On the Applicability of Prototypical Part Learning in Medical Images: Breast Masses Classification Using ProtoPNet

Gianluca Carloni[1,2,4][0000−0002−5774−361X], Andrea Berti[1,2,4][0000−0002−0089−6420], Chiara Iacconi[3][0000−0002−0729−4166], Maria Antonietta Pascali[1][0000−0001−7742−8126], and Sara Colantonio[1][0000−0003−2022−0804]

[1] Institute of Information Science and Technologies (ISTI), National Research Council of Italy (CNR), Pisa, Italy.
name.surname@isti.cnr.it

[2] Department of Information Engineering, University of Pisa, Pisa, Italy.
{gianluca.carloni, andrea.berti}@phd.unipi.it

[3] UOSD breast radiology, territorial area of Massa Carrara, Azienda USL Toscana Nord-Ovest, Carrara, Italy.

[4] These authors share the first authorship.

Abstract. Deep learning models have become state-of-the-art in many areas, ranging from computer vision to agriculture research. However, concerns have been raised with respect to the transparency of their decisions, especially in the image domain. In this regard, Explainable Artificial Intelligence has been gaining popularity in recent years. The ProtoPNet model, which breaks down an image into prototypes and uses evidence gathered from the prototypes to classify an image, represents an appealing approach. Still, questions regarding its effectiveness arise when the application domain changes from real-world natural images to gray-scale medical images. This work explores the applicability of prototypical part learning in medical imaging by experimenting with ProtoPNet on a breast masses classification task. The two considered aspects were the classification capabilities and the validity of explanations. We looked for the optimal model's hyperparameter configuration via a random search. We trained the model in a five-fold CV supervised framework, with mammogram images cropped around the lesions and ground-truth labels of benign/malignant masses. Then, we compared the performance metrics of ProtoPNet to that of the corresponding base architecture, which was ResNet18, trained under the same framework. In addition, an experienced radiologist provided a clinical viewpoint on the quality of the learned prototypes, the patch activations, and the global explanations. We achieved a Recall of 0.769 and an area under the receiver operating characteristic curve of 0.719 in our experiments. Even though our findings are non-optimal for entering the clinical practice yet, the radiologist found ProtoPNet's explanations very intuitive, reporting a high level of satisfaction. Therefore, we believe that prototypical part learning offers a reasonable and promising trade-off between classification performance and the quality of the related explanation.

Keywords: ProtoPNet · Breast masses · Classification · Deep learning · Explainable Artificial Intelligence.

1 Introduction

Today's world of information research is largely dominated by artificial intelligence (AI) technologies. In particular, deep learning (DL) models are being deployed transversely across many sectors, revealing a great added value to humans in many of them. Some examples are autonomous driving [8] and smart agriculture [16]. Although DL models usually outperform humans at many levels, performance is not all we need. Indeed, industrial and research communities demand more explainable and trustworthy DL models. These needs emerge from the user's difficulty in understanding the internal mechanisms of an intelligent agent that led to a decision. Based on this degree of understanding, the user often decides whether to trust the output of a model.

Explainable AI (XAI) plays a pivotal role in this scenario. Research is now focusing on developing methods to explain the behavior and reasoning of deep models. Explanation methods developed so far can be divided into two major classes: *post-hoc* explanations and *ante-hoc* explanations. The first class comprises solutions that are based on separate models that are supposed to replicate most of the behavior of the black-box model. Their major advantage is that they can be applied to an already existing and well-performing model. However, in approximating the outcome, they may not reproduce the same calculations of the original model. Among this family of explanations we find global/local approximations, saliency maps and derivatives. By contrast, the second class of explanation methods, also known as *explaining by design*, comprises inherently interpretable models that provide their explanations in the same way the model computes its decisions. Indeed, training, inference, and explanation of the outcome are intrinsically linked. Examples of such methods are Deep k-Nearest Neighbors [17] and Logic Explained Networks [7].

Regarding the image domain, a substantial body of DL literature concerns classification tasks [14, 11]. When it comes to image classification, one of the most familiar approaches humans exploit is to analyze the image and, by similarity, identify the previously seen instances of a certain class. A line of DL research focuses on models that mimic this type of reasoning, which is called prototypical learning [20, 13]. The key feature of this class of learning algorithms is to compare one whole image to another whole image. Instead, one could wish to understand what are the relevant parts of the input image that led to a specific class prediction. In other words, parts of observations could be compared to parts of other observations. In the attempt to build a DL model that resembles this kind of logic, Chen et al. [6] proposed prototypical part network (ProtoPNet).

ProtoPNet breaks down an image into prototypes and uses evidence gathered from the prototypes to qualify the image. Thus, the model's reasoning is qualitatively similar to that of ornithologists, physicians, and others on the image classification task. At training time, the network uses only image-level la-

bels without fine-annotated images. At inference time, the network predicts the image class by comparing its patches with the learned prototypes. The model provides an explanation visually by indicating the most informative parts of the image w.r.t. the output class. This allows the user to qualitatively evaluate how reasonable and trustworthy the prediction is according to the user domain knowledge.

ProtoPNet posed brilliant promises in classification domains regarding natural images (e.g., birds and cars classification [6], video deep-fake detection [23]). On the other hand, the applicability of this type of reasoning to medical images is still in its infancy. When presented with a new case, radiologists use to compare the images with previously experienced ones. They recall visual features that are specific to a particular disease, recognize them in the image at hand, and provide a diagnosis. For this reason, medical imaging seems to be suitable for prototypical part-based explanations. Nevertheless, some critical issues can arise when bringing technologies from other domains – like computer vision - into the medical world. Unlike natural images, usually characterized by three channels (e.g., RGB, CYM), conventional medical images feature single-channel gray-scales. For this reason, pixels contain a lower amount of information. Furthermore, x-ray images represent a body's projection and therefore are flat and bi-dimensional. As a result, objects in the field of view could not be as separable and distinguishable as in real-world natural images. Such issues might be detrimental to the application of these methodologies. In addition, the scarcity of labeled examples available for supervised training undermines the generalization capability achievable by complex models. This lack of labeled data is mainly due to the low prevalence of certain diseases, the time required for labeling, and privacy issues. Moreover, additional problems include the anatomical variability across patients and the image quality variability across different imaging scanners.

This work aims to investigate the applicability of ProtoPNet in mammogram images for the automatic and explainable malignancy classification of breast masses. The assessment of applicability was based on two aspects: the ability of the model in facing the task (i.e., classification metrics), and the ability of the model to provide end-users with plausible explanations. The novelty of this work stems from both the application of ProtoPNet to the classification of breast masses without fine-annotated images, and the clinical viewpoint provided for ProtoPNet's explanations.

2 Related Works

Several works in the literature have applied DL algorithms, and convolutional neural network (CNN) architectures in particular, to automatically classify benign/malignant breast masses from x-ray mammogram images. By contrast, only few works explored the applicability of ProtoPNet to the medical domain and, more specifically, on breast masses classification.

Concerning the use of CNNs for this task on the Curated Breast Imaging Subset of DDSM (CBIS-DDSM) [12] dataset some works follow. Tsochatzidis et al. [24] explored various popular CNN architectures, by using both randomly initialized weights and pre-trained weights from ImageNet. With ResNet50 and pre-trained weights they obtained an accuracy of 0.749. Alkhaleefah et al. [1] investigated the influence of data augmentation techniques on classification performance. When using ResNet50, they achieved 0.676 and 0.802 before and after augmentation, respectively. Arora et al. [3] proposed a two-stage classification system. First, they exploited an ensemble of five CNN models to extract features from breast mass images and then concatenated the five feature vectors into a single one. In the second stage, they trained a two-layered feed-forward network to classify mammogram images. With this approach, they achieved an accuracy of 0.880. They also reported the performance obtained with each individual sub-architecture of the ensemble, achieving an accuracy of 0.780 with ResNet18. Ragab et al. [19] also experimented with multiple CNN models to classify mass images. Among the experiments, they obtained an accuracy of 0.722, 0.711 and 0.715 when applying ResNet18, ResNet50 and ResNet101, respectively. Finally, Ansar et al. [2] introduced a novel architecture based on MobileNet and transfer learning to classify mass images. They benchmarked their model with other popular networks, among which ResNet50 led to an accuracy of 0.637.

Regarding the application of ProtoPNet to the medical domain, only few attempts have investigated it to date. Mohammadjafari et al. [15] applied ProtoPNet to Alzheimer's Disease detection on brain magnetic resonance images from two publicly available datasets. As a result, they found an accuracy of 0.91 with ProtoPNet, which is comparable to or marginally worse than that obtained with state-of-the-art black-box models. Singh et al. [22, 21] proposed two works utilizing ProtoPNet on chest X-ray images of Covid-19 patients, pneumonia patients, and healthy people for Covid-19 identification. In [22] they slightly modified the weight initialization in the model to emphasize the effect of differences between image parts and prototypes in the classification process, achieving an accuracy of 0.89. In [21] they modified the metrics used in the model's classification process to select prototypes of varying dimensions, and obtained the best accuracy of 0.87.

To the best of our knowledge, the only application of prototypical part learning to the classification of benign/malignant masses in mammogram images was provided by Barnett et al. [4]. They introduced a new model, IAIA-BL, derived from ProtoPNet, utilizing a private dataset with further annotations by experts in training data. They included both pixel-wise masks to consider clinically significant regions in images and mass margin characteristics (spiculated, circumscribed, microlobulated, obscured, and indistinct). On the one hand, annotation masks of clinically significant regions were exploited at training time in conjunction with a modified loss function to penalize prototype activations on medically irrelevant areas. On the other hand, they employed annotations of mass margins as an additional label for each image and divided the inference process into two phases: first, the model determines the mass margin feature

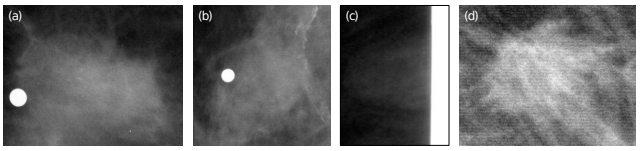

Fig. 1. Examples of images from the original CBIS-DDSM dataset that were removed due to artifacts. (a)-(b): annotation spot next to or within the mass; (c): white-band artifact; (d) horizontal-pattern artifact.

and then predicts malignancy based on that information. For this purpose, they added a fully-connected (FC) layer to convert the mass margin score to the malignancy score. With that architecture, they managed to achieve an AUROC of 0.84.

3 Materials and Methods

In this work, we trained a ProtoPNet model to classify benign/malignant breast masses from mammogram images on a publicly available dataset. We compared its performance to the baseline model on which ProtoPNet is based. We conducted a random search independently on both models with five-fold cross-validation (CV) to optimize the respective hyperparameters.

3.1 Dataset

In our study, we used images from CBIS-DDSM [12]. The dataset is composed of scanned film mammography studies from 1566 breast cases (i.e., patients). For each patient, two views (i.e., MLO and CC) of the full mammogram images are provided. In addition, the collection comes with the region of interest (ROI)-cropped images for each lesion. Each breast image has its annotations given by experts, including the ground truth for the type of cancer (benign, malignant, or no-callback) and the type of lesion (calcification or mass). Only the ROI-cropped images of benign and malignant masses for each patient were used in this study. As a first step, we performed a cleaning process of the dataset by removing images with artifacts and annotation spots next to or within the mass region (Fig. 1). We then converted DICOM images of the cleaned dataset into PNG files. The training and test split of the cohort was already provided in the data collection. To obtain a balanced dataset, we randomly selected the exceeding elements from the most numerous class and excluded them from the cohort.

3.2 ProtoPNet

Architecture and functioning ProtoPNet, introduced in [6], comprises three main blocks: a CNN, a prototype layer, and an FC layer. As for the CNN block,

it consists of a feature extractor, which can be chosen from many of the popular models competing on ImageNet challenges (VGGs, ResNets, DenseNets), and a series of add-on convolutional layers. This block extracts features from an input RGB image of size 224×224. Given this input size, the convolutional output has size $7 \times 7 \times D$, where D is the number of output filters of the CNN block. ReLU is used to activate all convolutional layers, except the last one that utilizes the sigmoid activation. The prototype layer that follows comprises two 1×1 convolutional layers with ReLU activation. It learns m prototypes, whose shape is $1 \times 1 \times D$. Each prototype embodies a prototypical activation pattern in one area of the convolutional output, which itself refers to a prototypical image in the original pixel space. Thus, we can say that each prototype is a latent representation of some prototypical element of an image.

At inference time, the prototype layer computes a similarity score as the inverted squared L^2 distance between each prototype and all patches of the convolutional output. For each prototype, this produces an activation map of similarity score whose values quantify the presence of that prototypical part in the image. This map is up-sampled to the size of the input image and presented as an overlayed heat map highlighting the part of the input image that mostly resembles the learned prototype. The activation map for each prototype is then reduced using global max pooling to a single similarity score. A predetermined number of prototypes represents each class in the final model. In the end, the classification is performed by multiplying the similarity score of each prototype by the weights of the FC layer.

Prototype learning process The learning process begins with the stochastic gradient descent of all the layers before the FC layer (joint epochs). Then, prototypes are projected onto the closest latent representation of training images' patches. Finally, the optimization of the FC layer is carried out. It is possible to cycle through these three stages more than once.

Differences in our implementation Differences exist between the original paper introducing ProtoPNet [6] and our work. Firstly and more importantly, we conceived a hold-out test set to assess the final models' performance, after the models were trained using CV. In the original paper, instead, both the selection of the best model and the evaluation of its performance were carried out on the same set, i.e., validation and test sets were the same.

In addition, since ProtoPNet works with three-channel images, we modified the one-channel gray-scale input images by copying the information codified in the single channel to the other two. Then, we set the number of classes for the classification task to two instead of 200. Finally, to reduce overfitting when training a large model using a limited dataset, we introduced a 2D dropout layer and a 2D batch-normalization layer after each add-on convolutional layer of the model. An overview of our implementation of ProtoPNet architecture and its inference process is depicted in Fig. 2, taking the classification of a correctly classified malignant mass as an example.

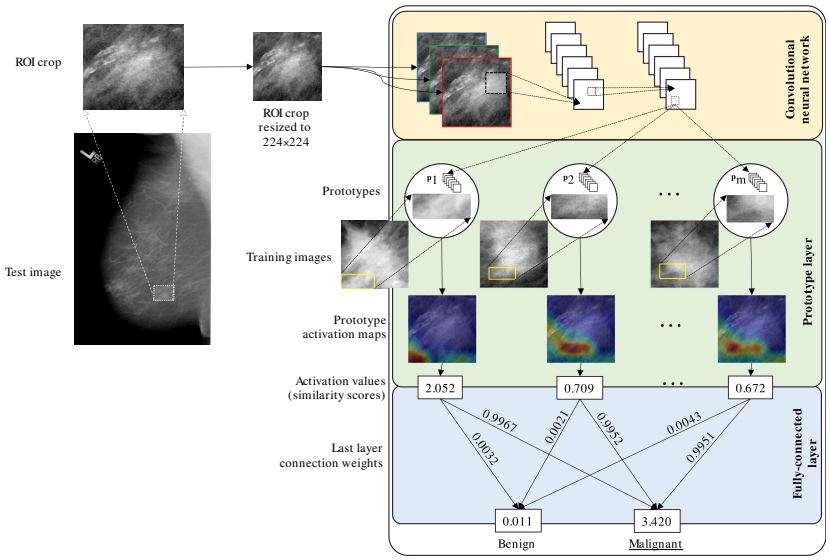

Fig. 2. Inference process through ProtoPNet: classification of a breast mass by means of the activation of pre-learned prototypes within the test image.

3.3 Experiment with ProtoPNet

As for the CNN block of ProtoPNet, the residual network ResNet18 with weights pre-trained on the ImageNet dataset was used in this experiment. Images were resized to a dimension of 224×224 pixels and their values were normalized with *mean* and standard deviation (*std*) equal to 0.5 for the three channels. As a result, image values range between -1 and $+1$ and this helps to improve the training process.

We then performed a random search to optimize the model's hyperparameters. For each configuration, we built a five-fold CV framework for training lesions, creating the internal-training and internal-validation subsets with an 80-20% proportion. We performed the splitting in both class-balanced and patient-stratified fashion; this way, we maintained the balance between the classes and we associated lesions of the same patients to the same subset (internal-training or internal-validation) for each CV fold. We employed the StratifiedGroupKFold function from the scikit-learn library [18] for this purpose.

Given the large number of hyperparameters in ProtoPNet that can be optimized, we investigated only a fraction of them in this work. In particular, we examined the learning rate (LR) at joint epochs, the weight decay (WD), the batch size of the internal-training subset, the coefficients of the ProtoPNet loss function terms, and the number of prototypes per class. Their possible values are reported in Table 1. Among the resulting 2592 configurations, 30 were randomly selected and used for training. The remaining hyperparameters were chosen with fixed values instead. The ones different from the origi-

nal ProtoPNet paper follow: $dropout\_proportion = 0.4$; $add\_on\_layers\_type = bottleneck$; $num\_filters = 512$; $validation\_batch\_size = 2$; $push\_batch\_size = 40$; $warm\_optimizer\_lrs = \{add\_on\_layers : 1e\text{-}6, prototype\_vectors : 1e\text{-}6\}$; and $last\_layer\_optimizer\_lr = 1e\text{-}6$.

At training time, we performed data augmentation on the internal-training subset by adding slightly modified copies of already existing data. Typically, this procedure reduces overfitting when training a machine learning model and acts as regularization. We adopted the following transformations: (i) images underwent rotation around their center by an angle randomly picked in the range $[-10°, +10°]$; (ii) images were perspective skewed, that is, transforming the image so that it appears as if it was viewed from a different angle; the magnitude was randomly drawn from a value up to 0.2; (iii) images were stretched by shear along one of their sides, with a random angle within the range $[-10°, +10°]$; images were mirrored (iv) from left to right along y-axis and (v) from top to bottom along x-axis. Among the presented transformations, those based on a random initialization of certain parameters were repeated ten times each to further augment the number of instances. As a result, considering also the original ones, the number of internal-training images was totally increased by a factor of 33. For such augmentation we exploited the Python Augmentor Library [5], which has been designed to permit rotations of the images limiting the degree of distortion.

Differently from the original study, we used fixed LR values instead of an LR scheduler, and we framed the training process within an early stopping (ES) setting rather than a 1000-epochs one. In particular, we checked the trend of the loss function for ES. We exploited a moving average with $window = 5$ and $stride = 5$ to reduce the influence of noise in contiguous loss values at joint epochs. At every push epoch, a discrete derivative was computed on the two averaged values resulting from the ten joint epochs preceding that push epoch. A non-negative derivative was the condition to be checked. If the condition persisted for the following 30 joint epochs (patience), ES occurred, and the training process stopped. The considered model was the one saved before the 30 patience epochs.

Table 1. Values of the ProtoPNet Hyperparameters for the Random Search

Parameter	Domain
$lr\_features$	$[1e\text{-}7, 1e\text{-}6]$
lr_add_on	$[1e\text{-}7, 1e\text{-}6]$
lr_prot_vector	$[1e\text{-}7, 1e\text{-}6]$
WD	$[1e\text{-}3, 1e\text{-}2]$
$train\_batch\_size$	$[20, 40]$
$clst$	$[0.6, 0.8, 0.9]$
sep	$[-0.1, -0.08, -0.05]$
$l1$	$[1e\text{-}5, 1e\text{-}4, 1e\text{-}3]$
$num\_prots\_per\_class$	$[5, 20, 40]$

Following the random search, we chose the best-performing configuration based on the metrics reported in section 3.4. Hence, we re-trained the model on the whole training set with the selected configuration for as many epochs as the average maximum epoch in the CV folds. We then performed a prototype pruning process, as suggested in the workflow of the original paper [6]. We did that to exclude, from the set of learned prototypes, those that potentially regard background and generic regions in favor of more class-specific ones. Finally, we evaluated the final model on test set images.

In the end, we compared ProtoPNet with a simpler, conventional black-box model. Since our ProtoPNet uses ResNet18 as the CNN block, we repeated the classification task with the same pre-processed dataset using a ResNet18 with weights pre-trained on ImageNet.

We conceived the training framework as a fine-tuning of the last convolutional layers. The fine-tuning was performed under the same five-fold CV settings and with the same data augmentation operations. To reduce the overfitting during training, we also inserted a dropout layer before the final FC layer.

Provided that ProtoPNet and ResNet18 have globally different hyperparameters, an independent random search was performed. The subset of investigated hyperparameters follows: number of re-trained last convolutional layers $= [1, 2, 3, 4, 5, 10, 20]$; LR = [$1e$-7, $1e$-6]; WD = [$1e$-3, $1e$-2, $1e$-1]; and dropout proportion $= [0, 0.2, 0.4]$. Among the 126 possible configurations, 50 were randomly selected for training.

Following the random search, we selected the top-performing configuration according to the metrics outlined in Section 3.4. Accordingly, we re-trained the model on the entire training set with the chosen configuration for a number of epochs equal to the average maximum epoch in the CV folds. Lastly, we evaluated the final model on the test set images.

3.4 Evaluation Metrics

We used both quantitative metrics and a qualitative assessment to evaluate the performance of the models at training time. As for quantitative metrics, we computed the accuracy value and stored it for both the internal training and the internal-validation subsets at each epoch for each CV fold of a given configuration. We then obtained the configuration accuracy with its standard deviation by averaging the best validation accuracy values across the CV folds.

Even though some CV folds might reach high validation accuracy values at some epochs, the overall trend of the validation learning curves could be erratic and noisy over epochs. Hence, we computed the learning curves of accuracy and loss for each configuration and collected them for both internal-training and internal-validation subsets at each CV fold. Then, these curves were averaged epoch-wise to obtain an average learning curve and standard deviation values for each epoch.

We used a qualitative assessment of the average learning curves in combination with quantitative metrics to verify the correctness of the training phase. In this regard, we considered a globally non-increasing or with a high standard

deviation trend as unjustifiable. We then selected the best performing configuration of hyperparameters based on both the configuration accuracy and the quality assessment. When evaluating the model on the test set, we assessed its performance through Accuracy, Precision, Recall, F1 score, F2 score, and AUROC.

3.5 Implementation Environment

All the experiments in this study ran on the AI@Edge cluster of ISTI-CNR, composed by four nodes, each with the following specifications: 1× NVIDIA® A100 40 GB Tensor Core, 2× AMD - Epyc 24-Core 7352 2.30 Ghz 128 MB, 16 x DDR4-3200 Reg. ECC 32 GB module = 512 GB.

We implemented the presented work using Python 3.9.7 on the CentOS 8 operating system and back-end libraries of PyTorch (version 1.9.1, build py3.9-cuda11.1-cudnn8005). In addition, to ensure reproducibility, we set a common seed for the random sequence generator of all the random processes and PyTorch functions.

4 Results

4.1 CBIS-DDSM dataset

The original dataset consisted of 577 benign and 637 malignant masses in the training set and 194 benign and 147 malignant masses in the test set. As a result of the cleaning process, we removed 49 benign and 60 malignant masses from the training set and 48 benign and 16 malignant masses from the test set. Next, based on the more prevalent class in each set, we removed 49 malignant masses from the training set and 15 benign masses from the test set to balance the resulting dataset. Therefore, the final number of utilized masses was 528 for each label in the training set and 131 for each label in the test set.

4.2 Experiment with ProtoPNet

As a result of the internal-training and internal-validation split, each CV fold consisted of 844 and 210 original images, respectively. Then, as a result of the data augmentation, the internal-training subset consisted of 27852 images.

The random search with five-fold CV on the specified hyperparameters yielded the results reported in Table 2. There, values in each configuration belong to the hyperparameter domain of Table 1, and are listed in the same order. For each configuration, we reported the values of mean and standard deviation accuracy across the CV folds.

Based on those values, the best-performing model was obtained in configuration 28, which has the following hyperparameter values: $lr\_features = 1e\text{-}6$; $lr\_add\_on = 1e\text{-}6$; $lr\_prot\_vector = 1e\text{-}6$; $WD = 1e\text{-}3$; $train\_batch\_size = 20$; $clst = 0.8$; $sep = -0.05$; $l1 = 1e\text{-}4$; and $num\_prots\_per\_class = 20$. With this model, the validation accuracy was 0.763 ± 0.034.

The selected model also satisfied goodness of the learning curves, according to the quality assessment (Fig. 3). During the training phase, the ES condition was triggered at epoch 30. Nevertheless, 60 epochs are reported in the plot because of the 30 patience interval epochs.

Table 2. Accuracy Results for the Random Search on ProtoPNet's Configurations

Configuration	$mean \pm std$
0 : [1e-6, 1e-7, 1e-7, 1e-3, 40, 0.6, −0.1, 1e-4, 5]	0.718 ± 0.069
1 : [1e-6, 1e-6, 1e-6, 1e-3, 20, 0.8, −0.1, 1e-3, 40]	0.753 ± 0.038
2 : [1e-6, 1e-7, 1e-7, 1e-3, 20, 0.9, −0.05, 1e-4, 20]	0.746 ± 0.043
3 : [1e-6, 1e-6, 1e-6, 1e-3, 20, 0.9, −0.08, 1e-5, 40]	0.743 ± 0.042
4 : [1e-6, 1e-6, 1e-6, 1e-3, 20, 0.9, −0.05, 1e-5, 40]	0.759 ± 0.035
5 : [1e-7, 1e-6, 1e-6, 1e-3, 40, 0.8, −0.08, 1e-3, 20]	0.706 ± 0.056
6 : [1e-7, 1e-6, 1e-6, 1e-2, 20, 0.8, −0.05, 1e-5, 5]	0.624 ± 0.045
7 : [1e-7, 1e-6, 1e-6, 1e-2, 20, 0.8, −0.1, 1e-3, 20]	0.698 ± 0.082
8 : [1e-7, 1e-6, 1e-6, 1e-2, 20, 0.6, −0.08, 1e-3, 5]	0.700 ± 0.037
9 : [1e-7, 1e-6, 1e-7, 1e-3, 20, 0.6, −0.05, 1e-5, 40]	0.713 ± 0.058
10 : [1e-7, 1e-6, 1e-7, 1e-2, 40, 0.9, −0.05, 1e-5, 5]	0.683 ± 0.042
11 : [1e-7, 1e-6, 1e-7, 1e-2, 40, 0.6, −0.08, 1e-3, 40]	0.697 ± 0.057
12 : [1e-7, 1e-6, 1e-7, 1e-2, 20, 0.6, −0.05, 1e-5, 40]	0.697 ± 0.066
13 : [1e-7, 1e-7, 1e-6, 1e-3, 40, 0.6, −0.08, 1e-4, 5]	0.591 ± 0.055
14 : [1e-7, 1e-7, 1e-6, 1e-3, 20, 0.8, −0.08, 1e-4, 20]	0.683 ± 0.067
15 : [1e-7, 1e-7, 1e-6, 1e-2, 20, 0.9, −0.08, 1e-3, 5]	0.668 ± 0.032
16 : [1e-7, 1e-7, 1e-7, 1e-3, 40, 0.6, −0.05, 1e-4, 5]	0.574 ± 0.030
17 : [1e-7, 1e-7, 1e-7, 1e-3, 20, 0.6, −0.1, 1e-4, 5]	0.679 ± 0.045
18 : [1e-7, 1e-7, 1e-7, 1e-2, 40, 0.6, −0.08, 1e-3, 20]	0.668 ± 0.041
19 : [1e-6, 1e-6, 1e-6, 1e-2, 20, 0.8, −0.05, 1e-5, 5]	0.748 ± 0.019
20 : [1e-6, 1e-6, 1e-6, 1e-3, 40, 0.9, −0.05, 1e-4, 20]	0.736 ± 0.039
21 : [1e-6, 1e-6, 1e-7, 1e-3, 40, 0.6, −0.08, 1e-5, 5]	0.757 ± 0.023
22 : [1e-6, 1e-6, 1e-7, 1e-3, 20, 0.8, −0.05, 1e-3, 20]	0.722 ± 0.018
23 : [1e-6, 1e-6, 1e-7, 1e-3, 20, 0.6, −0.1, 1e-3, 40]	0.762 ± 0.036
24 : [1e-6, 1e-6, 1e-7, 1e-2, 40, 0.6, −0.05, 1e-4, 20]	0.757 ± 0.038
25 : [1e-6, 1e-7, 1e-6, 1e-2, 40, 0.9, −0.1, 1e-3, 20]	0.732 ± 0.055
26 : [1e-6, 1e-7, 1e-6, 1e-2, 40, 0.6, −0.1, 1e-3, 40]	0.745 ± 0.028
27 : [1e-6, 1e-6, 1e-6, 1e-3, 40, 0.8, −0.08, 1e-5, 40]	0.743 ± 0.042
28 : [1e-6, 1e-6, 1e-6, 1e-3, 20, 0.8, −0.05, 1e-4, 20]	$\mathbf{0.763 \pm 0.034}$
29 : [1e-6, 1e-7, 1e-6, 1e-2, 20, 0.9, −0.1, 1e-4, 20]	0.741 ± 0.040

According to the training curves in Fig. 3, we re-trained the selected model on the training set for 30 epochs. After pruning, 9 and 2 prototypes were removed from the benign and the malignant classes, respectively. As a result, 29 final prototypes were retained. Then, we assessed this model on the test set.

Finally, regarding the comparison with ResNet18, we obtained the following results. Among the 50 explored configurations, the best performing model was found with the following hyperparameters: number of re-trained last con-

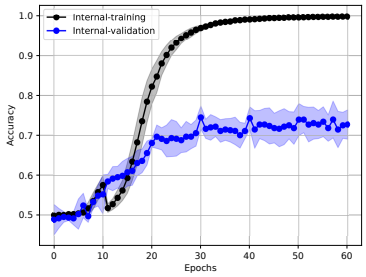

Fig. 3. Average accuracy curves across the five CV folds for the selected ProtoPNet model's configuration. Shaded regions represent $\pm 1 \cdot std$ interval for each epoch.

volutional layers = 3, LR = 1e-6, WD = 1e-3, dropout rate = 0.4. This model reached an average validation accuracy across the five CV folds of 0.776 ± 0.026. After re-training the model on the whole training set for 20 epochs, we evaluated it on the test set images.

The test-set metrics yielded by ProtoPNet and ResNet18 in their independent experiments are reported in Table 3. In Fig. 4, we report an example of an explanation provided by ProtoPNet for a test image of a correctly classified malignant mass. Similarities with prototypes recognized by the model are listed from top to bottom according to decreasing similarity score of the activation. Note that the top activated prototypes correctly derive from training images of malignant masses. Instead, towards the lower scores, prototypes originating from other classes might be activated, in this case of benign masses.

5 ProtoPNet's prototypes: a clinical viewpoint

Specific domain knowledge is necessary to understand and interpret explanations provided by models such as ProtoPNet when applied to medical images. The validity of provided visual explanations is hardly evaluable by someone without a background in the specific task. Furthermore, explanations can be misleading or confusing when analyzed by non-experts.

When dealing with explainable models, one of the first concerns is to assure that explanations are based on correct information. Also, for such models to be interpretable and hence helpful in the medical practice, their explanations should use intuitions that somewhat resemble the reasoning process of a physician. In this regard, we asked a radiologist with 16 years of experience for a clinical

Table 3. Test Set Metrics With Best-performing Models

Model	Accuracy	Precision	Recall	F1	F2	AUROC
ProtoPNet	0.685	0.658	0.769	0.709	0.744	0.719
ResNet18	0.654	0.667	0.615	0.640	0.625	0.671

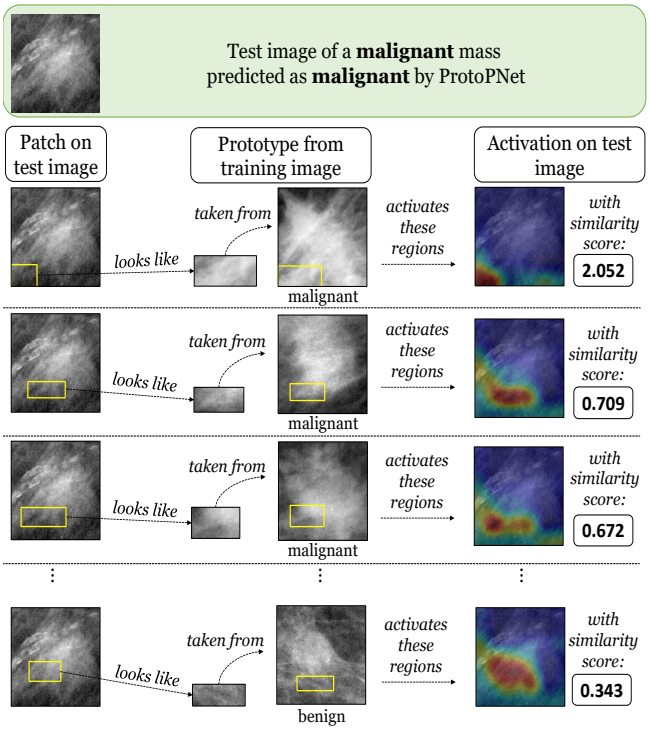

Fig. 4. The test image of a malignant mass is correctly classified as malignant by the model. Each row of this image represents the activation process of a certain prototype. In the first column there is the patch found on the test image, in the second column the activated prototype is shown together with the training image from which it originated, in the third column is shown the activation map with the corresponding similarity score.

viewpoint on the outputs of the selected model on a random subset of test images (15 benign, 15 malignant). In particular, we conceived three tasks (i.e., Task 1, 2 and 3).

As before stated, ProtoPNet bases the classification outcome and the explanation on patch similarities with a set of learned prototypes. Therefore, we first wish to understand whether good-quality prototypes were learned and used to characterize each class. This was done in Task 1. We presented the radiologist with a series of images representing the learned prototypes from both classes and the images from which they were extracted. We asked her to rate how much each prototype was specific for its corresponding class on a scale from one to five. Lower scores would be assigned to generic, not clinically significant prototypes, while class-specific, meaningful prototypes would receive higher scores. As a result of Task 1, only 50% of the benign prototypes were considered to be of acceptable quality, while about 88% of malignant prototypes were deemed good by the radiologist.

Next, we would like to check that ProtoPNet was capable of learning a meaningful concept of similarity. In this sense, image regions that the model recognized as similar should contain comparable clinical information. Therefore, in Task 2, we asked the radiologist to rate the activation of the most activated prototype w.r.t each image in the selected subset. For each case, the activated patch on the test image and the corresponding activating prototype were given. The rating was expressed on a scale from one to five. Activations that shared mutual clinical information would receive higher scores. Regarding Task 2, among the 30 activated patches of the test images, 20 resulted as clinically similar to the activating prototypes according to radiologist's feedback.

Finally, we wished to figure out the degree of satisfaction in medical end-users for the explanations provided. This was carried out in Task 3. We presented the radiologist with test images, each labeled with the classification yielded by ProtoPNet, along with the explanation based on the two most activated prototypes. She provided scores on a scale from one to five for the overall satisfaction of such explanations. A lower score would be assigned to explanations that highlighted non-relevant regions or did not highlight regions on which the radiologist would focus. Instead, if the radiologist believed the explanation to be convincing and complete (i.e., all the relevant regions are identified), she would have returned a higher score. The analysis on Task 3 showed that the radiologist recognized explanations for benign-predicted masses as sufficiently satisfying only in 50% of the cases. On the other hand, explanations for malignant-predicted masses were convincing 89% of the times.

This investigation of the explanation quality of the proposed method, both on the detection of prototypes and the activations correctness, is preliminary. As a by-product, the expert radiologist's feedback is a precious contribution for the design, in the near future, of other tests to assess both the explanation's correctness and of explanation's acceptance by end-users.

6 Discussion

Historically, not knowing precisely why DL models provide their predictions has been one of the biggest concerns raised by the scientific community. Healthcare, in particular, is one of the areas massively impacted by the lack of transparency of such black-box models. That is especially relevant for automatic medical image classification, which medical practice still strives to adopt. Explainable and interpretable AI might overcome this issue by getting insights into models' reasoning. In this regard, a promising approach is that of ProtoPNet [6], an explainable-by-design model firstly introduced in the natural images domain.

Our work aimed at exploring the applicability of prototypical part learning in medical images and, in particular, in the classification of benign/malignant breast masses from mammogram images. We assessed the applicability based on two aspects: the ability of the model to face the task (i.e., classification metrics) and the ability of the model to provide end-users with plausible explanations. We trained a ProtoPNet model and optimized its hyperparameters in a random

search with five-fold CV. Then, we compared its performance to that obtained with an independently optimized ResNet18 model. We selected images from CBIS-DDSM [12], a publicly available dataset of scanned mammogram images. After, came a cleaning and balancing process to obtain the final study cohort. As opposed to the original paper, we utilized a hold-out test independent from the internal-validation subset used at training time to assess the final performance. In addition, we introduced two-dimensional dropout and batch-normalization after each add-on convolutional layer in the ProtoPNet architecture.

Evaluation metrics resulting from the best performing ProtoPNet model seem mostly higher than with the ResNet18 architecture. In particular, we observed the most substantial improvement in the Recall, which is of considerable interest for this specific task. Indeed, it represents the capacity of the model to detect positive cases: a high Recall means that the model correctly identifies the majority of malignant masses. In addition, ProtoPNet provides a level of transparency that is completely missing from ResNet18. That said, it is well known that neural networks often use context or confounding information instead of the information that a human would use to solve the same problem in both medical [25] and non-medical applications [10].

We believe a large amount of prior domain knowledge is necessary to evaluate ProtoPNet's explanations. Without domain knowledge, its results are likely to be misinterpreted. Moreover, such knowledge would be necessary to properly select the number of prototypes for each class, instead of empirically derive it from a hyperparameter optimization. To prevent explanations to be based on irrelevant regions of the images, we asked for the radiologist's viewpoint. In this regard, she provided some helpful insights into the models' outputs. From Task 1, it seems reasonable to assume that ProtoPNet manages to learn more relevant prototypes for malignant masses similar to radiologists. As in actual practice, a suspicious finding (a non-circumscribed contour, whether microlobulated, masked, indistinct, or spiculated), even only in one projection, results easy to detect and justifies a recall for further assessment. On the other hand, a benign judgment requires an accurate bi-dimensional analysis of typical benign findings in both projections and differential diagnoses with overlapping tissue. From Task 2, it appears that the model's mathematical concept of similarity differs from how a radiologist would deem two regions clinically similar. The reason behind this may be that the radiologist recalls specific features from past experience, possibly consisting of other exams aside from mammography and biopsy results alone. This is way broader than the dataset the network uses for training, which strictly consists of image-biopsy label pairs. Finally, from Task 3, results that explanations for images classified as malignant are, in general, more likely to be more convincing to the radiologist. Notably, this behavior goes in the same direction as the low clinical relevance of benign prototypes from Task 1. Overall, the radiologist found ProtoPNet's explanations very intuitive and hence reported a high level of satisfaction. This is remarkably important because we were interested in the right level of abstraction for explanations to foster human interpretability.

Comparing our work with previous studies is not straightforward: no other work with prototypical part learning has been done on the CBIS-DDSM dataset and benign/malignant mass classification task. Nevertheless, we hereafter compare our results with previous works utilizing ResNets on the same dataset and task, albeit some of them in slightly different ways. In the comparison, we report the accuracy as the common performance metric across these studies. In our experiments we achieved an accuracy of 0.654 with ResNet18 and of 0.685 with ProtoPNet. Among the ones using ResNet18, Arora et al. [3] and Ragab et al. [19] achieved an accuracy of 0.780 ad 0.722, respectively. Instead, among the works using different ResNet architectures, Ragab et al. [19] achieved an accuracy of 0.711 and 0.715 when using ResNet50 and ResNet101, respectively. Tsochatzidis et al. [24] deployed ResNet50 obtaining an accuracy of 0.749. Also Alkhaleefah et al. [1] experimented with ResNet50 in different scenarios and achieved accuracy values between 0.676 and 0.802. Finally, Ansar et al. [2] reported an accuracy of 0.637 by using ResNet50. Although performance metrics reported in the previous works are in line with ours, they are, in general, higher.

Regarding previous studies adopting a prototypical part learning scheme to the mass classification task, not much work has been done. To the best of our knowledge, the work by Barnett et al. [4] is the only one, even though the authors utilized a different (and private) dataset and a different novel architecture, derived from ProtoPNet. For these reasons, a fair comparison may not be feasible. Besides, we achieved an AUROC of 0.719 with ProtoPNet, which is lower than theirs (0.840). The authors used images in combination with a dedicated fine-annotation of relevant regions and mass margins, and their model heavily exploits that information for its conclusions. We point out that this is different from our work, where ProtoPNet uses only image-level labels without annotated images to resemble the experimental setup of the original work on bird classification [6]. This is probably one of the reasons for the performance discrepancies between the two studies. However, fine-annotated images needed in their methodology require a massive intervention by clinical experts. Also, intending to deploy such models to fast assist radiologists in the classification of a new image, we believe their approach to be too dependent on annotations, therefore, our approach may be preferable. We likely obtained acceptable results without the complexity of the model and of the dataset of [4].

Interestingly, the performance in [4] is somewhat similar to that obtained on the bird classification task of the original work introducing ProtoPNet [6]. The inclusion of information regarding relevant regions and mass margins annotations might have been the key to achieve such high results on the mass classification task. However, our work shows that, by taking the same annotation-free approach of [6], lower results might be obtained for this task. According to our results, without additional information to complement images, the task to be solved is more challenging, and the problem covers a higher level of complexity. Specifically, in images acquired by projection, planes at different depths are fused in a single bi-dimensional representation. That makes object separation es-

pecially hard for these images. This implies that answering our research question may not be as straightforward as for the ornithology task.

Our work comes with limitations. Firstly, given the large number of hyper-parameters in ProtoPNet, we selected a subset of them for the optimization process. Moreover, of all the possible configurations obtainable with the chosen subset of hyper-parameters, we evaluated the model only on a random selection of them. That likely had an impact on the discovery of the optimal configuration. Secondly, due to the limited size of the utilized dataset, our models were prone to overfitting, which affects the generalization capabilities on new images. That is particularly true for ProtoPNet, where the entire architecture has to be re-trained. That happened even though we took several actions to counteract the issue. Specifically, we selected a shallower ResNet architecture, deployed WD, and introduced dropout and batch-normalization layers. In addition, we provided the clinical viewpoint of a single radiologist. We are aware that this clashes somewhat with the subjective nature of such views: a group of differently experienced radiologists should have been included to reach more robust conclusions.

7 Conclusion and Future Work

Our research question was to investigate the applicability of ProtoPNet to the automatic classification of breast masses from mammogram images. Although a clear-cut answer might not have been provided, this exploratory work allowed us to assess the advantages and the weak points of this kind of approach. The two aspects we considered to evaluate the applicability of this approach were the classification capabilities and the validity of explanations. Classification results were acceptable but insufficient for this method to enter the clinical practice. Based on the clinical assessment, we may say that explanations provided for malignant masses were highly plausible, valuable, and intuitive to a radiologist. However, this is not true for benign masses yet, and this currently invalidates the applicability of ProtoPNet in real clinical contexts. On the other hand, this behavior is comparable to that of a radiologist, who, typically, finds it easier to recognize malignant masses' characteristics. Nevertheless, our findings are promising and suggest that ProtoPNet may represent a compelling approach that still requires further investigation. We believe that training this model on more images or performing a more extensive optimization of the model's architecture may bring improved classification performance. That might also increase the ability of the model to deliver plausible explanations for benign cases.

Future work would include combining several ProtoPNet models with different base architectures together in an ensemble fashion or choosing a Vision Transformer architecture [9] instead of a CNN model at the core of ProtoPNet. In addition, a different initialization for the filter values could be adopted, for example, with values learned on the same dataset using the corresponding base architecture instead of those pre-trained on ImageNet. Moreover, in addition to geometrical transformations, one could also exploit intensity-based transforma-

tions to try improving the networks' generalization capabilities on images possibly obtained with different acquisition settings. These may include histogram equalization and random brightness modification. Also, one could utilize a combination of different mammogram images datasets to augment diversity in the data cohort. On top of that, a dataset comprising digital breast tomosynthesis images instead of conventional digital mammogram images could be used. That is a pseudo-3D imaging technique based on a series of low-dose breast acquisitions from different angles, which has the potential to overcome the tissue superposition issue and thus improve the detection of breast lesions. From a broader point of view, we see the customization of ProtoPNet functioning to produce explanations grounded in causality, instead of correlation, as a promising future work.

Acknowledgment

The research leading to these results has received funding from the Regional Project PAR FAS Tuscany - PRAMA and from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952159 (ProCAncer-I). The funders had no role in the design of the study, collection, analysis and interpretation of data, or writing the manuscript.

References

1. Alkhaleefah, M., Chittem, P.K., Achhannagari, V.P., Ma, S.C., Chang, Y.L.: The influence of image augmentation on breast lesion classification using transfer learning. In: 2020 International Conference on Artificial Intelligence and Signal Processing (AISP). pp. 1–5. IEEE (2020)
2. Ansar, W., Shahid, A.R., Raza, B., Dar, A.H.: Breast cancer detection and localization using mobilenet based transfer learning for mammograms. In: International symposium on intelligent computing systems. pp. 11–21. Springer (2020)
3. Arora, R., Rai, P.K., Raman, B.: Deep feature–based automatic classification of mammograms. Medical & biological engineering & computing **58**(6), 1199–1211 (2020)
4. Barnett, A.J., Schwartz, F.R., Tao, C., Chen, C., Ren, Y., Lo, J.Y., Rudin, C.: A case-based interpretable deep learning model for classification of mass lesions in digital mammography. Nature Machine Intelligence **3**(12), 1061–1070 (2021)
5. Bloice, M.D., Stocker, C., Holzinger, A.: Augmentor: an image augmentation library for machine learning. arXiv preprint arXiv:1708.04680 (2017)
6. Chen, C., Li, O., Tao, D., Barnett, A., Rudin, C., Su, J.K.: This looks like that: deep learning for interpretable image recognition. Advances in neural information processing systems **32** (2019)
7. Ciravegna, G., Barbiero, P., Giannini, F., Gori, M., Lió, P., Maggini, M., Melacci, S.: Logic explained networks. arXiv preprint arXiv:2108.05149 (2021)
8. Cui, Y., Chen, R., Chu, W., Chen, L., Tian, D., Li, Y., Cao, D.: Deep learning for image and point cloud fusion in autonomous driving: A review. IEEE Transactions on Intelligent Transportation Systems (2021)

9. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
10. Hu, S., Ma, Y., Liu, X., Wei, Y., Bai, S.: Stratified rule-aware network for abstract visual reasoning. arXiv preprint arXiv:2002.06838 (2020)
11. Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., Muller, P.A.: Deep learning for time series classification: a review. Data mining and knowledge discovery **33**(4), 917–963 (2019)
12. Lee, R.S., Gimenez, F., Hoogi, A., Miyake, K.K., Gorovoy, M., Rubin, D.L.: A curated mammography data set for use in computer-aided detection and diagnosis research. Scientific data **4**(1), 1–9 (2017)
13. Li, O., Liu, H., Chen, C., Rudin, C.: Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 32 (2018)
14. Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., Gao, J.: Deep learning–based text classification: a comprehensive review. ACM Computing Surveys (CSUR) **54**(3), 1–40 (2021)
15. Mohammadjafari, S., Cevik, M., Thanabalasingam, M., Basar, A.: Using protopnet for interpretable alzheimer's disease classification. In: Proceedings of the Canadian Conference on Artificial Intelligence doi. vol. 10 (2021)
16. Pandey, C., Sethy, P.K., Behera, S.K., Vishwakarma, J., Tande, V.: Smart agriculture: Technological advancements on agriculture—a systematical review. Deep Learning for Sustainable Agriculture pp. 1–56 (2022)
17. Papernot, N., McDaniel, P.: Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning. arXiv preprint arXiv:1803.04765 (2018)
18. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)
19. Ragab, D.A., Attallah, O., Sharkas, M., Ren, J., Marshall, S.: A framework for breast cancer classification using multi-dcnns. Computers in Biology and Medicine **131**, 104245 (2021)
20. Rudin, C., Chen, C., Chen, Z., Huang, H., Semenova, L., Zhong, C.: Interpretable machine learning: Fundamental principles and 10 grand challenges. Statistics Surveys **16**, 1–85 (2022)
21. Singh, G., Yow, K.C.: An interpretable deep learning model for covid-19 detection with chest x-ray images. Ieee Access **9**, 85198–85208 (2021)
22. Singh, G., Yow, K.C.: These do not look like those: An interpretable deep learning model for image recognition. IEEE Access **9**, 41482–41493 (2021)
23. Trinh, L., Tsang, M., Rambhatla, S., Liu, Y.: Interpretable and trustworthy deepfake detection via dynamic prototypes. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1973–1983 (2021)
24. Tsochatzidis, L., Costaridou, L., Pratikakis, I.: Deep learning for breast cancer diagnosis from mammograms—a comparative study. Journal of Imaging **5**(3), 37 (2019)
25. Wang, H., Wu, Z., Xing, E.P.: Removing confounding factors associated weights in deep neural networks improves the prediction accuracy for healthcare applications. In: BIOCOMPUTING 2019: Proceedings of the Pacific Symposium. pp. 54–65. World Scientific (2018)