International Journal of Innovative Research in Computer Science and Technology (IJIRCST)
ISSN: 2347-5552, Volume-12, Issue-6, November 2024
https://doi.org/10.55524/ijircst.2024.12.6.11
Article ID IRP-1573, Pages 81-88
www.ijircst.org

AI-Assisted Triage Using CT Head Scans For Emergency Diagnosis

Ahmet Egesoy[1], and Gulce Leylek[2]

[1] Assistant Professor, Department of Computer Engineering, Ege University, Izmir, Turkiye
[2] Student, Department of Computer Engineering, Ege University, Izmir, Turkiye

Correspondence should be addressed to Ahmet Egesoy ahmet.egesoy@ege.edu.tr

Received 21 October 2024; Revised 4 November 2024; Accepted 18 November 2024

ABSTRACT- The main goal of CT-based triage is to shorten the time it takes to reach the expert opinion for patients in emergency situations, especially in cases of cranial fractures and intracranial hemorrhage. Increasing the performance in this regard may be possible with the use of artificial intelligence-supported software that may pre-scan the images and put them in order of urgency before the human triage officer is able to evaluate them. The project involves the development software that quickly classifies cases into one of two groups, urgent or non-urgent, by analysing the computerized tomography (CT) image of the brain taken without the administration of intravenous contrast material. In this way, more effective triage is aimed. The software developed for this purpose was observed to have high performances in two separate machine learning models. Additionally, a visual interface that allows viewing DICOM files was developed within the scope of the project.

KEYWORDS- AI, DenseNet, Machine Learning, Triage, VGG-16

I. INTRODUCTION

Medical imaging encompasses a set of technologies that are indispensable for the detection and diagnosis of diseases and abnormalities. The computer-based nature of imaging technologies, which allows for the generation of numerical data, enhances their effectiveness in terms of speed and reliability when supported by artificial intelligence (AI). AI assistance plays a critical role in two main areas: computer-aided detection (CADe) and computer-aided diagnosis (CADx) [1].

Theoretically, there is no known obstacle preventing a neural network implementation with sufficient complexity and information processing capacity from performing any well-defined pattern recognition task at a level that approaches or even surpasses the performance of a human expert. Significant technological advancements are expected in all areas that can be classified as pattern recognition, where human experts still demonstrate superior performance. In fact, the realization of these advancements likely awaits the resolution of surmountable technical issues, such as the development of accurate models for each area, the attainment of a higher level of hardware performance, or the finding of practical solutions to some key technical challenges related to the application of neural network technology (or a similar technology) in the relevant field.

Artificial intelligence is revolutionizing the healthcare industry and its potential to automate medical triage is also very promising. AI algorithms can assess the severity of a patient's condition much faster than a human specialist and enable healthcare professionals to prioritize the most critical examinations and treatments, allowing for more efficient use of all resources, especially time. Such use of automation will significantly reduce patient waiting times, allowing critically ill patients to receive immediate intervention. Moreover, AI-powered triage systems can help identify potential complications early and suggest proactive life-saving interventions.

The aim of this work is to develop intelligent software that can provide a rapid triage of brain CT scans taken in emergency conditions before radiologist evaluation. It is aimed to reduce patient morbidity and mortality by prioritizing emergency patients in the doctor's work list through this automation. With the use of this software brain CT scans can be analyzed beforehand and placed in the appropriate urgency order. It is hoped that this way, urgent CT scans will be read by radiologists before less urgent CT scans and the patient will be treated more quickly and effectively.

The remainder of this paper is organized as follows: Section II provides general information about the domain and reviews the existing literature. Section III introduces the data source used in this study. Section IV describes the methodology, and Section V presents an assessment of the results. Finally, Section VI concludes the paper.

II. GENERAL INFORMATION

A. Domain Knowledge: AI in Medicine

Artificial Intelligence is revolutionizing many fields, and medical imaging is one of the areas most affected by technological advances. The integration of AI into medical imaging has significant potential to increase diagnostic accuracy, improve the treatment process, and streamline healthcare operations [2]

The most critical contribution of AI to medical imaging is the ability to increase diagnostic accuracy. AI technologies, especially those based on machine learning, can analyze large amounts of imaging data faster and more accurately than human radiologists. Studies have shown that AI can

match or even surpass human performance in detecting diseases such as breast cancer, lung cancer, and diabetic retinopathy [3]. For example, AI systems trained on mammography images have demonstrated greater sensitivity in detecting breast cancer, often identifying subtle patterns that the human eye may miss [4].

The use of AI also improves the workflow efficiency in medical imaging organizations. Traditional imaging workflows that involve human-based image analysis, reporting, and management tasks are time-consuming processes. AI automates many steps of these processes, reducing radiologists' workload and allowing them to focus on more complex cases. AI-powered image analysis tools can pre-scan images, highlight areas of concern, and prioritize cases that require urgent intervention [5]. This automated triage capability will improve patient care by ensuring critical cases are addressed immediately. Additionally, AI can save time and significantly reduce the risk of human error by automating the creation of preliminary reports [6].

AI technologies are also useful in predictive analytics and personalized medicine. AI helps predict disease progression and patient outcomes by analyzing imaging data together with other clinical data. A well-known application is predicting cancer prognosis and treatment responses by evaluating tumor growth patterns obtained from sequential imaging studies [7]. This predictive capability allows for more personalized treatment plans tailored to the specific needs of individual patients for more effective treatments and better overall patient management.

B. State of the Art

Previous studies in the literature have demonstrated the application of machine learning techniques on CT images. One notable project, conducted by Titano et al. [8], evaluated brain computed tomography images using a three-dimensional convolutional neural network model. This study employed deep learning technology, using a 50-layer ResNet50 architecture to address the gradient loss problem, a common challenge in deep learning. Although the accuracy results were modest (*ACC = 0.55, AUC = 0.73*), simulation tests indicated that such automated support could significantly enhance the efficiency of triage processes.

In studies that involve deep learning, two-dimensional image processing is much more common. In some studies, instead of directly processing the image, statistical models of the image are used. For example, Da and his colleagues [9] applied deep learning using the features such as *energy*, *contrast*, *homogeneity*, *correlation*, *entropy*, *variance*, etc. obtained through a statistical analysis known as *GLCL (Gray-Level Co-occurrence Matrix)*, which is widely used due to its simplicity and information density. With this method, they processed brain tomography images by filtering them at different gray depths and reached high accuracy levels (the highest being at 16 gray depths). They also observed that within the framework of their approach, although the learning depth increased the network stability, it did not contribute to the degree of accuracy.

Another noteworthy study is the study by Grewal et. al. [10] who, instead of depicting 3D brain tomography images as 3D matrices consisting of conventional *voxels (3D pixels)*, treated them as discrete but interrelated 2D matrices, as human experts do. This study achieved successful results (81.82% accuracy) despite the small number of training data (329 data in total) by using Densenet [11] and LSTM (Long-Short Term Memory) layers. This study can be considered as an experimental study due to the fact that it only targeted hemorrhage cases and that the test dataset was rather small (*77* images).

A more comprehensive study was conducted by Chilamkurthy and colleagues using a different deep learning architecture consisting of ResNet variants for each type of disorder and pixel-level labeled data in places [12],[13]. This study used a large amount of well-labeled data. In this approach, a separately labeled 2D image set for each investigated pathology was fed into a *Convolutional Neural Network (CNN)*. Specific architectural details such as the number of layers, layer types (convolutional, pooling, fully connected), and activation functions were optimized for the task. The models were trained to identify multiple critical findings simultaneously using multitask learning techniques. The models were trained using a detailed labeled training dataset. The dataset used in the study consisted of *313,318* CT scans, which when multiplied by the number of slices contained in each scan, would create a dataset of millions of images. Data augmentation techniques such as rotation, translation and scaling were applied to further (artificially) increase the size (and diversity) of the training set, further increasing the power of the models.

In this study, deep learning algorithms achieved high sensitivity and specificity in detecting critical cases. Although challenges arose when testing with different datasets (for instance, when sensitivity was increased for detecting intra-parenchymal hemorrhage, specificity dropped to 0.60) cases where both sensitivity and specificity were low simultaneously were rare. One example is the detection of tumors and abscesses, reported as mass effect, where sensitivity rose to 0.86 while specificity remained at 0.61. The study underscores the potential of these algorithms to function as diagnostic aids, particularly in resource-limited environments with restricted access to radiologists.

Due to its fragmented design, comprising different sections that could each serve as standalone projects, this approach is hard to be compared with similar studies or adapted to emerging technologies. This study combines multiple tasks, each addressing another unique problem and offering distinct solutions. Additionally, approaches requiring detailed labeling of data incur significant costs. Given the millions of images (some labeled at the pixel level and others at the slice level) reproducing the success achieved in this study is difficult. Notably, the success rate has been lower for certain conditions, such as tumors and abscesses.

A study conducted by Xiaohong W. Gao and colleagues in 2016 [14] obtained encouraging results by using 2D and 3D CNN networks together and limiting the domain to Alzheimer's and lesion detection. Since the data they used were 2D images of just 16 or 33 slices, the effect of the third dimension on the system performance is limited. The dataset they used consists of a total of *285* (3D) images. The approach is interesting as it uses 3D and 2D convolutional networks together. Hosseini-Asl et al. [15] achieved similar success in Alzheimer's diagnosis using 3D CNN.

Another important study was conducted by Wang and colleagues [16] with the aim of detecting intracranial

hemorrhage (*ICH*), a critical condition requiring immediate medical intervention and typically diagnosed with non-contrast head CT scans. Diagnostic variability among radiologists, interpretation challenges, and increased workloads underscore the value of machine assistance in this area. The study used the *2019-RSNA Brain CT Hemorrhage Detection Challenge* dataset, which includes over *25,000* CT scans. A 2D CNN was combined with two sequence models to detect and classify acute ICH and its subtypes (EDH, IPH, IVH, SAH, and SDH), employing a multi-label classification scheme with rigorous training and validation on a large dataset. High performance was achieved on external datasets (*PhysioNet-ICH and CQ500*), with AUC values of 0.988 for ICH and high values in external validation (*0.964* for *PhysioNet-ICH* and *0.949* for *CQ500*), confirming the model's reliability. The study demonstrated high accuracy in detecting and classifying acute ICH, suggesting that the developed software could be a valuable aid to radiologists in clinical settings. However, the study focused exclusively on intracranial hemorrhage and relied on a large amount of labeled data.

Another successful study aiming *Intracranial Hemorrhage* (*ICH*) detection was conducted by Kuo and his colleagues [17]. This study aimed to identify small abnormalities in noisy, low-contrast CT images using *4,396* head CT scans (1,131 positive and *3,265* negative for *ICH*) provided by *UCSF* and affiliated hospitals. A primary test set of *200* head CT scans (*25 positive, 175 negative*), representing a range of scanner types and conditions, was used. Subsequently, a multi-class prediction was performed with *4,766* labeled scans to identify various types of hemorrhage. The classification model chosen was a fully convolutional neural network (FCN) named *PatchFCN*. Supervised learning with pixel-level labels was carried out by optimizing cross-entropy loss using stochastic gradient descent (SGD).

The study demonstrated an *AUC of 0.99* for detecting *acute intracranial hemorrhage* (ICH), surpassing the performance of 2 out of 4 consulting radiologists involved in the study. This level of performance was achieved by applying image preprocessing that excluded the skull, allowing the model to focus on intracranial structures during the learning phase. High accuracy and reliable localization were attained with a relatively small training dataset, outperforming previous methods, including those based on weaker supervision learning and Mask R-CNN.

Most studies on automatic diagnosis from head CT scans focus on *intracranial hemorrhage*. A study by Cortés-Ferre et al. [18] similarly aimed to develop an original deep learning model for ICH detection. The *RSNA Intracranial Hemorrhage Challenge* dataset from *Kaggle*, containing *752,799* scan slices from *18,938* patients, was used. This dataset was divided into training (*90%*), validation (*5%*), and test (*5%*) sets, with attention to balanced class distribution. The study combined *EfficientDet* and *ResNet* architectures to create an integrated model named *EfficientClass (EffClass)*. For model training, *195,050* slices were used, with *10,802* slices for validation and *10,014* for testing. The model achieved an *AUC-ROC* of *0.978*. Additionally, *87.5%* accuracy was reached with a sensitivity of *100%*. In patient-based evaluation, *100%* sensitivity and *100%* accuracy were achieved using a *10%* threshold. These values were obtained from an external test dataset of *55* cases, collected from two hospitals in Seville, Spain. This external set included *47* patients with *ICH* and *8* healthy patients, presenting a serious imbalance in medical condition.

III. DATA SOURCE

In this study a small open dataset was used from a public source called *PhysioNet* [19]. The formal name of the dataset is *Computed Tomography Images for Intracranial Hemorrhage Detection and Segmentation* (dataset identifier: *ct-ich*) [20]. Although the name suggests otherwise the dataset contains not only hemorrhage but also cases of various fractures.

PhysioNet is an important resource in the biomedical research community and is known for its comprehensive collection of freely accessible datasets and software tools that support complex physiological studies. *PhysioNet* was established as part of the *Research Resource for Complex Physiologic Signals*, created by the *Massachusetts Institute of Technology* (*MIT*) and supported by the *National Institutes of Health* [19]. It has made various contributions to the development of fields such as artificial intelligence and data science, and has impacted areas such as medical research, education, and the development of diagnostic algorithms.

The head trauma dataset was created using *82* CT scans. Among them, *36* scans belonged to patients diagnosed with the condition. Each CT scan contained approximately *30* slices with a slice thickness of *5* mm. The mean and standard deviation of the patients were *27.8* and *19.5* years, respectively. *46* of the patients were male and *36* were female. Each slice of the non-contrast CT scans was performed by two radiologists who recorded possible bleeding and fractures along with their types. Labeling was performed by consensus among the radiologists. The radiologists did not have access to the patients' clinical history and performed evaluations based on a rough version of the CT scan [20].

The cross-sectional images of the dataset are labeled according to diagnoses. In our study to ensure this information is easily accessible to the entire development team, it was planned to encode diagnostic information into the file names. For this purpose, a naming format has been designed to standardize data collected from various sources. The format is as follows:

P<Patient No>_S<Cross-Section No>_D<Diagnosis No>.

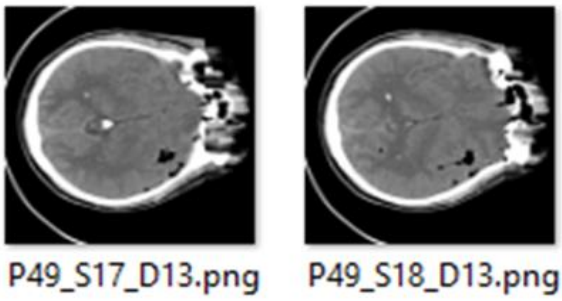

Figure 1: Codified Names of Actual Files

Figure 1 shows some files with codified names. A challenging aspect of this labeling process has been that multiple diagnoses can be associated with a single image. For instance, cases may involve both a skull fracture and a

specific type of bleeding, or two or more types of bleeding occurring together. In such cases, a coding table that includes all possible diagnosis combinations would be required, making file naming too complex to manage effectively.

This problem was resolved using basic arithmetic. In our method, each individual diagnostic label is assigned a unique prime number. Each image is then encoded with a value generated by multiplying the prime numbers corresponding to its diagnoses. This allows for a simple divisibility test to be performed on the file name to verify the presence of any specific diagnosis. The diagnostic labels and their corresponding prime factors are shown in Table 1.

Table 1: Diagnostic Codification Key Prime Factors

Diagnosis	Key P.F.
Intraventicular	2
Intraparenchymal	3
Subarachnoid	5
Epidural	7
Subdural	11
No_Hemorrhage	13
Fracture	17

To extract all diagnostic information from a file name in this format, divide the number following the letter *D (diagnosis)* into its prime factors. For example, an image labeled *D221* indicates a non-bleeding fracture, as $221 = 17 \times 13$ (where *17* is the prime factor for 'fracture' and 13 represents 'no bleeding'). Similarly, *D51* indicates a 'fracture' with *'intraparenchymal bleeding'* as $51 = 17 \times 3$.

IV. METHOD

A. The Software Platform

Software development was conducted using the Anaconda package and the Python programming language. Python's popularity in machine learning is closely tied to its flexibility, extensive library ecosystem, and ease of learning. With its user-friendly syntax, Python also facilitates rapid prototyping and testing. Additionally, powerful libraries and frameworks like *TensorFlow, Keras, PyTorch*, and scikit-learn enable the efficient development of machine learning models.

One of the important stages of the project is to provide the function of accessing and displaying *DICOM* files, which is a common radiology imaging format and can contain large amounts of data. It was necessary to quickly verify that *DICOM* content was interpreted correctly and converted into numeric information. For this purpose, a visual application called *Metrik-G* was developed. This application allows medical images, which by their nature contain a large amount of detail, to be displayed through a narrow contrast window in accordance with current computer monitor technology, and to adjust the width of this contrast window and its position over the entire illumination spectrum as desired. This image processing technique is called *windowing*.

The contrast resolution of medical imaging devices far exceeds both the brightness spectrum that monitors can display and the capabilities of human vision. This is particularly true for computed tomography (*CT*) devices, which use high-frequency radiation in imaging. Regardless of the imaging technique used, it is generally impossible to view all details in a typical *CT* scan simultaneously. As a result, the examination of medical images necessarily involves image processing techniques. One common technique is *windowing*, which is used to obtain an image viewable on a monitor. *Windowing* involves selecting a sub-range of the brightness that falls within the monitor's displayable range, and mapping only the pixels within this range to values displayable gray-scale values. In other words, a narrow brightness 'window' is chosen from a much larger brightness range, and only this range is displayed. Pixels outside this window appear fully white if they are brighter than the range, and fully black if they are darker.

In medical imaging, it is critical to adjust the windowing process to the physical properties of the tissue to be imaged. Different tissues absorb different amounts of radiation, resulting in varying density values in the image. Without proper windowing, it will be impossible to distinguish subtle differences in tissue density.

There are two parameters to the windowing process: *Window Width (WW)* and *Window Level (WL)* [21]. By adjusting the window width and level, medical personnel can customize the image display to highlight specific areas of interest.

Window Width (WW): This parameter determines the range of intensity values displayed in the image. A narrow window width increases contrast for structures within a certain intensity range, but may result in loss of detail in areas outside this range. These will be painted either completely black or completely white, effectively removing them from the image.

Window Level (WL): The location of the window over the entire brightness spectrum. Also known as the *window center*, this parameter locates the midpoint of the intensity range to be displayed. Adjusting the window level shifts the range of values displayed up or down the intensity scale.

By optimizing contrast and brightness with windowing, the differences between tissues with similar densities can be better separated. Improved visualization achieved with the right parameters helps radiologists and clinicians make more accurate diagnoses and assessments. For example, the *Lung window (WW=1500, WL=-600)* used in chest *CT* scanning is used to increase the visibility of the lung parenchyma. The *Mediastinum window (WW=350, WL=50)* that can be used on the same scan is used to better visualize the mediastinum structures and blood vessels. There are four different windows commonly used in head images: *Brain Window (WW=80-100, WL=40), Bone Window (WW=2500-4000, WL=300-500), Subdural Window (WW=200-300, WL=50-100) and Stroke Window (WW=8-40, WL=32-40).*

B. DenseNet-121

DenseNet-*121* (*Densely Connected Convolutional Networks*) is a type of *Convolutional Neural Network* (*CNN*) architecture designed to improve the flow of information and gradients throughout the network by connecting each layer to other layers in a feedforward manner. [11] In *DenseNet-121*, each layer takes input from all previous layers and passes its own feature maps to all subsequent layers, which improves feature reuse and reduces the number of parameters compared to traditional *CNN*s. This architecture helps alleviate the vanishing gradient problem, provides more efficient training, and achieves high performance in image classification tasks.

C. VGG-16

Developed by the *Visual Geometry Group* (*VGG*) at the University of Oxford, *VGG-16* is a *Convolutional Neural Network* (*CNN*) architecture that achieves significant performance in image classification tasks. It consists of *16* weight layers: *13* convolutional layers and *3* fully connected layers, totaling approximately *138* million parameters. The architecture uses small *3x3* convolutional filters that are stacked to increase the depth of the network, allowing it to learn complex features. *VGG-16* is known for its simplicity and efficiency and has played a significant role in the advancement of deep learning research in computer vision [22].

D. ResNet50 and ResNet152V2

ResNet-50 is a *Convolutional Neural Network* (*CNN*) architecture that is part of the *ResNet* (*Residual Network*) family. Developed by Kaiming He et al., *ResNet-50* contains *50* layers and uses the deep residual learning framework. The key feature of *ResNet* is the introduction of residual blocks, where shortcut connections (skip connections) skip one or more layers. This design helps to cope with the vanishing gradient problem, enabling very deep networks to be trained. *ResNet-50*, like other *ResNet* versions, has shown high performance in image classification tasks and has been widely used for various computer vision applications [23].

ResNet-152V2 is an advanced version of the original *ResNet* (*Residual Network*) architecture, specifically designed to improve the training of very deep neural networks. It contains *152* layers and uses residual learning, where shortcut connections are used to skip one or more layers. The "*V2*" version includes improvements such as pre-activation residual blocks, which further improve the training process by normalizing the input before applying the activation function. *ResNet-152V2* demonstrates good performance in image classification and other computer vision tasks [23].

E. Xception

Xception is a *Convolutional Neural Network* architecture that stands for "*Extreme Inception*". It was developed by François Chollet and is based on the *Inception architecture*, but with significant changes. In *Xception*, instead of the standard Inception modules, there are *depth-separable* convolutions which is a type of *factorized convolution*. This change reduces the number of parameters and computational cost while maintaining or improving performance. *Xception* consists of *36* convolutional layers that form the feature extraction base of the network, followed by a fully connected layer for classification. This architecture is known for its efficiency and effectiveness in image classification and other computer vision tasks [24].

V. ASSESSMENT OF RESULTS

A. Performance Metrics

Performance evaluation of AI-based classification models is critical to ensure reliability and effectiveness in various applications, especially in sensitive areas such as healthcare and autonomous driving.

The most basic tool used to evaluate the performance of classification models is the complexity matrix. The complexity matrix consists of four basic terms: True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). These terms facilitate the definition of various performance metrics that provide different insights into the predictive capabilities of the model [25]. In the medical domain, True Positives (TP) occur when a test correctly identifies patients who have a disease. True Negatives (TN) happen when the test correctly identifies healthy individuals without the disease. False Positives (FP) are cases where the test incorrectly indicates disease in healthy individuals. False Negatives (FN) occur when the test fails to detect the disease in patients who actually have it. From these values, several statistical metrics can be derived to quantify the success of the tests:

Precision: It is also known as Positive Predictive Value. It is defined as the ratio of true positive predictions to the total number of positive predictions made by the model (i.e., TP / (TP + FP)). Precision indicates the proportion of positive diagnoses that are actually correct. High precision is especially important in scenarios such as disease diagnosis, where the cost of false positives is high [26]. If this value is low, there is a risk that an expensive treatment or examination will be applied to a healthy person when it is not necessary.

Accuracy: The ratio of correctly predicted examples (both true positives and true negatives) to the total number of examples (i.e. (TP + TN) / (TP + TN + FP + FN)). Accuracy provides an overall measure of how often the classifier is correct, but can be misleading with imbalanced datasets where one class heavily dominates [27].

Sensitivity: Also known as Recall or True Positive Rate. It is defined as the ratio of true positive predictions to the total number of true positives (i.e. TP/(TP+FN)). Sensitivity measures the ability of the model to identify all relevant samples. High sensitivity is very important in situations where missing a positive sample (false negative) has serious consequences, such as in cancer screening [28].

F1-score: Metrics known as F-Metrics, especially the F1-score, provide a harmonic mean of precision and accuracy, creating a single measure that takes both concerns into account and balances them (F1 = 2 * (Precision * Precision) / (Precision + Precision)). The F1-score is particularly useful when the balance between precision and accuracy is critical, and provides a more comprehensive assessment of model performance in scenarios where both false positives and false negatives are important. Other F-metrics calculate the harmonic mean of these two measures with different weights (weighted harmonic mean). In this way, it is possible to prioritize precision over precision if desired.

B. Initial Slice-based Performance

In the initial experiments, slice based labeling was disregarded at the learning phase, though performance evaluations were conducted entirely on a slice basis. After fine-tuning, training was performed with 20 epochs on the DenseNet-121, VGG-16, and ResNet152V2 models, followed by completion of training with 100 epochs.

The evaluation of the algorithms has a two-fold nature. The first aspect is assessing the discriminatory power each algorithm achieves in classifying head CT scans. This type of evaluation involves measuring the algorithms' (or models') ability to classify images and is based on a cross-

sectional approach, utilizing a large dataset of 1,047 images. However, such a comparison serves primarily to benchmark theoretical image classification algorithms and has limited practical benefit. In real medical applications, it is not individual cross-sections that need classification, but rather the patients as a whole. Therefore, the evaluation that holds true importance is the one conducted on a patient (case) basis.

On the other hand, because case-by-case classification requires more detailed experimentation, it is necessary first to identify models that show promise in evaluating the images. Table 2 presents a comparison of models produced by different architectures. This approach allows the effective algorithms and architectures to be selected initially, followed by more detailed examination.

Table 2: Slice Image Classification Performances

Model	Prec.	Acc.	Rec.	F1
DenseNet-121	0.593	0.784	0.526	0.557
VGG-16	0.569	0.779	0.599	0.583
ResNet152	0	0.741	0	0
Xception	0	0.741	0	0
ResNet50	0.527	0.760	0.711	0.605
VGG-16/S	0.540	0.899	0.530	0.535
DenseNet-121/S	0.395	0.860	0.522	0.449

Seven models are compared in Table 2: *DenseNet-121*, *VGG-16*, *ResNet152*, *Xception*, *ResNet50*, *VGG-16/S*, and *DenseNet-121/S;* in terms of *Precision*, *Accuracy*, *Recall*, and *F1* metrics. *VGG-16/S* and *DenseNet-121/S* are slice-based trained versions of the respective models
ResNet, *VGG-16/S*, and *DenseNet-121/S* performed well. In contrast, the *ResNet152 and Xception* algorithms failed to classify any images as positive (patient), rendering their classifications ineffective, at least for this small dataset. Consequently, the obtained accuracy values for these models are not meaningful.

C. Case-based Performance

Case-based evaluation aims to analyze the distribution of slice classes within each scan and proceed to case classification based on these results. There are several possible approaches to achieve this challenge. Following a brief review, it became evident that using a simple threshold value for slice classifications is an effective and straightforward method. For larger datasets or larger DICOM files with more slices per scan, alternative methods or algorithms could be developed. Under current conditions, however, the threshold method appears sufficient. This study, conducted on slice-labeled versions of the *VGG-16* and *DenseNet-121* models, continued by establishing three threshold values for each algorithm.

Table 3: Distribution of Slice Classes Over Scans

Model	Real	Pos. Slices	Neg. Slices	Pos. Ratio %
VGG-16/S	*Unhealthy*	5.55	4.91	49.24
	Healthy	1.79	30.35	5.57
	Total	2.82	23.35	17.58
DenseNet-121/S	*Unhealthy*	5.45	5.00	55.64
	Healthy	3.17	29.10	9.76
	Total	3.80	22.47	22.38

In Table 3, the two most successful architectures (*VGG-16/S and DenseNet-121/S*) are compared based on the average values of slices per scan. All values in the table represent averages. Separate rows show the averages for unhealthy individuals, healthy individuals, and all scans combined. In the first column, the average number of slices classified as *positive* (marked by algorithm as *unhealthy*) is presented as three separate averages, as noted. The second column provides the averages for slices classified as *negative* (marked by algorithm as *healthy*) in the same format. The last column displays the average positive section ratios as percentages. It is important to note that these percentages represent slice ratios per scan, averaged across all scans, so the values for patients and healthy individuals may not add up to *100%.* Additionally, the sum of the Pos. Slice Count and Neg. Slice Count does not equal the total slice count per scan, as these values are the average counts within each scan.

The dataset contains a total of *1419* cross-sectional images of healthy views and *1395* cross-sectional images for pathologies. The data is divided into *70%* for training and *30%* for testing, ensuring that no cross-sectional images from a patient in the test set are present in the training set. In other words, all images from a single patient are included entirely in either the training set or the test set. This separation is crucial to prevent the learning algorithms from inadvertently identifying patients based on personal physiological features, rather than diagnosing the disease. Without this separation, artificial intelligence could attempt to recognize patients individually by using subtle traits, such as bone thickness or unique bone angles, as shortcuts. Although this could yield misleadingly high performance, the model would likely fail to generalize when encountering new patients. To avoid this issue, all cross-sectional images from each patient are assigned exclusively to either the training or the test set.

Table 4: Case Classification Performances

Model	Prec.	Acc.	Rec.	F1
VGG-16/S (20%)	0.800	0.875	0.727	0.762
DenseNet-121/S (10%)	0.524	0.750	1.000	0.688
VGG-16/S (30%)	1.000	0.900	0.636	0.778
DenseNet-121/S (20%)	0.667	0.825	0.727	0.696
VGG-16/S (50%)	1.000	0.875	0.545	0.706
DenseNet-121/S (10%)	1.000	0.900	0.636	0.778

In Table 4, *VGG-16* and *DenseNet-121* models trained with only slice-based data were evaluated using different threshold values on performance metrics. The leap from *slice-based* evaluation to *scan-based* evaluation is performed using a very simple method. The number of slices evaluated as positive for each patient is compared with a certain threshold value, and if the patient exceeds this value, they are classified as positive (*unhealthy*), and if they remain below, they are classified as negative (*healthy*). Since each scan consists of a different number of slices, a healthy borderline can only be expressed as a fraction. These rates are given as percentage values in the table. For *VGG-16*, the *20%*, *30%* and *50%* border values, and for *DenseNet-121*, the *10%*, *20%* and *50%* values were found to be significant in terms of effective discrimination.

Using this method, case-based performance evaluation yields improved values compared to slice-based evaluation. The highest accuracy (*90%*) is achieved by the *DenseNet-*

121 model with a *50%* threshold and the *VGG-16* model with a *30%* threshold. These two configurations also produced the best *F1* metric values. If sensitivity (recall) is prioritized, DenseNet-121 with a 10% threshold can be selected.

In automated diagnosis tasks, the cost of missing a sick individual (*false negative*) is typically much higher than misidentifying a healthy individual as sick (*false positive*). Therefore, prioritizing maximum sensitivity with an acceptable level of accuracy can be considered as a rational policy, particularly when the primary goal is triage.

Among similar studies, those that achieved better performance results generally focused on a limited diagnostic area, such as *intracranial hemorrhage* (brain hemorrhage and related disorders). In contrast, the study by Chilamkurthy et al. [12],[13], which addressed a broader diagnostic scope, is not fully comparable because it applied different solutions to each disorder, achieving high performance in some cases but not in others. Additionally, their study used a very large, detailed dataset with separate labeling for each diagnostic purpose. Our requirements and approach differ: we developed a faster, lighter classification software for computer-aided rapid triage, using a smaller dataset. Despite this limitation, we demonstrated that *100%* precision could be achieved with *90%* accuracy. This performance could potentially be improved with access to larger, labeled datasets.

VI. CONCLUSION

As our main purpose is optimizing the triage order, all the methods in Table 3 appear to be sufficiently effective. It should be noted, that these results were obtained using a very small dataset. With more slice-labeled data, the learning process would likely become more efficient, and performance metrics would further improve. Additionally, the current methods have been tested on scans with slice counts ranging from *4* to *39*, though CT scans typically contain more slices.

In the experiments, *DenseNet-121* and *VGG-16* architectures have shown high performance with slice-based labeled data and certain limit values. These results were obtained with small amounts of unbalanced data, and data acquisition studies are ongoing for further improvements.

Integrating machine learning algorithms such as *VGG-16* and *DenseNet-121* into the analysis of CT scans, along with appropriate cross-sectional assessment methods, has the potential to significantly improve the medical triage process. Machine learning models can rapidly and successfully identify and prioritize critical conditions from CT images, thereby accelerating diagnostic timelines and improving patient management. Advanced automated analysis capabilities will not only facilitate workflow efficiency in emergency situations, but will also ensure that patients receive timely and appropriate care according to the severity of their condition. As a result, it is believed that the use of machine learning in CT scan analysis will further advance emergency medical services by providing more precise and efficient patient assessments. This project is a useful and successful step in this direction.

When moving from the section-based classification stage to the case-based classification level, using methods other than the cut-off value may be one of the topics to be investigated in the future. There are many complex algorithms that can be considered, and creating and competing them may yield better results.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

ACKNOWLEDGMENT

This work was supported by Ege University Scientific Research Projects Coordination Unit. Project Number: 21422.

REFERENCES

[1] J. Gao, Q. Jiang, B. Zhou, and D. Chen, "Convolutional neural networks for computer-aided detection or diagnosis in medical image analysis: An overview," *Math. Biosci. Eng.*, vol. 16, no. 6, pp. 6536–6561, 2019, Available from: https://dx.doi.org/ 10.3934/mbe.2019326

[2] R. Najjar, "Redefining radiology: A review of artificial intelligence integration in medical imaging," *Diagnostics*, vol. 13, no. 17, p. 2760, 2023. Available from: https://dx.doi.org/10.3390/diagnostics13172760

[3] S. M. McKinney, M. Sieniek, V. Godbole, J. Godwin, N. Antropova, H. Ashrafian, et al., "International evaluation of an AI system for breast cancer screening," *Nature*, vol. 577, no. 7788, pp. 89–94, 2020. Available from: https://doi.org/10.1038/s41586-019-1799-6

[4] A. Rodríguez-Ruiz, K. Lång, A. Gubern-Mérida, M. Broeders, G. Gennaro, P. Clauser, et al., "Standalone artificial intelligence for breast cancer detection in mammography: Comparison with 101 radiologists," *JAMA Oncol.*, vol. 5, no. 9, pp. 1397–1404, 2019. Available from: http://dx.doi.org/10.1093/jnci/djy222

[5] V. Makeeva, *An Essential Roadmap for AI in Radiology* Reston, VA: American College of Radiology, 2022. Available from: https://shorturl.at/V2lrQ

[6] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, et al., "A survey on deep learning in medical image analysis," *Med. Image Anal.*, vol. 42, pp. 60–88, 2017. Available from: https://doi.org/10.48550/arXiv.1702.05747

[7] F. Jiang, Y. Jiang, H. Zhi, Y. Dong, H. Li, S. Ma, et al., "Artificial intelligence in healthcare: Past, present and future," *Stroke Vasc. Neurol.*, vol. 2, no. 4, pp. 230–243, 2017. Available from: https://doi.org/10.1016/j.ibmed.2021.100047

[8] J. J. Titano, M. Badgeley, J. Schefflein, et al., "Automated deep-neural-network surveillance of cranial images for acute neurologic events," *Nat. Med.*, vol. 24, pp. 1337–1341, 2018, Available from: https://dx.doi.org/ 10.1038/s41591-018-0147-y

[9] C. Da, H. Zhang, and Y. Sang, "Brain CT image classification with deep neural networks," in *Proc. 18th Asia Pacific Symp. Intelligent Evol. Syst.*, vol. 1, H. Handa, H. Ishibuchi, Y. S. Ong, and K. Tan, Eds., Cham: Springer, 2015. Available from: http://dx.doi.org/10.1007/978-3-319-13359-1_50

[10] M. Grewal, M. M. Srivastava, P. Kumar, and S. Varadarajan, "RADNET: Radiologist level accuracy using deep learning for hemorrhage detection in CT scans," *arXiv*, 2017. [Accessed: Aug. 20, 2018]. Available from: https://arxiv.org/abs/1710.04934

[11] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 4700–4708, Available from: https://dx.doi.org/10.1109/CVPR.2017.243

[12] S. Chilamkurthy, R. Ghosh, S. Tanamala, M. Biviji, N. G. Campeau, V. K. Venugopal, V. Mahajan, P. Rao, and P.

Warier, "Development and validation of deep learning algorithms for detection of critical findings in head CT scans," Apr. 2018. *arXiv*, vol. abs/1803.05854 Available from: https://doi.org/10.48550/arXiv.1803.05854
[13] S. Chilamkurthy, R. Ghosh, S. Tanamala, M. Biviji, N. G. Campeau, V. K. Venugopal, V. Mahajan, P. Rao, and P. Warier, "Deep learning algorithms for detection of critical findings in head CT scans: A retrospective study," *The Lancet*, Dec. 2018, Available from: https://dx.doi.org/10.1016/s0140-6736(18)31645-3
[14] X. W. Gao, R. Hui, and Z. Tian, "Classification of CT brain images based on deep learning networks," *Comput. Methods Programs Biomed.*, vol. 138, pp. 49–56, 2017. Available from: https://https://doi.org/10.1109/SAI.2016.7555958
[15] E. Hosseini-Asl, M. Ghazal, A. Mahmoud, et al., "Alzheimer's disease diagnostics by a 3D deeply supervised adaptable convolutional network," *Front. Biosci. (Landmark Ed)*, vol. 23, pp. 584–596, 2018. Available from: https://doi.org/10.48550/arXiv.1607.00556
[16] X. Wang, T. Shen, S. Yang, J. Lan, Y. Xu, M. Wang, J. Zhang, and X. Han, "A deep learning algorithm for automatic detection and classification of acute intracranial hemorrhages in head CT scans," *NeuroImage: Clin.*, vol. 32, Article 102785, 2021. Available: https://doi.org/10.1016/j.nicl.2021.102785.
[17] W. Kuo, C. Häne, P. Mukherjee, J. Malik, and E. Yuh, "Expert-level detection of acute intracranial hemorrhage on head computed tomography using deep learning," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 116, no. 35, p. 201908021, 2019. Available from: https://dx.doi.org/10.1073/pnas.1908021116
[18] L. Cortés-Ferre, M. A. Gutiérrez-Naranjo, J. J. Egea-Guerrero, S. Pérez-Sánchez, and M. Balcerzyk, "Deep learning applied to intracranial hemorrhage detection," *J. Imaging*, vol. 9, p. 37, 2023, Available from: https://dx.doi.org/10.3390/jimaging9020037
[19] A. Goldberger, L. Amaral, L. Glass, J. Hausdorff, P. C. Ivanov, R. Mark, et al., "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000. Available from: http://dx.doi.org/10.1161/01.CIR.101.23.e215
[20] M. Hssayeni, "Computed tomography images for intracranial hemorrhage detection and segmentation (version 1.3.1)," *PhysioNet*, 2020, Available from: https://dx.doi.org/10.13026/4nae-zg36
[21] F. Gaillard, "CT window and algorithm effects," *Radiopaedia.org*, Sep. 23, 2017. [Accessed: Jul. 30, 2024]. Available from: https://doi.org/10.53347/rID-55748
[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv* preprint arXiv:1409.1556, 2014. Available from: https://arxiv.org/abs/1409.1556
[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 770–778. Available: https://doi.org/10.1109/CVPR.2016.90
[24] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 1251–1258. Available: https://doi.org/10.1109/CVPR.2017.195
[25] R. Kohavi and F. Provost, "Glossary of terms," *Mach. Learn.*, vol. 2, pp. 271–274, 1998, Available from: https://dx.doi.org/10.1023/A:1017181826899
[26] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Inf. Process. Manage.*, vol. 45, no. 4, pp. 427–437, Jul. 2009, Available from: https://dx.doi.org/10.1016/j.ipm.2009.03.002
[27] D. M. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation," *J. Mach. Learn. Technol.*, vol. 2, no. 1, pp. 37–63, Mar. 2011, Available from: https://dx.doi.org/10.48550/arXiv.2010.16061
[28] J. Davis and M. Goadrich, "The relationship between precision-recall and ROC curves," in *Proc. 23rd Int. Conf. Mach. Learn.*, ACM, 2006. Available from: https://dl.acm.org/doi/10.1145/1143844.1143874

ABOUT THE AUTHORS

Ahmet Egesoy (PhD in Computer Engineering) is an instructor and Assistant Professor in Computer Engineering Department of Ege University Izmir, Turkey. Research interests include Object-Oriented Programming, Design Patterns, Model-Driven Software Development, Programming Languages and related paradigms, Philosophy of The Language, Semiotics And Knowledge Representation.

Gülce Leylek (M.S. in Computer Engineering) is a System Engineer in Kentkart Izmir, Turkey. Completed her master degree in Ege University Department of Computer Engineering. Research interests include Machine Learning, Image Processing, Image Enhancement and Behaviour Driven Development.