore, *LU decomposition* on the observation matrix has shown to reduce computation complexity to $\mathcal{O}(D)$ and $\mathcal{O}(D^2)$ in the forward and backwards pass, respectively [31]. Element-wise and linear flows are not expressive enough to capture complex distributions. The output of these flows are always in the same family of distributions as the input. This is because the flows are not able to expand and contract densities. Planar flows and radial flows [32] are able to do this. The bijection of the planar flow can be written as

$$g(\mathbf{x}) = \mathbf{x} + \mathbf{u}h(\mathbf{w}^T\mathbf{x} + \mathbf{b}). \tag{10}$$

For radial flows, the expansion and contraction is around a particular point as

$$g(\mathbf{x}) = \mathbf{x} + \frac{\beta}{\alpha\left|\mathbf{x} - \mathbf{x}_0\right|}(\mathbf{x} - \mathbf{x}_0). \tag{11}$$

Both flows need many consecutive bijections in order to model complex distributions. This can get computationally expensive. Also, the invertibility of the flow holds for only specific conditions. Their practicality has mostly shown in variational inference, as is done in the cited publication.

4.2.2. Coupling Flows

Dinh *et al.* [33] introduced a method for highly expressive transformations referred to as *coupling* flows. Before passing through the bijection, the input $\mathbf{x} \in \mathbb{R}$ will be split into two

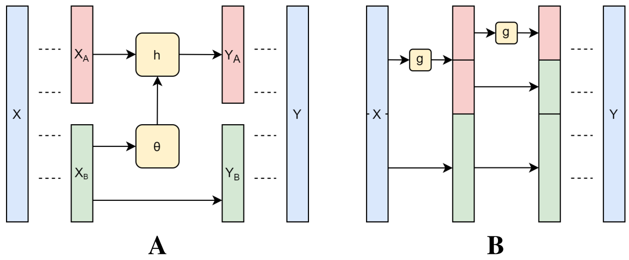


Fig. 2: (A) Conventional and (B) multi-scale coupling architecture in the normalizing direction

subspaces $\mathbf{x}_A \in \mathbb{R}^d$ and $\mathbf{x}_B \in \mathbb{R}^{D-d}$. The bijection function is $h(\bullet, \theta(\bullet)) : \mathbb{R}^d \mapsto \mathbb{R}^d$. With the *conditioner* θ being any arbitrary function, we can regard the coupling flow $g : \mathbb{R}^D \mapsto \mathbb{R}^D$ as functions

$$\begin{aligned} \mathbf{y}_A &= h(\mathbf{x}_A, \theta(\mathbf{x}_B)) \\ \mathbf{y}_B &= \mathbf{x}_B. \end{aligned} \tag{12}$$

For the invertibility of the total flow we require the conditioner to be invertible as well. The inverse can be stated as

$$\begin{aligned} \mathbf{x}_A &= h^{-1}(\mathbf{y}_A, \theta(\mathbf{x}_B)) \\ \mathbf{x}_B &= \mathbf{y}_B. \end{aligned} \tag{13}$$

If we consider the Jacobian

$$\underline{\mathbf{J}}_h = \begin{bmatrix} \frac{\partial \mathbf{y}_A}{\partial \mathbf{x}_A} & \frac{\partial \mathbf{y}_A}{\partial x_B} \\ 0 & \mathbb{I}_d \end{bmatrix} \tag{14}$$

it can be seen that the determinant equals $\det \underline{\mathbf{J}}_h = \frac{\partial \mathbf{y}_A}{\partial \mathbf{x}_B}$. The Jacobian is a triangular matrix in the case of a element-wise operating conditioner. See Figure 2 for an illustration of the conventional coupling flow. Most state-of-the-art models are affine coupling-based NFs such as Glow (section 4.3) and variants of it (Section 4.4). Other methods such as Non-linear squared, Continuous Mixed CDF, Spline, Neural Autoregressive, Sum-of-Square and Real-and-Discrete coupling flows [11, 21, 34–39] exist but has not seen equal success.

In regular coupling flows, the conditioner is always dependent on the split vector $\mathbf{x}_B$. It is possible to have the conditioner be independent of $\mathbf{x}_B$ to develop something called a *multi-scale* flow [40] (Figure 2). A *multi-scale* flow introduces dimension to a bijective transformation g in the generative direction. In the normalizing direction, the dimension of the vector that undergoes the bijection g decreases. This is analogous to some degree with filters in convolutional neural networks (CNN) when the coupling vector becomes smaller as we traverse deeper in the network where it enables the network to capture more multi-scale information. This works especially well with natural images of landscapes because these images often contain a general structure over the whole canvas.

4.3. Glow

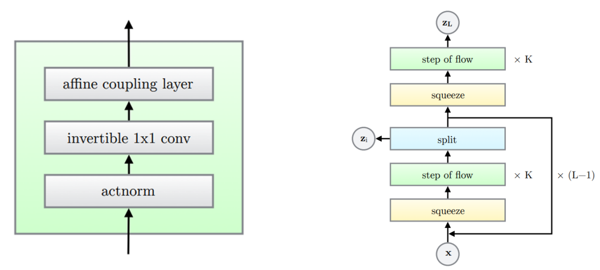
GLOW [8] is a coupling NF which has proven to obtain very competitive results compared to other generative models. GLOW builds on improvements introduced by NICE and RealNVP [33, 40]. RealNVP improved upon the NICE architecture by introducing data dependent shift and translation parameters $s, t : \mathbb{R}^d \mapsto \mathbb{R}^{D-d}$ to use in the coupling layer as

$$\begin{aligned} \mathbf{y}_A &= \mathbf{x}_A \cdot e^{s(\mathbf{x}_B)} + t(\mathbf{x}_B) \\ \mathbf{y}_B &= \mathbf{x}_B \end{aligned} \tag{15}$$

with the inverse then being

$$\begin{aligned} \mathbf{x}_A &= (\mathbf{y}_A - t(\mathbf{x}_B)) \cdot e^{-s(\mathbf{x}_B)} \\ \mathbf{x}_B &= \mathbf{y}_B. \end{aligned} \tag{16}$$

This is contrary to NICE, where instead $\mathbf{y}_B$ is directly passed through an arbitrary conditioner function and added to its counterpart $\mathbf{y}_A$. There are three key concepts that GLOW introduces to improve upon RealNVP. The first concept concerns the reduction of noise variance in the activations. Usually, deep models are trained with small batch sizes. This especially holds for training with high-dimensional datasets due to the available memory in the GPUs. To compensate for this, small minibatches are used. However, this introduces greater noise variation in the batch normalization step. The GLOW paper suggest adding an *actnorm* layer that initializes the model such that the activations have zero mean and unit variance and thus reducing the activation noise variation. After initialization, the parameters will adapt during training like the other parameters of the model. To achieve the goal of normalization, the actnorm layer introduces affine transformations using per-channel scale and bias parameters with function and inverse as



A A single flow step **B** Multi-scale GLOW architecture

Fig. 3: The GLOW architecture as proposed by Kingma *et al.* [8].

$$\mathbf{y}_{i,j} = \mathbf{s} \cdot \mathbf{x}_{i,j} + \mathbf{b} \tag{17a}$$

$$\mathbf{x}_{i,j} = \frac{\mathbf{y}_{i,j} - \mathbf{b}}{\mathbf{s}} \tag{17b}$$

and log determinant

$$h \cdot w \cdot \mathrm{sum}(\log|\mathbf{s}|). \tag{18}$$

Secondly, GLOW suggest using invertible 1x1 convolutional layers with equal input and output channels (which is a generalization of a permutation) instead of reversing the orders of the channels. Now, the permutations can be learned during training. The permutations are done to ensure that every dimension will have the ability to be altered. In combination with *LU decomposition* the computational costs of the log Jacobian determinant can be reduced from $\mathcal{O}(c^3)$ to $\mathcal{O}(c^2)$ as

$$\log|\det(\underline{\mathbf{W}})| = \text{sum}(\log|\mathbf{s}|). \tag{19}$$

The final improvement of GLOW is concerning the multi-scale design approach. The GLOW architecture (Figure 3) consist of a squeezing function, applying the above mentioned flow for K number of times and then a splitting layer. These three steps are repeated L times and are followed by a final squeeze and flow step. The flow itself first uses the actnorm layer then the invertible 1x1 convolution and finally the affine coupling layer. The paper compares GLOW against it predecessor and concludes that it outperforms them by great margin on public benchmark datasets such as CIFAR10, ImageNet and LSUN in both image generation and density estimation.

4.4. Wavelet Flow

In a recent publication by Yu *et al.* [10], they introduce Wavelet Flow, a flow architecture for fast training of high resolution NFs. They use NFs in combination with Haar wavelets in a multiresolution-like architecture i.e. cascaded discrete wavelet transforms (DWT). In a similar manner, it utilizes these Haar wavelets to downsample and obtains detail coefficients from the input image. The architecture consists of multiple levels, where at each level the DC component of the Haar basis function is used for the low pass component h_l, and the vertical, horizontal and diagonal filters are used to extract high frequency detail coefficients denoted as h_d. See Figure 4 for a detailed depiction of the transformation from image $\mathbf{I}$ to the Haar coefficients $\mathbf{D}$. The first filter is the DC component i.e. the low pass filter of the wavelet transform. During training, a NF that closely resembles the GLOW architecture is used to model h_d conditioned on h_l. Here, the multi-scale squeeze/split architecture is not used since the wavelets are used to decompose the image at multiple scales. The total multilayer architecture is give in Figure 5. This architecture enables both density estimation and sampling with the additional benefits of:

- Possibility to do image generation and likelihood estimation at each scale
- Training and likelihood estimation at each scale *independently*. However, sampling is still sequential.
- Data driven upsampling of lower resolution images
- Up to 15x faster training time
- Competitive with other models in BPD evaluations

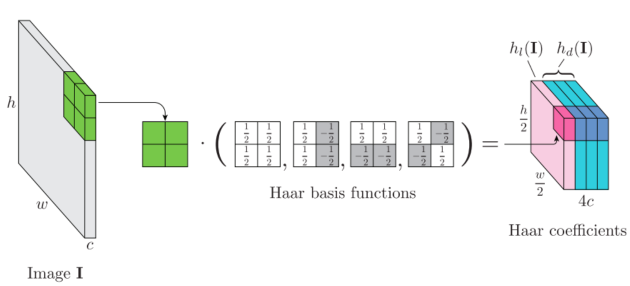


Fig. 4: Example transformation of an image **I** in the Wavelet Flow architecture with the four Haar basis kernels [10].

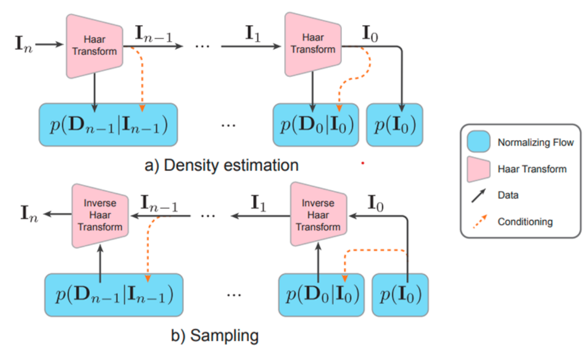


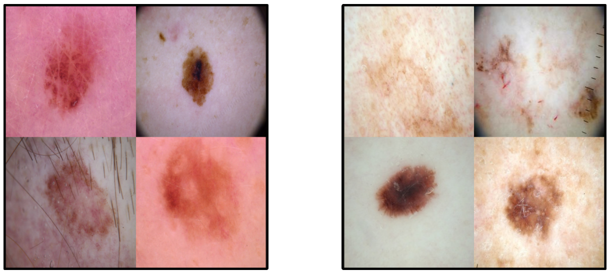
Fig. 5: The Wavelet Flow [10] architecture in the normalizing (a) as well as the generative (b) direction. The ith level of this model consists of a Haar transform and a Normalizing Flow. The lowest level consists of only a normalizing flow.

The summed losses of each individual layer results in the total loss of the Wavelet Flow as

$$\log p(\mathbf{I}) = \log p_0(\mathbf{I_0}) + \sum_{i=0}^{n-1} \log p(\mathbf{D_i} \mid \mathbf{I_i}). \tag{20}$$

5. DATASET

In this research a processed version of the ISIC dataset is used. The dataset consists of 1800 images (224x224) of both benign as well as malignant melanoma. Some example images are depicted in Figure 6. The train and test set of the benign images consists of 1440 and 360 images, respectively.



A Benign **B** Malignant

Fig. 6: Example images from the adjusted ISIC dataset of melanoma used for OOD detection.

6. RESULTS

We present our results after experimenting with GLOW and Wavelet Flow. GLOW with parameter settings $L = 3$ and $K = 32$ is trained on downsampled 64x64 images of the ISIC dataset on a NVIDIA RTX 2080 SUPER. Initially, the model is trained with a batch size of 16, learning rate of 5e-4, weight decay of 5e-5 and 256 hidden channels in the convolutional neural network (CNN) that is responsible for estimating parameters s and t. For masking in the coupling layer, we use the channel-wise method. We present our results after 1000 epochs in Figure 7 with its corresponding Receiver Operating Characteristic (ROC) curve in Figure 8. Furthermore, we experimented with channel-wise, checkerboard and cycle masking from Ren et al. [2020]. Cycling is only done for one iteration due to limited dedicated GPU memory. For the shift and translate parameters we use a single hidden layer convolutional neural network (CNN) with ReLU activation except on the output layer. We show the hyperparameter settings in Table 1.

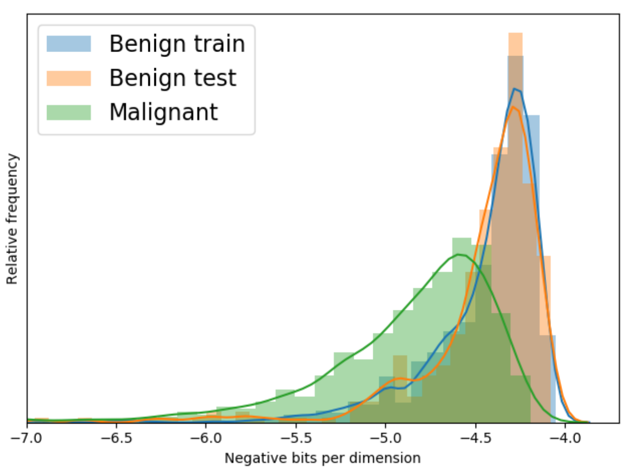


Fig. 7: Likelihood estimation of GLOW masked channel-wise and trained on ISIC benign images.

We compare in dataset training data with out of dataset

	channel-wise	**checker**	**cycle-1**
epochs	1000	350	500
batch size	16	8	8
learning rate	0.0005	0.0001	0.0001
weight decay	0.001	0.001	0.001
hidden channels conv.	256	512	128

Table 1: Hyperparameter settings of the three different masking models.

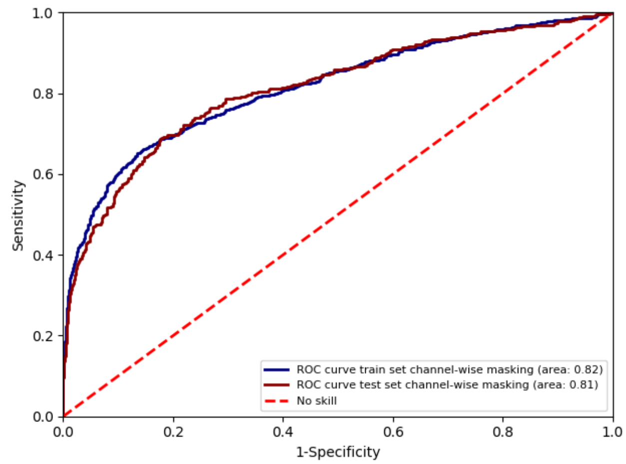
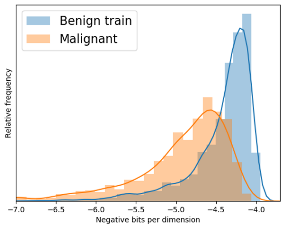


Fig. 8: ROC curves of channel-wise masked GLOW trained on Benign images. The train and test distributions are compared to the malignant out-of-dataset images.

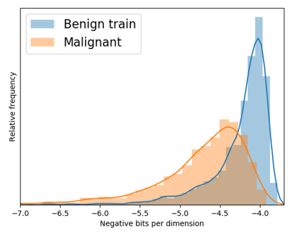
data in Figure 9 for the distributions with channel-wise, checkerboard and 1-cycle masking. We find their ROCs to be 0.81, 0.82 and 0.81.

During training and preparation of Wavelet Flow, our computation power and a time constraint limited the number of training iterations for each layer. For full architectural details we suggest reading the original paper. We use the same hyperparameter settings as is given in their *GitHub* repository [41] except for the fact that we do not randomly crop the images for data augmentation in the final levels and that the training batch size of the final level is 16 instead of 64 due to computational limitations. The number of iterations per layer from 0 to 6 are 1630000, 1130000, 400000, 530000, 510000, 940000 and 840000, respectively. Figure 10 depicts the distribution of the train, test and OOD images. The ROC curve of the test and OOD images is shown in Figure 11. The respective ROC curves of the different models are given in Figure 8.

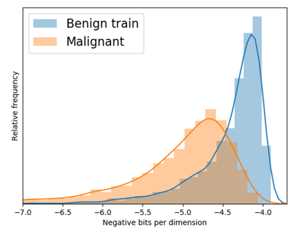
With the frequency dependent analyses, the first two levels are not considered since its contribution to the BPD is insignificant (>0.02) (Appendix B). The frequency dependent analysis for the benign test images is given for regular and



A channel-wise

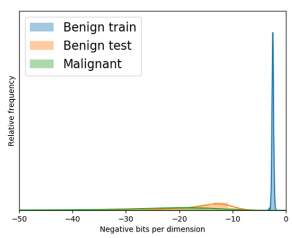


B checkerboard

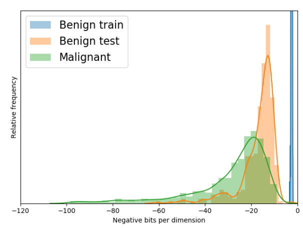


C 1-cycle

Fig. 9: Density estimation of in distribution benign train and out-of-dataset malignant images for multiple masking techniques.



A Full view



B Enhanced

Fig. 10: Likelihood distribution of Wavelet Flow with the ISIC dataset

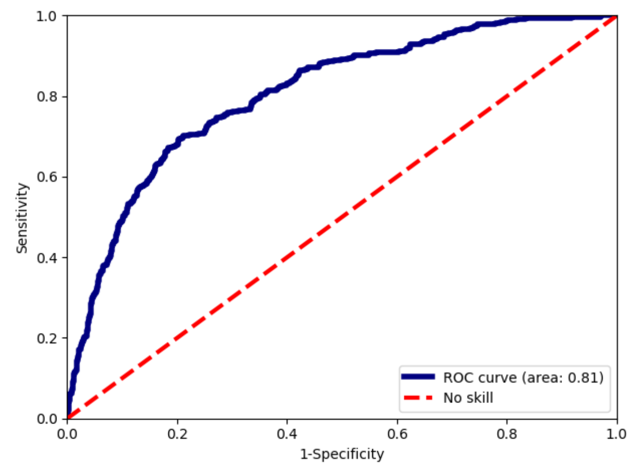


Fig. 11: ROC curve of Wavelet Flow with Benign images from the test set and OOD malignant images.

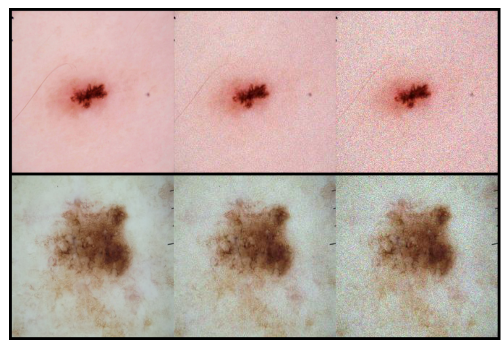


Fig. 12: Benign and malignant images with additive Gaussian noise with variance 0, 0.05^2 and 0.1^2

noisy images. The distribution of images with multiplicative Gaussian noise with variance of 0.05^2 and 0.1^2 (see Figure 12) is compared against the original images. We show the same for the malignant OOD data and present the results in Figures 14, 15 and 16. We also take the average Power Spectral Density (PSD) (Figure 13) of all benign images to identify which level of the Wavelet Flow should theoretically shift the most. The frequency range which the individual levels cover are bounded by vertical lines from left to right. For completeness, we also compared the generative capabilities of both GLOW and Wavelet Flow (see appendix C).

7. DISCUSSION

We employed GLOW and Wavelet Flow for OOD detection on the ISIC melanoma dataset and find clear differences in performance. As shown in Figure 7, GLOW generalized well given the large overlap between the train and test distribution. We draw an ROC curve against the malignant distribution that yields an area of 0.82 and 0.81 for the train and test set, respectively. Our initial hypothesis was that the BPD of the train images will be much less than what is actually observed in the results of our experiments as this is found in literature. We also hypothesized that the train set ROC area would have been closer to 1 while the test distribution being less than that (shifted more to the right). This suggests that GLOW generalizes well on the benign set but still finds large similarities with the malignant images and as such, fails to accurately detect them as OOD. Furthermore, we implemented channel-wise, checkerboard and 1-cycle masking and have shown the likelihood distributions of the benign train and malignant images. This was to find if using masking strategies can lead to better OOD performance. Using the different masking methods the ROC curves of the distributions are 0.81, 0.82 and 0.81, respectively. We do not see any significant change in the ROC curve area by introducing the checkerboard or cycle mask in the coupling layers. The results of our experiments confirm the lack of OOD detection capabilities of GLOW and show that different masking strategies are not the direction to move towards for improvement of it. We believe it is because GLOW is unable to distinguish between semantic and

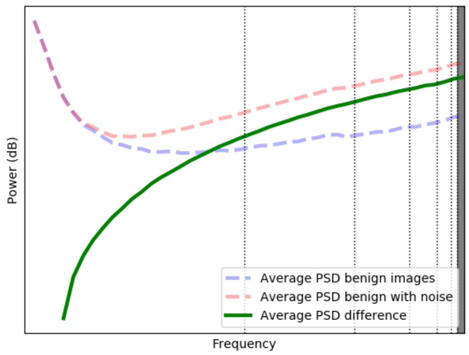


Fig. 13: Average PSD of normal and noisy benign images with multiplicative Gaussian noise of 0.1^2 variance.

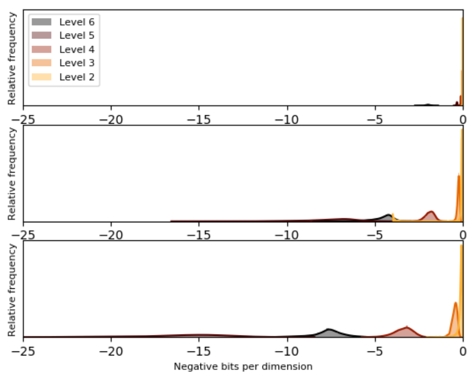


Fig. 14: Per level likelihood estimation of the benign train images with multiplicative Gaussian noise of 0, 0.05^2 and 0.1^2 variance from top to bottom. Excluded are level 0 and 1 because their contribution to the total loss is insignificant.

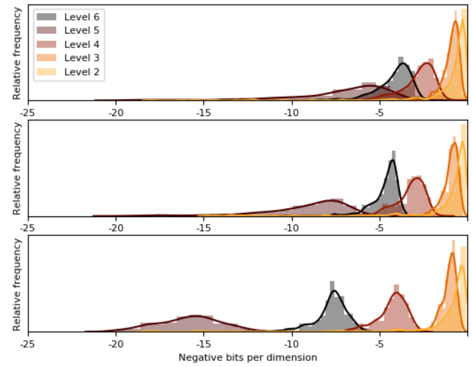


Fig. 15: Per level likelihood estimation of the benign test images with multiplicative Gaussian noise of 0, 0.05^2 and 0.1^2 variance from top to bottom. Excluded are level 0 and 1 because their contribution to the total loss is insignificant.

background information as also suggested by previous work. Although these experiments can be improved by having access to better computational power, we do not suggest this approach should be further research as we believe augmentation methods or architectural modifications could potentially lead to more improvement of OOD detection in GLOW. A modified version of GLOW, Wavelet Flow has recently reported in great improvement over regular GLOW in terms of BPD on the training set. However, from our findings the ROC curve of the test and out-of-dataset data has an area of 0.8 which is similar to GLOW. Furthermore, the training set distribution is very far from the test set distribution. This implies that the model has not generalized well since the train and test images are from the same class. We think enabling random cropping as suggested by the paper can serve as a good regularizer during training. We do not think simply training for more iterations will improve OOD detection. Given that the BPD on the train set is much lower, training Wavelet Flow is much more efficient on high dimensional data and the fact that the ROC curve is similar to GLOW, one can suggest that learning data distributions per frequency component might potentially result in better OOD performance on all datasets compared to GLOW. It might be even more beneficial to consider only particular frequency components since not every layer contributes to the loss equally. In the Wavelet Flow implementation, we investigated the likelihood distributions of each wavelet decomposition level independently. As expected, we see that when we add multiplicative Gaussian noise to the benign test images, the higher level curves shift to increasing BPD while lower level curves are almost unaffected. We observe that the likelihood distribution of level 5 shifted the most with increasing noise. The average PSD of the noisy benign images suggests that the sixth level should be subjective to the most shift. This could be the result of the limited time and resource available during the training of the final layer and would then also explain the blurriness of our samples (appendix C) since the highest level 6 is responsible for the highest image coefficients. Thus, longer training (potential better convergence) and more consistency between the number of iterations between each layer should translate into more accurate OOD detection of noise and higher quality sampled images. This is in line with literature, since the original paper presents the results with levels trained for up to 5 times more iterations on several datasets. We see for the lower levels (appendix B) that only level 1 is affected and as a consequence slightly shifts and flattens while level 0 stays the same in the training set. For the test and out-of-dataset images there is no change to be observed. An additional interesting investigation would be to evaluate the effect of the variance in the higher frequency ranges and the eventual likelihood dis-

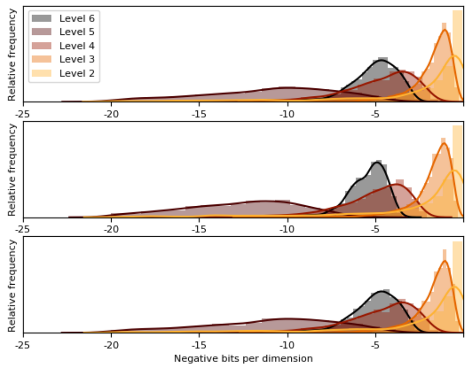


Fig. 16: Per level likelihood estimation of the malignant OOD images with multiplicative Gaussian noise of 0, 0.05^2 and 0.1^2 variance from top to bottom. Excluded are level 0 and 1 because their contribution to the total loss is insignificant.

tributions at each of the individual Wavelet Flow levels. The degree of correlation between these two statistics could potentially serve as a benchmark for frequency dependent likelihood estimation in images. When the same approach was taken on the malignant data, we found that noise did not impact likelihood estimation of it. Ideally, we would like the likelihood of malignant images to decrease as well with increasing noise. This would then imply that the model has learned the semantics of melanoma very well, understanding that malignant images are not noise or benign but something in between. Our suggestions made for improvement of the combined levels also hold for the frequency dependent OOD detection and encourage for it be explored in future work. We believe that Wavelet Flow holds promise for OOD detection but needs further improvement. Namely, the test should be closer to the train set distribution. Given the sub-optimal performance of GLOW in our findings, we do not think that the NF used in Wavelet Flow is a big contributor to the improvements made in OOD detection. We see many areas that can be researched to further understand which underlying mechanics enable Wavelet Flow to perform better than GLOW in BPD evaluation and how to improve its OOD detection capability. As mentioned by the authors of Wavelet Flow, Haar wavelets were used but other wavelets should be investigated as well. Also, a different NF architecture, for estimating the detail coefficients, that performs similar or better than GLOW in BPD evaluation can be implemented. If similar or better performance is found, it confirms our hypothesis that the improvements are not due to the choice of NF architecture but, in fact, as a result of the usages of wavelet transformations. If it results in worse results then it means that the decision of choosing a GLOW inspired NF was essential for the performance of the model. Furthermore, using a more state-of-the-art NF architectures might improve OOD detection even further. From literature we know that OOD detection often fails due to the lack of semantic capture and the exploitation of background components of the image. We believe that employing a method that enables the NF model to distinguish between semantic and background features will enable OOD detection using NF the most. We have found an interesting direction to head for towards tractable likelihood estimation for medical images by implementing frequency based OOD detection for the first time to the best of our knowledge and we mentioned multiple areas for further exploration to gain a better understanding of OOD detection on frequency basis.

8. CONCLUSION

In this study we apply GLOW and Wavelet Flow for the task of OOD detection on the ISIC dataset. In line with similar findings, we confirm that GLOW lacks ability to reliably detect out-of-distribution data. As further research suggests, we attempt to improve the OOD detection capabilities by employing different masking strategies in the coupling layers of GLOW. These improvements yield little-to-no change in the ability to detect OOD data using GLOW. We further implemented Wavelet Flow which was originally introduced due to the computational efficiency and competitiveness with GLOW in bits per dimension. We have observed a big improvement in BPD evaluation and find similar OOD detection performance. Furthermore, we hypothesize that its ability to capture semantics is better than that of GLOW and therefore see much potential for this model. We have looked at the density and likelihood estimation capabilities of each individual layer in Wavelet Flow and analyzed the effect of random noise on the distributions in order to better understand how well the model generalized. We have done OOD detection on medical images but the applications for this technique is not limited to medical imaging but can be used in other domains as well such as health monitoring or video surveillance. We find NFs in combination with Wavelet decomposition to be very promising given the demand for explainable and interpretable machine learning and suggest further research in to Normalizing Flows with wavelets for density estimation.

References

[1] D. P. Kingma and M. Welling, *Auto-encoding variational bayes*, 2014. arXiv: 1312.6114 [stat.ML].

[2] A. van den Oord, N. Kalchbrenner, and K. Kavukcuoglu, *Pixel recurrent neural networks*, 2016. arXiv: 1601.06759 [cs.CV].

[3] M. Arjovsky and L. Bottou, *Towards principled methods for training generative adversarial networks*, 2017. arXiv: 1701.04862 [stat.ML].

[4] K. Roth, A. Lucchi, S. Nowozin, and T. Hofmann, *Stabilizing training of generative adversarial networks through regularization*, 2017. arXiv: 1705.09367 [cs.LG].

[5] V. Bogachev, A. Kolesnikov, and K. Medvedev, "Triangular transformations of measures," *Sbornik: Mathematics*, vol. 196, p. 309, Oct. 2007. DOI: 10.1070/SM2005v196n03ABEH000882.

[6] K. Medvedev, "Certain properties of triangular transformations of measures," *Theory of Stochastic Processes*, vol. 1, Jan. 2008.

[7] C. Villani, *Topics in Optimal Transportation*. 2003, p. 58.

[8] D. P. Kingma and P. Dhariwal, *Glow: Generative flow with invertible 1x1 convolutions*, 2018. arXiv: 1807.03039 [stat.ML].

[9] R. T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud, *Neural ordinary differential equations*, 2019. arXiv: 1806.07366 [cs.LG].

[10] J. J. Yu, K. G. Derpanis, and M. A. Brubaker, *Wavelet flow: Fast training of high resolution normalizing flows*, 2020. arXiv: 2010.13821 [cs.CV].

[11] J. Ho, X. Chen, A. Srinivas, Y. Duan, and P. Abbeel, *Flow++: Improving flow-based generative models with variational dequantization and architecture design*, 2019. arXiv: 1902.00275 [cs.LG].

[12] A. Abdelhamed, M. A. Brubaker, and M. S. Brown, *Noise flow: Noise modeling with conditional normalizing flows*, 2019. arXiv: 1908.08453 [cs.CV].

[13] M. Kumar, M. Babaeizadeh, D. Erhan, C. Finn, S. Levine, L. Dinh, and D. Kingma, *Videoflow: A conditional flow-based model for stochastic video generation*, 2020. arXiv: 1903.01434 [cs.CV].

[14] P. Esling, N. Masuda, A. Bardet, R. Despres, and A. Chemla–Romeu-Santos, *Universal audio synthesizer control with normalizing flows*, 2019. arXiv: 1907.00971 [cs.LG].

[15] S. Kim, S.-g. Lee, J. Song, J. Kim, and S. Yoon, *Flowavenet : A generative flow for raw audio*, 2019. arXiv: 1811.02155 [cs.SD].

[16] R. Prenger, R. Valle, and B. Catanzaro, *Waveglow: A flow-based generative network for speech synthesis*, 2018. arXiv: 1811.00002 [cs.SD].

[17] K. Madhawa, K. Ishiguro, K. Nakago, and M. Abe, *Graphnvp: An invertible flow model for generating molecular graphs*, 2019. arXiv: 1905.11600 [stat.ML].

[18] B. Mazoure, T. Doan, A. Durand, R. D. Hjelm, and J. Pineau, *Leveraging exploration in off-policy algorithms via normalizing flows*, 2019. arXiv: 1905.06893 [cs.LG].

[19] P. N. Ward, A. Smofsky, and A. J. Bose, *Improving exploration in soft-actor-critic with normalizing flows policies*, 2019. arXiv: 1906.02771 [cs.LG].

[20] A. Touati, H. Satija, J. Romoff, J. Pineau, and P. Vincent, *Randomized value functions via multiplicative normalizing flows*, 2019. arXiv: 1806.02315 [cs.LG].

[21] T. Müller, B. McWilliams, F. Rousselle, M. Gross, and J. Novák, *Neural importance sampling*, 2019. arXiv: 1808.03856 [cs.LG].

[22] G. Kanwar, M. S. Albergo, D. Boyda, K. Cranmer, D. C. Hackett, S. Racanière, D. J. Rezende, and P. E. Shanahan, "Equivariant flow-based sampling for lattice gauge theory," *Physical Review Letters*, vol. 125, no. 12, Sep. 2020, ISSN: 1079-7114. DOI: 10.1103/physrevlett.125.121601. [Online]. Available: http://dx.doi.org/10.1103/PhysRevLett.125.121601.

[23] J. Köhler, L. Klein, and F. Noé, *Equivariant flows: Sampling configurations for multi-body systems with symmetric energies*, 2019. arXiv: 1910.00753 [stat.ML].

[24] F. Noé, S. Olsson, J. Köhler, and H. Wu, *Boltzmann generators – sampling equilibrium states of many-body systems with deep learning*, 2019. arXiv: 1812.01729 [stat.ML].

[25] P. Wirnsberger, A. J. Ballard, G. Papamakarios, S. Abercrombie, S. Racanière, A. Pritzel, D. Jimenez Rezende, and C. Blundell, "Targeted free energy estimation via learned mappings," *The Journal of Chemical Physics*, vol. 153, no. 14, p. 144 112, Oct. 2020, ISSN: 1089-7690. DOI: 10.1063/5.0018903. [Online]. Available: http://dx.doi.org/10.1063/5.0018903.

[26] K. W. K. Wong, G. Contardo, and S. Ho, "Gravitational-wave population inference with deep flow-based generative network," *Physical Review D*, vol. 101, no. 12, Jun. 2020, ISSN: 2470-0029. DOI: 10.1103/physrevd.101.123005. [Online]. Available: http://dx.doi.org/10.1103/PhysRevD.101.123005.

[27] *The international skin imaging collaboration archive*, http://https://www.isic-archive.com/, Accessed: 2021-01-18.

[28] P. Kirichenko, P. Izmailov, and A. G. Wilson, *Why normalizing flows fail to detect out-of-distribution data*, 2020. arXiv: 2006.08545 [stat.ML].

[29] J. Ren, P. J. Liu, E. Fertig, J. Snoek, R. Poplin, M. A. DePristo, J. V. Dillon, and B. Lakshminarayanan, *Likelihood ratios for out-of-distribution detection*, 2019. arXiv: 1906.02845 [stat.ML].

[30] *Normalizing flows*, https://janosh.dev/blog/normalizing-flows, Accessed: 2021-01-19.

[31] J. M. Tomczak and M. Welling, *Improving variational auto-encoders using householder flow*, 2017. arXiv: 1611.09630 [cs.LG].

[32] D. J. Rezende and S. Mohamed, *Variational inference with normalizing flows*, 2016. arXiv: 1505.05770 [stat.ML].

[33] L. Dinh, D. Krueger, and Y. Bengio, *Nice: Non-linear independent components estimation*, 2015. arXiv: 1410.8516 [cs.LG].

[34] Z. M. Ziegler and A. M. Rush, *Latent normalizing flows for discrete sequences*, 2019. arXiv: 1901.10548 [stat.ML].

[35] C. Durkan, A. Bekasov, I. Murray, and G. Papamakarios, *Cubic-spline flows*, 2019. arXiv: 1906.02145 [stat.ML].

[36] ——, *Neural spline flows*, 2019. arXiv: 1906.04032 [stat.ML].

[37] C.-W. Huang, D. Krueger, A. Lacoste, and A. Courville, *Neural autoregressive flows*, J. Dy and A. Krause, Eds., Stockholmsmässan, Stockholm Sweden, Jul. 2018. [Online]. Available: http://proceedings.mlr.press/v80/huang18d.html.

[38] P. Jaini, K. A. Selby, and Y. Yu, *Sum-of-squares polynomial flow*, 2019. arXiv: 1905.02325 [cs.LG].

[39] L. Dinh, J. Sohl-Dickstein, H. Larochelle, and R. Pascanu, *A rad approach to deep mixture models*, 2020. arXiv: 1903.07714 [cs.LG].

[40] L. Dinh, J. Sohl-Dickstein, and S. Bengio, *Density estimation using real nvp*, 2017. arXiv: 1605.08803 [cs.LG].

[41] *Wavelet flow github repository*, https://github.com/YorkUCVIL/Wavelet-Flow/, Accessed: 2021-01-22.

Appendix

A. NORMALIZING FLOWS DERIVATION

A bijection between distributions should adhere to the law of total probability as

$$\int p(x)dx = \int p(y)dy. \tag{21}$$

The absolute change in volume is only relevant. In this fashion we can represent a sample in Y in terms of a sample in X as

$$p(y) = p(x)\left|\frac{dx}{dy}\right|. \tag{22}$$

We now observe two variables in this transformation. We have a Uniform(0,1) distribution being linearly transformed in to any arbitrary parallelogram without shift with the matrix.

$$T = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \tag{23}$$

Now, the change in absolute volume of the parallelogram is

$$|ad - cb| = |\det(T)|. \tag{24}$$

Now for a multidimensional non-linear transformation, one approximates this with infinite, infinitesimally parallelograms to track the global change in volume. This local-linear change approximation is in fact the Jacobian determinant. Given that $y = f(x)$ thus $x = f^{-1}(y)$ we can write

$$p_Y(y) = p_X(f^{-1}(y))\left|\det\frac{\partial f^{-1}}{\partial x}\right|. \tag{25}$$

Based on a chain of these operations as has been shown in Figure 1 one can write

$$\mathbf{z}_{i-1} \sim p_{i-1}(\mathbf{z}_{i-1}) \tag{26a}$$
$$\mathbf{z}_i = f_i(\mathbf{z}_{i-1}), \mathbf{z}_{i-1} = f_i^{-1}(\mathbf{z}_i) \tag{26b}$$

and thus estimate the probability as

$$p_i(\mathbf{z}_i) = p_{i-1}\left(f_i^{-1}(\mathbf{z}_i)\right)\left|\det\frac{df_i^{-1}}{d\mathbf{z}_i}\right|. \tag{27}$$

The inverse function theorem and the property $\det(\underline{\mathbf{M}}^{-1}) = \det(\underline{\mathbf{M}})^{-1}$ for invertible matrices $\underline{\mathbf{M}}$ allows us to rewrite the transformation as

$$p_i(\mathbf{z}_i) = p_{i-1}\left(f_i^{-1}(\mathbf{z}_i)\right)\left|\det\frac{df_i^{-1}}{d\mathbf{z}_i}\right| \tag{28a}$$
$$= p_{i-1}(\mathbf{z}_{i-1})\left|\det\left(\frac{df_i}{d\mathbf{z}_{i-1}}\right)^{-1}\right| \tag{28b}$$
$$= p_{i-1}(\mathbf{z}_{i-1})\left|\det\frac{df_i}{d\mathbf{z}_{i-1}}\right|^{-1} \tag{28c}$$

and we can state the likelihood of a transformation as

$$\log p_i(\mathbf{z}_i) = \log p_{i-1}(\mathbf{z}_{i-1}) - \log\left|\det\frac{df_i}{d\mathbf{z}_{i-1}}\right|. \tag{29}$$

Given the chain of the transformations

$$\mathbf{x} = \mathbf{z}_K = f_K \circ f_{K-1} \circ \cdots f_1(\mathbf{z}_0) \tag{30}$$

and the log-likelihood of the observed data $\mathbf{x}$

$$\begin{aligned}
\log p(\mathbf{x}) = \log p_K\left(\mathbf{z}_K\right) &= \log p_{K-1}\left(\mathbf{z}_{K-1}\right) - \log\left|\det \frac{df_K}{d\mathbf{z}_{K-1}}\right| && \text{(31a)} \\
&= \log p_{K-2}\left(\mathbf{z}_{K-2}\right) - \log\left|\det \frac{df_{K-1}}{d\mathbf{z}_{K-2}}\right| - \log\left|\det \frac{df_K}{d\mathbf{z}_{K-1}}\right| && \text{(31b)} \\
&= \ldots
\end{aligned}$$

or concisely written as

$$\log p(\mathbf{x}) = \log p_0\left(\mathbf{z}_0\right) - \sum_{i=1}^{K} \log\left|\det \frac{df_i}{d\mathbf{z}_{i-1}}\right| \tag{32}$$

B. LOW LEVEL FREQUENCY DEPENDENT DISTRIBUTIONS

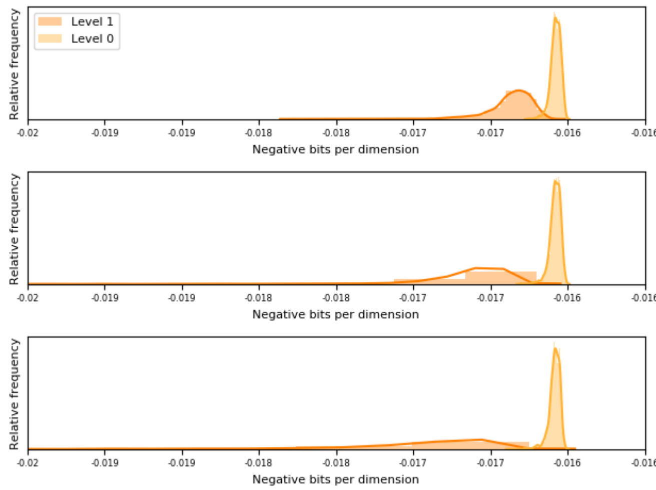


Fig. 17: Low level likelihood estimations of the malignant out-of-dataset images with multiplicative Gaussian noise of 0, 0.05^2 and 0.1^2 variance from top to bottom.

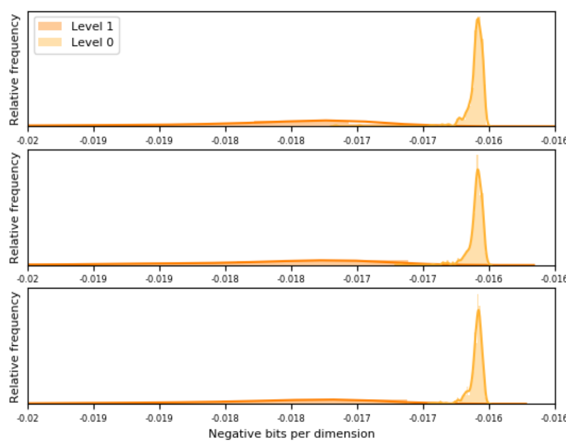


Fig. 18: Low level likelihood estimations of the malignant out-of-dataset images with multiplicative Gaussian noise of 0, 0.05^2 and 0.1^2 variance from top to bottom.

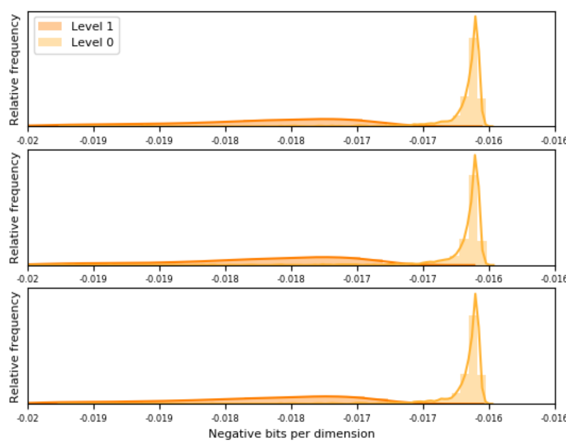
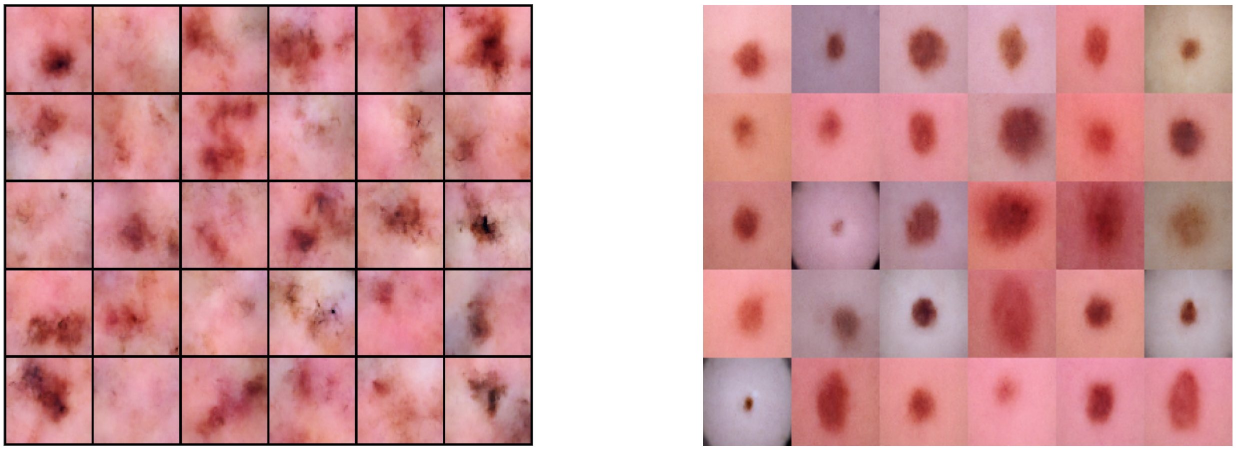


Fig. 19: Low level likelihood estimations of the malignant out-of-dataset images with multiplicative Gaussian noise of 0, 0.05^2 and 0.1^2 variance from top to bottom.

C. SAMPLES FROM GLOW AND WAVELET FLOW



A GLOW

B Wavelet Flow

Fig. 20: Generated images representing benign melanoma