scientific reports

Check for updates

OPEN

A comparative evaluation of deep learning approaches for ophthalmology

Glenn Linde[1], Waldir Rodrigues de Souza Jr[2,4], Renoh Chalakkal[1]✉, Helen V. Danesh-Meyer[3], Ben O'Keeffe[1] & Sheng Chiong Hong[1,2]

There is a growing number of publicly available ophthalmic imaging datasets and open-source code for Machine Learning algorithms. This allows ophthalmic researchers and practitioners to independently perform various deep-learning tasks. With the advancement in artificial intelligence (AI) and in the field of imaging, the choice of the most appropriate AI architecture for different tasks will vary greatly. The best-performing AI-dataset combination will depend on the specific problem that needs to be solved and the type of data available. The article discusses different machine learning models and deep learning architectures currently used for various ophthalmic imaging modalities and for different machine learning tasks. It also proposes the most appropriate models based on accuracy and other important factors such as training time, the ability to deploy the model on clinical devices/smartphones, heatmaps that enhance the self-explanatory nature of classification decisions, and the ability to train/adapt on small image datasets to determine if further data collection is worthwhile. The article extensively reviews the existing state-of-the-art AI methods focused on useful machine-learning applications for ophthalmology. It estimates their performance and viability through training and evaluating architectures with different public and private image datasets of different modalities, such as full-color retinal images, OCT images, and 3D OCT scans. The article is expected to benefit the readers by enriching their knowledge of artificial intelligence applied to ophthalmology.

Keywords Diabetic retinopathy, Fundus imaging, Ophthalmoscopy, Deep learning, Artificial intelligence

The global rise of Artificial Intelligence (AI) shows no signs of slowing down[1]. As AI technologies continue to advance, their potential to revolutionize various industries, including healthcare, is becoming increasingly apparent. Ophthalmology, in particular, stands to benefit significantly from AI advancements, which promise to enhance diagnostic accuracy, personalize treatment plans, and streamline the management of eye diseases.

While not all AI is created equal[2], the industry is becoming increasingly consistent and organized. Key developments contributing to this include the establishment of reporting guidelines[3,4], specialist guidance on the safe and effective adoption of AI[2], and government-led best practice initiatives[5]. These efforts are crucial in ensuring that AI is integrated into ophthalmology in a manner that maximizes its benefits while minimizing potential risks. Additionally, the rise of drag-and-drop AI platforms[6] has made AI more accessible to a broader audience, including users with varying levels of coding expertise.

Transparency in AI development is also advancing, driven by the availability of open-source scripts and a growing number of publicly accessible datasets featuring various imaging modalities[7]. These resources are essential for training machine learning (ML) algorithms, particularly in ophthalmology, where they aid in developing tools for detecting and classifying eye pathologies. A significant contribution to this transparency is the *Papers with Code*, which provides a collection of AI models and their implementations, along with benchmarks for different tasks[8].

Ophthalmic imaging modalities and AI applications

Ophthalmic imaging plays a critical role in diagnosing and monitoring eye diseases. Fundus photography and Optical Coherence Tomography (OCT) are two key modalities widely used in clinical practice. Fundus photography provides high-resolution images of the retina, aiding in the identification of various pathologies.

[1]oDocs Eye Care Research, Dunedin, New Zealand. [2]Department of Ophthalmology, Dunedin Hospital, Te Whatu Ora Southern, Dunedin, New Zealand. [3]Department of Ophthalmology, University of Auckland, Auckland, New Zealand. [4] Department of Medicine, Ophthalmology Section, University of Otago, Dunedin, New Zealand. ✉email: renohcj@odocs-tech.com

OCT, on the other hand, creates high-resolution cross-sectional images of the retina, offering detailed visualization of retinal layers. These imaging techniques generate extensive datasets, which, when paired with corresponding diagnostic ground truths, serve as the foundation for training deep learning algorithms.

Machine learning tasks in ophthalmology

Several machine learning tasks can be performed on ophthalmic datasets, including classification, grading, heatmap generation, and quantization. For instance, deep learning algorithms trained on image data from fundus cameras and OCT scanners can predict pathologies such as glaucoma with high accuracy. Classification, in particular, has been well-documented, with Lily Peng et al.[9] demonstrating high specificity and sensitivity in detecting Diabetic Retinopathy (DR) using the CNN InceptionV3 model on Eyepacs DR-graded images. Their AI model outperformed ophthalmologists in classifying the same dataset, underscoring the potential of AI in enhancing diagnostic accuracy.

Similarly, Cecilia Lee et al.[10] achieved high ROC values for Age-Related Macular Degeneration (AMD) using 2D OCT image slices. Other studies, such as those by Barros et al.[11], Singh et al.[12], and Jiang et al.[13], have successfully used CNNs to classify glaucoma and other pathologies like optic disc edema, papillitis, and ARMD, with performance comparable to that of board-certified ophthalmologists.

While classification of 2D OCT slices is achievable with traditional classifiers, training on entire 3D OCT scans is expected to yield better results. This can be accomplished using 3D CNNs or transformers, which can process 3D volumes. However, a challenge with training 3D CNNs is the need to downsize the resolution to fit GPU memory, often requiring a low batch number.

Heatmaps are another important tool in ophthalmic AI applications. During inference, heatmaps can reveal the regions of an image that the classifier deemed important when making its decision. For example, in glaucoma classification, a heatmap might highlight the optic disc region. Heatmaps are part of Explainable Artificial Intelligence (XAI), which aims to make machine learning decisions more interpretable[7].

Quantization is also of interest to ophthalmic researchers, as it allows the reduction of a trained classifier model's size, enabling deployment on small devices like smartphones. Quantization involves converting model parameters from floating-point to integer values, which not only shrinks the model but also speeds up computations by using integer operations.

Ophthalmic image data is typically labeled by class, such as DR vs. normal or glaucoma vs. normal. However, some datasets, like the publicly available OCT2017 dataset, contain images labeled with multiple pathologies, including Diabetic Macular Edema (DME), Cytomegalovirus Retinitis (CMV), and Drusen. Additionally, some datasets, like Eyepacs, offer images labeled by grade rather than by class. Training models on graded data can be more advantageous, as demonstrated by Yijin Huang[14], who achieved superior results using Mean Squared Error (MSE) loss compared to cross-entropy loss when training on Eyepacs data.

AI architectures: CNNs and transformers

The architectures discussed in this paper fall into two primary categories: Convolutional Neural Networks (CNNs) and transformers. CNNs are specifically designed for image processing and are commonly used in tasks requiring classification into multiple categories, such as 'normal', 'glaucoma', or 'DR'. CNNs leverage convolutions to identify image features using spatially aware filters, learning structured representations that enable accurate categorization or grading. Notably, traditional CNNs have been employed in 2D OCT slice classification, but the potential for improved results lies in training on entire 3D OCT volumes. This can be achieved with 3D CNNs, which use 3D convolutions, although this approach requires careful management of GPU memory due to the higher data demands.

Transformers, originally developed for Natural Language Processing (NLP) tasks, have recently been adapted for image-related tasks with considerable success[15]. Unlike CNNs, transformers can capture long-term dependencies within an image, identifying non-local correlations that CNNs may overlook. This capability has led to transformers outperforming CNNs in image classification tasks, as evidenced by their superior performance in ImageNet rankings[16]. One reason for their better performance is that transformers can see long-term dependencies within an image as non-local correlations of objects, which are often ignored by CNNs[17]. Moreover, transformers have shown versatility across various data formats, such as images, video, sound, and text, making them particularly promising for multimodal applications in ophthalmology[18].

Recent advancements have also led to the development of hybrid models that combine CNNs and transformers, aiming to leverage the strengths of both architectures. For instance, a hybrid approach can use CNNs for local feature extraction and transformers for capturing global context, leading to enhanced performance in tasks like retinal disease classification[8].

Given these advancements, this study investigates the performance of both CNN and transformer architectures in ophthalmic applications, utilizing public and private datasets that represent various ophthalmic modalities. The performance of these architectures is evaluated not only by accuracy but also by factors such as training time, quantization efficiency, and the ability to generate interpretable heatmaps. The ultimate goal of this paper is to identify the most effective AI models for diagnosing and managing eye diseases, thereby advancing the field of ophthalmology.

Methodology

The proposed method is divided into three broad categories: Fundus image, 2D OCT image, and 3D OCT volume-based classifiers. Within the Fundus image-based classification, further evaluation and analysis is performed based on the type of classifier required for the targeted pathology, such as a multiclass classifier for detecting DR (e.g., DR vs Healthy) and grading classifier for exclusively grading the pathology (e.g., classification into different grades of DR ranging from healthy to severe DR).

Classification of fundus images

In the classification of fundus retinal images, two distinct types of classifiers hold prominence: the multiclass classifiers and the grading classifiers. These two classifiers are extensively expounded upon in the subsequent subsections, offering valuable insights into their diverse applications and significance in the field.

Multiclass classifiers

Datasets. In order to determine the best-performing classification architectures for fundus images, we utilized four publicly available datasets containing pathologies such as diabetic retinopathy (DR) and glaucoma. Specifically, we employed the Eyepacs dataset[19], which includes retinal images categorized into four different grades of diabetic retinopathy. Grade 0 comprises 25810 images, grade 1 comprises 2443 images, grade 2 comprises 5292 images, and grade 3 comprises 873 images. The three grades were merged into a single DR class for the classification task, while healthy images were placed in the normal class.

The Messidor dataset[20] contains 1200 images related to diabetic retinopathy, consisting of 788 normal and 172 DR images. This dataset also includes an exclusive test set comprising 182 normal and 54 DR images. The Messidor-2 dataset[21] is another collection of DR-related images, featuring 1368 normal images and 380 DR images. The ACRIMA dataset[22] is a glaucoma dataset comprising 309 normal and 396 glaucomatous images.

Architectures. A promising architecture for ophthalmic tasks needs to consider several factors. These factors include the accuracy of the architecture when training on fundus datasets, the speed at which the model can be trained, the model's ability to be trained on small datasets, the size of the model to fit on small ophthalmic imaging/triaging devices, and the ability to create heatmaps from the model. Different factors may hold varying levels of importance for specific ophthalmic tasks, but overall accuracy is generally considered the most important parameter.

The architectures we examined were the best-performing ones described in Papers with Code[23]. In cases where Papers with Code did not show fundus image datasets for a specific eye pathology, we selected architectures based on the best performers in the Papers with Code ImageNet leaderboard[24]. The top performers included ViT, EfficientNet, VOLO, Beit, and RegNet. Additionally, we included InceptionV3 from Lily Peng's 2016 paper[9] for historical reasons, even though it did not rank as a top performer. All the architectures were pre-trained with the ImageNet dataset[24]. Image augmentation techniques, such as modifying contrast, aspect ratio, flipping, and brightness, were also utilized during training to reduce overfitting.

The architectures used in the study included mainly transformers, CNN, or a combination of both. Pure transformers such as the ViT classifier, a scaled vision transformer[25,26], were utilized to extend NLP for images using patching to reduce the sequence size. This approach is known for achieving top performance on ImageNet. The GitHub vit-keras codebase[27] was employed with an image size of 384x384. Additionally, the transformer VOLO[28] was tested. VOLO uses fine-level features or tokens that are often overlooked in self-attention methods. The Keras CV Attention GitHub repository (KCAC)[29] was used for the implementation, with the VOLOd5 variant and an image size of 224x224. BEIT[30], another transformer architecture, tokenizes the image and applies masks to patches. The study utilized the KCAC[29] variant BeitBasePatch16 with an image size of 224. DaViT[31] is a simple visual transformer that captures global context by leveraging self-attention mechanisms with both spatial tokens and channel tokens. In this study, the KCAC variant DaViTS[29] was used, with an image size of 224x224.

Unlike the previously described architectures, CotNet[32] is a hybrid of transformer and CNN that utilizes convolutions and employs attention on 2D feature maps. It utilized the KCAC variant CotNetSE152D[29] with an image size of 320. Another CNN/transformer hybrid, CoAtNet[33], utilizes depthwise convolution and self-attention. The KCAC variant CoAtNet0[29] was used with an image size of 224x224. ResNeSt Split-Attention Networks[34] is another CNN/transformer that combines CNN with attention mechanisms. It features split attention, which enables cross-feature interactions. The variant used is ResNest269 with KCAC[29], and the image size is 416x416. MLP-Mixer[35] has an unconventional architecture as it is neither a CNN nor a transformer. It is a multilayer perceptron with no CNNs, transformers, or attention mechanisms. The implementation uses the MLPMixerL16 variant with KCAC[29] and an image size of 224x224.

The rest of the architectures mentioned are CNNs. RegNet[36], which belongs to the ResNeSt family, is a CNN with shortcuts to prevent vanishing gradients. Unlike other models, RegNet features a regulator module for improved complementary features. The variant used is RegNetZ with KCAC[29], and the image size is 256x256. Additionally, Normalizer-Free ResNeSts[37] is a CNN that eliminates batch normalization, which can be computationally expensive. It uses the NFNetF2 variant with KCAC[29] and an image size of 352x352.

InceptionV3[38] is a classic CNN that utilizes factorized convolutions, wherein multiple filters are applied simultaneously to a channel, and label smoothing, which compensates for errors in ground truths. The implementation was carried out using Tensorflow[39], with an image size of 299x299. EfficientNet[38] is a CNN that determines the width, resolution, and depth of a CNN through 'compound scaling', adjusting the width and depth for a given resolution rather than doing so arbitrarily. It employs KCAC[29] and uses the variant EfficientNetV2S with an image size of 384x384.

These architectures were trained on four public datasets, partitioning the data into 80% for training and 20% for validation. Each dataset was balanced, ensuring an equal number of images for each class. The accuracy scores were calculated by inferring from the 20% validation set, image by image, in order to obtain the final accuracy score. Since multiclass classification is not a regression problem, only accuracy was calculated.

The training was stopped when the validation accuracy failed to increase after more than ten epochs. The architectures were trained on NVidia T4 (2560 cores) GPUs, except for the InceptionV3, which was trained using a Geforce GTX960 (1024 cores). The performance of different CNN architectures on various datasets with corresponding accuracies is reflected in Table 1 and Fig. 1.

Article	Architecture	Eyepacs 34418 images	Messidor2 1748 images	Messidor 1200 images	ACRIMA 705 images	Average Accuracy	Quantized	Heatmap
[38]	EfficientNet	85	94	93	97.5	92.3	✓	✓
[36]	RegNet	82	97	90	98	91.8	✓	✓
[33]	CoatNet	77	95	96	96	91	✓	✓
[31]	DaVit	78.5	86	96	100	90	✓	✓
[34]	ResNeSt	78	94	90	61	80.8	✓	✓
[32]	Cotnet	75	75	90	89	82.2	✓	✓
[35]	MLP family	65	88	86	98.7	84.4	✓	✓
[38]	Inceptionv3	66	90	87	87	82.5	✓	✓
[37]	NFNet	63	70	85	95	78.2	✓	✓
[30]	Beit	51	80	65	96	73	✓	✓
[26]	ViT	53	72	60	98.5	70.9	✗	✗
[28]	VOLO	50	50	80	50	57.5	✓	✓

Table 1. List of architectures and accuracy (in %) of each dataset for multiclass problem.

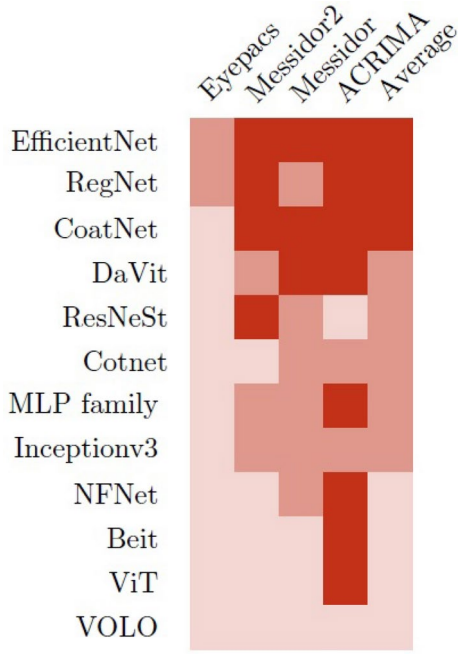

Fig. 1. Accuracy of each architecture for each dataset for multiclass problem (> 90% dark, > 80% medium otherwise light).

Training time. The training time for each architecture was determined based on the processing time per image trained on the Eyepacs dataset with an NVidia T4 GPU. The batch number, calculated as the number of steps per epoch divided by the time to train each epoch, provides the number of images processed per second. Architectures that require high memory will have a lower batch number, resulting in fewer processed images per second. Similarly, a deep architecture with a wide field of view will also process fewer images per second due to the higher number of architectural parameters. The most efficient training times were observed with the EfficientNet, RegNet, CoatNet, and InceptionV3 architectures.

Quantization. Quantization[40] is a crucial process for deploying trained models onto small devices. It involves converting the floating point values in the model to 8-byte integers (uint8), resulting in a smaller model size and faster computation when used on Advanced Reduced Instruction Set Computing Machines (ARM) chips-based devices like many smartphones. Quantization requires determining the range of the input data so that the integers can be scaled correctly, which may involve clipping.

EfficientNet, Beit, CotNet, and ResNeSt families were effectively quantized using the Keras TFLiteConverter. For example, an EfficientNet model trained on Eyepacs size was reduced from 245 meg to 80 meg, with the accuracy decreasing from 85% to 82% after quantization. Additionally, a quantized model for RegNet was trialed

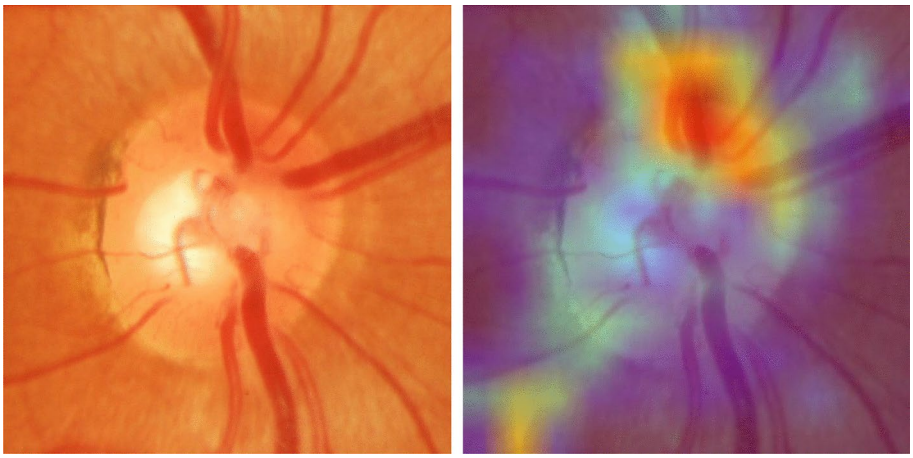

Fig. 2. Original ACRIMA glaucoma image and the heatmap generated from EfficientNet.

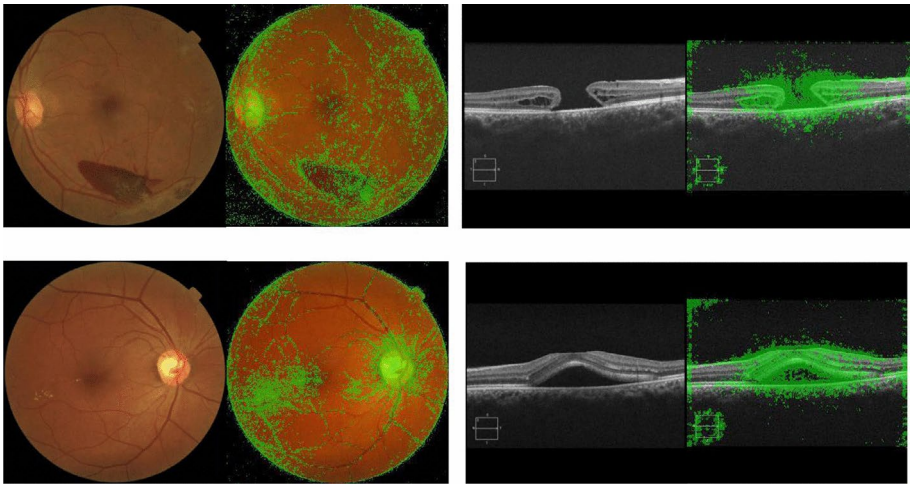

Fig. 3. Heatmap generated for InceptionV3-trained models using guided backpropagation.

on an iPhone XR using a TensorFlow repository for real-time classification using the iPhone's camera and could perform multiple classifications per second. The RegNet uint8 quantized model took 250 milliseconds per image inference, while a floating point version of the same model took 400 milliseconds on the iPhone.

The InceptionV3 models were quantized by first converting the model variables to constants using TensorFlow's "convert variables to constants"function. The quantized model could then be deployed on Android and iPhone devices using the TensorFlow Android repository[41] and iOS repository[42].

Heatmaps. Heatmaps provide insights into classification decisions by highlighting important regions of an image that were influential in a specific diagnostic inference. They commonly employ the Grad-CAM technique[43], which utilizes the gradients of the target class, such as glaucoma, in a classification network and feeds them into the last CNN layer. This process generates a coarse localization map of the critical regions used to make the prediction, followed by backward propagation for reconstruction into a DeconvNet, which produces the final heatmap image.

An EfficientNet model, trained with the ACRIMA glaucoma dataset, was utilized to create heatmaps using the Grad-CAM technique[44] from the Keras repository[45]. The resulting heatmap image (Fig. 2) highlights the critical parts of the optic disc that the classifier identified as significant during the classification decision.

Heatmaps were also created for InceptionV3-trained models using a Grad-CAM repository[46]. Figure 3 shows a heatmap generated using guided backpropagation[6]. Due to the disappointing performance of transformers with small datasets, no attempt was made to generate heatmaps for transformers like ViT, even though they can be created using the same method[25].

Discussion. The performance metrics for various CNNs for the multiclass classification problem are summarized in Table 1. EfficientNet demonstrates strong performance across different datasets with a remarkable training speed of 31 images per second. On the other hand, Transformers exhibit lower accuracy and slow

training time. Considering factors such as accuracy, training speed, the ability to quantize, and the generation of heatmaps, EfficientNet emerges as the top performer overall. Transformers tend to over-fit on smaller datasets[47], displaying poor performance even on the largest ophthalmic dataset, Eyepacs (containing over 25810 images). This suggests that more than tens of thousands of images are required to mitigate over-fitting. Transformers trained using the ImageNet dataset of over 14 million images[8], which may explain why they perform well on the ImageNet leaderboard. Additionally, architectures that excel with larger datasets also tend to perform well with smaller datasets. For instance, EfficientNet performs well with Eyepacs and Messidor, while the MLP family yields inferior results on the same datasets.

Despite the small size of the datasets used (consensus is less than 4000 images[48]), it was observed that the accuracy was quite high, even though overfitting would typically be expected to affect accuracy. Overfitting in small datasets can be mitigated by techniques such as image augmentation, which involves rotating, flipping, and cropping images to make it more challenging for the classifier to memorize the training data and instead generalize. Image augmentation is commonly integrated into deep learning frameworks such as Keras (Keras ImageDataGenerator augments with parameters including shearrange, zoomrange).

The use of pre-trained models is also beneficial for smaller datasets, as these models have been trained on millions of ImageNet images, and the learned filters can be reused for new images, including fundus images. Without pre-trained models, datasets would need to be much larger to train new filters. Additionally, a dropout layer can help reduce overfitting and is commonly included in CNN models such as EfficientNet. Dropout randomizes weights to varying degrees, reducing the likelihood of data memorization and encouraging generalization.

Using smaller models also helps mitigate overfitting, as fewer parameters make it harder for the model to learn the training data[49]. This may explain why EfficientNet, despite its small size (20.33 million parameters), performed well compared to VOLO, which has 296 million parameters. The poor performance of VOLO and DavitS may be attributed to overfitting, resulting in memorization of the training data rather than generalization. The high number of parameters also explains why transformers took longer to train compared to CNNs, with ViT and VOLO having the slowest training times.

The underperformance of transformers with smaller datasets has been observed in studies such as those by Chen et al.[47] and Zhu et al.[50]. Zhu et al. argue that the VIT transformer's lower performance on small datasets may be due to a "lack of inductive bias of locality with lower layers, where Vit cannot learn the local relations with a small amount of data." This poor performance may not only be attributed to the larger parameter size but also to the transformer architecture itself. Further research into improving transformers' ability to train on smaller datasets would be beneficial.

Grading classifiers

When labeling datasets based on grades rather than classes, like the four grades of DR in the Eyepacs dataset[19], a standard multiclass classifier is not suitable. The multiclass classifier requires modification to produce a single-grade output ranging from 0 to 1.

For example, in the case of InceptionV3, the number of labels was reduced to one using the TensorFlow tf.reshape function in the last layer. The loss function was changed to Mean Squared Error (MSE) loss, replacing softmax since the grading does not use cross-entropy. Similarly, in the case of InceptionV3, the slim Euclidian loss (MSE) replaced the slim softmax loss in the model. This modification was also made for EfficientNet and RegNet models using the sigmoid activation.

Zhang et al.[51] took a different approach by using a deep graph correlation network (DGCN) consisting of multiple CNNs that are correlated through a graph. They claimed that the performance was close to that of specialists' results. However, they did not compare the performance of a DGCN to that of a single modified CNN, so it is unclear whether it is superior to a single CNN.

Datasets. The datasets used to test these architectures included Eyepacs, which contained four grades of DR scaled between 0 and 1. For Messidor[20] and Messidor-2[21], a grade of 0 was assigned to healthy images and a grade of 1 was assigned to DR images.

Architectures. There are no examples of grading architectures in the papers with code[8]. Therefore, architectures were selected based on the performance of previously examined ones. The chosen architectures include EfficientNet, RegNet, and InceptionV3. Each architecture was adjusted to use a single output with mean squared error (MSE) loss, instead of using softmax.

We used 80% of the data for training and 20% for validation. Training accuracy alone cannot ensure better class prediction, so we need to calculate accuracy differently. A prediction is considered correct if the predicted value and the ground truth are both less than 0.5, or if both are over 0.5. Since grading is a regression problem, AUC, precision, and recall were also calculated, with 0.5 as the midpoint. Each modified CNN architecture utilized for grading underwent training on different datasets, and their performances are detailed in Table 2. The grading accuracies are also depicted in Fig. 4 as a heat grid.

Heatmaps, quantization, training time. Heatmaps and quantization were performed similarly to multiclass classifiers due to the use of the same architectures as with classification (except for the last layer). It was assumed that the training time was the same because of the identical architectures being used.

Discussion. The performance metrics for various grading classifiers are presented in Table 2. Among the three datasets tested, RegNet demonstrated the best performance for grading applications. RegNet, Inceptionv3, and

Article	Architecture	Eyepacs 34418 images (Accuracy in %)	AUC (%)	Messidor2 1748 images (Accuracy in %)	AUC (%)	Messidor 1200 images (Accuracy in %)	AUC (%)	Average Accuracy (%)	Quantization	Heatmap
[36]	RegNet	88	89	95.9	99	90	92	91.3	✓	✓
[38]	EfficientNet	88	89	89	98	89	91	88.6	✓	✓
[38]	Inceptionv3	79	68	85	73	83	68	82.3	✓	✓

Table 2. Performance metrics of grading classifiers.

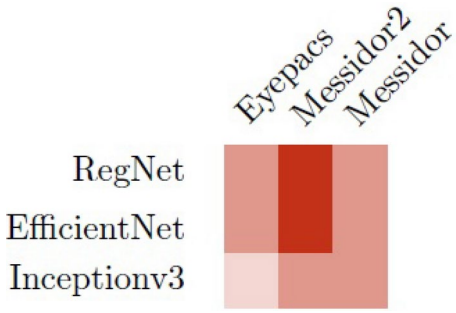

Fig. 4. Grading classifier accuracy of each CNN architecture on different datasets (> 90% dark, > 80% medium otherwise light).

EfficientNet displayed similar capabilities for generating heatmaps and quantization, likely due to their use of the same architectures as in the multiclass classification problem.

The AUC (Area Under the Curve) is a helpful metric when dealing with unbalanced data and was used to evaluate the performance of the grading classifiers. The AUC values showed a strong correlation with accuracy, with RegNet achieving the highest AUC values, similar to its accuracy performance. RegNet also demonstrated the highest precision, averaging 91% across the three datasets, while EfficientNet averaged 85% and InceptionV3 averaged 76%. In terms of recall, EfficientNet averaged 78%, compared to 76% for RegNet and 53% for InceptionV3. The fact that precision is higher than recall for the grading classifiers indicates that the models are better at predicting when a subject truly has a condition, but less effective at predicting when a patient does not have a condition. However, the threshold of 0.5 could be adjusted to balance recall and precision.

RegNet's superior performance in regression compared to other models was also noted by Maddury et al.[52]. The study indicated that, across different regression problems, RegNet outperformed EfficientNet. However, the paper did not provide any explanations as to why RegNet may have outperformed other models in regression. RegNet incorporates a regulatory module that controls the flow of information between layers, preventing early block information from being forgotten in later blocks, whereas EfficientNet optimally scales depth and width. It is possible that RegNet's regulatory module is better suited for regression tasks.

The comparison showed that there are few disadvantages to using grading over multiclass classification, especially since the accuracy is similar (85% for EfficientNet multiclass Eyepacs versus 88% for grading). Additionally, grading had similar training times and the ability to generate heatmaps and freezing compared to multiclass. Moreover, grading offers the advantage of providing a probability of a condition instead of a discrete multiclass prediction, which may be more useful in a clinical situation.

Classification OCT 2D images

In the context of OCT 2D slices (as opposed to fundus camera images), the most effective architectures were studied using two publicly available OCT image datasets, OCT2017 and OCTID. TSuji et al.[53] also demonstrated the effectiveness of training OCT data (compared to the fundus images) with CNNs for pathologies CNV, DME, and Drusen, achieving close to 100% accuracy.

Datasets

The dataset named OCT2017[54] contains 2D cross sections of sagittal slices of the retina. It includes four image classes: Choroidal Neovascularization (CNV) (37205 images), Diabetic Macular Edema (DME) (11348 images), drusen (8616 images), and healthy (26315 images). The CNV images display the neovascular membrane and associated subretinal fluid. Meanwhile, the images for DME images depict retinal thickening associated with intraretinal fluid, along with multiple drusen present in early AMD.

The OCTID dataset[55] consists of slices displaying various eye pathologies, including: normal (200 images), macular holes (100 images), macular degeneration, and retinopathy (100 images).

The EIA2020 dataset includes 200 normal and 200 glaucomatous optic disc cube OCT volumes from 200 participants, with 100 of them diagnosed with glaucoma and the other 100 being normal controls. All 2D images

Article	Architecture	OCT2017 CNV DME Drusen 83,484 images	OCTID Macular hole 300 images	OCTID MD 300 images	OCTID Retinopathy 300 images	EIA2020 enface 93760 images	EIA2020 side 40400 images	Quantized	Heatmap
[32]	CotNet	95.9	100	100	100	91	68	✓	✓
[38]	Inceptionv3	98.9	100	100	100	75	75	✓	✓
[38]	EfficientNet	99.6	100	100	100	75	61	✓	✓
[36]	RegNet	99.8	100	100	100	73	70	✓	✓
[34]	ResNeSt	99.3	100	100	100	50	77	✓	✓

Table 3. Accuracy (in %) of CNN classifiers on different 2D OCT datasets.

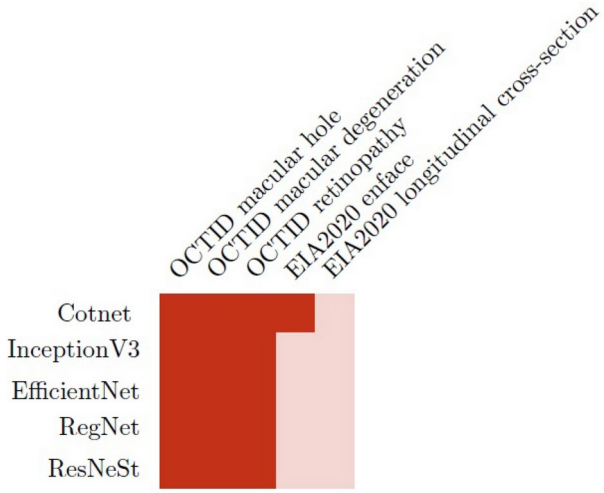

Fig. 5. Heatgrid showing the accuracy of each CNN architecture for different OCT datasets (> 90% dark, > 80% medium otherwise light).

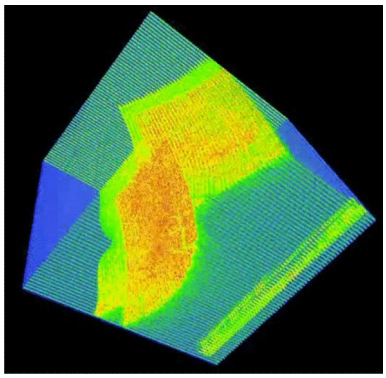

Fig. 6. 3D OCT volume.

from the 200 participants were categorized into glaucoma and non-glaucoma multiclass groups. This dataset comprises 93760 Enface slices and 40400 longitudinal cross-sectional slices of the optic nerve head.

Architectures

The leaderboard on papers with code for the dataset OCT2017[56] is publicly available. However, because the accuracy for each architecture is close to 100%, it's challenging to determine which architectures performed the best. Therefore, the ones that showed the best performance for fundus images were chosen. These architectures include EfficientNet, RegNet, ResNeSt, Cotnet, and InceptionV3.

Similar to the training of fundus images, 80% of the dataset's image data is used for training and 20% for validation. Each architecture is trained using the relevant dataset, and accuracy is calculated in the same manner as for fundus images. The results are presented in Table 3 and Fig. 5.

Quantization, heatmaps and training time
Because these architectures are the same as the fundus images, each architecture can be quantized, and heatmaps generated with the same technique as the fundus images. Training times were also calculated in the same way.

Discussion
According to the leaderboard on papers with code[56], determining the best-performing architecture for 2D OCT images is challenging because accuracies are close to 100% for OCT2017 and OCTID. For EIA2020, Enface showed the best performance with CotNet, while the accuracy was mixed for EIA2020.

Midena[57] suggests that the high accuracy observed when training on OCT datasets is because OCT images contain more information on eye structures compared to fundus images. The article describes how OCT images include eye structures that are not visible in fundus images. It might be suspected that the high accuracy is due to overfitting, but the accuracy is calculated using a separate validation set. The high accuracy of OCT images was observed for all models tested, indicating that OCT images are better for predicting pathology compared to fundus images.

Classification of 3D OCT volumes

When training with individual 2D OCT images using a 2D classifier, we achieved almost 100% accuracy with the datasets we used. However, training on an entire 3D OCT volume is expected to yield even better results. To accomplish this, we experimented with 3D CNNs and transformers.

Dataset
As with our previous 2D classifier, we utilized the EIA2020 dataset. However, this time, we employed the entire 3D volume of the Optic Disc Cube for classification. Figure 6 shows a sample 3D OCT volume.

Architectures
For the targeted application, no papers with code 3D classifier architecture leaderboards were available. Hence, potential classifiers were tested from GitHub, which included three 3D CNNs and two transformers. The 3D CNN architecture for volumetric data was used, with voxels instead of 2D points, as specified by Ahmed et al.[58]. The 3D CNNs used are less deep and wide compared to 2D CNNs due to memory constraints from the extra dimension of volumetric data. All architectures were trained using the EIA-2020 dataset, with the Optic Disc Cube in the Enface orientation; these OCT volumes were 128x128x64 for each patient.

The CNN-3D-images-Tensorflow repository[59] is similar to a 2D CNN but includes two Conv3D layers instead of multiple conv2d layers. It comprises a ReLU layer, followed by fully connected layers with two Conv3D layers (32, 64) and dropout. The 3D CNN in the Keras io repository[60] is deeper than the previous architecture, with four Conv3D layers (64, 64, 128, 256) and dropout. The 3D-CNN-Keras repository[61] has just one layer by default but was modified to have five layers, and it includes batch normalization.

The Perceiver transformer[62] was tested using the Keras perceiver code[63], which is designed to train on images with three channels (RGB). However, the three channels were replaced with a stack of 64 deep grayscale 128x128 OCT images, forming a volume of 128x128x64. The perceiver is a transformer, as opposed to a 3D CNN, and is capable of processing data in various formats, including audio, video, 3D volumes and images. It utilizes attention with keys and query sizes that are unrelated to the input size, allowing it to conserve memory as compared to traditional transformers for the same input size.

The second transformer trialed was the ViT transformer[26], and implemented using vit-keras[27]. It was originally designed for 2D classification. However, similar to the perceiver, it was modified to have a depth of 64 layers and trained with three channels to produce a volume of 128x128x64. As with 2D classification, 80% of the data is used for training and 20% for validation. Table 4 displays the accuracy and classification time of each classifier on the trialed dataset. Figure 7 depicts the heatgrid of different CNNs for classification.

Heatmaps
Similar to 2D classifiers, heatmaps can be generated when inference is performed on sample OCT volumes using Github code from Mehanna[64], which was modified to work in 3D on 3D-CNN-Keras[61]. Similar to 2D heatmaps, the technique uses GradCAM.

In Fig. 8, a sample glaucoma OCT volume from the EIA-2020 dataset, highlighting the area around the optic disc, is shown. Heatmaps were only generated for 3D-CNN-Keras. These steps can be applied to the other 3D CNN architectures as well.

Article	Architecture	EIA-2020 400 volumes	Images/s
[58]	CNN-3D-images-Tensorflow	93	1.21
[58]	3D-CNN-Keras	91	36
[26]	ViT	90	26
[62]	Perceiver	81	16
[58]	Keras io	80	4.6

Table 4. Accuracy (in %) of different CNN architecture on EIA-2020 dataset.

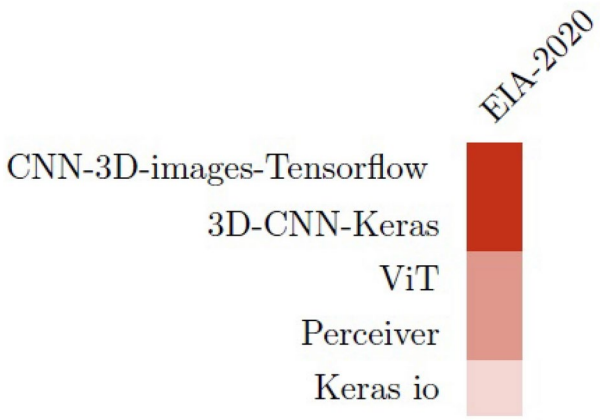

Fig. 7. Accuracy of different architectures on EIA2020 dataset (> 90% dark, > 80% medium otherwise light).

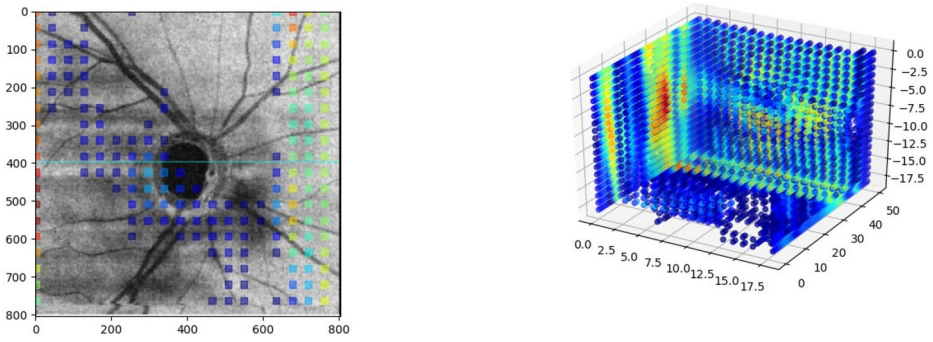

Fig. 8. A single slice from the 3D OCT volume with the corresponding slices displayed in 3D.

Training time
Training time was calculated in the same way as for 2D images. It was estimated from the processing time per volume, with the batch number multiplied by the number of steps per epoch and then divided by the epoch time.

Discussion
It's important to note that the CNN-3D architecture showed the highest accuracy, even outperforming keras io, ViT, and the perceiver models. When tested on the MosMed dataset[65], the CNN-3D architecture achieved an accuracy of 88%, while keras io scored 68%, ViT scored 48%, and the perceiver scored 55%. Despite achieving the highest accuracy, the CNN-3D architecture also had the slowest training speed, whereas the 3D-CNN-Keras model was the fastest.

We found that the CNN-3D model performs better than the same slices trained in 2D. We organized the glaucoma slices from each patient of the EIA-2020 data into one group and the normal slices into another group. This resulted in two groups, each containing over 40000 images (49001, 44761). The two groups were trained using an InceptionV3 classifier. The CNN-3D model was 93% accurate, while the InceptionV3 model was 78% accurate. This demonstrates the advantage of training on an entire volume rather than individual slices.

Quantizing 3D classifiers is impractical because performing inference on extensive 3D volumetric data, such as OCT scans, on a smartphone is not feasible due to hardware constraints. As a result, we did not attempt to quantize 3D classifiers, although it can be done in the same way as 2D classifiers. Also, due to the limitation of having only a single dataset (EIA-2020), we were unable to compare the performance of different 3D CNN architectures with datasets of different sizes.

Conclusion, limitations, and future works

We experimented with several CNN architectures for various image types, such as transformers, transformer hybrids, and CNNs. Transformers, which have been widely discussed in architectures like ChatGPT, were included in our trials. After conducting our study, we determined the best architectures for different image modalities. We found that EfficientNet performed the best in terms of accuracy, training time, and its ability to work with smaller datasets for classifying color fundus and OCT images. For grading, we found that RegNet was the most effective, and for OCT 3D volumes, we found that 3D-CNN was the best performer, despite not being the fastest. While transformers have received significant attention recently[66,67], our study found that they were outperformed by CNNs, which is consistent with prior research indicating that transformers rely on

large datasets[26] to achieve desirable performance and have a tendency to overfit on smaller datasets[47]. These limitations favour their use by those who have access to big data and high computing power.

The proposed study has some limitations. The method we are proposing uses models that rely on publicly available datasets for training, testing, and validation. However, publicly available healthcare datasets often have limitations, such as restricted clinical information. These datasets tend to focus on single diagnoses without providing broader comorbidity data. Additionally, disease status labels, indicating whether a person is positive or negative for a particular disease, may come from a single diagnostician, potentially introducing significant bias.

Another issue with these datasets is that they often exhibit biases toward Western sources, which can be attributed to data availability, dominant platforms, and the prevalence of English-language content. Furthermore, datasets obtained from private healthcare providers may lead to an under-representation of patients with lower income and from ethnic minorities[68,69].

It is vital to have comprehensive whole-person clinical data to understand complex patient conditions. Simply having large datasets is not enough to address the issue of generalizability. It is important to also have diversity and cross-population validation to ensure that models can be used in real-world scenarios. Using multimodal approaches that combine different types of data (such as imaging, genomic, and clinical data) is likely to improve model performance and generalizability. Robust real-world testing and diverse datasets are necessary to ensure that AI systems are effective across various clinical settings and patient populations, addressing both technical and equitable healthcare challenges.

This paper adds to our understanding of different AI approaches for ophthalmological applications. It compares the performance of various combinations of CNN architectures and image modalities, highlighting differences in their accuracy and ability to perform various machine-learning tasks. It emphasizes the importance of heatmaps in providing transparency into the decision-making process of CNNs by highlighting areas of interest in an image[2,7]. The study makes a significant contribution to the journey of AI development, providing detailed information for those involved in integrating these algorithms into medical devices[2,5].

However, there are still areas that require attention in future studies. One such area is the absence of 3D OCT datasets graded for glaucoma severity. Due to the difficulty in collecting data and the limited reliability of smaller datasets, there is a need for more specific research on hybrid architectures. These architectures could combine the strengths of transformers and CNNs, while also establishing continuous systems to self-monitor performance and refine new approaches for effectively handling smaller datasets.

It is well established that many systemic diseases can be detected through observable changes in the retina[70]. AI technology is at the forefront of using the eye to gain insights into overall health[71,72]. Therefore, future research has significant potential in examining retinal markers of systemic conditions using large cohort datasets containing extensive ophthalmic imaging and comprehensive longitudinal data on various comorbidities.

Data availibility

Publicly available datasets used are listed below: Keras CV attention models (https://github.com/leondgarse/keras_cv_attention_models) . Vision Transformers (ViT) in Image Recognition (https://viso.ai/deep-learning/vision-transformer-vit). Vit-keras (https://github.com/faustomorales/vit-keras). InceptionV3 Code (https://github.com/tensorflow/tensorflow/tree/r1.4/tensorflow/examples/image_retraining). Quantization (https://intellabs.github.io/distiller/quantization.html). TensorFlow Lite image classification iOS example application (https://github.com/tensorflow/examples/tree/master/lite/examples/image_classification/ios). TensorFlow Android Camera Demo (https://github.com/tensorflow/tensorflow/tree/r1.4/tensorflow/examples/android). TensorFlow iOS Examples (https://github.com/tensorflow/tensorflow/tree/r1.4/tensorflow/examples/ios). keras cv attention models visualizing (https://github.com/leondgarse/keras_cv_attention_models/tree/main/keras_cv_attention_models/visualizing). Grad-CAM class activation visualization (https://keras.io/examples/vision/grad_cam/). Guided back prop (https://github.com/hummat/saliency/blob/master/guided_backprop.py). Keras IO 3D Image Classification (https://github.com/keras-team/keras-io/blob/master/examples/vision/3D_image_classification.py). CNN 3D Images using Tensorflow (https://github.com/jibikbam/CNN-3D-images-Tensorflow). 3D-CNN-Keras (https://github.com/Ectsang/3D-CNN-Keras). Perceiver image classification (https://github.com/keras-team/keras-io/blob/master/examples/vision/perceiver_image_classification.py). Strip Unused (https://github.com/tensorflow/tensorflow/blob/r1.4/tensorflow/python/tools/strip_unused.py) . Open-source architectures listed below: Eyepacs (https://www.kaggle.com/c/diabetic-retinopathy-detection/data). Messidor (https://www.adcis.net/en/third-party/messidor/). Messidor-2 (https://www.adcis.net/en/third-party/messidor2/). ACRIMA (https://www.kaggle.com/sshikamaru/glaucoma-detection). OCT2017 (https://www.kaggle.com/paultimothymooney/kermany2018#OCT2017.zip). OCTID (https://dataverse.scholarsportal.info/dataverse/OCTID). Drions DB (http://www.ia.uned.es/~ejcarmona/DRIONS-DB.html). Refuge (https://refuge.grand-challenge.org/).

Received: 25 July 2023; Accepted: 9 September 2024
Published online: 18 September 2024

References

1. Adam, B., & Kaveh, M. The rise of artificial intelligence in healthcare applications. In: *Artificial Intelligence in healthcare* 25–60 (Elsevier, 2020).
2. Saria, S. Not all ai is created equal: Strategies for safe and effective adoption. *NEJM Catal. Innov. Care Deliv.* **3**(2), https://catalyst.nejm.org/doi/full/10.1056/CAT.22.0075 (2022).
3. Decide-ai: New reporting guidelines to bridge the development-to-implementation gap in clinical artificial intelligence. *Nat. Med.* **27**(2), 186–187 (2021).

4. Liu, X. *et al.* Reporting guidelines for clinical trial reports for interventions involving artificial intelligence: The consort-ai extension. *Lancet Digit. Health* **2**(10), e537–e548 (2020).
5. Food and Drug Administration and others, Proposed regulatory framework for modifications to artificial intelligence/machine learning (ai/ml)-based software as a medical device (samd) (2019).
6. Korot, E. et al. Code-free deep learning for multi-modality medical image classification. *Nat. Mach. Intelligen.* **3**(4), 288–298. https://www.nature.com/articles/s42256-021-00305-2 (2021).
7. Yang, G., Ye, Q. & Xia, J. Unbox the black-box for the medical explainable ai via multi-modal and multi-centre data fusion: A mini-review, two showcases and beyond. *Inf. Fusion* **77**, 29–52 (2022).
8. Papers with code (2022). https://paperswithcode.com
9. Varun, G. et al. Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *jama* **316**(22), 2402–2410. https://jamanetwork.com/journals/jama/fullarticle/2588763/ (2016).
10. Lee, C., Baughman, D. & Lee, A. Deep learning is effective for the classification of OCT images of normal versus age-related macular degeneration (2017).
11. Barros, D. *et al.* Machine learning applied to retinal image processing for glaucoma detection: Review and perspective. *Biomed. Eng. Online* **19**(1), 1–21 (2020).
12. Singh, L. K., Khanna, M., Thawkar, S. & Singh, R. A novel hybridized feature selection strategy for the effective prediction of glaucoma in retinal fundus images. *Multimed. Tools Appl.* **83**(15), 46087–46159 (2024).
13. Jiang, P., Dou, Q. & Shi, L. Ophthalmologist-level classification of fundus disease with deep neural networks. *Transl. Vis. Sci. Technol.* **9**(2), 39–39 (2020).
14. Yijin, H., Lina, L., Pujin, C., Junyan, L. & Xiaoying, T. Identifying the key components in resnet-50 for diabetic retinopathy grading from fundus images: A systematic investigation. *Diagnostics* **13**(10), 1664. https://www.mdpi.com/2075-4418/13/10/1664 (2021).
15. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. & Polosukhin, I. Attention is all you need *Adv. Neural Inf. Process. Syst.* **30**(1), 261–272. https://user.phil.hhu.de/~cwurm/wpcontent/uploads/2020/01/7181-attention-is-all-you-need.pdf (2017).
16. Imagenet rank (2022). https://paperswithcode.com/sota/image-classification-on-imagenet
17. He, K., Gan, C., Li, Z., Rekik, I., Yin, Z., Ji, W., Gao, Y., Wang, Q., Zhang, J. & Shen, D. Transformers in medical image analysis: A review *Intell. Med.* **3**(1), 59–78. https://www.sciencedirect.com/science/article/pii/S2667102622000717 (2022).
18. Korngiebel, D. M. & Mooney, S. D. Considering the possibilities and pitfalls of generative pre-trained transformer 3 (gpt-3) in healthcare delivery. *NPJ Digit. Med.* **4**(1), 93 (2021).
19. Eyepacs (2022). https://www.kaggle.com/c/diabetic-retinopathy-detection/data
20. Messidor (2022). https://www.adcis.net/en/third-party/messidor/
21. Messidor-2 (2022). https://www.adcis.net/en/third-party/messidor2/
22. Acrima (2022). https://www.kaggle.com/sshikamaru/glaucoma-detection
23. Papers with code on imagenet (2022). https://paperswithcode.com/sota/image-classification-on-imagenet
24. Imagenet (2022). https://www.image-net.org/
25. Boesch, G. Vision transformers (vit) in image recognition–2022 guide, viso. ai (2022). https://viso.ai/deep-learning/vision-transformer-vit/
26. Dosovitskiy, A. *et al.* An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020).
27. vit-keras (2022). https://github.com/faustomorales/vit-keras
28. Yuan, L., Hou, Q., Jiang, Z., Feng, J. & Yan, S. Volo: Vision outlooker for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **45**(5), 6575–6586 (2022).
29. d Garse, L. Keras cv attention models (2022). URL: hhttps://github.com/leondgarse/keras_cv_attention_models
30. Bao, H., Dong, L., Piao, S. & Wei, F. Beit: Bert pre-training of image transformers. arXiv preprint arXiv:2106.08254 (2021).
31. Ding, M., Xiao, B., Codella, N., Luo, P., Wang, J. & Yuan, L. Davit: Dual attention vision transformers. In *European Conference on Computer Vision* 74–92 (Springer, 2022).
32. Li, Y., Yao, T., Pan, Y. & Mei, T. Contextual transformer networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **45**(2), 1489–1500 (2022).
33. Dai, Z., Liu, H., Le, Q. V. & Tan, M. Coatnet: Marrying convolution and attention for all data sizes. *Adv. Neural Inf. Process. Syst.* **34**, 3965–3977 (2021).
34. Zhang, H., Wu, C., Zhang, Z., Zhu, Y., Lin, H., Zhang, Z., Sun, Y., He, T., Mueller, J., Manmatha, R. *et al.* Resnest: Split-attention networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* 2736–2746 (2022).
35. Tolstikhin, I. O. *et al.* Mlp-mixer: An all-mlp architecture for vision. *Adv. Neural Inf. Process. Syst.* **34**, 24261–24272 (2021).
36. Xu, J., Pan, Y., Pan, X., Hoi, S., Yi, Z. & Xu, Z. Regnet: Self-regulated network for image classification. *IEEE Trans. Neural Netw. Learn. Syst.* **34**(11), 9562–9567. https://ieeexplore.ieee.org/abstract/document/9743274/ (2022).
37. Brock, A., De, S., Smith, S. L. & Simonyan, K. High-performance large-scale image recognition without normalization. In *International Conference on Machine Learning* 1059–1071 (PMLR, 2021).
38. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. & Wojna, Z. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 2818–2826 (2016).
39. Inceptionv3 code (2022). https://github.com/tensorflow/tensorflow/tree/r1.4/tensorflow/examples/image_retraining
40. Quantization (2022). https://intellabs.github.io/distiller/quantization.html
41. Tensorflow android camera demo (2017). https://github.com/tensorflow/tensorflow/tree/r1.4/tensorflow/examples/android
42. Tensorflow ios examples (2017). https://github.com/tensorflow/tensorflow/tree/r1.4/tensorflow/examples/ios
43. Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D. & Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision* 618–626 (2017).
44. Chollet, F. Grad-cam class activation visualization (2021). https://keras.io/examples/vision/grad_cam/
45. keras cv attention models visualizing (2022). https://github.com/leondgarse/keras_cv_attention_models/tree/main/keras_cv_attention_models/visualizing
46. Guided back prop (2020). https://github.com/hummat/saliency/blob/master/guided_backprop.py
47. Chen, Z., Xie, L., Niu, J., Liu, X., Wei, L. & Tian, Q. Visformer: The vision-friendly transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* 589–598 (2021).
48. François Chollet, J. A. Image classification on small datasets with keras, Posit AI Blog (2017). https://blogs.rstudio.com/ai/posts/2017-12-14-image-classification-on-small-datasets/
49. Dijkinga, F. J. Methods to avoid overfitting in artificial neural networks, Medium (2023). https://medium.com/@fernando.dijkinga/methods-to-avoid-overfitting-in-artificial-neural-networks-7564518bf65d
50. Zhu, H., Chen, B. & Yang, C. Understanding why vit trains badly on small datasets: An intuitive perspective. arXiv preprint arXiv:2302.03751 (2023).
51. Zhang, G. *et al.* Diabetic retinopathy grading by deep graph correlation network on retinal images without manual annotations. *Front. Med.* **9**, 872214 (2022).
52. Maddury, S. & Desai, K. Deepad: A deep learning application for predicting amyloid standardized uptake value ratio through pet for alzheimer's prognosis. *Front. Artif. Intell.* **6**, 1091506 (2023).
53. Tsuji, T. *et al.* Classification of optical coherence tomography images using a capsule network. *BMC Ophthalmol.* **20**(1), 1–9 (2020).

54. OCT2017 (2017). https://www.kaggle.com/paultimothymooney/kermany2018#OCT2017.zip
55. OCTID (2018). https://dataverse.scholarsportal.info/dataverse/OCTID
56. Retinal oct disease classification on oct2017 (2022). https://paperswithcode.com/sota/retinal-oct-disease-classification-on-oct2017
57. Midena, E. *et al.* Optical coherence tomography and color fundus photography in the screening of age-related macular degeneration: A comparative, population-based study. *Plos One* **15**(8), e0237352 (2020).
58. Ahmed, E., Saint, A., El Rahman Shabayek, A., Cherenkova, K., Das, R., Gusev, G., Aouada, D. & Ottersten, B. A survey on deep learning advances on different 3d data representations. arXiv e-prints (2018) arXiv–1808.
59. Cnn 3d images using tensorflow (2019). https://github.com/jibikbam/CNN-3D-images-Tensorflow
60. Keras io 3d image classification (2021). https://github.com/keras-team/keras-io/blob/master/examples/vision/3D_image_classification.py
61. 3d-cnn-keras (2016). https://github.com/Ectsang/3D-CNN-Keras
62. Jaegle, A., Gimeno, F., Brock, A., Vinyals, O., Zisserman, A. & Carreira, J. Perceiver: General perception with iterative attention. In *International Conference on Machine Learning* 4651–4664 (PMLR, 2021).
63. Perceiver image classification (2019). https://github.com/keras-team/keras-io/blob/master/examples/vision/perceiver_image_classification.py
64. Mehanna, N. Visualizing convolutional neural networks outputs (2018). https://naifmehanna.com/2018-09-14-visualizing-convolutional-neural-networks-outputs-part-1/
65. Covid-19 imaging datasets (2021). https://www.eibir.org/covid-19-imaging-datasets/
66. Lee, H. The rise of chatgpt: Exploring its potential in medical education. *Anat. Sci. Educ.* (2023).
67. Artificial intelligence and human rights (2023). https://www.judiciary.senate.gov/committee-activity/hearings/artificial-intelligence-and-human-rights
68. Chia, M. A. *et al.* Validation of a deep learning system for the detection of diabetic retinopathy in indigenous Australians. *Br. J. Ophthalmol.* **108**(2), 268–273 (2024).
69. Obermeyer, Z., Powers, B., Vogeli, C. & Mullainathan, S. Dissecting racial bias in an algorithm used to manage the health of populations. *Science* **366**(6464), 447–453 (2019).
70. Wu, J.-H. & Liu, T. Y. A. Application of deep learning to retinal-image-based oculomics for evaluation of systemic health: A review. *J. Clin. Med.* **12**(1), 152 (2022).
71. Balaskas, K. Oculomics: The eye as a window to systemic disease. *Acta Ophthalmol.* 100(S275). https://doi.org/10.1111/j.1755-3768.2022.15399 (2022).
72. MunishKhanna, Singh, L. K. & Garg, H. A novel approach for human diseases prediction using nature inspired computing & machine learning approach. *Multimed. Tools Appl.* **83**(6), 17773–17809 (2024).

Acknowledgements

We are grateful to Helen V Danesh-Meyer for her contribution in making the EIA2020 dataset available for the study. Waldir Rodrigues de Souza Jr received funding from the Gordon Sanderson Scholarship to conduct the data collection for the EIA2020 dataset.

Author contributions

Glenn Linde, Waldir Rodrigues de Souza Jr, and Renoh Chalakkal all made equal and substantial contributions to the study. This included conception and design, data collection, data analysis, manuscript writing, and final editing. Sheng Chiong Hong, Helen V Danesh-Meyer, and Ben O'Keeffe provided valuable supervision and offered insightful feedback on the manuscript, which enhanced its quality and scholarly rigor. Renoh Chalakkal also provided overall supervision throughout the project and oversaw the submission process.

Competing Interests

The authors declare NO competing interests.

Additional information

Correspondence and requests for materials should be addressed to R.C.