A Review of Capsule Networks in Medical Image Analysis

A Review of Capsule Networks in Medical Image Analysis

Heba El-Shimy[1 [0000−0001−8465−8050]⋆], Hind Zantout[1 [0000−0002−3804−0513]], Michael Lones[2 [0000−0002−2745−9896]], and Neamat El Gayar[1 [0000−0003−1467−1115]]

[1] Heriot-Watt University, Dubai, UAE
[2] Heriot-Watt University, Edinburgh, Scotland

Abstract. Computer-aided diagnosis technologies are gaining increased focus within the medical field due to their role in assisting physicians in their diagnostic decision-making through the ability to recognise patterns in medical images. Such technologies started showing promising results in their ability to match or outperform physicians in certain specialities and improve the quality of medical diagnosis. Convolutional neural networks are one state-of-the-art technique to use for disease detection and diagnosis in medical images. However, capsule networks aim to improve over these by preserving part-whole relationships between an object and its sub-components leading to better interpretability, an important characteristic for applications in the medical domain. In this paper, we review the latest applications of capsule networks in computer-aided diagnosis from medical images and compare their results with those of convolutional neural networks employed for the same tasks. Our findings support the use of Capsule Networks over Convolutional Neural Networks for Computer-Aided Diagnosis due to their superiority in performance but more importantly for their better interpretability and their ability to achieve such performance on small datasets.

Keywords: Capsule Networks · Deep Learning · CADx · Computer-Aided Detection · Computer-Aided Diagnosis · Medical Imaging

1 Introduction

The concepts of Computer-Aided Detection (CADe) and Computer-Aided Diagnosis (CADx) started developing since the advent of digital image processing techniques and entered the stage of large-scale, systematic research in the 1980s. These systems are able to learn patterns in images that could at times be difficult to detect by the human eye and thus can help clinicians detect and diagnose diseases in medical images. CADe and CADx systems act as a way for physicians to get a second-opinion before making a final decision [1]. Early CADe/x systems used traditional machine learning (ML) techniques which required a manual or separate feature extraction step to be performed by a domain-expert

⋆ he12@hw.ac.uk

before feeding those features into the model as input. These hand-crafted features resulted in a limitation for feature-based ML techniques due to possible human perceptual biases [2]. In recent years, there has been a growing interest in Deep Learning (DL) techniques and the possibilities that they carry for utilising CADe/x systems in the field of medical image analysis. One of the significant advantages of DL techniques over traditional ML is that they do not require prior feature extraction as that is done inherently during model training. DL models can analyse images in more detail and can train millions of parameters which often makes them superior to traditional ML techniques, at least in terms of predictive accuracy, when a lot of data is available.

Tasks that are common in CADe/x systems are image classification, image segmentation or object localisation within a frame or image. All of these tasks have been successfully accomplished using Convolutional Neural Networks (CNNs) with remarkable performance in terms of high accuracy and low false negatives. CNNs can handle high dimensional data such as images, and they are capable of efficiently extracting and learning features that are determinant for the relevant classes. Countless architectures have been proposed for CNNs, but the basic building blocks are convolutional and pooling layers. While convolutional layers act as the feature extractors and provide translational equivariance, the pooling layers act as dimensionality reduction steps via feature aggregation, thus providing spatial invariance. It is this pooling process that Sabour et al. [3] argued causes loss of valuable information and should be replaced by another mechanism leading the authors to propose Capsule Networks (CapsNets) as a novel approach to overcome the limitations of CNNs.

In this paper, we review the most recent literature about applications of CapsNets in CADe/x systems for the detection and diagnosis of diseases in medical images. Our main contributions are: 1) exploring the latest publications on CapsNets and selecting for review the ones that match the criteria of being applied in a medical domain for diagnostic purposes and of using image data, i.e., techniques using signals, text and other formats as input data were excluded; 2) we compare CapsNets performance with that of CNNs where possible; and 3) we draw a conclusion of the best technique to use for CADe/x systems, comment on the limitations of the chosen technique and recommend directions for future research. To the best of our knowledge, this work is the only review of CapsNets applications in the medical domain. The rest of this paper is structured as follows: Section 2 will provide a theoretical background on CapsNets and their advantages over CNNs. In Section 3, we report and summarise the various approaches in the literature utilising CapsNets for CADe/x tasks organised by type of disease and affected organ. In Section 4, we discuss our findings and comment on the potential and limitations of CapsNets based on our research, and finally, Section 5 concludes the paper.

2 Background on Capsule Networks

2.1 Limitations of CNNs

It has been shown in numerous studies that CNNs have important limitations. For instance, they generally require large datasets in order to achieve acceptable performance; this is particularly significant within a medical context, where the collection of large datasets is often clinically infeasible. Another important limitation is their inability to understand global objects, shapes and poses; hence, they tend to perform less well with shape-based tasks than with texture-based tasks [4]. Additionally, [3] argue that there is some loss of information that happens in CNN pooling layers, as they ignore all but areas with the highest response to the convolution operation in layers that preceded the pooling layer. Pooling is an efficient dimensionality reduction method, but it comes at the cost of some information loss, as noted by [3] and [5]. Finally, CNNs are not intrinsically interpretable and are commonly referred to as "black boxes" [6, 7]. Moreover, techniques used to explain CNNs are not always accurate or faithful to the original model [6]. This *opaqueness* of CNNs makes them unfavourable for medical applications where the clinicians need to rationalise a model's decision and provide the patient's right to an explanation for any medical decisions. In conclusion, CNNs can understand objects at the local level, but not as a whole or how the detected features spatially relate to each other.

2.2 Advantages of Capsule Networks

In computer graphics, a program starts with the instantiation parameters of an object, such as the position, size, orientation, deformation, velocity, hue, texture, etc., and uses these parameters to draw the object. CapsNets are able to invert this process by learning from objects in input images their instantiation parameters. CapsNets can thus achieve equivariance. Another advantage of CapsNets is the dynamic routing algorithm that acts as a disentanglement technique to "explain-away" part-whole relationships using a structure that resembles a *parse-tree*. Dynamic routing in capsule networks can be similar to the purpose of self-attention in transformers in trying to understand part-whole relationships [8]. Attention maps in transformers can be mapped to routing coefficients in CapsNets; the difference is that in transformers the attention is computed top-down while in CapsNets the routing coefficients are computed bottom-up [9]. This quality of CapsNet of understanding part-whole relationships allows for better interpretability that is intrinsic in the network and which allows rationalising a CapsNet model's decision, a process that is crucial to medical applications.

2.3 Capsule Network Architecture

A CapsNet consists of two components: an encoder and a decoder, as shown in Figure 1. The first layer in the network is a regular convolutional layer, as in any CNN, followed by a second convolution operation where the output feature

maps are reshaped into blocks of width, height and depth, and containing groups of neurons called *capsules* which are the basic building block for CapsNets. The intuition behind having these capsules is to learn the instantiation parameters of entities in an image. The output of each capsule is a vector containing the encoded properties learnt about a single entity. The probability of the existence of an entity is represented by the length of the output vector. A *squashing* function (Eq. 1) is used on each capsule to introduce non-linearity by driving the vector length value closer to 0 for short vectors and closer to 1 for long vectors. The first layer with capsules is called the "Primary Capsules" layer.

$$v_j = \frac{\|s_j\|^2}{1 + \|s_j\|^2} \frac{s_j}{\|s_j\|} \tag{1}$$

as in [3], where s_j represents a given capsule, j, and v_j is the output of capsule j after normalising or *squashing*.

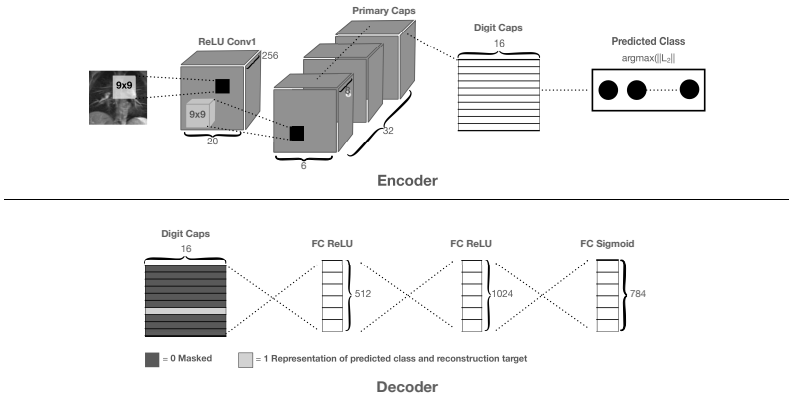

Fig. 1. Architecture of a Capsule Network adapted from [3]

The second capsule layer as in [3] is called "Digit Capsules." It is responsible for the prediction and is expected to have a number of capsules equalling that of the classes in the dataset. Routing between primary capsules and digit capsules is what [3] named "Dynamic Routing" where lower layer capsules *choose* which higher layer capsules to send their information to. In dynamic routing, the lower layer capsules try to predict the output of every higher layer capsule by using a transformation matrix. By multiplying the weights in the transformation matrix with the lower layer capsule output, we get the predicted output of higher layer capsules as in Eq. 2. The predicted output is used to calculate *routing coefficients* c_{ij} which represent the likelihood that a lower layer capsule connects with a higher layer one.

$$\hat{u}_{j|i} = W_{ij} u_i \tag{2}$$

Routing coefficients of a certain layer should sum up to 1, hence c_{ij} is calculated as a softmax of log prior probabilities b_{ij} (Eq. 3). Log priors b_{ij} are initialised to zeros and are learnt over several iterations discriminatively along with other weights in the network and is dependent on the capsules types and locations. They are refined by adding a scalar value representing the *agreement* between two capsules, which can be calculated as a similarity score between the predicted outputs and the activity vector of a higher layer capsule as in Eq. 4. By the end of the training process, the Digit Capsules will contain the instantiation parameters of each class and the probability of the existence of a certain class can be calculated as the magnitude of its corresponding vector. The output vector with highest magnitude is the detected class.

$$c_{ij} = \frac{exp(b_{ij})}{\sum_k exp(b_{ik})} \tag{3}$$

$$b_{ij} = b_{ij} + \hat{u}_{j|i}.v_j \tag{4}$$

The second part of the network is the decoder. In this part, the decoder takes the output vector of the detected class from the encoder while other capsules are masked, and feeds that to a series of fully connected layers to reconstruct the original image. The loss of the network is calculated over two stages: firstly, the Margin Loss is calculated for the encoder, and secondly, a Reconstruction Loss is calculated for the decoder. The margin loss (Eq. 5) is calculated for each output capsule and aims to allow a certain capsule of class k to have a long instantiation vector if and only if the entities of that class exist in the image. The total margin loss is the sum of all digit capsule losses. The reconstruction loss on the other hand is calculated using mean squared error (MSE) between the reconstructed image and the original. The total loss of a Capsule Network is the sum of the margin and the reconstruction losses with a down-weighing factor for the decoder loss (0.0005) so as not to dominate the margin loss.

$$L_k = T_k \max(0, m^+ - \|\mathbf{v}_k\|)^2 + \lambda(1 - T_k)\max(0, \|\mathbf{v}_k\| - m^-)^2 \tag{5}$$

as in [3] where $T_k = 1$, $m^+ = 0.9$, $m^- = 0.1$, $\lambda = 0.5$ is a weighing factor to minimise the effect of loss of absent digits from shrinking the length of other digits' activity vectors.

3 Applications of Capsule Networks on Medical Images

In this section, we will review the latest approaches in the literature on using CapsNets for CADe/x tasks. We organise the literature reviewed by the part of the body and the type of disease on which CapsNets have been used for detection and diagnosis. This is not a comprehensive review, and the techniques reviewed below are only examples representative of CapsNets potential for CADe/x tasks.

3.1 Brain Injuries and Tumours

In a work by Cheng et al. [10], a novel approach for classifying brain tumours in Magnetic Resonance Images (MRIs) was proposed. The authors used a CapsNet for identifying existing tumours and predicting their types from one of three categories: meningioma, glioma, or pituitary tumour. The dataset used contained 3,064 images and is imbalanced. The proposed model accepts two inputs; the first input is a 512×512 full image of a brain MRI scan, and the second input is an image with only the brain tumour region with a size of 128×128. A max-pooling operation was applied to the larger input to reduce its size to 128×128 to decrease the number of parameters in the image. Then, two convolutional layers were applied to each input separately to extract a feature map for each. After that, the resulting feature maps were merged depth-wise and were used as the input for Primary Caps. The output of the network is produced by a "Data capsule" layer which directly succeeds the Primary Caps layer and consists of three capsules representing the three possible classes. Another novelty in this work was the proposal of a modified total loss function where the authors replaced the reconstruction loss with a classification loss for the last layer in the network and calculated that as a cross-entropy loss. The authors argue that the reconstruction loss has no significant effect on the classification, thus they removed it from the total loss function for the model. The reported results of this approach were very good, with an average performance of 93.5% accuracy. The authors compared their model to other architectures they implemented including the original CapsNet as in [3] and a CNN as in [11], and the proposed approach outperformed both. Commenting on the results reported in this work, more metrics were needed to better evaluate the model's performance such as sensitivity, specificity and F1-score, especially since the dataset was imbalanced. Moreover, the authors applied a max-pooling operation on one of the inputs which opposes the concept of CapsNets which refrains from using pooling operations to avoid loss of information.

Afshar et al. [12] started a series of works on using CapsNets for the classification of brain tumours in MRI scans. The dataset used for all works was the same and consisted of 3,064 images of three malignancy categories in addition to the tumour region coordinates for each image. In their first paper [12], they explored different CapsNets architectures that could potentially perform best on the given task. They used the original CapsNet architecture as in [3] but changed some of its hyperparameters. The reported results showed that CapsNets with feature maps reduced from 256 down to 64 resulted in the highest accuracy. They also reported that CapsNets outperform a CNN architecture adopted from [11]. In a later work [13], the authors explored the effect of feeding tumour region coordinates to CapsNets alongside the full brain scan. They were motivated by the hypothesis that CapsNets are sensitive to background noise in images and have the tendency to model everything, and with the high level of detail captured in MRIs, this could impede CapsNets performance. They fed the full brain scans into a CapsNet based on an architecture that they developed in their previous work [12]. The tumour boundary coordinates were concatenated

with the unmasked output vector, i.e., the vector of the highest magnitude, and were fed into two fully connected layers. The final layer of the proposed architecture was a softmax layer which output the probability of each class being present. The rationale for concatenating the tumour boundary coordinates with the last capsule layer output was to get the network to pay more attention to the tumour region and not get distracted by extra details in the image. The results for this approach showed an enhanced performance over the authors' previous work [12] with 90.89% accuracy. Additionally, this approach outperformed feeding segmented tumours to the same CapsNet with an increase of 4.33% in accuracy. The results only reported the accuracy of the model with no tracking of other metrics. In a later paper [14], the authors explored the interpretability of CapsNet by examining the learned features and determining whether they are discriminative of the three types of brain tumours using T-distributed Stochastic Neighbour Embedding (t-SNE) visualisation technique as in [15]. The authors' findings were that the CapsNet was able to successfully distinguish between the classes with good separability. However, the authors provided a visualisation of the input features and there was minimum overlap between the data points from different classes suggesting ease of separability even with simpler models. The authors went on to reconstruct the input from the learned features of the predicted class. To understand what features have been learned by the CapsNet the authors repeatedly tweaked the output features in the last capsule layer and reconstructed the input. Then, they tried to relate the tweaked and reconstructed inputs to hand-crafted features that human experts use to characterise a tumour such as tumour size, maximum horizontal and vertical diameter and tumour centre. They found that the features learned by the CapsNet were correlated with the hand-crafted features. This finding suggests that CapsNets provide better interpretability than CNNs and can be used for clinical applications where the workings of a model are as important as the prediction. In their final work in this series [16], the authors proposed a novel architecture of a CapsNet with an internal boosting mechanism. Boosting is a committee-based machine learning technique that starts with a weak learner, i.e., model, and repeatedly trains this model while adding more weight to misclassified samples at each iteration, resulting in a better performing model at the end of the training cycle. The authors built on their previous work [13] and reused the CapsNet architecture; the only difference was the training process as described. The results of this work provided more insight into the model's performance as the authors tracked accuracy, sensitivity, specificity and area under the ROC curve (AUC) for the receiver operating characteristic (ROC) curves for each class. They compared their proposed model to the basic CapsNet as in [3] and the results showed that their model outperformed CapsNet with 92.45% accuracy. BoostCaps had better specificity per-class than CapsNet with an average increase of 4.33% . However, the BoostCaps network was behind in sensitivity and AUC for one class, Glioma, and had the same AUC score as that of CapsNet for the Pituitary class. The authors did not investigate the features learned by BoostCaps and did not com-

ment on whether the ensemble technique they used resulted in better learning of tumour characteristics in the images.

3.2 Ophthalmology

The work by Koresh et al. [17] used a CapsNet with its original architecture as in [3] to classify images of the cornea into one of two categories: noiseless or noisy. The dataset used in this work was captured by an Optical Coherence Tomography (OCT) device which captures a cross section of the cornea for thickness assessment. The dataset was small consisting of only 579 images, and no specific preprocessing was mentioned. The authors tracked the accuracy, specificity, sensitivity, positive predictive value (PPV) and negative predictive value (NPV) of the model and reported that the CapsNet achieved 95.65% accuracy, 96.08% sensitivity, 80% specificity, 99.42% PPV and 36.36% NPV on images belonging to all patients. The hyperparameters for the CapsNet were not given.

In another work by Tsuji et al. [18], a slightly modified CapsNet architecture was used for the classification of diabetic neuropathy as a screening system to avert the danger of vision loss in diabetic patients. The dataset consisted of 84,484 OCT images of four classes that are imbalanced. The images were resized to 512×512 and augmented by shifting by 16 pixels in each direction. The CapsNet architecture the authors proposed added four convolutional layers before the primary caps layer, then a final capsule layer with four capsules representing the four possible classes as output. The metrics tracked were only the loss and accuracy, and the CapsNet was compared to an Inception V3 model as in [19]. The results reported a higher accuracy for the Inception model 99.8% compared with 99.6% for the CapsNet model. The authors argue that the comparison is favourable to CapsNet due to its shallowness compared to the Inception model; however, more data were needed about trainable parameters for both models to support the authors' argument.

On the other hand, Koresh et al. [20] used a modified CapsNet for an image classification followed by an image segmentation task on corneal images from OCT devices. The dataset used in this work contained 579 images. The task was to assess corneal thickness prior to LASIK surgery, and for this task the proposed system needed to use two CapsNets over two stages: the first stage used a Class CapsNet as in [3] to classify noisy corneal images and filter them out, and in the second stage, another CapsNet architecture called SegCaps as in [21, 22] with a slight modification was employed to segment the corneal layer in the image for the human expert to measure its thickness. The Class CapsNet scored 96.41%, 96.83%, 83.33%, 99.46%. 45.45%, 98.12% on accuracy, sensitivity, specificity, PPV, NPV and F1-score, respectively. The modified SegCaps had an average Dice coefficient accuracy of 97.17%. In the classification task, the CapsNet outperformed other CNN models, and in the segmentation task, the SegCaps outperformed U-Net which is a CNN-based segmentation model. One statement by the authors that could be argued against is that CapsNet have the ability to "visualise" images even if there is heavy noise in them. This is contrary to repeated findings in the literature that one of the limitations of CapsNets

is their attempt to model everything in an image, hence, being affected by any noise.

3.3 Cardiac Diseases

In a work by He et al. [23], a novel capsule-based automated segmentation technique was proposed. The authors used cardiac MRIs to segment the Left Ventricle (LV) which plays a key role in quantifying the volumetric functions of the heart. The dataset they used consisted of 1,720 images and they applied a series of transformations to the images prior to feeding them to the network. The preprocessing transformations start with Fast Fourier Transform (FFT) and inverse FFT followed by Canny edge detection and finally, Hough circle detection and Gaussian kernel. After that, the transformed images are fed to SegCaps as in [22] which is a CapsNet architecture modified for image segmentation tasks. The output of SegCaps undergoes further processing by applying Otsu adaptive thresholding algorithm as in [24] to remove residual noise from the segmentation step. The metrics used to measure the model's performance were Dice coefficient, Jacquard coefficient, Average Symmetric Distance (ADS) and Hausdorff Distance (HD). The authors' proposed approach achieved the best results with 92.2% Dice score, 85.9% Jacquard score, 0.226 mm ASD and 3.541 mm HD and a P value of $< 10^{-6}$ and $< 10^{-13}$ when comparing against other approaches such as SegCaps with preprocessing only and SegCaps with postprocessing only, respectively. The reported results supported by the P values of comparative tests are significant and the authors' approach was successful in segmenting LV in cardiac MRIs; however, as the authors mentioned, the results were not compared with other LV segmentation techniques in the literature. Additionally, the authors did not include clinical measures or detect multiple objects in an image.

Bargsten et al. [25] presented an optimised CapsNet for segmentation of blood vessel lumen and wall in Intravascular Ultrasound (IVUS) images. There were two datasets used for this work: one containing 435 annotated frames with a size of 384×384, and the second dataset contained 77 images with a size of 512×512. The authors preprocessed all the images by resizing them into 256×256 and applying random rotations and flips on-the-fly during training. The authors used a CapsNet architecture called Matwo-Capsnet cited in [21] intended for image segmentation tasks. Matwo-CapsNet hyperparameters were optimised to work on IVUS images, as these kind of images usually contain shadows of artefacts that may obscure the target tissues as well as containing speckle noise which tends to make borders between tissues unclear and hard to detect. The objective function used was margin loss as in [3]. The authors used Dice coefficient and Hausdorff distance for evaluating the model's performance. The network that had the best performance had 420,000 trainable parameters, implemented dual routing with three iterations and applied normalisation for the pose matrix as proposed in [21]. The best performing network scored 73.09% on Dice coefficient for vessel wall and 90.84% on the lumen, which is higher than that of a U-Net model cited in [26], trained on the same data, by 2.83% and 0.27%, respectively. The best model's results on average Hausdorff distance were 0.085 mm on the vessel wall

segmentation and 0.022 mm on lumen segmentation. The Hausdorff results were better than those for the U-Net model, which suggests better accuracy for the CapsNet over CNN-based segmentation techniques. The authors also noted that CapsNet models had smaller standard deviations, suggesting better stability across trainings.

3.4 Pulmonary Diseases

In a work by Tiwari et al. [27], CapsNet was integrated with other state-of-the-art CNN architectures such as DenseNet as in [28], VGG16 as in [29], ResNet-50 as in [30] and MobileNet as in [31]. The proposed models were named DenseCapsNet, VGGCapsNet, ResCapsNet and MobileCapsNet and were used to detect COVID-19 in lung CT scans. The dataset used contained 2,482 images of two balanced classes. The images were resized to 128×128. Each of the CNN architectures was pretrained and the full structure was kept except the last dense layer. The weights of all layers were frozen, then the resulting feature maps were passed to the standard CapsNet architecture by [3]. The authors reported that the basic CapsNet performed poorly on lung CT scans and that the proposed architectures outperformed CapsNet. All models had an accuracy of 99%, sensitivity of 99% and F1-score of 99% except ResCapsNet. In similar work by Yousra et al. [32], an ImageNet-pretrained VGG19 network was fused with a CapsNet to detect COVID-19 in chest X-rays. The dataset contained 3,310 images of three imbalanced classes. The authors resized the images to 224×224, normalised and augmented them by applying random rotation, horizontal flips and scaling. The model's performance was evaluated using accuracy, precision and F1-score on which the model scored 94%, 95% and 94%, respectively.

Karnkawinpong et al. [33] used CapsNet for detecting Tuberculosis (TB) in chest X-rays. The images were resized to 32×32 pixels and randomly rotated by 10 degree in each direction to increase the number of images in the dataset. The model architecture was similar to the basic CapsNet in [3] but with two convolutional layers at the beginning of the network instead of one. The model performed worse (86.86% accuracy) than CNN-based models in comparison such as VGG16 (90.79% accuracy) when tested on images with the same rotation range of the training set. However, the performance of the CNN-based models quickly deteriorated (74.17% accuracy) when tested on images with larger rotations of 30 degrees. CapsNet, on the other hand, showed better robustness towards image rotation and outperformed models in comparison (80.06% accuracy) by a large margin. This result evidences the ability of CapsNet to provide equivariance better than CNNs. It should be noted that significant information would have been lost when the images were downsized, and this may have limited the performance of the CapsNets in this study.

The work by LaLonde et al. [34] was one of the first and few works to investigate the instantiation parameters learned by CapsNets. Their intuition was to examine the ability of capsules to model specific visual attributes within the learned instantiation parameters. The authors proposed a novel architecture, X-Caps, to classify lung nodules in CT scans into one of six classes based on their

visual attributes as described by expert radiologists. The dataset contained 1018 CT scans, no preprocessing was applied to the images but downsizing to 32×32. The number of the output classes was set to six, representing the number of visual attributes needed to characterise a tumour such as sphericity, margin, lobulation, texture, subtlety and spiculation. The authors tried to keep the architecture of X-Caps similar to that of the standard CapsNet in [3] but only introduced a few modifications. Firstly, they modified the dynamic routing algorithm to allow child capsules to route information to all parent capsules instead of letting them "choose" one as in the standard CapsNet. This modification allowed a nodule to score high or low on multiple visual attributes at once. Secondly in a branch, the authors attached a fully connected layer to the class capsules, i.e., visual attributes capsules, which served as supervised labels for the X-Caps output vectors. The output of this fully connected layer were a range of scores (1-5) for each visual attribute with 1 interpreted as low evidence of the corresponding attribute and 5 being high evidence. Finally, the authors reconstructed the input images using the X-Caps output vectors and applying subtle variations each dimension to ensure that the desired attributes are being modeled correctly. The results of their work in terms of accuracy in detecting a tumour was higher than other comparable CNN techniques and the standard CapsNet. More importantly in terms of accuracy for each visual attribute, their work outperformed that of an explainable CNN-based technique, 3D Dual-Path HSCNN as in [35], in every attribute by a margin of >10%. This work was able to provide evidence that CapsNets are more interpretable than their counterparts.

A novel approach was proposed by Afshar et al. [36] for the detection of lung nodule malignancy in CT scans and was called 3D-MCN. The authors used three independent CapsNets and fed each one of them inputs from three different sets of training data at different scales. The output vectors of the three CapsNets were concatenated and fed into a fusion module consisting of fully connected layers. The final output of the proposed model is the probability of a nodule being benign or malignant, a binary classification problem based on the information learnt from the three input images scales. The proposed model was evaluated based on accuracy, specificity, sensitivity and AUC, and was compared to other CNN architectures including a 3D-CNN model that the authors developed to mimic their CapsNet-based model in having three independent networks fused. The accuracy of the 3D-MCN model at the final layer was 93.12%, and sensitivity and specificity were 94.94% and 90%, respectively. Although the proposed model was outperformed by a CNN trained with expert hand-crafted features, the 3D-MCN had the highest sensitivity, which is often more important in medical diagnosis as it means the model rarely misclassifies patients who do have a tumour.

4 Discussion

CapsNets have been evaluated on medical images in a number of works in the literature and the results suggest they can have significant benefits over their CNN

counterparts. CapsNets were successfully applied to different imaging modalities with diverse underlying features and quite different tasks. Standard CapsNets as cited in [3] might not be the best architecture to use when working on medical images classification tasks that require preprocessing, for example to remove noise, and the addition of several convolutional layers before the Primary Caps may be beneficial as the results in the literature show. For segmentation tasks, several architectures have been proposed to solve these tasks and have outperformed well-known CNN-based segmentation architectures. We have summarised the reviewed techniques based on CapsNets and their performance compared to CNN-based techniques in Tables 1 and 2 below.

Table 1: Summary of CapsNet-based architectures for classification of medical images and comparison with CNN-based techniques

Technique	**Use Case**	**Dataset**	**Preprocessing**	**Hyperparameters (CapsNet)**	**Accuracy**	**Comparison**
Multi-input CapsNet by [10]	Diagnosing brain tumours in MRIs	3,064 images, 3 classes, imbalanced	N/A	Batch size: 30 Learning rate: 0.0001 Routing iterations: 4	**93.5%**	CNN by [11], 72.12% accuracy
CapsNet by [12]	Diagnosing brain tumours in MRIs	3,064 images, 3 classes, imbalanced	N/A	Optimizer: Adam Epochs: 10	**78%**	CNN by [11], 61.97% accuracy
CapsNet by [13]	Diagnosing brain tumours in MRIs	3,064 images + tumour coordinates, 3 classes, imbalanced	N/A	Batch size: 16 Learning rate: 0.01 Routing iteration: 3 Optimizer: Adam Epochs: 50 LR decay: 0.9	**90.89%**	Modified CNN based on [11], 88.33% accuracy
BoostCaps by [16]	Diagnosing brain tumours in MRIs	3,064 images + tumour coordinates, 3 classes, imbalanced	Not detailed, input size = 128×128	Batch size: 16 Routing iterations: 3 Epochs: 100	**92.45%**	-
CapsNet by [17]	Filtering corneal images in OCTs	579 images, 2 classes	N/A	N/A	**95.65%**	CNN (MATLAB/ unspecified), 87.5% accuracy
Modified CapsNet by [18]	Diabetic retinopathy screening	84,484 images, 4 classes imbalanced	Resizing to 512×512, shifting up to 16px all directions	Batch size: 128 Optimizer: Adam Epochs: 50	99.6%	Inception V3 as as in [19], **99.8%** accuracy
Mobile-CapsNet by [27]	COVID-19 Detection in CT scans	2,482 images, 2 classes, balanced	Resizing to 128×128	Optimizer: Adam Learning rate: 0.0015 Batch size: 64 Epochs: 100	**99%**	MobileNet as in [31], 98% accuracy
CNN-	COVID-19	3,966	Resizing	Optimizer: SGD	94%	-

Continued on next page

Table 1 – continued from previous page

Technique	Use Case	Dataset	Preprocessing	Hyperparameters (CapsNet)	Accuracy	Comparison
CapsNet by [32]	Detection in X-rays	images, 3 classes, balanced	to 224×224, normalisation, rotation, horizontal flips, scaling	Learning rate: 1e-5 Batch size: 64 Epochs: 120 Dropout: 0.25		
Modified CapsNet by [33]	TB Detection in X-rays	3,310 images, 2 classes, imbalanced	Resizing to 32×32, rotation ($\pm 10^{\circ}$)	Batch size: 64 Epochs: 52 Optimizer: Adam Learning rate: 0.001	80.06-86.68%	VGG16 as in [29], 74.17-90.79% accuracy
X-Caps by [34]	Lung Malignancy Detection in CT	1,018 CT scans, 2 classes (tumour=0/1), 6 visual attributes in second branch	N/A	Batch size: 16 Optimizer: Adam Learning rate: 0.02 LR decay: 0.1	**86.39%** in tumour detection	3D DualPath HSCNN% as in [35], 84.20% accuracy in tumour detection
3D-MCN by [36]	Lung Malignancy Detection in CT	2,283 images, 2 classes	Patch extraction with different scales, normalisation, random flipping	Optimizer: Adam	**93.12%**	ResNet as in [30], 89.90% accuracy

Table 2: Summary of CapsNet-based architectures for segmentation of medical images and comparison with CNN-based techniques

Technique	Use Case	Dataset	Preprocessing	Hyperparameters (CapsNet)	Dice Score	Comparison
SegCaps by [20]	Corneal image Segmentation in OCTs	579 images, 2 classes	De-noising, grayscale, extracting region-of-interest	N/A	**97.17%**	U-Net as in [26], 96.06% Dice score
Modified CapsNet by [23]	Segmentation of Left Ventricle	1,720 images, 2 classes	Random translation, rotation, scaling, sheer	Optimizer: Adam Epochs: 10,000 Learning rate: 0.001 LR decay: 0.05	**92.2%**	-
Optimised Matwo-CapsNet by [25]	Segmentation of Intravascular Ultrasound images	512 images, 3 classes	Resized to 256×256, random rotation, flipping	Optimizer: Adam Learning rate: 0.001 Epochs: 200	avg. **81.96%**	U-Net as in [26], avg. 80.42% Dice score

5 Conclusions and Recommendations for Future Work

Capsule networks have been proposed as an improvement over CNNs and with the promise of providing equivariance without the loss of information that might happen during the pooling operation in CNNs. In this work, we reviewed approaches in the literature that utilised CapsNets for disease diagnosis in medical images. CapsNet have continued to show improved results over CNNs and outperformed more complex and deeper architectures with significantly smaller datasets. Moreover, CapsNets are showing a potential for increased model interpretability due to the learned instantiation parameters embedded in the capsules. Both of the mentioned characteristics of CapsNets—being able to learn from small datasets and their interpretability—are important for models operating in the medical field, and we can conclude that CapsNets could be more suitable than CNNs for being deployed in CADe/x systems. There are a few limitations for CapsNets: a) they are very computationally expensive to train and, in some cases, to run, which might hinder their deployment on conventional machines found in medical institutions; and b) they require some image preprocessing to remove noise in the input images due to their tendency to model everything which might impact their performance. We recommend more work on improving CapsNet architecture, more efficient architectures are needed that have far less trainable parameters without impact on the performance. This will help with deploying CapsNets on various machines and not being limited to ones with powerful hardware. Finally, we recommend more efforts are directed towards CaspNets interpretability and investigating what these networks learn and how we can seize this potential into advancing the eXplainable Artificial Intelligence (XAI) and Trustworthy AI fields.

References

1. Doi, K.: Computer-aided diagnosis in medical imaging: Historical review, current status and future potential. Computerized Medical Imaging and Graphics **31**(4), 198–211 (2007). https://doi.org/10.1016/j.compmedimag.2007.02.002
2. Byrne, M. F., Chapados, N., Soudan, F., Oertel, C., Linares Pérez, M., Kelly, R., Iqbal, N., Chandelier, F., Rex, D. K.: Real-time differentiation of adenomatous and hyperplastic diminutive colorectal polyps during analysis of unaltered videos of standard colonoscopy using a deep learning model. Gut **68**, 94–100 (2019). https://doi.org/10.1136/gutjnl-2017-314547
3. Sabour, S., Frosst, N., Hinton, G.E.: Dynamic routing between capsules. In: Advances in Neural Information Processing Systems. **2017-December**, pp. 3857–3867. Neural information processing systems foundation (2017).
4. Geirhos, R., Michaelis, C., Wichmann, F.A., Rubisch, P., Bethge, M., Brendel, W.: Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In: 7th International Conference on Learning Representations, ICLR 2019. ICLR (2019). https://doi.org/10.1136/gutjnl-2017-314547
5. Nguyen, H.P., Ribeiro, B.: Advanced Capsule Networks via Context Awareness. In: Tetko, I., Kůrková, V., Karpov, P., Theis, F. (eds) Artificial Neural Networks and Machine Learning – ICANN 2019: Theoretical Neural Computation. ICANN 2019.

Lecture Notes in Computer Science **11727**, pp. 166–177. Springer International Publishing (2019).
6. Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature Machine Intelligence **1**(5), 206–215 (2019). https://doi.org/10.1038/s42256-019-0048-x
7. London, A. J.: Artificial Intelligence and Black-Box Medical Decisions: Accuracy versus Explainability. Hastings Center Report **49**, 15–21 (2019). https://doi.org/10.1002/hast.973
8. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems **2017-December**, pp. 5999–6009. Neural information processing systems foundation (2017).
9. Abnar, S.: From Attention in Transformers to Dynamic Routing in Capsule Nets. Samira Abnar (blog), https://samiraabnar.github.io/articles/2019-03/capsule. Last accessed 30 Aug 2022
10. Cheng, Y., Qin, G., Zhao, R., Liang, Y., Sun, M.: ConvCaps: Multi-input Capsule Network for Brain Tumor Classification. In: Gedeon, T., Wong, K. W., Lee, M. (eds) Neural Information Processing. ICONIP 2019. Lecture Notes in Computer Science **11953**, pp. 524–534. Springer International Publishing (2019). https://doi.org/10.1007/978-3-030-36708-4_43
11. Paul, J. S., Plassard, A. J., Landman, B. A., Fabbri, D.: Deep learning for brain tumor classification. In: Andrzej, K., Barjor, G. (eds) Progress in Biomedical Optics and Imaging – Proceedings of SPIE **10137**, pp. 1013710. SPIE (2017). https://doi.org/10.1117/12.2254195
12. Afshar, P., Mohammadi, A., Plataniotis, K.N.: Brain Tumor Type Classification via Capsule Networks. In: 25th IEEE International Conference on Image Processing – ICIP 2018. ICIP 2018. Pp. 3129–3133. IEEE Computer Society (2018). https://doi.org/doi:10.1109/ICIP.2018.8451379
13. Afshar, P., Plataniotis, K.N., Mohammadi, A.: Capsule Networks for Brain Tumor Classification Based on MRI Images and Coarse Tumor Boundaries. In: 44th IEEE International Conference on Acoustics, Speech and Signal Processing – ICASSP 2019. ICASSP 2019. **2019-May**, pp. 1368–1372. Institute of Electrical and Electronics Engineers Inc. (2019). https://doi.org/10.1109/ICASSP.2019.8683759
14. Afshar, P., Plataniotis, K.N., Mohammadi, A.: Capsule Networks' Interpretability for Brain Tumor Classification Via Radiomics Analyses. In: 26th IEEE International Conference on Image Processing – ICIP 2019. ICIP 2019. **2019-September**, pp. 3816–3820. IEEE Computer Society (2019). https://doi.org/10.1109/ICIP.2019.8803615
15. Van Der Maaten, L., Hinton, G.: Visualizing data using t-SNE. Journal of Machine Learning Research **9**, 2579–2625 (2008).
16. Afshar, P., Plataniotis, K.N., Mohammadi, A.: BoostCaps: A Boosted Capsule Network for Brain Tumor Classification. In: 42nd Annual International Conferences of the IEEE Engineering in Medicine and Biology Society – EMBC 2020. EMBC 2020. **2020-July**, pp. 1075–107. Institute of Electrical and Electronics Engineers Inc. (2020). https://doi.org/10.1109/EMBC44109.2020.9175922
17. Koresh, H.J.D., Chacko, S.: Classification of noiseless corneal image using capsule networks. Soft Comput **24**, 16201–16211 (2020). https://doi.org/10.1007/s00500-020-04933-5
18. Tsuji, T., Hirose, Y., Fujimori, K., Hirose, T., Oyama, A., Saikawa, Y., Mimura, T., Shiraishi, K., Kobayashi, T., Mizota, A., Kotoku, J.: Classification of optical

coherence tomography images using a capsule network. BMC Ophthalmology **20**, 114 (2020). https://doi.org/10.1186/s12886-020-01382-4
19. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the Inception Architecture for Computer Vision. In: 29th IEEE Conference on Computer Vision and Pattern Recognition – CVPR 2016. CVPR 2016. **2016-December**, pp. 2818–2826. IEEE Computer Society (2016). https://doi.org/10.1109/CVPR.2016.308
20. Koresh, H.J.D., Chacko, S., Periyanayagi, M.: A modified capsule network algorithm for oct corneal image segmentation. Pattern Recognition Letters **143**, pp.104–112 (2021). https://doi.org/10.1016/j.patrec.2021.01.005
21. Bonheur, S., Štern, D., Payer, C., Pienn, M., Olschewski, H., Urschler, M.: MatwoCapsNet: A Multi-label Semantic Segmentation Capsules Network. In: Shen, D. et al. (eds) Medical Image Computing and Computer Assisted Intervention – MICCAI 2019. MICCAI 2019. Lecture Notes in Computer Science, **11768**, pp. 664–672. Springer International Publishing (2019).
22. LaLonde, R., Xu, Z., Irmakci, I., Jain, S., Bagci, U.: Capsules for biomedical image segmentation. Medical Image Analysis **68**, 101889 (2021). https://doi.org/10.1016/j.media.2020.101889
23. He, Y., Qin, W., Wu, Y., Zhang, M., Yang, Y., Liu, X., Zheng, H., Liang, D., Hu, Z.: Automatic left ventricle segmentation from cardiac magnetic resonance images using a capsule network. Journal of X-Ray Science and Technology **28**, 541–553 (2020). https://doi.org/10.3233/XST-190621
24. Otsu, N.: A Threshold Selection Method from Gray-Level Histograms. IEEE Transactions on Systems, Man, and Cybernetics **9**, 62–66 (1979). https://doi.org/10.1109/TSMC.1979.4310076
25. Bargsten, L., Raschka, S., Schlaefer, A.: Capsule networks for segmentation of small intravascular ultrasound image datasets. Int J CARS **16**(8), 1243–1254 (2021). https://doi.org/10.1007/s11548-021-02417-x
26. Hssayeni, M.D., Croock, M.S., Salman, A.D., Al-khafaji, H.F., Yahya, Z.A., Ghoraani, B.: Intracranial Hemorrhage Segmentation Using a Deep Convolutional Model. Data **5**, 14 (2020). https://doi.org/10.3390/data5010014
27. Tiwari, S., Jain, A.: A lightweight capsule network architecture for detection of COVID-19 from lung CT scans. Int J Imaging Syst Technol. **32**(2), 419–434 (2022). https://doi.org/10.1002/ima.22706
28. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely Connected Convolutional Networks. In: 30th IEEE Conference on Computer Vision and Pattern Recognition – CVPR 2017. CVPR 2017. **2017-January**, pp. 2261–2269 (2017). https://doi.org/10.1109/CVPR.2017.243
29. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: 3rd International Conference on Learning Representations – ICLR 2015. ICLR 2015. (2015).
30. He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition. In: 29th IEEE Conference on Computer Vision and Pattern Recognition – CVPR 2016. CVPR 2016. **2016-December**, pp. 770–778 (2016). https://doi.org/10.1109/CVPR.2016.90
31. Srinivasu, P. N., Sivasai, J.G., Ijaz, M.F., Bhoi, A.K., Kim, W., King, J.J.: Classification of Skin Disease Using Deep Learning Neural Networks with MobileNet V2 and LSTM. Sensors **21**(8), 2852 (2021). https://doi.org/10.3390/s21082852
32. Yousra, D., Abdelhakim, A.B., Mohamed, B.A.: A Novel Model for Detection and Classification Coronavirus (COVID-19) Based on Chest X-Ray Images Using CNN-

CapsNet. In: Corchado, J.M., Trabelsi, S. (eds) Sustainable Smart Cities and Territories. SSCTIC 2021. Lecture Notes in Networks and Systems **253**, pp. 187–199 (2021). https://doi.org/10.1007/978-3-030-78901-5_17

33. Karnkawinpong, T., Limpiyakorn, Y.: Chest X-Ray Analysis of Tuberculosis by Convolutional Neural Networks with Affine Transforms. In: 2nd International Conference on Computer Science and Artificial Intelligence – CSAI 2018. CSAI 2018. Pp. 90–93. Association for Computing Machinery (2018). https://doi.org/10.1145/3297156.3297251
34. LaLonde, R., Torigian, D., Bagci, U.: Encoding Visual Attributes in Capsules for Explainable Medical Diagnoses. In: 23rd International Conference on Medical Image Computing and Computer Assisted Intervention – MICCAI 2020. MICCAI 2020. Lecture Notes in Computer Science **12261**, pp. 294–304 (2020). https://doi.org/10.1007/978-3-030-59710-8_29
35. Shen, S., Han, S.X., Aberle, D.R., Bui, A.A., Hsu, W.: An interpretable deep hierarchical semantic convolutional neural network for lung nodule malignancy classification. Expert Systems with Applications **128**, 84–95 (2019). https://doi.org/10.1016/j.eswa.2019.01.048
36. Afshar, P., Oikonomou, A., Naderkhani, F., Tyrrell, P.N., Plataniotis, K.N., Farahani, K., Mohammadi, A.: A 3D Multi-scale Capsule Network for Lung Nodule Malignancy Prediction. Sci Rep **10**, 7948 (2020). https://doi.org/10.1038/s41598-020-64824-5