# Anomaly detection for skin lesion images using replicator neural networks

Fabrizio Nunnari[1 [0000−0002−1596−4043]], Hasan Md Tusfiqur Alam[1 [0000−0003−1479−7690]], and Daniel Sonntag[1,2 [0000−0002−8857−8709]]

[1] German Research Center for Artificial Intelligence (DFKI)
Saarland Informatics Campus D3 2
{fabrizio.nunnari,hasan_md_tusfiqur.alam,daniel.sonntag}@dfki.de
[2] Oldenburg University

**Abstract.** This paper presents an investigation on the task of anomaly detection for images of skin lesions. The goal is to provide a decision support system with an extra filtering layer to inform users if a classifier should not be used for a given sample. We tested anomaly detectors based on autoencoders and three discrimination methods: feature vector distance, replicator neural networks, and support vector data description fine-tuning. Results show that neural-based detectors can perfectly discriminate between skin lesions and open world images, but class discrimination cannot easily be accomplished and requires further investigation.

**Keywords:** skin cancer · anomaly detection · autoencoders · replicator neural networks · SVDD

## 1 Introduction

Clinical decision support systems (CDSS) for skin cancer detection, based on deep neural networks, have proven to be effective and in some cases surpass human performances [9,1,14,2].

To foster research in this direction, from 2016 on, the International Society for Digital Imaging of the Skin[3] organizes the ISIC[4] challenge for the development of computer vision systems supporting clinical decision in the field of skin lesions. The tasks considered in the past editions include classification [6,7], lesion segmentation, and feature extraction [5].

The 2019 edition[5] contained, as an implicit task, anomaly detection. The training dataset provided for the ISIC 2019 challenge included images pertaining to 8 classes of skin lesions. However, the test dataset contained also images pertaining to none of those categories, named the *unknown* (UNK) class. In other words, as the training set was providing material for 8 known classes, the test phase asked for a classification into 9 classes (see figure 1).

One approach to solve this problem would be to inject random pictures of other known skin pathologies into the training, or random pictures from the real world, and

[3] https://isdis.org/
[4] https://www.isic-archive.com/
[5] https://challenge.isic-archive.com/landing/2019

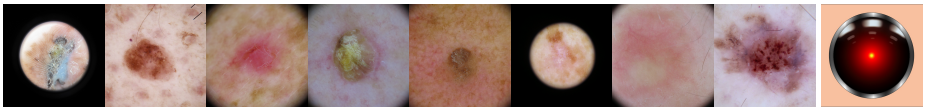

**Fig. 1.** A sample for each of the nine classes in the ISIC 2019 dataset. From left to right: Melanoma, Melanocytic nevus, Basal cell carcinoma, Actinic keratosis, Benign keratosis, Dermatofibroma, Vascular lesion, and Squamous cell carcinoma, followed by a sample of the test set clearly belonging to the UNK class.

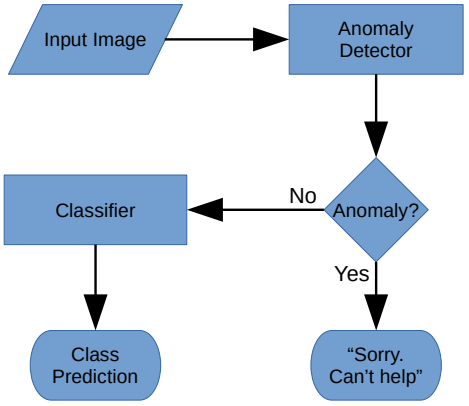


**Fig. 2.** The classification chain based on the concatenation of an anomaly detector and a standard classifier.

mark them as UNK. However, the choice of such extra training images would be arbitrary and possibly not reflect the selection criteria used for the preparation of the test set.

An alternative approach would be to chain two models: the first dedicated to performing *anomaly detection*, followed by a classification model (see figure 2). Hence, a new sample would be first filtered by the anomaly detector. If detected as not-pertaining to any of the 8 classes, it would be marked as UNK, or continue through the classification model otherwise.

In general, anomaly detection, in our approach also known as 1-class classification, is the task of discriminating if a given sample pertains to the same distribution of a reference set. Such a pre-filtering strategy would help circumventing the critical limitation of classifiers, which are unable to output choices beyond the closed-list of classes provided at training time.

Here, the purpose would be to enhance clinical decision support systems to provide answers like “I cannot take a decision: this system was not prepared for this kind of input image”. Another possible application would be of an automatic filtering during the automated collection of images, for example, from the web.

Despite the potential advantages of anomaly detection in the field of skin lesions, from the results of the ISIC 2019 challenge, it emerges that none of the participants was

able to reach satisfactory specificity for the UNK class, with some of the participants ignoring the problem as a whole.

Hence, in this paper, we report on a post-challenge investigation that we conducted to measure the effectiveness of deep-learning-based anomaly detection on skin lesion images.

From a survey on the ISIC 2019 reports, it looks like all of the participants addressed the problem of anomaly detection through a statistical analysis of the softmax output of their classifiers. The work we present here seems to be the first one to tackle the problem of anomaly detection using deep neural networks configured as autoencoders. Our results do not show major gains in classification performance, i.e., discrimination methods based on feature vector distance, Replicator Neural Networks, and Support Vector Data Description do not perform as good as they do on other domains. Nevertheless, we contribute with several hints when dealing with anomaly detection for (skin lesion) images and an investigation methodology that could be used as starting point for future work in this field or related imaging task.

## 2 Related Work

Anomaly detection (aka 1-class classification, outlier detection, novelty detection) refers to the task of discriminating between samples pertaining to a reference *target* distribution and samples coming from whatever kind of other distribution, and identify them as *anomalies*, or *outliers*. See Chandola et al. [3] for a comprehensive review.

Anomaly detection presents distinct problem complexities compared to the majority of analytical and learning problems. Pang et al. [20] discuss some unique problem complexities like unknowness, heterogeneous anomaly classes, rarity and class imbalance and the diverseness in the types of anomaly that results in largely unsolved challenges.

The One-Class SVM [22] is a popular solution for anomaly detection based on the SVM method. The drawback is that it doesn't scale with the number of features, and is thus not applicable to CNN-driven image classification, where the number of features describing a sample before the softmax stage is above 1000.

When using CNN-based classificators, an approach that reaches state-of-the-art performance comes from Lee et al. [17], who proposed a method for detecting out of distribution (OOD) samples where class conditional Gaussian distributions with respect to the features of the deep models are obtained under Gaussian discriminant analysis. Then, the confidence score are obtained by using the Mahalanobis distance metric. Their method considers both the final softmax scores and the intermediate features of internal hidden layers.

In the context of dermatoscopy, Li et al. [18] proposed a non-parametric deep isolation forest (DeepIF) as a modification of the method from Lee et al. [17] in order to take into account the huge intra-class diversity of skin disease images. With this approach they reach an average 0.71 ROC on intra-class discrimination on the HAM10000 dataset [24].

As a new approach, the tests reported in this paper use Replicator Neural Networks [12], which are based on the training of an autoencoder on the target set and a measurement of the reconstruction error between an input image and the encoded-decoded

output image. The hypothesis is that an autoencoder "specialized" in compressing and decompressing a certain type of images will show a higher reconstruction error if applied to images never used during the training phase.

Additionally, we test the effectiveness of the deep support vector data description (SVDD) technique proposed by Ruff et al. [21], who used neural-based anomaly detectors on images of digits as well as on open space images. The SVDD optimization technique is a post-training, fine-tuning technique increasing the accuracy of the detection through an analysis of the internal feature vector of the autoencoder.

A closer look at the results of the ISIC2019 challenge[6] (see table 1) denotes that the classification for the UNK class was poor, and in some cases the problem was ignored as a whole. For the UNK class, only four teams reached a sensitivity above 0.1.

**Table 1.** Results of the top 10 performers of the ISIC2019 challenge. The *Acc.* column refers to the Balanced Multiclass Accuracy (i.e. average sensitivity among all classes) which is the main ranking metric of the challenge.

| Team | Acc. | Ext. data | UNK Acc. | UNK Sens. | UNK Spec. | UNK AUC |
|---|---|---|---|---|---|---|
| DAISYLab | 0.636 | Yes | 0.808 | 0.002 | 0.999 | 0.808 |
| DysionAI | 0.606 | No | 0.798 | 0.179 | 0.946 | 0.562 |
| AImageLab | 0.592 | No | 0.808 | 0.004 | 0.999 | 0.502 |
| DermaCode | 0.578 | No | 0.807 | 0.012 | 0.997 | 0.642 |
| Nurithm Labs | 0.569 | Yes | 0.806 | 0.002 | 0.997 | 0.551 |
| Torus Actions | 0.563 | No | 0.808 | 0.000 | 1.000 | 0.500 |
| BITDeeper | 0.558 | No | 0.729 | 0.390 | 0.810 | 0.705 |
| SYSU-MIA-Group | 0.557 | No | 0.801 | 0.272 | 0.920 | 0.600 |
| MelanoNorm_IITRopar | 0.546 | No | 0.802 | 0.004 | 0.992 | 0.496 |
| MH_team | 0.544 | No | 0.799 | 0.118 | 0.961 | 0.556 |

For example, the first in the rank (DAISYLab) [10], who reached a balanced multiclass accuracy of 0.636, achieved only 0.002 sensitivity for the UNK class. Their strategy was to train directly a classifier on 9 classes, injecting in the training set a collection of 2334 images from other datasets, including healthy skin.

Among the best performers MH.team (ranked 10th with accuracy 0.544) performed a post-prediction analysis using the minimum, maximum and standard deviation of the softmax output of each sample. By cross-validating on 7 classes against the others (eight times), they manually selected the discrimation thresholds. With this approach they reached 0.118 sensitivity for UNK.

DysionAI (ranked 2nd with 0.607 accuracy) achieved an UNK sensitivity of 0.179 by training as 9-class classification with 0 images for UNK class. During prediction, they assign the input sample to UNK if its softmax probability is greater than a threshold set to 0.35.

The SYSU-MIA-Group (8th with 0.557 accuracy) computed the entropy of a softmax prediction on 8 classes. They interpret entropy as the inverse of confidence when

[6] https://challenge.isic-archive.com/leaderboards/2019

the classification network makes a prediction. If the confidence is below a certain threshold, the sample is marked as UNK. The threshold was manually set during internal tests by using two under-represented classes (AK and VASC) as UNK class. With this approach they reached 0.272 sensitivity for UNK.

Finally, the highest sensitivity for the UNK class (0.390) was achieved by the BIT-Deeper team (7th with 0.557 accuracy). They trained a multi-class classifier in parallel with a multi-label classifier (actually implemented via 8 binary classifiers) on the 8 known classes. The output for the UNK class is computed as a class-wise combination of the 8 softmax (multi-class) and the 8 sigmoid (multi-label) outputs. However, the choice of the combination formula and its parameter values is not explicitly motivated.

## 3 Method

The goal is to build an anomaly detection system that, given the image of a skin lesion as input, outputs a binary decision stating whether the input pertains to the *target* distribution, i.e., the same class of images on which the model was trained (negative case), or it is an outlier (positive case).

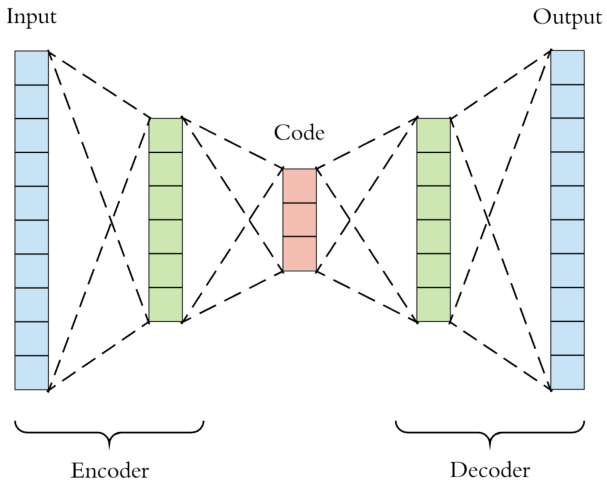


**Fig. 3.** The autoencoder architecture used to train the anomaly detection model.

As already introduced, we build an anomaly detection system based on a deep convolutional neural network autoencoder. The configuration, training, and testing procedures work as follows:

1. **Configure**. Figure 3 shows the general structure of an autoencoder. Our goal is to configure an autoencoder based on a convolutional architecture composed by a sequence `Conv :` $[D_c$ `:]` $F$ `[:` $D_d]$ `: Deconv`, where $F$ is a central dense layer with the *code* or *features* of input images, while the (optional) $D_c$ and

$D_d$ are dense layers connecting the last convolution stage to $F$ and the same to the first deconvolution stage;

2. **Train** the autoencoder $f$ using a *target* set $S_{train}$ of $m$ images, where $f(x; w)$ takes as input an image $x$ and the encoder weights $w$ and outputs another image after encoding and decoding steps. The objective function for training the autoencoder is:

$$\min_{W} \quad \frac{1}{m} \sum_{i=1}^{m} \|f(x_i; W) - x_i\|^2 \tag{1}$$

where $x_i \in S_{train}$ is an input image and $W$ are the initial pre-trained parameters (weights) of the deep autoencoder. In other words, the goal is to minimize the $l^2$-norm computed on the pixel-wise difference between the original and the reconstructed image. After training, $W^*$ are parameters of the trained model;

3. Test method $l^2$**-norm**. Given $\phi(x; w)$ the function that computes the feature vector of an image $x$ for the weights $w$, find the center $c$ of the hypersphere for the training set in the feature space:

$$c = \frac{\sum \phi(x, W^*)}{m}, x \in S_{train} \tag{2}$$

and $d_{std}$ as the standard deviation of the $l^2$-norm between the feature vector of every sample and the center:

$$d_{std} = \sqrt{\frac{\sum \|\phi(x; W^*) - c\|^2}{m}}, x \in S_{train} \tag{3}$$

Test using the discrimination formula that marks a sample $x$ as *anomaly* if

$$\|\phi(x; W^*) - c\|^2 > d_{std} * T \tag{4}$$

where $T > 0$ is a multiplier which sets the "threshold" for the discrimination.

4. Test method **Err**. Define the reconstruction error $E$ of an image $x$ as:

$$E(x) = \|x - f(x; W^*)\|^2 \tag{5}$$

Mean and standard deviation of the reconstruction error $E$ of train set images are used to determine the binary classification:

$$E_m = \frac{\sum E(x)}{m}, x \in S_{train} \tag{6}$$

$$E_{std} = \sqrt{\frac{\sum (x - E_m)^2}{m}}, x \in S_{train} \tag{7}$$

Test using the discrimination formula:

$$\|E(x) - E_m\|^2 > E_{std} * T \tag{8}$$

5. Test method **SVDD**. Fine-tune the `Conv` stage using the Deep Support Vector Data Description (SVDD) method [21], which consists of training further the `Conv` : $[D_c$ `:]` $F$ part of the model with the following objective:

$$\min_{W} \quad \frac{1}{m}\sum_{i=1}^{m}\|\phi(x_i;W)-c\|^2+\frac{\lambda}{2}\sum_{l=1}^{L}\|\mathbf{W}^l\|_F^2 \tag{9}$$

where $c$ is the center of the learned hypersphere that represents the training set in the feature space, $L \in \mathbb{N}$ is total number of hidden layers and $\lambda > 0$ is the weight decay regularization parameter.
Then, test using the same formulas of method $l^2$-norm (Equations 2, 3, and 4).

*Architecture configuration* We used two backbone CNN architectures for our tests, where the plain convolution stage was used as encoder and its transpose for the decoding part. The first backbone CNN architecture is VGG16 [23], which has proven to be sufficiently accurate in the classification skin lesions during previous ISIC challenges as well as still relatively fast to train. The second architecture is LeNet [15], which was successfully used by Ruff et al. [21] in the anomaly detection applied to the MINST [16] and CIFAR-10[7] datasets.

We tried both networks together with several configurations for the internal dense layers (hence, the number of features describing an image) and optionally the optimization method SVDD. As an additional hyper-parameter, we optionally frozen the parameters of both the `Conv` and `Deconv` stages instead of training the whole autoencoder. We also tried a combination of freezing the encoder and training the decoder together with the dense layers, but we did not observe any significant improvement, hence, results on this combination will not be reported

*Dataset* Training stages were performed on the ISIC2019 dataset ($S$), which consists of 25331 images pertaining to 8 classes. Table 2 shows the class frequencies. To conduct our studies, we selected the *nevus* ($S^{NV}$) as target class, as it contains the highest number of samples. The dataset $S^{NV}$ was further split into $S^{NV}_{train}$, $S^{NV}_{val}$, and $S^{NV}_{test}$, where the two last subsets included 2500 images each.

**Table 2.** Class frequency for the ISIC2019 dataset.

| Lesion | MEL | NV | BCC | AK | BKL | DF | VASC | SCC | Tot |
|---|---|---|---|---|---|---|---|---|---|
| Pct. | 17.8% | 50.8% | 13.1% | 3.4% | 10.4% | 1.0% | 1.0% | 2.5% | 100% |
| Count | 4522 | 12875 | 3323 | 867 | 2624 | 239 | 253 | 628 | 25331 |

*Training* It has to be noted that while training for the ISIC2019, using randomly initialized weights couldn't converge. We had to use a *double transfer* approach. First, a classifier based on the VGG16 architecture was initialized with the weights computed

[7] https://www.cs.toronto.edu/~kriz/cifar.html

for the ImageNet dataset [8]. Second, the dense layers were substituted with a 2X 2048 nodes dense layers, followed by a final 8-level softmax output and the model trained on an $S_{train}$ set. This model scored 0.91 accuracy and 0.53 sensitivity in the ISIC 2019 challenge. The resulting weights were then used to initialize both the `Conv` and `Deconv` stages of the VGG16-based autoencoder.

After initialization, we also distinguished between training the full autoencoder or only the internal dense layers (All vs. Dense-only).

The structure of SVDD is identical to the encoder part of the autoencoder along with the final representation layer and the initial weights of SVDD architecure are transferred from the trained autoencoder part and further optimization is done using the objective function 9.

*Testing* We tested our architectures using three test sets. The first $T_{7cls}$ is composed by the union of $S^{NV}_{test}$ with the remaining seven classes of the ISIC 2019 set ($S$ - $S^{NV}$), for a total of 4154 samples. As the nevus class is already contained in the $S^{NV}_{train}$ set, the goal was to discriminate from nevus as target and melanoma as anomaly. The second test set $T_{MedNode}$ is the MedNode dataset [11], which contains 100 images for nevi and 70 melanomas. Finally, the third test set $T_{coco}$ is composed by the union of $S^{NV}_{test}$ with a selection of 4989 random images from the COCO dataset [19]. The goal here is to set a baseline for the discrimination between skin lesion images and random "outside-world" ones.

## 4 Results

Table 3 show the test results for several combination of hyperparameters and test sets. The positive case (i.e., high sensitivity) is associated with the capability of detecting an anomaly.

In addition to the reference CNN architecture (base arch.) and the configuration of the dense layers (dense layers), we test the difference between training the whole autoencoder vs. training only the internal dense layers (trained layers) and use different norm and three discrimination methods: feature vector distance, (Err, $l^2$, and SVDD). The AUC is computed considering all of the samples of the test set, and gives an indication on the capability of the method into discriminating between the target and the anomaly classes. However, the AUC does *not* suggest what would be a proper distance (or error) threshold value $T$ for deploying the system in real settings.

Hence, with reference to equation 8, we tested the performances of the anomaly detector using two threshold $T$ values: 1 and 3. With $T = 1$, our hypothesis is that the hypersphere including the target samples would be very narrow, thus including only some of the target samples, but no anomaly samples. Differently, with $T = 3$, which for normal distributions would include 99.7% of the samples, our hypothesis is to have a discriminator which retain most of the target samples at a risk of missing many anomalies.

The top section of table 3 reports results for the LeNet architecture. From the AUC measurement, we can see that the network is perfectly able to detect COCO classes, but the discrimination with 7cls fails ($AUC \simeq 0.5$). This is reflected in the high sensitivity couple with a very low specificity.

**Table 3.** A selection of the tests of different architectures against other 7classes and COCO datasets.

| Test # | Base arch. | Dense nodes | Train layers | Test method | Test set | AUC | T = 1 acc. | T = 1 spec. | T = 1 sens. | T = 3 acc. | T = 3 spec. | T = 3 sens. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | LeNet | 128 | all | Err | 7cls | 0.49 | 0.39 | 0.15 | 0.86 | 0.35 | 0.03 | 0.98 |
| 2 | LeNet | 128 | all | Err | coco | 1 | 0.96 | 1 | 0.86 | 0.99 | 1 | 0.98 |
| 3 | LeNet | 128 | all | SVDD | 7cls | 0.49 | 0.38 | 0.12 | 0.88 | 0.35 | 0.3 | 0.97 |
| 4 | LeNet | 128 | all | SVDD | coco | 1 | 0.7 | 1 | 0 | 0.99 | 1 | 0.97 |
| 5 | VGG16 | 1960:1960:1960 | dense | Err | 7cls | 0.51 | 0.45 | 0.34 | 0.66 | 0.34 | 0 | 1 |
| 6 | VGG16 | 1960:1960:1960 | dense | Err | coco | 1 | 0.9 | 1 | 0.66 | 1 | 1 | 1 |
| 7 | VGG16 | 1960:1960:1960 | dense | SVDD | 7cls | 0.49 | 0.39 | 0.15 | 0.85 | 0.37 | 0.07 | 0.94 |
| 8 | VGG16 | 1960:1960:1960 | dense | SVDD | coco | 0.99 | 0.96 | 1 | 0.85 | 0.98 | 1 | 0.94 |
| 9 | VGG16 | 1960:1960:1960 | all | Err | 7cls | 0.49 | 0.4 | 0.17 | 0.82 | 0.36 | 0.04 | 0.96 |
| 10 | VGG16 | 1960:1960:1960 | all | Err | coco | 1 | 0.95 | 1 | 0.82 | 0.99 | 1 | 0.96 |
| 11 | VGG16 | 1960:1960:1960 | all | SVDD | 7cls | 0.5 | 0.66 | 1 | 0 | 0.34 | 0 | 1 |
| 12 | VGG16 | 1960:1960:1960 | all | SVDD | coco | 1 | 0.7 | 1 | 0 | 1 | 1 | 1 |
| 13 | VGG16 | 1960X2:980:1960X2 | all | Err | coco | 0.95 | 0.91 | 0.97 | 0.77 | 0.39 | 0.1 | 0.99 |
| 14 | VGG16 | 3920 | all | Err | coco | 0.93 | 0.88 | 0.91 | 0.83 | 0.63 | 0.47 | 0.96 |
| 15 | VGG16 | 490 | all | SVDD | coco | 0.92 | 0.87 | 0.89 | 0.84 | 0.59 | 0.41 | 0.96 |
| 16 | VGG16 | 980 | all | SVDD | coco | 0.92 | 0.87 | 0.89 | 0.84 | 0.58 | 0.4 | 0.96 |
| 17 | VGG16 | 147 | all | Err | coco | 0.92 | 0.87 | 0.88 | 0.83 | 0.58 | 0.4 | 0.96 |
| 18 | VGG16 | 1960:980:1960 | all | Err | coco | 0.9 | 0.86 | 0.87 | 0.82 | 0.5 | 0.29 | 0.95 |
| 19 | VGG16 | 1960:do(0.5):980:1960 | all | Err | coco | 0.9 | 0.86 | 0.87 | 0.82 | 0.5 | 0.29 | 0.95 |
| 20 | VGG16 | 1960:1960:1960 | all | $l^2$-norm | 7cls | 0.5 | 0.41 | 0.22 | 0.77 | 0.34 | 0 | 0.99 |
| 21 | VGG16 | 1960:1960:1960 | all | $l^2$-norm | coco | 0.73 | 0.34 | 0.16 | 0.77 | 0.30 | 0 | 0.99 |

Therefore, we configured a more powerful autoencoder, based on the VGG16 architecture, experimenting with several configurations for the internal dense layers. In table 3, lines from 13 to 19 show the test results for several combinations of dense layers. Such configurations where not able to reach AUC 1.0 even on the COCO dataset. Lines 20–21 show the results for the $l^2$-norm method, which was, too, unable to reach AUC 1.0 on the $S_{coco}$ test set.

The perfect detection of COCO images is achieved by the dense nodes configuration $L_{d_c} = 1960 : L_f = 1960 : L_{d_d} = 1960$ (Table 3, lines 5–12). However, in spite of the similarity with the original classifier (`Conv : 2048 : 2048 : softmax`), the test on the 7cls dataset lead to an $AUC \simeq 0.5$. It can be seen that tests on thresholds 1 and 3 lead (in some cases) to opposite results in terms of sensitivity and specificity.

To better understand the behaviour of the discriminator as function of the threshold $T$, we computed the quality metrics for different values of $T$, ranging from 0 to 8 with increments of 0.1. This last procedure is also essential for fixing the $T$ parameter to a value that should include most (or better all) of the samples of the target distribution, and be ready to intercept anomalies in a real application scenario, such as an online web service, where input samples can come from unpredictable distributions.

Figures 4 and 5 show the results for tests number 5–6 and 9–10, respectively (The other configurations show a similar behaviour). The top-left plots show the variation when testing NV against the other 7 classes. As the threshold increases, the sensitivity

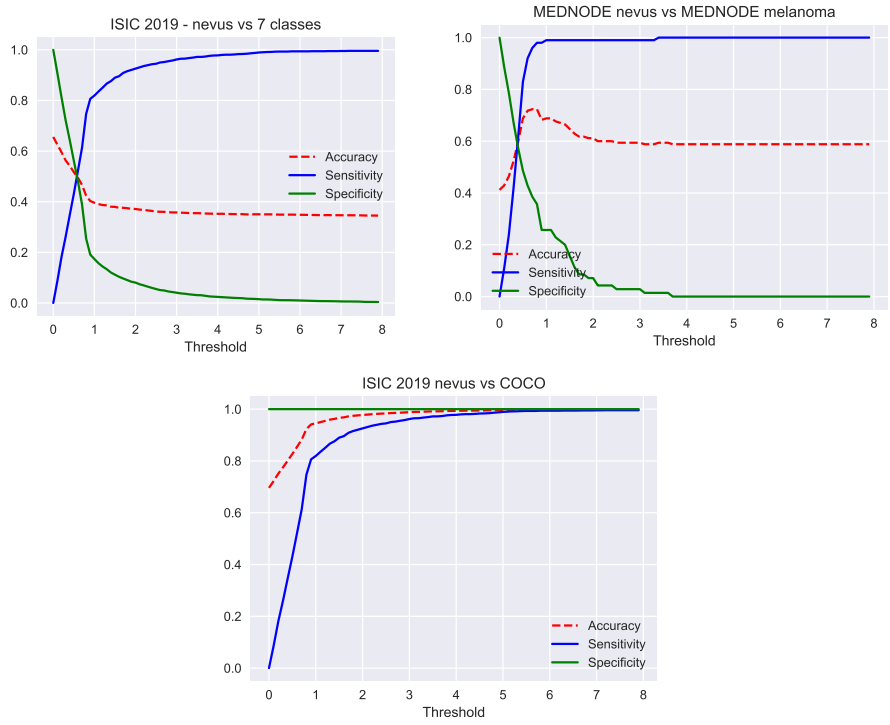


**Fig. 4.** Performance metrics as function of the threshold for tests 5 and 6 (training all layers), and the same architecture tested on the $T_{MedNode}$ test set.

(i.e., the capability to detect an anomaly reaches 1.0). However, the specificity drops to 0.0, meaning that the system is not able to discriminate at all. The accuracy reflects this behaviour and converges to the class proportions ratio. The top-right plots show the same behaviour when trying to discriminate against the melanoma class in the MedNode dataset. Finally, the bottom plots show positive results when testing against the COCO dataset. By setting $T = 6$ for full training, and $T = 3$ for only-dense layers training, we reach accuracy 1.0. When comparing the two configurations, it means that by training only the dense layers, the target samples are closer to the center of the hypershpere, potentially meaning that the discrimination among classes can be more difficult.

To better inspect the behaviour when applying the SVDD technique, we plotted the metrics variation for tests 11–12 (which correspond to the non-SVDD test 5–6 of Figure 4). Figure 6 shows that when testing against 7-classes and against melanoma, around $T = 1.5$ there is a sudden inversion between specificity and sensitivity. It suggests that, as is the purpose of SVDD, the sample features space is contracted towards the center of the target hypersphere, reducing the range of the distribution. However, this leads to poor results also when testing against the COCO images, meaning that also the feature vectors of fairly different images are collapsing together with the target lesion images.

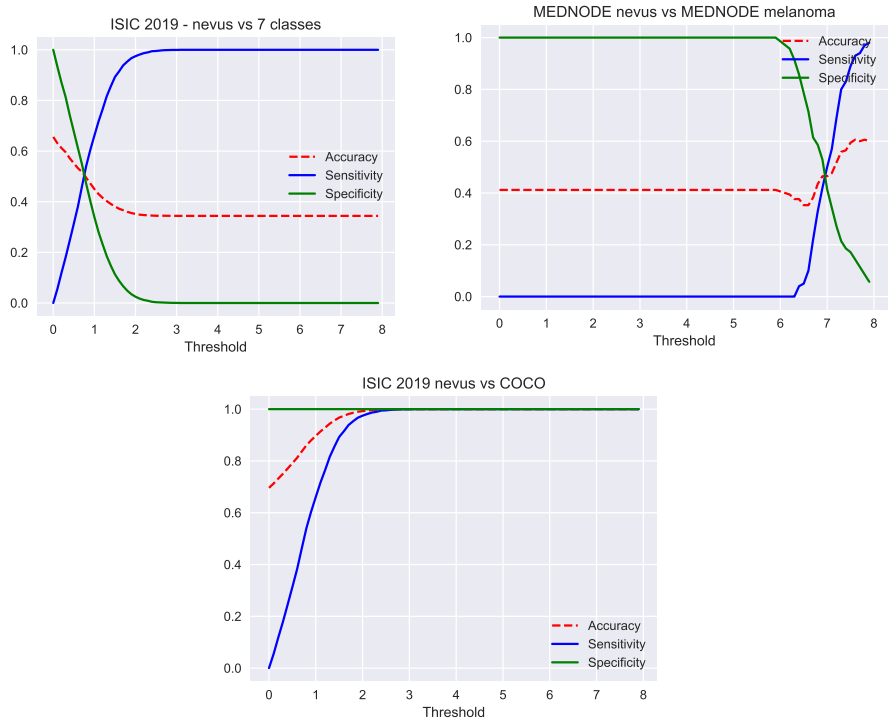


**Fig. 5.** Performance metrics as function of the threshold for tests 9 and 10 (training only dense layers), and the same architecture tested on the $T_{MedNode}$ test set.

The coherence of this last results with other configurations, led us to mark the SVDD method as ineffective for the skin lesion domain.

The discrimination between targets and anomalies is based on the measurement of the error $E$ between the original and the reconstructed image. Here, the idea is that during the training the autoencoder specializes in encoding images of the target set (low MSE), but is not able to encode images from other distributions (high MSE).

So far, these results suggests that the reconstruction error $E$, measured between the original and the reconstructed image, is similar for nevus as well as for the other 7 classes, but differs for the COCO classes. To visually verify this hypothesis, we plotted the distribution of the errors for the samples for $S^{NV}_{test}$, $T_{7cls}$, and $T_{coco}$ (see figure 7). The histogram shows (with some approximation) that there is indeed an overlap between the error scores between the nevus class and the other 7 classes, while samples of the COCO dataset are well distanced. Finally, to understand if there would be the possibility to discriminate between the nevus class and any other the 7 other classes, we plotted the error distribution for the 7 classes separately. Figure 8 shows that the error distribution of all classes overlaps with the error distribution for NV class, hindering the capability to perform a discrimination based on error analysis.

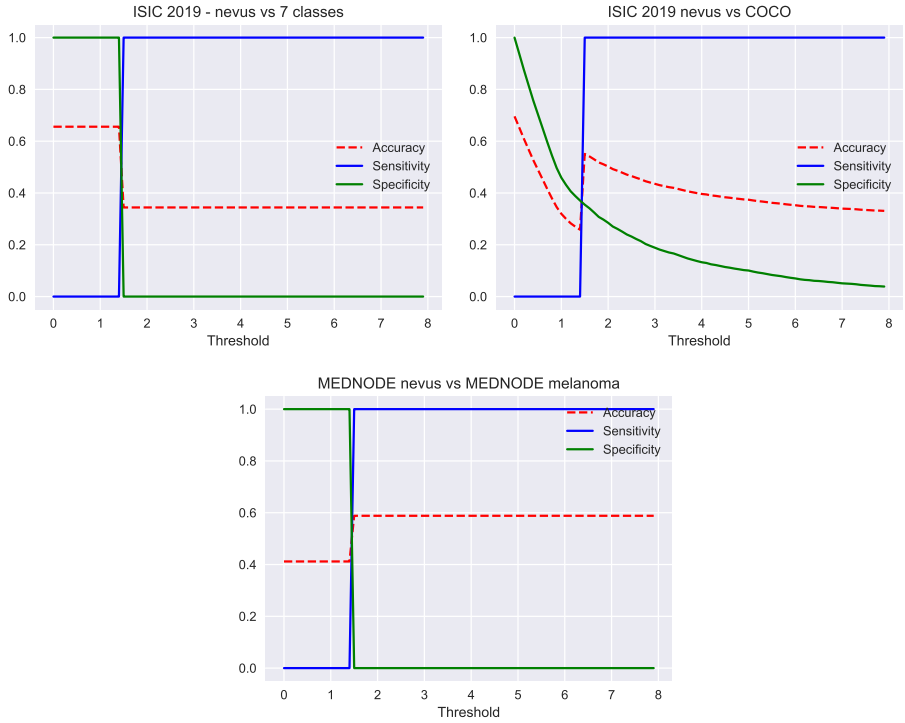


**Fig. 6.** Performance metrics as function of the threshold for tests 11 and 12 (training all layers, plus SVDD), and the same architecture tested on the $T_{MedNode}$ test set.

## 5 Conclusions

The results of our tests show that anomaly detectors based on replicator neural networks, initially trained as autoencoders, can distinguish skin lesions from random images of the outside world very well when the discrimination is based on the encoding/decoding reconstruction error. This discrimination technique should be preferred over $l^2$-norm or SVDD methods.

However, the discrimination among classes of skin lesions still leads to random selection. We suspect that this is the case because the VGG16 architecture is learning features that are common to all lesions. Hence, while the same architecture, trained on all classes, can be effective as classifier, it doesn't allow for setting a discrimination threshold when trained on a single class. More tests should be conducted to check whether the same applies when changing the target class, from nevus to any of the other seven.

Future work can be done in several directions: i) explore more hyperparameters, ii) try with more powerful networks, iii) solve the limitations recently addressed on SVDD [4]. We also aim to investigate in the direction of information fusion and explainable AI by incorporating multi-modal embeddings with Graph Neural Networks [13]

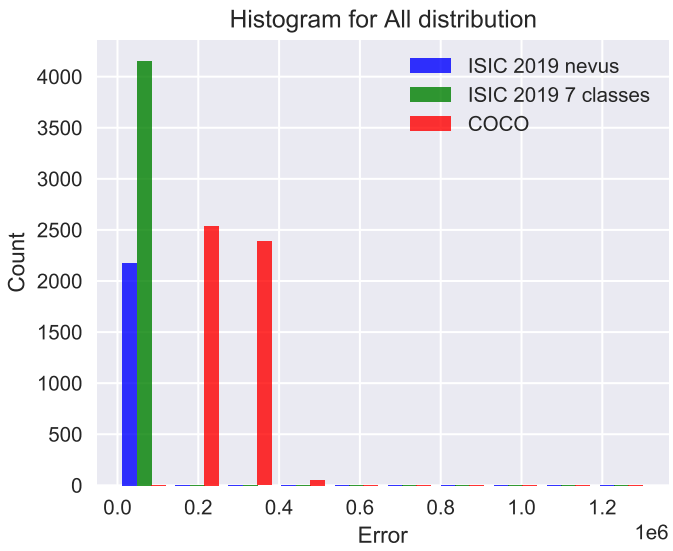


**Fig. 7.** Distribution of the MSEs for the CNN configuration used in tests 5 and 6.

## Acknowledgements

This research is partly funded by the pAItient project (BMG) and the Endowed Chair of Applied Artificial Intelligence (Oldenburg University).

## References

1. Brinker, T.J., Hekler, A., Enk, A.H., et al.: Deep learning outperformed 136 of 157 dermatologists in a head-to-head dermoscopic melanoma image classification task. European Journal of Cancer **113**, 47–54 (May 2019). https://doi.org/10.1016/j.ejca.2019.04.001, https://linkinghub.elsevier.com/retrieve/pii/S0959804919302217 1
2. Celebi, M.E., Codella, N., Halpern, A.: Dermoscopy Image Analysis: Overview and Future Directions. IEEE Journal of Biomedical and Health Informatics **23**(2), 474–478 (Mar 2019). https://doi.org/10.1109/JBHI.2019.2895803, https://ieeexplore.ieee.org/document/8627921/ 1
3. Chandola, V., Banerjee, A., Kumar, V.: Anomaly Detection: A Survey. ACM Comput. Surv. **41**(3), 15:1–15:58 (Jul 2009). https://doi.org/10.1145/1541880.1541882, http://doi.acm.org/10.1145/1541880.1541882 3
4. Chong, P., Ruff, L., Kloft, M., Binder, A.: Simple and effective prevention of mode collapse in deep one-class classification. 2020 International Joint Conference on Neural Networks (IJCNN) (Jul 2020). https://doi.org/10.1109/ijcnn48605.2020.9207209, http://dx.doi.org/10.1109/IJCNN48605.2020.9207209 12

5. Codella, N., Rotemberg, V., Tschandl, P., Celebi, M.E., Dusza, S., Gutman, D., Helba, B., Kalloo, A., Liopyris, K., Marchetti, M., Kittler, H., Halpern, A.: Skin Lesion Analysis Toward Melanoma Detection 2018: A Challenge Hosted by the International Skin Imaging Collaboration (ISIC). arXiv:1902.03368 [cs] (Feb 2019), http://arxiv.org/abs/1902.03368, arXiv: 1902.03368 1
6. Codella, N.C.F., Gutman, D., Celebi, M.E., Helba, B., Marchetti, M.A., Dusza, S.W., Kalloo, A., Liopyris, K., Mishra, N., Kittler, H., Halpern, A.: Skin Lesion Analysis Toward Melanoma Detection: A Challenge at the 2017 International Symposium on Biomedical Imaging (ISBI), Hosted by the International Skin Imaging Collaboration (ISIC). arXiv:1710.05006 [cs] (Oct 2017), http://arxiv.org/abs/1710.05006, arXiv: 1710.05006 1
7. Codella, N.C.F., Gutman, D., Celebi, M.E., Helba, B., Marchetti, M.A., Dusza, S.W., Kalloo, A., Liopyris, K., Mishra, N., Kittler, H., Halpern, A.: Skin lesion analysis toward melanoma detection: A challenge at the 2017 International symposium on biomedical imaging (ISBI), hosted by the international skin imaging collaboration (ISIC). In: 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018). pp. 168–172. IEEE, Washington, DC (Apr 2018). https://doi.org/10.1109/ISBI.2018.8363547, https://ieeexplore.ieee.org/document/8363547/ 1
8. Deng, J., Dong, W., Socher, R., Li, L.J., Kai Li, Li Fei-Fei: ImageNet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255. IEEE, Miami, FL (Jun 2009). https://doi.org/10.1109/CVPR.2009.5206848, http://ieeexplore.ieee.org/document/5206848/ 8
9. Esteva, A., Kuprel, B., Novoa, R.A., Ko, J., Swetter, S.M., Blau, H.M., Thrun, S.: Dermatologist-level classification of skin cancer with deep neural networks. Nature **542**, 115 (Jan 2017), https://doi.org/10.1038/nature21056 1
10. Gessert, N., Nielsen, M., Shaikh, M., Werner, R., Schlaefer, A.: Skin lesion classification using ensembles of multi-resolution EfficientNets with meta data. MethodsX **7**, 100864 (2020). https://doi.org/https://doi.org/10.1016/j.mex.2020.100864, https://www.sciencedirect.com/science/article/pii/S2215016120300832 4
11. Giotis, I., Molders, N., Land, S., Biehl, M., Jonkman, M.F., Petkov, N.: MED-NODE: A computer-assisted melanoma diagnosis system using non-dermoscopic images. Expert Systems with Applications **42**(19), 6578–6585 (Nov 2015). https://doi.org/10.1016/j.eswa.2015.04.034, https://linkinghub.elsevier.com/retrieve/pii/S0957417415002705 8
12. Hawkins, S., He, H., Williams, G., Baxter, R.: Outlier Detection Using Replicator Neural Networks. In: Data Warehousing and Knowledge Discovery, vol. 2454, pp. 170–180. Springer Berlin Heidelberg, Berlin, Heidelberg (2002). https://doi.org/10.1007/3-540-46145-0_17, series Title: Lecture Notes in Computer Science 3
13. Holzinger, A., Malle, B., Saranti, A., Pfeifer, B.: Towards multi-modal causability with graph neural networks enabling information fusion for explainable AI. Inf. Fusion **71**, 28–37 (2021). https://doi.org/10.1016/j.inffus.2021.01.008, https://doi.org/10.1016/j.inffus.2021.01.008 12
14. Kawahara, J., Hamarneh, G.: Visual Diagnosis of Dermatological Disorders: Human and Machine Performance. arXiv:1906.01256 [cs] (Jun 2019), http://arxiv.org/abs/1906.01256, arXiv: 1906.01256 1
15. LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Backpropagation Applied to Handwritten Zip Code Recognition. Neural Computation **1**(4), 541–551 (Dec 1989). https://doi.org/10.1162/neco.1989.1.4.541, https://www.mitpressjournals.org/doi/abs/10.1162/neco.1989.1.4.541 7

16. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (Nov 1998). https://doi.org/10.1109/5.726791, http://ieeexplore.ieee.org/document/726791/ 7
17. Lee, K., Lee, K., Lee, H., Shin, J.: A simple unified framework for detecting out-of-distribution samples and adversarial attacks (2018) 3
18. Li, X., Lu, Y., Desrosiers, C., Liu, X.: Out-of-Distribution Detection for Skin Lesion Images with Deep Isolation Forest. In: Liu, M., Yan, P., Lian, C., Cao, X. (eds.) Machine Learning in Medical Imaging. pp. 91–100. Springer International Publishing, Cham (2020) 3
19. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common Objects in Context. In: Computer Vision – ECCV 2014, vol. 8693, pp. 740–755. Springer International Publishing, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48, series Title: Lecture Notes in Computer Science 8
20. Pang, G., Shen, C., Cao, L., Hengel, A.V.D.: Deep learning for anomaly detection. ACM Computing Surveys **54**(2), 1–38 (Mar 2021). https://doi.org/10.1145/3439950, http://dx.doi.org/10.1145/3439950 3
21. Ruff, L., Vandermeulen, R., Goernitz, N., Deecke, L., Siddiqui, S.A., Binder, A., Müller, E., Kloft, M.: Deep One-Class Classification. In: Proceedings of the 35th International Conference on Machine Learning. vol. 80, pp. 4393–4402. PMLR, Stockholm, Sweden (Jul 2018), http://proceedings.mlr.press/v80/ruff18a.html 4, 7
22. Schölkopf, B., Platt, J.C., Shawe-Taylor, J., Smola, A.J., Williamson, R.C.: Estimating the Support of a High-Dimensional Distribution. Neural Computation **13**(7), 1443–1471 (Jul 2001). https://doi.org/10.1162/089976601750264965, https://direct.mit.edu/neco/article/13/7/1443-1471/6529 3
23. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv:1409.1556 [cs] (Sep 2014), http://arxiv.org/abs/1409.1556, arXiv: 1409.1556 7
24. Tschandl, P., Rosendahl, C., Kittler, H.: The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. Scientific Data **5**(1) (Dec 2018). https://doi.org/10.1038/sdata.2018.161, http://www.nature.com/articles/sdata2018161 3

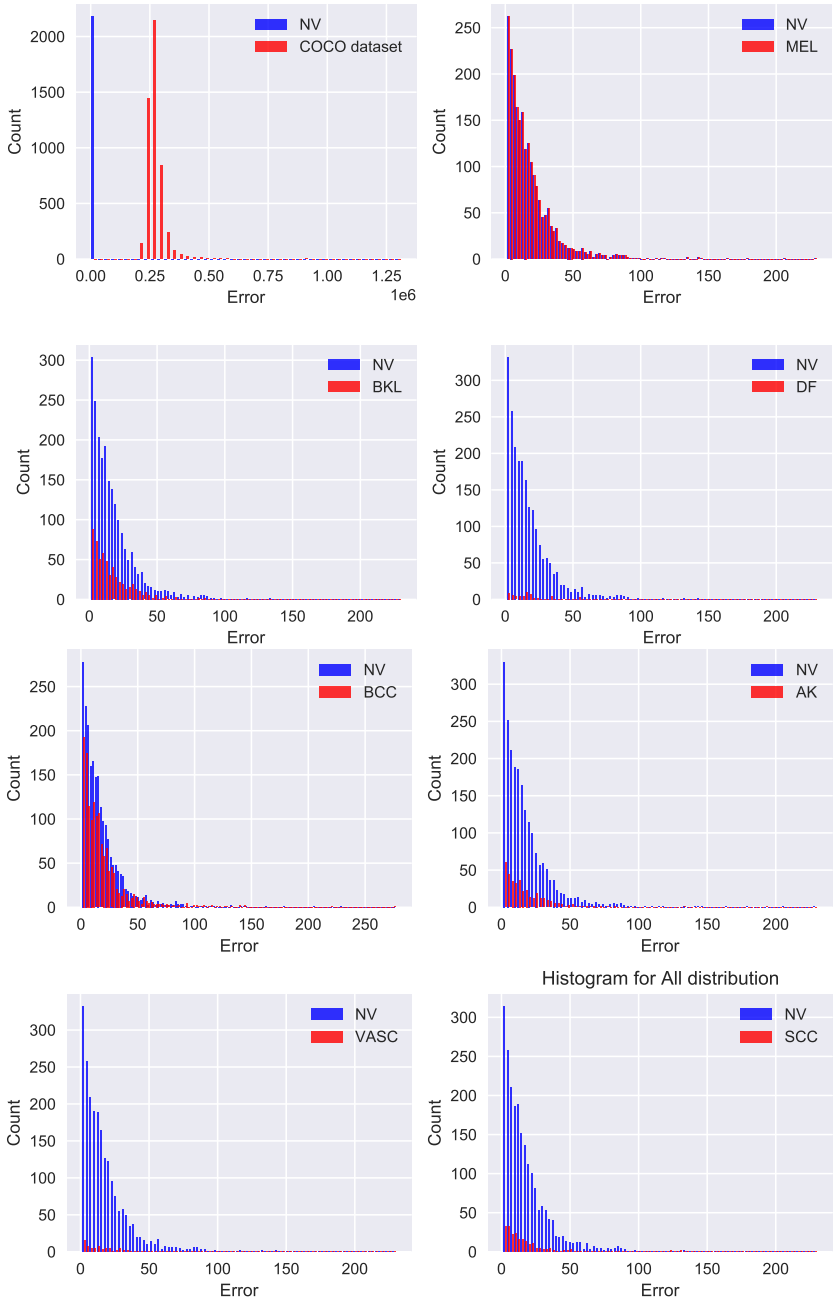


**Fig. 8.** Distribution of reconstruction errors for nevus and all of the other classes separately.