IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 29 March 2024, accepted 25 April 2024, date of publication 6 May 2024, date of current version 15 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3396999

METHODS

Automated Detection of Spinal Lesions From CT Scans via Deep Transfer Learning

ANDREA CAMISA[1], GIOVANNI MONTANARI[2], ANDREA TESTA[3], (Member, IEEE), LUIGI FALZETTI[4], SOFIA AVNET[5], NICOLA BALDINI[4,5], AND GIUSEPPE NOTARSTEFANO[3], (Member, IEEE)

[1]Helsing, W1T 3BL London, U.K.
[2]Salcef Group, 00138 Rome, Italy
[3]Department of Electrical, Electronic and Information Engineering, University of Bologna, 40126 Bologna, Italy
[4]Biomedical Science and Technologies and Nanobiotechnology Laboratory, IRCCS Istituto Ortopedico Rizzoli, 40136 Bologna, Italy
[5]Department of Biomedical and Neuromotor Sciences, University of Bologna, 40126 Bologna, Italy

Corresponding author: Andrea Testa (a.testa@unibo.it)

ABSTRACT Convolutional Neural Networks are being increasingly applied to the detection of anomalies in Computed Tomographies (CTs). The goal of this paper is to implement an automated Computer-Aided Detection (CADe) system for spinal lesions using CTs and Convolutional Neural Networks pre-trained on commercial datasets. The proposed pipeline works as follows. The CADe takes in input CT scans and is equipped with an intuitive GUI to allow physicians to use it as a support tool for diagnoses. From the CT scans, the CADe selects volumes containing the vertebrae and extracts 2D slices of these volumes. These slices are pre-processed and then analyzed using a VGG19 Convolutional Neural Network and a tailored, binary classifier. The neural network identifies healthy vertebrae and vertebrae containing lesions (e.g. metastases, primary tumors, lytic and sclerotic lesions). For training and testing purposes, we generated a dataset from CTs of vertebrae from patients treated at IRCCS Rizzoli Orthopaedics Institute of Bologna, Italy, between 2009 and 2019. Both healthy and lesioned vertebrae retrieved with different tomography machines and setups are considered. The dataset has been enlarged using data augmentation techniques and subsequently used to train a wide range of deep learning models. We perform an in-depth benchmark study to assess the performance of the considered classifier against different Convolutional Neural Networks pre-trained on the ImageNet dataset and exploiting Transfer Learning techniques. Leveraging Transfer Learning techniques, we reached 93.43% accuracy and a recall of 92.99%.

INDEX TERMS CT scan, deep learning, lesion detection, neural networks, spinal lesions, transfer learning.

I. INTRODUCTION

Computer Aided Detection (CADe) techniques for the analysis of medical images have been rapidly growing in both importance and sophistication, see, e.g., [1] and references therein. Diagnostic tools can be life-saving, especially when dealing with extensive exams such as Computed Tomographies (CTs) that include a substantial section of the patient's body and require great attention from medical staff. In recent times, there has been a growing attention for automated diagnosis tools stemming from Artificial Intelligence. Important techniques to elaborate CTs, MRIs, ultrasound, and radiographs are Convolutional Neural Networks (CNNs). These networks proved to be excellently reliable for image analysis, reaching the state of the art for several medical tasks such as disease detection and classification in the spinal column. Detection and diagnosis of pathologies such as spinal metastases (i.e. tumors that have origin in organs and then extend to other locations) represent a critical task to be accomplished. Spinal metastases are indeed very frequent. It is estimated that 50% to 70% of cancer patients have bone metastases [2]. However, when dealing with spine tumors, diagnosis through 3D imaging (e.g. CT or MRI) is way more delicate, due to the complexity of the vertebral body and the considerable number of slices to examine. Thereby, the development of a fully automated method for the detection of pathological features represents a timely challenge. In this paper, we consider the development

The associate editor coordinating the review of this manuscript and approving it for publication was Paolo Crippa.

of an automated detection system that detects possible abnormalities in vertebrae, leveraging deep transfer learning methodologies. The proposed tool automatically extracts slices of the vertebral body and detects possible anomalies. The system is equipped with a graphical interface to ease its usage as a support tool for diagnoses.

A. RELATED WORK

Deep learning methodologies have shown promising results in cancer detection, segmentation, and classification, see, e.g., the recent survey [3]. Examples of the different cases of study to which deep learning has been applied include the detection, segmentation, and classification of breast cancer [4], lung cancer [5], brain tumors [6], [7], skin lesion [8], liver tumor [9], [10], and musculoskeletal disesases [11]. Studies about viral and bacterial pneumonia detection from X-rays were made [12]. Latest research focuses on COVID-19 detection from thoracic CT scans [13], [14], [15]. These works consider different imaging problems, in which the inputs come from CT scans [5], [9], [10], [13], [14], [15], X-Rays [12], MRI [6], [7], or RGB images [8]. These methods use open-source networks like, e.g., UNet [16] and SegNet [17] suitably modified. When dealing with small datasets like in medical applications, training can be a difficult task. For this reason, one of the latest approaches is *Transfer Learning*, i.e. the practice of using state-of-the-art networks that have already been trained from huge datasets and then transferring the achieved knowledge to another domain. Inspired by these approaches, in this paper, we consider the development of a deep-learning-based CADe system for spine lesions in CT scans. Focusing on the spine, a considerable amount of studies regard the identification of the vertebral body volume, see, e.g., the work in [18]. In these works, the goal is to identify and label vertebrae, rather than classify eventual diseases. Several works focus on scoliosis and spondylolisthesis detection and classification, see the survey [19], and the recent works [20], [21], [22]. Differently from our work, these approaches do not focus on vertebral lesions. Moreover, papers in [21] and [22] consider X-Ray images, and the work [20] considers ultrasound data. On the other hand, we consider CT scans. As for the segmentation of intervertebral disks, authors in [23] propose a harmonization approach based on GANs. Differently from our approach, they do not consider lesion classification. Moreover, their method is tailored for MRI images. Lumbar diseases are instead considered in [24]. Similarly to our method, authors propose a classification scheme based on VGG networks. However, they mainly focus on MRI images and do not consider vertebral lesions. A grading system for intervertebral disk degeneration is proposed in [25]. Similarly to the work in [24], authors do not consider vertebral lesions, and their approach is tailored for MRI images.

As regards spine tumors, the attention is entirely on spine metastases. Several works consider deep learning models for the detection of abnormalities in the spinal column. In [26], authors propose a study of the performances of a Faster R-CNN network on MRI images. In [27] authors exploit Siamese networks to detect spine metastases from MRI images. Despite the high performance, the slices have to be manually extracted from the MRI sequences. In [28] MRIs are exploited to label vertebrae and detect the presence of tumors. However, only 2D slices from the sagittal plane are considered, making small or non-centered tumors hard to detect. As regards spine metastasis segmentation, [29], [30] use the UNet neural network on MRI images. Differently from these approaches, in this paper, we consider instead an automatic tool to detect spinal lesions from CT scans rather than MRIs. Also, the proposed methodology does not require manual annotations.

As for deep learning methodologies for detection in CT scans, authors in [31] propose an unbalanced UNet for labeling and segmentation of vertebrae, without a particular focus on abnormalities. The work in [32] proposes a pipeline based on a region of interest detector and a multi-modal classifier for spinal metastases. Differently from our work, the paper focuses on metastatic epidural spinal cord compression. In [33] instead, authors propose a diagnosis method for benign and malignant vertebral fractures leveraging the ResNet50 network. Differently from our proposed scheme, the pipeline is based on sagittal scans and requires human intervention to select a first region of interest. In [34], authors exploit a CNN to implement a spinal metastasis "hybrid" detection-segmentation tool from CT scans. This fully automatic detection system showed good results, even though tests were made on patients with a high amount of metastases, while no control (i.e., healthy) patients were considered. In our work instead, we consider both healthy and non-healthy vertebrae. In [35], the focus is on CT for the segmentation of metastases from manually extracted volumes of interest. Our methodology instead does not require manual extractions. Authors in [36] consider a detection system based on a CNN for lytic spinal lesions. Differently from our approach, authors here consider only lytic lesions. Moreover, they do not propose a CADe tool to automatize the diagnosis process. As for CADe systems for automated detection of metastases, in [37] authors develop an iterative approach, based on random forests, for the detection and classification of metastatic lesions in sagittal scans. The results have good sensitivity, but a large number of false positives. Detection (and segmentation) of sclerotic metastases is also addressed in [38] from CT exams of the thoracolumbar section of the spine exploiting a watershed algorithm. Authors in [39] build a cascade system including the pipeline in [38] and an ad-hoc CNN trained on CIFAR-10. In our pipeline instead, we do not rely on external CADe systems, and the detection is based on a state-of-the-art CNN pre-trained on a larger dataset.

B. CONTRIBUTIONS

The main contributions can be summarized as follows:

- we develop a CADe system for automatic extraction and detection of abnormalities in vertebrae;

- we develop a Graphical User Interface to make the CADe available to the medical staff;
- we benchmark the proposed methodology against existing methods in literature and other state-of-the-art networks.

More in detail, we develop a CADe (Computer-Aided Detection) system that automatically extracts vertebrae from CT scans and detects possible abnormalities to be further evaluated by the medical staff. We conceptualize and implement a system to automatize the steps for the lesion detection process by providing utilities for the extraction of the vertebrae from CTs and their subsequent analysis. The developed detection tool turns out to be robust with respect to different tomography machines and setups. In addition, the detection is not only focused on metastases but also on primary tumors (chordomas, aggressive hemangiomas) and lesions (either lytic or sclerotic), i.e., cases from our dataset required a biopsy to determine the nature of the lesion. The detection system, based on a VGG19 Convolutional Neural Network for binary classification, predicts whether the considered slices contain abnormalities (i.e. lesions) and give a confidence index of the prediction. To train and test the CADe, we selected 256 vertebrae, from a large group of CT scans performed on a continuous series of 105 patients with bone lesions (either primitive or metastatic) or affected by degenerative condition of the spine treated at IRCCS Rizzoli Orthopaedics Institute (IOR) of Bologna, Italy, between 2009 and 2019. We excluded patients with spinal internal fixation and vertebrae from the cervical tract. The study was approved by the local Ethical Committee (CE-AVEC 867/2021/Oss/IOR). We benchmark the chosen CNN against other state-of-the-art CNNs, i.e., VGG-16 [40], Inception V3 [41] and ResNet 50 [42]. Moreover, we benchmark the proposed method against other methods in the literature. The considered approach outperforms the other in all the considered metrics. The study highlights excellent performance when implementing Transfer Learning techniques, both in computation time and accuracy, and shows that VGG-19 is the most suitable network for our task. Finally, we developed a Graphical User Interface (GUI) that allows radiologists and other medical staff to adopt the developed toolbox through simple operations. The paper unfolds as follows. Data description and preliminaries on spinal lesions and CT are provided in Section II. In Section III, we present the developed system and its Graphical User Interface. In Section IV, we provide details about the dataset preprocessing and data augmentation techniques employed in the algorithm. Training, test, and benchmark results for the proposed CADe are discussed in Section V.

II. DATA DESCRIPTION AND PREPROCESSING

A. SPINAL LESIONS

Spine tumors are classified into two categories, depending on their nature. Primary tumors originate directly in the spine or its adjacent tissues while secondary ones, also called metastases, derive from previously existing tumors located in organs. The first ones have an unknown incidence due to their rarity and asymptomatic nature, even though some estimates have been done concerning hemangiomas [2] (the most common primary tumors of the spine) resulting in an 11% to 14% incidence. On the other hand, metastases are very frequent, accounting for 97% of spine tumors with respect to primary ones. The latter occurs in 50-70% of patients with previous cancers. Statistics are different when considering breast cancer, which alone accounts for 85% of occurrence of bone metastases. Spinal lesions that characterize tumors can be often distinguished as lytic if they involve bone degradation and sclerotic (or blastic) when an abnormal increase in bone density can be appreciated. Sometimes lesions are mixed, involving both lytic and sclerotic behaviors. The frequency of metastases also depends on the location inside the spine. Indeed, the most frequent locations are thoracic and thoracolumbar tracts (70%), while the lumbar one together with the sacrum hosts on average 20% of metastases. The cervical spine is a less recurrent candidate location.

B. COMPUTED TOMOGRAPHIES FOR SPINE TUMORS

Computed Tomography is an advanced and precise technique, that exploits a narrow beam of X-rays that rotates around the patient to reconstruct a 3-plane view of the body. Each full rotation of the beam generates a *slice*, namely a section of the body with a certain thickness (or depth, representing the spacing between slices). The resulting slices compose a volume whose units are called *voxels* (the 3D counterpart of pixels). Each voxel has a value that is directly proportional to the density of the represented tissue. In particular, the scanner measures the attenuation coefficient μ of the X-ray beam after it has pierced the body tissues. The intensity of the corresponding voxel is measured in *Hounsfield Units* (HU) which is a transformation scale of μ. The relationship between HU values and the attenuation coefficient is $\text{HU} = 1000 \cdot \frac{\mu - \mu_{\text{water}}}{\mu_{\text{water}} - \mu_{\text{air}}}$, where μ_{water} is the radiodensity of water and μ_{air} is the radiodensity of air at standard pressure and temperature. This quantity is computed by the scanner and the output values are typically in the range $[-1000, 3000]$. The slices are captured and stacked in one anatomical projection, usually axial (or transverse) but also in the sagittal one. Then, through Multi-Planar Reconstruction (MPR) techniques, the other two views are recreated by processing the slices. Modern software allows for a complete 3-dimensional reconstruction. The CTs used in this work were stored in the Rizzoli Institute's database. In particular, we used CT scans from 105 patients from 2009 to 2019. Among the different CTs, the 40% contains metastases. About the 43% of CTs instead contains other lesions such as, e.g., osteolytic lesions, benign lesions, and chordoma. The remaining 17% instead does not contain any lesions. Male and female patients are equally distributed, and different ranges of age have been considered. About 28% of the patients have been examined via CT-guided ago-biopsies, 21% whole-body scans while the remaining data come from lumbar or thoracic CTs.

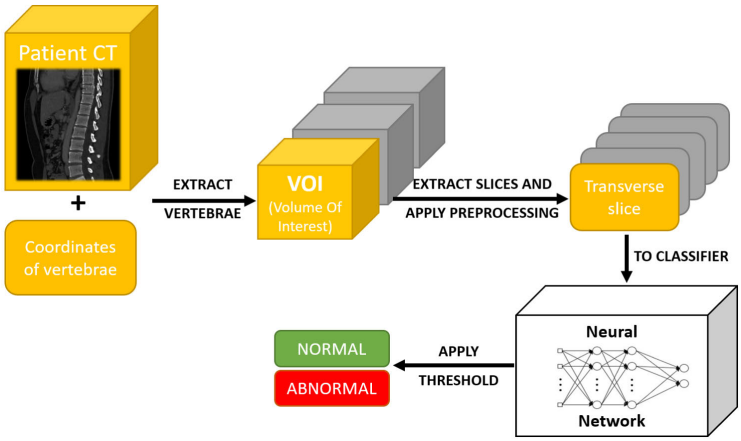


FIGURE 1. Illustration of the framework for automated detection.

Multiple vertebrae for each patient have been used. Thus, the dataset contains equally distributed L1 to L5 and T1 to T12 vertebra. In the proposed system, it is necessary to specify the center coordinates of each vertebra in the CT. For simplicity, we assume that they are already provided by using an automated state-of-the-art technique for localizing vertebrae from CTs [18], [43]. Coordinates are given with respect to the extracted CT volume, corresponding to the i-th slice for each dimension. Each tuple (X_i, Y_i, Z_i, *label*) refers to the center of the vertebra under consideration. Annotations are saved in JSON format, where X corresponds to the coronal view of the CT, Y to the sagittal view, and Z to the transverse one. The *label* entry identifies the vertebrae.

III. COMPUTER-AIDED DIAGNOSIS SYSTEM VIA CONVOLUTIONAL NEURAL NETWORKS

In this section, we describe the main components of the proposed CADe. The goal of the proposed system is to alert the medical staff if an anomaly is found in the CT scans. To this end, as detailed in Section IV, we trained a binary classification network that can distinguish if there is an anomaly in the scan. This methodology requires in general less training data with respect to, e.g., a multi-class classification network. As we show in Section V, the proposed system achieves high performance in this task.

A. ALGORITHM STRUCTURE

In this work, we develop a CADe system for automatic analysis of vertebrae from CTs to highlight the presence of abnormalities (i.e. tumors and lesions of other nature) within the vertebral body. To achieve this objective, we propose a comprehensive methodology outlined in Algorithm 1, with a graphical sketch provided in Figure 1. We now explain in detail the main steps of the algorithm.

Initialization: First, the whole volume composing the CT scan is taken into account for the desired patient. Then, the coordinates of the centers of the vertebrae are found (e.g., using the approach in [18], [43]). At this point, each Volume Of Interest (VOI) containing a single vertebra is extracted from the scan. The size of each VOI is computed using CT metadata, in particular leveraging image spacing information. This approach makes the system adaptable to various CTs and setups of the scanner (we considered whole-body, thoracic, abdominal CTs and CT-guided ago-biopsies from several machines with different resolutions and settings). The transverse slices composing the VOI are adequately processed to improve the contrast and fit the input size of the detector.

Start detection: Once the slices of the VOI are processed, the CNN classifies them individually, associating to each image x_i a prediction $p(x_i) \in [0, 1]$ corresponding to the probability of x_i being abnormal. Then, a threshold set at 0.5 is applied. Predictions above this value are labeled as positive (i.e. abnormal), while predictions below this value are labeled as normal. The considered CNN is the VGG19 neural network, with a suitable set of output layers for binary classification. In particular, it consists of 16 convolutional layers and 5 MaxPool layers, 1 Global Average Pooling layer and 1 dense layer with sigmoid activation function, and takes in input 224×224 images. As for the Convolutional layers, the total number of parameters is 20024384. We refer the reader to [40] for the shape of each layer. The Global Average Pooling returns a vector of size 512. The Dense layer has instead 513 parameters and returns a scalar value in [0, 1]. We refer the reader to Section V for more details on the performance of this network.

Output: The CADe finally generates a list of the extracted vertebrae, associated with the predictions of the slices composing the VOIs. At this point, classification results are available for further analysis.

B. GRAPHICAL USER INTERFACE

We now provide a detailed overview of the Graphical User Interface (GUI), developed to allow easy access to the automated detection system and the training utilities.

The GUI is made using the *DearPyGUI* Python library, and new CT can be imported using the `NIfTI` format. We refer

TABLE 1. Automated detection of lesions.

Algorithm 1 Automated detection of lesions

Initialization:
$size_{vertebra} \leftarrow [dim_x, dim_y, dim_z]$
$(CT, metadata) \leftarrow$ loadCT($examPath$)
$size_{VOI} \leftarrow$computeVOIsize($metadata$,$size_{vertebra}$)
$vertebraeList \leftarrow$ loadCoordinates($examPath$)
$Labels \leftarrow$ Empty array
Start detection:
for each $v \in vertebraeList$ **do**
$VOI \leftarrow$ extractVOI(v, $size_{VOI}$)
$VOI \leftarrow$ contrastEnhancement(VOI)
$SLICES \leftarrow$ extractSlices(VOI)
for each $s \in SLICES$ **do**
$s \leftarrow$ Resize(s,$height = 224$,$width = 224$)
$s \leftarrow$ convertToRGB(s)
$Labels(v, s) \leftarrow$ NeuralNetwork(s)
end for
Output:
$Labels$

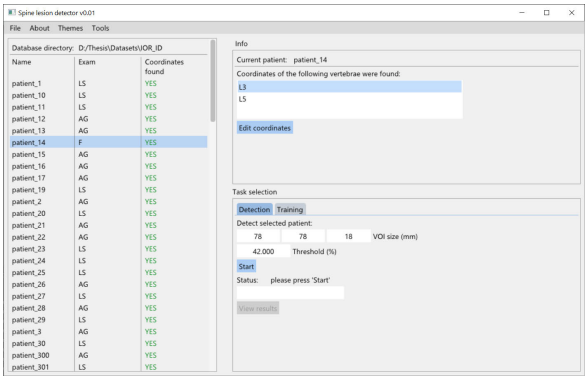


FIGURE 2. Main window of the graphical user interface.

the reader to Section IV-A for more details on this format. The GUI (Figure 2) includes several utilities. First, it allows visualization of the patient database within a given directory and creates annotation files for the vertebral coordinates. The features are divided into two main branches, i.e. detection and training process. The first one manages the lesion detection system and provides the visualization of the VGG-19 network predictions for a given patient. The second branch grants easy access to the training utilities. Indeed, the VOIs are extracted automatically, while the slices composing the volumes can be easily classified to create the ground truth. The totality of slices composes the training set. Then, a widely customizable data augmentation utility allows to improve and extend the set of slices. Training can be personalized through several parameters (e.g. the number of epochs or the optimizer's settings), including the ones for additional fine-tuning.

The main window is composed of three tabs, respectively to manage the patient database, for coordinate insertion, and finally to select the desired task, either training or detection. Patients are found automatically in the selected database, together with the type of exam that was found (''AG'' for CT-guided ago-biopsies, ''F'' for whole-body CTs, and ''LS'' for thoracic or lumbar CTs). The ''Coordinates found'' column states whether a coordinate annotation file was found for the respective patient. Once selected, the vertebrae which are already inserted are visible in a dedicated list.

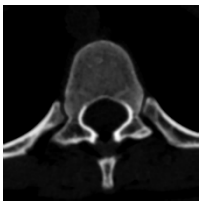
After selecting the desired patient on the left side of the main window, new vertebrae coordinates can be inserted by pressing the ''Edit coordinates'' button. A popup window will show up. Here, the coordinates of the vertebra under consideration can be adjusted by three sliders, one for each dimension of the volume. Coordinates are automatically saved in JSON format in the same directory of the patient's CT.

The system for lesion detection is managed in the ''Task selection'' tab. Here, vertebrae are extracted from the CT of the previously selected patient. The size of the VOI (i.e. the dimension of the vertebra volume) can be adjusted if needed. In addition, the slice threshold for which vertebrae are marked as normal or abnormal can be tuned. After pressing the ''Start'' button, the neural network (i.e. the VGG-19) will evaluate all the slices for each vertebra. The progress can be followed through a status bar, together with informative messages.

Once the system has finished the classification, results can be seen by pressing ''View results''. Thus, a new popup window will open. The results window is composed of three sections. On the left, a list of the evaluated vertebrae can be seen, together with their respective diagnosis. Abnormal vertebrae are given a red ''LESION'' label, while normal vertebrae an ''OK'' (green if zero slices are marked as abnormal, orange if there are abnormal slices, yet under the selected threshold). On the central tab, the selected vertebra is shown in detail (through a median view of the sagittal projection). Their disposition corresponds to their location on the transverse plane. The probability of the predictions is represented in grayscale by a bar contiguous to the rectangles. A slider allows one to select a particular slice for a deeper inspection. Indeed, the slice appears on the right tab, followed by its predicted class and a confidence score. This score is computed as follows; for a slice z that results abnormal, the confidence corresponds to the output $\sigma(z)$ of the sigmoid. Similarly, the confidence for normal slices is equal to $1-\sigma(z)$.

The training tab is composed of three sections reflecting the necessary steps for the training process. To perform a new training, the slices must be extracted from the VOIs first and then manually labeled to generate a dataset with the associated ground truth. The ''Slice labeling'' section is delegated to this step. Then, the ''Data augmentation'' tab allows to improve and extend the dataset through customizable random transformations. Finally, the training process is set and started in the ''Training setup'' section.

In the slice classification tab, a custom VOI size can be selected for extraction of the vertebrae, together with the clip parameter for contrast enhancement. The vertebrae that are still unlabeled are visible in a simple list. Then, the button ''Classify selected'' opens the slice labeling utility popup. Here, a mosaic view of the extracted slices for a given vertebra is listed. Slices can be easily classified through a list that opens by clicking on them. The user can choose to consider the slice for training by selecting either normal or abnormal.



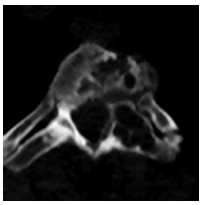


FIGURE 3. Comparison between two samples from the generated dataset. In (a) a slice from a vertebra labeled as *normal* is shown, while in (b) an *abnormal* sample from a patient affected by renal cell carcinoma is provided.

Contour slices, i.e. sections picturing inter-vertebral regions can be discarded. Also, images that were not clicked on are discarded by default.

The last section contains widgets to set up the whole training process for the CNN. The network to be trained is the VGG-19 which, as we detail in Section V, resulted as the best one for our task. The application of Transfer Learning comprises two steps. The first one serves to let the new output learn the features of the dataset, while the second one is optional and regards fine-tuning of the pre-trained convolutional layers. The GUI allows to set up both phases. In particular, for each step, the number of epochs and the optimizer's learning rate (i.e. stepsize) can be selected. The batch size and the model's name are common for both phases. Once the ''Start training'' button is pressed, the progress can be followed through text messages.

IV. TRAINING AND TESTING DATASET INITIALIZATION

In this section, we describe how we constructed a dataset to train and test the proposed CADe. The dataset is composed of slices (i.e. images of dimension $224 \times 224 \times 3$) from the transverse plane of the CT scans containing the vertebrae, either healthy or affected by diseases (Figure 3).

The pool of considered CTs was made of whole-body, abdominal, and thoracic scans and CT-guided ago-biopsies, executed on several machines and setups. Dataset generation is composed of three main phases. In the first one, the annotated VOIs are automatically extracted from the pool of CTs. Then, the slices composing the VOIs are saved as PNG images. A second phase where images are manually labeled is carried out. Slices that presented too small lesions were discarded to avoid erroneous learning. Finally, the third procedure generates a dataset by applying so-called data augmentation techniques. The dataset is finally split into training, validation, and test subsets.

Due to the nature of the extraction process, a random split related to single slices would create a bias in the network evaluation. Indeed, no slices belonging to the same vertebra should be found in both the training and test set, to preserve their uncorrelation. For this reason, the division was done vertebra-wise. In addition, the ratio between normal and abnormal vertebrae was kept as close as possible for both sets. More information is in Table 2.

TABLE 2. Dataset composition.

Set	Size (%)	N. of vertebrae	N. of images	normal/ abnormal
training	75.0	208	3316	2100/1216
test	25.0	48	837	523/314

TABLE 3. Composition of test and validation sets.

Set	Size (%)	N. of images	normal/ abnormal
training	80.0	2653	1680/973
validation	20.0	663	420/243

The training set was further split to generate a small validation set for evaluating the choice of the models and the hyperparameters. The ratio between normal and abnormal was maintained as in the previous procedure. The splitting was done slicewise (Table 3). Implementation details are provided in the following section.

A. DATASET PREPROCESSING

The CTs used in this work were stored in the Rizzoli Institute's database in `DICOM` format. `DICOM` (Digital Imaging and Communications in Medicine) is a standard for managing and storing information for medical imaging applications. Despite the presence of libraries for directly importing `DICOM` files in Python environments, it was preferred to convert only the interesting exams of each patient to a simpler format, named `NIfTI`. `NIfTI` (Neuroimaging Informatics Technology Initiative) was originally thought for neuroimaging but it also allows managing any CT scan in a simple way and preserving metadata (e.g. spatial information). In order to import the CT (already converted in `NIfTI`), a developed and sophisticated Python library named *NiBabel* is used. Once imported, the CT scan is as a 3D array representing the volume plus a header containing important information such as slice thickness with respect to each dimension, orientation as well as other data about the machine settings. It also contains an affine matrix mapping array coordinates to some world coordinate space, which however is not necessary for our system. Each CT scan has been loaded as a 3D `numpy` array, and the slice thickness (i.e. the space between the slices) along each coordinate as a list. The extracted VOIs, whose voxel values are represented in HU, are mapped to a $[0, 255]$ interval. Then, the contrast of the volumes is enhanced. Finally, for each VOI, every transverse slice is adequately pre-processed to fit the network inputs and improve its lesion detection capabilities. A volume-wise contrast enhancement technique is applied. We use Contrast Limited Adaptive Histogram Equalization (CLAHE). We found that a satisfying setting of the clip limit is 0.001 with kernel size equal to $1/8$ of the VOI. Figure 4 shows the application of CLAHE on a sample slice. Note that the modified version highlights important details such

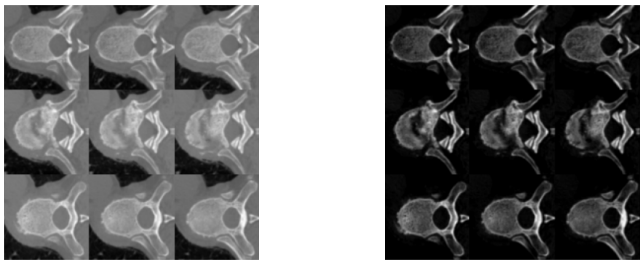


FIGURE 4. A series of slices from the same VOI is shown before and after preprocessing, respectively in (left) and (right).

as the bone structures (in particular the change in density, i.e. degradation or thickening) and drops uninteresting tissues (e.g. the soft ones).

Due to the networks requiring particular input sizes, images are resized to a 224×224 dimension. The main issue is represented by the fact that each VOI has a different size, depending on the spatial information contained in the CT header. Thus, it is necessary either to downsample or to upsample the slices. To achieve this, two different techniques were implemented. Indeed, upsampling is done with bicubic interpolation while downsampling with Lanczos kernel. Networks were trained with RGB images, hence every slice is converted from grayscale to RGB.

B. DATA AUGMENTATION

During the training, we applied data augmentation techniques. Data augmentation has been shown to improve the learning process [44], helping the network to be invariant to positional and unintended patterns. Moreover, a small dataset and its variability can be efficiently increased through this method.

Data augmentation was performed at runtime during the training, thus generating random images from dataset images at each epoch. Each image was processed with the following random transformations:

- vertical/horizontal flip,
- rotation in range $[-36, +36]$ degrees,
- width/height shift for a maximum of $\pm 10\%$ of 224 pixels,
- zoom in/out for a maximum of $\pm 10\%$ of 224 pixels.

Since some transformations (e.g., rotations) generate empty regions in the new image, it is necessary to perform padding to fill the empty regions. We chose the zero padding technique, where the pixels are filled with zeros.

V. EXPERIMENTAL RESULTS

In this section, we perform an in-depth benchmark study to assess the performance of the chosen neural network for the slice classification step of the proposed CADe. *Tensorflow* environment (version 2.9.1) was chosen for testing and structuring the CNNs. Tensorflow allows to efficiently set up training by offering a large set of utilities. In particular, instantiation of the networks was accomplished through *Keras*, a framework for designing efficient AI applications that use Tensorflow as back-end. Keras contains well-structured APIs for easy management of common layer structures. The training was carried out on two machines. The first one, a workstation with an Intel i9 octa-core processor, a total of 64 GB RAM, and 2 NVIDIA QUADRO P4000 GPUs was used for the most intensive computations. For less resource-expensive procedures a laptop with an hexa-core Intel i7 processor, 16 GB RAM, and a CUDA-enabled NVIDIA GTX 1050 was used. For this study, we considered the dataset described in Section IV. An example of such data is in Figures 3 and 4. First, common metrics will be introduced for evaluation and comparison purposes. Given a dataset, a binary classifier assigns two possible outcomes to each investigated sample, either Positive (P) or Negative (N). By comparing the results with the ground truth (i.e. the real class the samples belong to), we can define True Positives (TP) as the number of real positives that were correctly classified and False Positives (FP) as the ones that were mislabeled. Dually, True Negatives (TN) are the number of correctly labeled negatives while False Negatives (FN) are the number of negatives classified as positive. A binary classifier gives as output a value in range $[0, 1]$, namely a threshold (usually set at 0.5) is necessary to classify outputs above it as positive and vice-versa. In the following, we provide a list of common metrics.

- *Accuracy* is defined as

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}. \tag{1}$$

  It is the sum of the correctly classified samples over the dimension of the dataset. It is intuitive but not enough for evaluating a classifier, especially when classes are imbalanced (e.g. high accuracy is obtainable from a classifier that always predicts the bigger class).
- *Precision* or *Positive Predicted Value* (PPV) is

$$\text{PPV} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

  and measures how many samples predicted as positive are indeed positive.
- *Recall* or *Sensitivity* or *True Positive Rate* (TPR) is

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

  and represents how many samples out of all the positive ones were classified as positive.
- F_1 *score* is defined as the harmonic mean of precision and recall, namely

$$\text{F}_1 = 2\frac{\text{PPV} \cdot \text{TPR}}{\text{PPV} + \text{TPR}}. \tag{4}$$

  It is important because, unlike accuracy, it takes into account how data is distributed. Namely, it is very useful with imbalanced classes.
- *ROC (Receiver Operating Characteristic)* is a plot of the performance of the classification model at every

classification threshold. It shows the ratio between TPR and False Positive Rate (FPR), which is

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}, \tag{5}$$

with various threshold values. Another useful metric that can be obtained from ROC is the Area Under Curve (AUC), i.e. an aggregate measure of the classifier performance with all possible thresholds. A random classifier will present the ROC curve as a line, with $AUC = 0.5$.

- *Confusion Matrix* is a good method to represent TP, TN, FP, and FN predictions. These appear as a matrix where the horizontal axis shows the predicted classes, and the vertical axis shows the true classes.
- *Brier Score* is a metric that measures the accuracy of probabilistic predictions. For a binary classifier with output sigmoid activation function (as in the considered models), the score can be evaluated as

$$BS = \sum_{i=1}^{N_T} (f_i - y_i)^2, \tag{6}$$

where N_T is the number of the test data, f_i is the output of the network for the i-th sample and y_i is the real label of the sample.

A. NETWORK ARCHITECTURES AND TRANSFER LEARNING

To benchmark the performance of our CADe, we considered different state-of-the-art CNNs and applied Transfer Learning techniques to train them. Specifically, we trained the VGG-19 and tested it against VGG-16, Inception V3, and ResNet 50. Then, to have an experimental proof of the advantages of the technique, a neural network trained from scratch (i.e. with a random initialization of the parameters) is used as a baseline. The baseline is structured as in [45], because of the similarities between our approach and the one designed by authors for detecting brain tumors. The input shape of the network is $224 \times 224 \times 3$, thus compliant with the dataset described in Section II. The input is fed to three blocks, each of which is composed of two convolutional layers with a stride equal to 1, 3×3 kernel size, and no padding plus a pooling layer with a stride equal to 2. The convolutional layers of each block are characterized by an increasing number of filters, respectively 32, 64, and 128. The output of the third block is connected to a final multi-layer perceptron composed of two Dense layers with 550 neurons each, plus a single neuron that serves as a binary classifier. All the layers, either dense or convolutional are implemented with ReLU activaton functions. The only exception is represented by the classification layer which supplies an output between 0 and 1, using the sigmoid. Moreover, due to possible overfitting related to the small dataset, a dropout layer was inserted after each Dense layer.

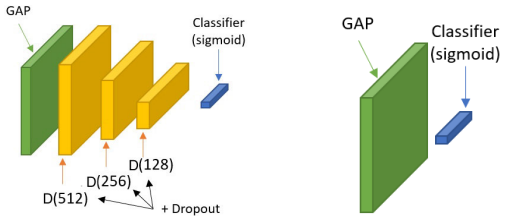


FIGURE 5. Composition of the two output classification layers. (left) is bigger than (right) which is the simplest output model. The label 'GAP' represents a Global Average Pooling layer while 'D(*n*)' a Dense layer with *n* neurons.

TABLE 4. Test results between (a) and (b) output models with VGG-19.

Output	Accuracy (%)	Precision (%)	Recall (%)	F_1 (%)	AUC	BS
(a)	92.47	88.15	92.36	90.20	0.98	0.069
(b)	**93.43**	**89.85**	**92.99**	**91.39**	**0.98**	**0.062**

B. OUTPUT CLASSIFICATION LAYERS

Binary classification can be achieved in several ways. As shown in [12], leveraging a single dense layer with softmax activation may not be sufficient in medical tasks. Therefore, it is necessary to study different classification models. In the following, we test two of the various classification structures proposed in [12].

The test was performed on the VGG-19 network to assess the best possible classifier. The first candidate is composed of a global average pooling layer, three cascaded Dense layers, and a dropout layer placed after each Dense layer. The last element is a single neuron with a sigmoid activation function giving a scalar output in [0, 1] where 0 is the normal class and 1 is the abnormal one). The second candidate is a simpler model composed only of a global average pooling layer followed by a single neuron with a sigmoid as activation. In Figure 5 we show the structure of the two classifiers.

After completing the training for both candidates, performance was assessed through the test set. The results are shown in Table 4. We can see that both candidates are comparable, with (b) performing slightly better than (a). The highest gain regards recall with $+1.8\%$ with respect to (a), while recall and AUC resulted almost equal. For the previous considerations, (b) was the chosen candidate.

After assessing the best classification layers, we applied Transfer Learning techniques by loading weights of training on the ImageNet dataset. Then, we substitute the final layers of each CNN with the model (b) layers.

C. DATA AUGMENTATION CONSISTENCY AND K-FOLD CROSS VALIDATION

We assess the performance of the considered model with and without data augmentation. This was made to ensure consistency of the dataset after employing data augmentation techniques as for Section IV-B. The results are in Table 5.

It is worth noting how the considered data augmentation scheme increases the performance of the predictor. While

TABLE 5. Test results with and without data augmentation.

Data Augm.	Accuracy (%)	Precision (%)	Recall (%)	F_1 (%)	AUC	BS
Yes	**93.43**	89.85	**92.99**	**91.39**	**0.98**	**0.062**
No	92.35	**91.12**	88.22	89.64	0.95	0.067

TABLE 6. Test results with and without k-fold cross validation.

Folds	Accuracy (%)	Precision (%)	Recall (%)	F_1 (%)	AUC	BS
1	93.43	89.85	92.99	91.39	0.98	0.062
10 (mean)	94.43	97.03	86.98	91.59	0.98	0.048
10 (std)	3.66	3.78	8.19	5.59	0.02	0.031

there is a slight drop in the precision, the recall is raised by more than 4.5%. This is an important feature since we are interested in maximizing the detection of anomalies.

Moreover, we include a k-fold cross-validation analysis for the chosen detector. That is, we split the dataset into k folds. We iteratively re-train the detector over $k-1$ folds and test it on the remaining fold. As suggested in recent literature (see, e.g. [46]) we choose $k = 10$. To avoid statistical correlation between testing and training, each patient is assigned to only one fold. Each fold contains the same amount of patients. In this setting, the ratio between normal and abnormal in each fold may not be identical. The amount of data in each fold may slightly differ, due to the different number of images for each patient. In the considered split, the ratio normal/abnormal is in mean 1.87 with a standard deviation of 0.76. Each fold has an average of 416 elements, with a standard deviation of 50. Results are reported in Table 6. Here we compare the results of the k-fold cross-validation with the ones in Table 4. For the cross-validation results, we provide both mean and standard deviation (std) results.

D. COMPARISON OF NETWORK ARCHITECTURES

For each network, the application of Transfer Learning was performed with a procedure composed of three steps:

- Step 0: The pre-trained network, i.e. the base model with no top layers was loaded and initialized with its weights, with input shape equal to $224 \times 224 \times 3$. Then, the output classification layers were added to the top of the base model.
- Step 1: Training of the custom output was run for 100 epochs, holding the rest of the network with frozen weights. The chosen optimizer (ADAM [47]) was set with a step-size $\alpha = 0.0001$ while the other parameters were kept with default values ($\beta_1 = 0.9$, $\beta_2 = 0.999$ and tolerance $\epsilon = 1 \cdot 10^{-7}$). The batch size was chosen equal to 32 samples.
- Step 2: Fine tuning was performed for 100 additional epochs, i.e. we performed additional training iterations with some layers of the base model unfrozen, depending on the considered network. The optimizer settings were kept as in Step 1 apart from the step-size α which was lowered to $5 \cdot 10^{-5}$.

The baseline was instead tuned with the following parameters; the batch size was chosen equal to 32 samples,

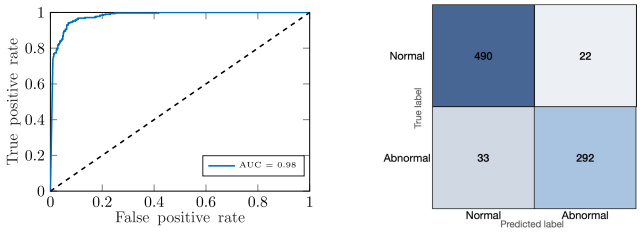


FIGURE 6. VGG-19 training. Left: ROC curve obtained from the test set results. Right: confusion matrix of the predicted labels.

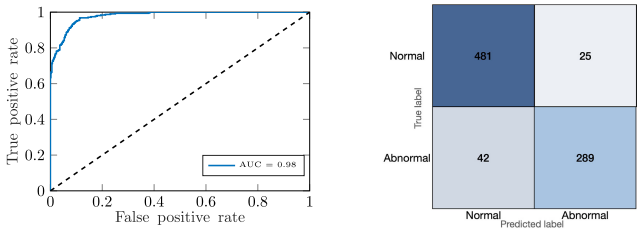


FIGURE 7. VGG-16 training. Left: ROC curve obtained from the test set results. Right: confusion matrix of the predicted labels.

while the ADAM optimizer was left with default settings, i.e. stepsize $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and tolerance $\epsilon = 1 \cdot 10^{-7}$. At the end, training was carried out for 100 epochs, while monitoring the loss and accuracy of the validation set. From these metrics, the network performance was shown to be increasing up to epoch 85, before starting to overfit the training samples. For this reason, results were computed using the weights from epoch 85 The confusion matrix and ROC curve computed from the test set for each network are shown below.

1) VGG-19

VGG-19 resulted in an AUC of 0.98 in the ROC graph, reported on the left of Figure 6. On the right the Confusion Matrix can be seen.

2) VGG-16

For VGG-16, the resulting ROC curve (Fig. 7) shows an AUC of 0.98. With respect to VGG19, the Confusion Matrix denotes a loss on performance when detecting abnormal images.

3) INCEPTION V3

Figure 8 shows the ROC curve for Inception V3, with AUC of 0.967 and the Confusion Matrix resulted from the test set.

4) ResNet50

ResNet50 was the worst performing network between the pre-trained ones in terms of accuracy, but has the highest Recall value (Figure 6).

5) BASELINE

The training phase is reported in Figure 10. The baseline showed a small AUC, equal to 0.84 (Figure 11).

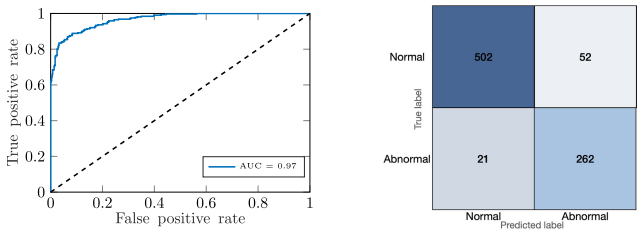


FIGURE 8. Inception V3 training. Left: ROC curve obtained from the test set results. Right: confusion matrix of the predicted labels.

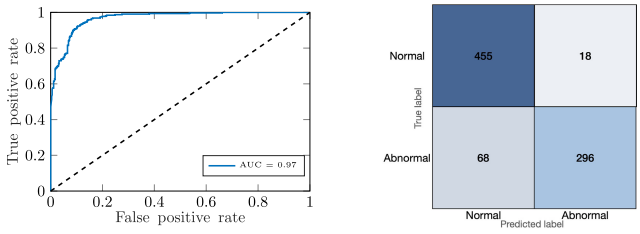


FIGURE 9. ResNet50 training. Left: ROC curve obtained from the test set results. Right: confusion matrix of the predicted labels.

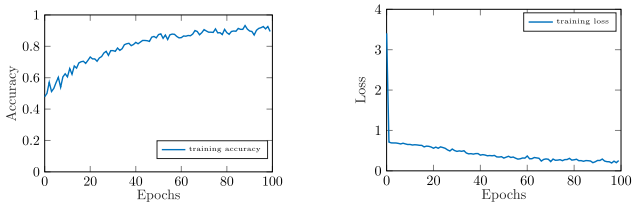


FIGURE 10. Baseline training. Left: the plot of the training accuracy is shown. Right: the graph of the training loss is provided.

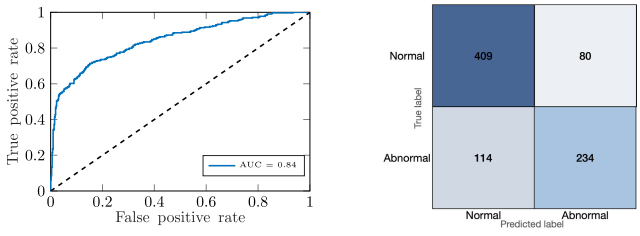


FIGURE 11. Baseline training. Left: the ROC curve obtained from the test set results. Right: the confusion matrix of the predicted labels.

To conclude, we run a calibration plot on the considered models before and after Logistic Regression. The results are in Figure 12.

E. COMPARISON WITH RELATED WORKS

To conclude, we perform a comparison with related works. In particular, we benchmark our results against the ones in [36] and [39]. The work in [36] proposes a detection scheme for lytic lesions using CNNs. The work in [39] instead proposes a CADe for automatic detection and classification of sclerotic lesions. Both works use CT scans as input data for the system. We compare the Specificity, Recall, and AUC metrics of our approach against the ones in [36] and [39].

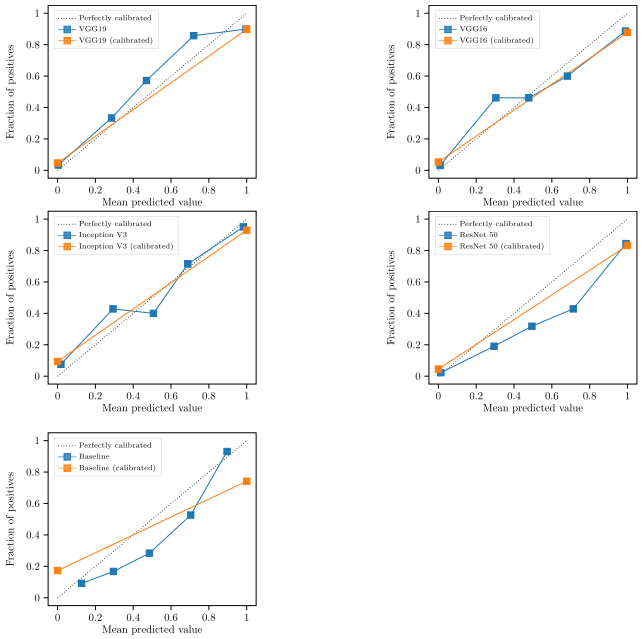


FIGURE 12. Calibration curves for the considered models before and after logistic regression.

TABLE 7. Comparison with related works in [36], [39].

Approach	Specificity (%)	Recall (%)	AUC
Our	**95.7**	**92.99**	**0.98**
[36]	64.3	90.6	n.a.
[39]	n.a.	70	0.83

As for the Specificity, it is defined as

$$\text{Specificity} = \frac{TN}{FP + TN}, \tag{7}$$

where we recall that TN are True Negative prediction while FP are False Positive prediction. The aforementioned works do not include further details on other metrics, so we are not able to report them. Moreover, the work in [36] does not include the AUC, while the one in [39] does not include the Specificity. The results are provided in Table 7, where it can be noticed that our approach performs better in all the considered metrics.

F. SUMMARY OF RESULTS

A table summarizing the benchmark results from Section V-D is in Table 8. Here, we provide the results for all the considered architectures taking into account Accuracy, Precision, Recall, F_1 score, AUC, and Brier Score (see Section V for the definition of these metrics).

The networks under test show how Transfer Learning is an excellent technique to deal with small datasets, even when the high-level features of the new domain are different from the original one. Indeed, Table 8 shows that the pre-trained networks (namely the considered VGG-19, Vgg-16, Inception V3, and ResNet50) outperform the baseline, with gains

TABLE 8. Comparison of the test results between VGG-19, VGG-16, Inception V3, ResNet50 with Transfer Learning and the network trained from scratch (the baseline).

Network	Accuracy	Precision	Recall	F_1	AUC	Brier Score
VGG-19	**93.43**	89.85	92.99	**91.39**	**0.980**	**0.062**
VGG-16	92.00	87.31	92.04	89.61	0.980	0.068
Inception V3	91.28	**92.58**	83.44	87.77	0.967	0.072
ResNet50	89.73	81.32	**94.27**	87.32	0.967	0.088
Baseline	76.82	67.24	74.52	70.69	0.845	0.160

in accuracy ranging from 13% to 16.6%. Similar results can be observed also for all the other metrics. This suggests that leveraging knowledge gained from training on large, different datasets offers a valuable starting point for tasks where labeled data is scarce.

The VGG-19 employed in Algorithm 1 outperforms all the other networks in terms of accuracy, F1-score, BS and AUC. It is worth noting that for this classification task, we are interested in detecting abnormalities. Thus, recall should be maximized. Among the different networks, VGG-19 and ResNet50 are the ones with the highest Recall. While ResNet50 has slightly higher Recall, its performance on the other metrics is inferior with respect to VGG-19. This confirms that the considered network is the most suited for the classification task. VGG-16 demonstrates performance comparable to the ones of VGG-19, although slightly inferior. Both VGG-19 and VGG-16 have the highest AUC value, which is almost 1. This value indicates that the networks can properly distinguish between normal and abnormal images. This aspect is further highlighted by the low Brier score, in which these networks are comparable. InceptionV3 has the highest precision values. Thus, it is the network that minimizes the number of false positives. This is a crucial point in medical imaging tasks where misdiagnoses can have serious consequences. However, its lower performance across other metrics makes InceptionV3 not suitable for this task. As an example, its Recall is 9.5% less than the considered VGG-19. To conclude, the decision to select VGG-19 for the lesion detection system is well-founded. Indeed, this network maximizes the Recall while maintaining high Accuracy and minimizing false positives. This choice reflects a careful consideration of the specific requirements and objectives of the medical classification task at hand. Moreover, given the high AUC and the almost-zero BS, VGG-19 can properly distinguish between normal and abnormal images. The k-fold cross validation in Table 6 further suggests that it achieves similar performance with new data. Moreover, as confirmed in Section V-E, the considered framework outperforms other state-of-the-art CADe systems for analogous tasks.

VI. CONCLUSION

Spinal metastases are indeed very frequent, so their detection and diagnosis represent a critical task. However, the complexity of the vertical body makes this task delicate and complex. Thus, in this paper, we developed a CADe system for automated lesion detection from CTs. We implemented an algorithmic process that pre-processes the CTs, extracts the volumes containing the vertebrae, and analyzes them to diagnose possible abnormalities. This approach exploits a VGG19 neural network to classify the slices into two classes, namely normal and abnormal. To train the CNN, we generated a dataset of slices from 256 vertebrae belonging to CT scans of 105 patients from the Rizzoli Orthopaedics Institute of Bologna, Italy. We apply Transfer Learning and Data Augmentation techniques to properly train the network. Finally, the toolbox was connected to a Graphical User Interface to ease the usability for the medical staff. We benchmark the considered VGG19 network with different state-of-the-art CNNs according to different performance metrics, i.e., accuracy, precision, recall, F1, AUC, and brier score. The chosen network reached 93.43% accuracy and F_1 score of 91.39%, outperforming the other ones also in terms of AUC and Brier score. Also, it achieved comparable performance in both precision and recall. Moreover, a comparison with related works shows that our approach has better performance. Future works include the analysis of a multi-class classification method for this task, as well as an extension of the framework to MRI and radiography images.

DECLARATION OF COMPETING INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

ETHICS

The study was approved by the local Ethical Committee (CE-AVEC 867/2021/Oss/IOR).

REFERENCES

[1] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1285–1298, May 2016.

[2] M. Ciftdemir, M. Kaya, E. Selcuk, and E. Yalniz, "Tumors of the spine," *World J. Orthopedics*, vol. 7, no. 2, p. 109, 2016.

[3] P. Celard, E. L. Iglesias, J. M. Sorribes-Fdez, R. Romero, A. S. Vieira, and L. Borrajo, "A survey on deep learning applied to medical images: From simple artificial neural networks to generative models," *Neural Comput. Appl.*, vol. 35, no. 3, pp. 2291–2323, Jan. 2023.

[4] N. Fatima, L. Liu, S. Hong, and H. Ahmed, "Prediction of breast cancer, comparative review of machine learning techniques, and their analysis," *IEEE Access*, vol. 8, pp. 150360–150376, 2020.

[5] O. Ozdemir, R. L. Russell, and A. A. Berlin, "A 3D probabilistic deep learning system for detection and diagnosis of lung cancer using low-dose CT scans," *IEEE Trans. Med. Imag.*, vol. 39, no. 5, pp. 1419–1429, May 2020.

[6] B. Yu, L. Zhou, L. Wang, W. Yang, M. Yang, P. Bourgeat, and J. Fripp, "SA-LuT-nets: Learning sample-adaptive intensity lookup tables for brain tumor segmentation," *IEEE Trans. Med. Imag.*, vol. 40, no. 5, pp. 1417–1427, May 2021.

[7] N. Noreen, S. Palaniappan, A. Qayyum, I. Ahmad, M. Imran, and M. Shoaib, "A deep learning model based on concatenation approach for the diagnosis of brain tumor," *IEEE Access*, vol. 8, pp. 55135–55144, 2020.

[8] P. Yao, S. Shen, M. Xu, P. Liu, F. Zhang, J. Xing, P. Shao, B. Kaffenberger, and R. X. Xu, "Single model deep learning on imbalanced small datasets for skin lesion classification," *IEEE Trans. Med. Imag.*, vol. 41, no. 5, pp. 1242–1254, May 2022.

[9] H. Seo, C. Huang, M. Bassenne, R. Xiao, and L. Xing, "Modified U-Net (mU-Net) with incorporation of object-dependent high level features for improved liver and liver-tumor segmentation in CT images," *IEEE Trans. Med. Imag.*, vol. 39, no. 5, pp. 1316–1325, May 2020.

[10] X. Dong, Y. Zhou, L. Wang, J. Peng, Y. Lou, and Y. Fan, "Liver cancer detection using hybridized fully convolutional neural network based on deep learning framework," *IEEE Access*, vol. 8, pp. 129889–129898, 2020.

[11] R. Kijowski, F. Liu, F. Caliva, and V. Pedoia, "Deep learning for lesion detection, progression, and prediction of musculoskeletal disease," *J. Magn. Reson. Imag.*, vol. 52, no. 6, pp. 1607–1619, Dec. 2020.

[12] S. S. Yadav and S. M. Jadhav, "Deep convolutional neural network based medical image classification for disease diagnosis," *J. Big Data*, vol. 6, no. 1, pp. 1–18, Dec. 2019.

[13] Y. Oh, S. Park, and J. C. Ye, "Deep learning COVID-19 features on CXR using limited training data sets," *IEEE Trans. Med. Imag.*, vol. 39, no. 8, pp. 2688–2700, Aug. 2020.

[14] Md. M. Islam, F. Karray, R. Alhajj, and J. Zeng, "A review on deep learning techniques for the diagnosis of novel coronavirus (COVID-19)," *IEEE Access*, vol. 9, pp. 30551–30572, 2021.

[15] S. Ahuja, B. K. Panigrahi, N. Dey, V. Rajinikanth, and T. K. Gandhi, "Deep transfer learning-based automated detection of COVID-19 from lung CT scan slices," *Int. J. Speech Technol.*, vol. 51, no. 1, pp. 571–585, Jan. 2021.

[16] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. 18th Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, vol. 9351. Munich, Germany: Springer, 2015, pp. 234–241.

[17] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder–decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[18] A. Sekuboyina, M. E. Husseini, A. Bayat, M. Löffler, H. Liebl, H. Li, G. Tetteh, J. Kukacka, C. Payer, and D. Stern, "VerSe: A vertebrae labelling and segmentation benchmark for multi-detector CT images," *Med. image Anal.*, vol. 73, Oct. 2021, Art. no. 102166.

[19] R. Kumar, M. Gupta, and A. Abraham, "A critical analysis on vertebra identification and cobb angle estimation using deep learning for scoliosis detection," *IEEE Access*, vol. 12, pp. 11170–11184, 2024.

[20] H. Chen, L. Qian, Y. Gao, J. Zhao, Y. Tang, J. Li, L. H. Le, E. Lou, and R. Zheng, "Development of automatic assessment framework for spine deformity using freehand 3-D ultrasound imaging system," *IEEE Trans. Ultrason., Ferroelectr., Freq. Control*, vol. 71, no. 3, pp. 408–422, Mar. 2024.

[21] A. A. A. Interiano, M. A. M. Palma, and K. M. R. Leiva, "Prediction of spinal abnormalities in neuroradiology images applying deep transfer learning," in *Proc. IEEE Int. Conf. Mach. Learn. Appl. Netw. Technol. (ICMLANT)*, Dec. 2023, pp. 1–7.

[22] R. R. Maaliw, J. A. B. Susa, A. S. Alon, A. C. Lagman, S. C. Ambat, M. B. Garcia, K. C. Piad, and M. C. Fernando-Raguro, "A deep learning approach for automatic scoliosis Cobb angle identification," in *Proc. IEEE World AI IoT Congr. (AIIoT)*, Jun. 2022, pp. 111–117.

[23] C. Kim, S.-M. Park, S. Lee, and D. Lee, "A deep learning harmonization of multi-vendor MRI for robust intervertebral disc segmentation," *IEEE Access*, vol. 12, pp. 19482–19499, 2024.

[24] B. M. Abuhayi, Y. A. Bezabh, and A. M. Ayalew, "Lumbar disease classification using an involutional neural based VGG nets (INVGG)," *IEEE Access*, vol. 12, pp. 27518–27529, 2024.

[25] N. Tong, S. Gou, Y. Yang, B. Liu, Y. Bai, J. Liu, and T. Ding, "Fully automatic fine-grained grading of lumbar intervertebral disc degeneration using regional feature recalibration," *IEEE J. Biomed. Health Informat.*, early access, 20, 2024, doi: 10.1109/JBHI.2024.3366780.

[26] H. Ouyang, F. Meng, J. Liu, X. Song, Y. Li, Y. Yuan, C. Wang, N. Lang, S. Tian, M. Yao, X. Liu, H. Yuan, S. Jiang, and L. Jiang, "Evaluation of deep learning-based automated detection of primary spine tumors on MRI using the Turing test," *Frontiers Oncol.*, vol. 12, Mar. 2022, Art. no. 814667.

[27] J. Wang, Z. Fang, N. Lang, H. Yuan, M.-Y. Su, and P. Baldi, "A multi-resolution approach for spinal metastasis detection using deep Siamese neural networks," *Comput. Biol. Med.*, vol. 84, pp. 137–146, May 2017.

[28] S. Zhao, B. Chen, H. Chang, X. Wu, and S. Li, "Discriminative dictionary-embedded network for comprehensive vertebrae tumor diagnosis," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Lima, Peru: Springer, 2020, pp. 691–701.

[29] G. Hille, M. Dünnwald, M. Becker, J. Steffen, S. Saalfeld, and K. Tönnies, "Segmentation of vertebral metastases in MRI using an U-Net like convolutional neural network," in *Bildverarbeitung Für die Medizin*. Lübeck, Germany: Springer, 2019, pp. 31–36.

[30] G. Hille, J. Steffen, M. Dünnwald, M. Becker, S. Saalfeld, and K. Tönnies, "Spinal metastases segmentation in MR imaging using deep convolutional neural networks," 2020, *arXiv:2001.05834*.

[31] I. D. Kawathekar, A. S. Areeckal, and V. Aparna, "A novel deep learning pipeline for vertebra labeling and segmentation of spinal computed tomography images," *IEEE Access*, vol. 12, pp. 15330–15347, 2024.

[32] J. T. P. D. Hallinan, L. Zhu, W. Zhang, T. Kuah, D. S. W. Lim, X. Z. Low, A. J. Cheng, S. E. Eide, H. Y. Ong, and F. E. M. Nor, "Deep learning model for grading metastatic epidural spinal cord compression on staging CT," *Cancers*, vol. 14, no. 13, p. 3219, Jun. 2022.

[33] Y. Li, Y. Zhang, E. Zhang, Y. Chen, Q. Wang, K. Liu, H. J. Yu, H. Yuan, N. Lang, and M.-Y. Su, "Differential diagnosis of benign and malignant vertebral fracture on CT using deep learning," *Eur. Radiol.*, vol. 31, no. 12, pp. 9612–9619, Dec. 2021.

[34] J. Chmelik, R. Jakubicek, P. Walek, J. Jan, P. Ourednicek, L. Lambert, E. Amadori, and G. Gavelli, "Deep convolutional neural network-based segmentation and classification of difficult to define metastatic spinal lesions in 3D CT data," *Med. Image Anal.*, vol. 49, pp. 76–88, Oct. 2018.

[35] Y. Song, H. Lu, H. Kim, S. Murakami, M. Ueno, T. Terasawa, and T. Aoki, "Segmentation of bone metastasis in CT images based on modified HED," in *Proc. 19th Int. Conf. Control, Autom. Syst. (ICCAS)*, Oct. 2019, pp. 812–815.

[36] C. Y. Chang, F. A. Huber, K. J. Yeh, C. Buckless, and M. Torriani, "Original research: Utilization of a convolutional neural network for automated detection of lytic spinal lesions on body CTs," *Skeletal Radiol.*, vol. 52, no. 7, pp. 1377–1384, Jul. 2023.

[37] M. Hammon, P. Dankerl, A. Tsymbal, M. Wels, M. Kelm, M. May, M. Suehling, M. Uder, and A. Cavallaro, "Automatic detection of lytic and blastic thoracolumbar spine metastases on computed tomography," *Eur. Radiol.*, vol. 23, no. 7, pp. 1862–1870, Jul. 2013.

[38] J. E. Burns, J. Yao, T. S. Wiese, H. E. Muñoz, E. C. Jones, and R. M. Summers, "Automated detection of sclerotic metastases in the thoracolumbar spine at CT," *Radiology*, vol. 268, no. 1, pp. 69–78, Jul. 2013.

[39] H. R. Roth, L. Lu, J. Liu, J. Yao, A. Seff, K. Cherry, L. Kim, and R. M. Summers, "Improving computer-aided detection using convolutional neural networks and random view aggregation," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1170–1181, May 2016.

[40] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. 3rd Int. Conf. Learn. Represent.*, 2015, pp. 1–14.

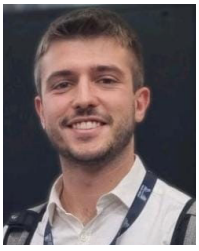
[41] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.

[42] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

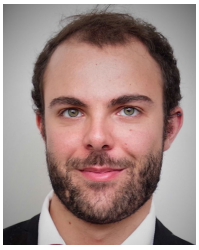
[43] R. Jakubicek, J. Chmelik, J. Jan, P. Ourednicek, L. Lambert, and G. Gavelli, "Learning–based vertebra localization and labeling in 3D CT data of possibly incomplete and pathological spines," *Comput. Methods Programs Biomed.*, vol. 183, Jan. 2020, Art. no. 105081.

[44] J. Wang and L. Perez, "The effectiveness of data augmentation in image classification using deep learning," *Convolutional Neural Netw. Vis. Recognit.*, vol. 11, pp. 1–8, Dec. 2017.
[45] S. Banerjee, S. Mitra, A. Sharma, and B. Uma Shankar, "A CADe system for gliomas in brain MRI using convolutional neural networks," 2018, *arXiv:1806.07589*.
[46] B. G. Marcot and A. M. Hanea, "What is an optimal value of k in k-fold cross-validation in discrete Bayesian network analysis?" *Comput. Statist.*, vol. 36, no. 3, pp. 2009–2031, Sep. 2021.
[47] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1–15.

ANDREA CAMISA received the Laurea degree (summa cum laude) in computer engineering from the University of Salento, Lecce, Italy, in 2017, the Licenza degree from the ISUFI Excellence School, Italy, in 2018, and the Ph.D. degree in biomedical, electrical, and systems engineering from the University of Bologna, Bologna, Italy, in 2021. He was a Visiting Student with the University of Stuttgart, in 2017 and 2018. At the time of submission of this paper, he was was a Postdoctoral Research Fellow and an Adjunct Professor with the University of Bologna. He is currently an AI Research Engineer at Helsing. His research interests include reinforcement learning, convex, distributed, and mixed-integer optimization.



GIOVANNI MONTANARI received the Laurea degree (summa cum laude) in automation engineering from the University of Bologna, Italy, in 2021. He is currently an Automation Team Leader with Salcef Group SpA.



ANDREA TESTA (Member, IEEE) received the Laurea degree (summa cum laude) in computer engineering and the Ph.D. degree in engineering of complex systems from the Università del Salento, Lecce, Italy, in 2016 and 2020, respectively.

He was a Visiting Scholar with LAAS-CNRS, Toulouse, from July 2015 to September 2015 and February 2016, and the Alma Mater Studiorum, University of Bologna, Bologna, Italy, from October 2018 to June 2019. He is currently a Research Fellow with the Alma Mater Studiorum, University of Bologna. His research interests include control of UAVs and distributed optimization.

Dr. Testa serves as an Associate Editor for the IEEE International Conference on Automation Science and Engineering (CASE).

LUIGI FALZETTI received the Laurea degree (summa cum laude) in medicine and surgery from the University of Bologna, Bologna, Italy, in 2018.

Since 2019, he has been with the Orthopedics and Traumatology Residency Program, University of Bologna. He was an Observer with the Sarcoma Unit, Royal National Orthopaedic Hospital, London, U.K. He was occupied as an International Orthopeadic Resident with the Emergency's Surgical Centre for War Victims, Kabul, Afghanistan. He is currently practicing as a Senior Resident with the Department of Spine Surgery, IRCCS Istituto Ortopedico Rizzoli, Bologna. His research interests include orthopedic oncology, trauma, and spine surgery.

SOFIA AVNET received the Ph.D. degree, in 2005, the degree in cellular and molecular biology, in 2010, and the Specialization degree in pharmacology and toxicology, in 2023.

She was a Medical Biotechnologist, in 2000. During the master's degree, she began research on bone tumors with the Kimmel Cancer Institute and the IRCCS Istituto Ortopedico Rizzoli. For the Ph.D. degree, she continued studying biological mechanisms of bone cancers with the Institute for Cancer Research, Candiolo. She also conducted research on bone pathophysiology as a Visiting Researcher with McGill University, in 2006, and a Research Fellow with the University of Turku, in 2007. From 2008 to 2019, she was a Biotechnologist with the Rizzoli Orthopedic Institute, focusing her research on tumor invasiveness, chemoresistance, microenvironment, and metabolism in bone cancers. Since 2019, she has been an Assistant Professor with the University of Bologna, where she develops 3D bone tissue and bone cancer models using spheroids, microfluidics, and bioprinting. Over her career, she has received several research grants as a Principal Investigator from the Italian Association of Cancer Research and the Ministry of Health.

Dr. Avnet has been the President and a Board Member of the International Society of Cancer Metabolism.

NICOLA BALDINI received the Laurea degree in medicine and surgery, the Certification degree in orthopaedics, and the Certification degree in oncology from the University of Bologna, Bologna, Italy, in 1981, 1987, and 1990, respectively.

From 1988 to 1989, he was a Research Fellow with Harvard University. From 2013 to 2018, he was a Visiting Professor with Kyoto Prefectural University of Medicine. He is currently a Professor with the Department of Biomedical and Neuromotor Sciences, Alma Mater Studiorum, University of Bologna, and the Director of the Department of Research, Innovation and Technology, IRCCS Istituto Ortopedico Rizzoli, Bologna. His research interests include the microenvironment of bone tissue under physiological and pathological conditions.

Prof. Baldini has served as the President for the Connective Tissue Oncology Society, from 1999 to 2000, the European Orthopaedic Research Society, from 2008 to 2012, and the International Society for Cancer Metabolism, from 2019 to 2020.

GIUSEPPE NOTARSTEFANO (Member, IEEE) received the Laurea degree (summa cum laude) in electronics engineering from the Università di Pisa, Pisa, Italy, in 2003, and the Ph.D. degree in automation and operation research from the Università di Padova, Padua, Italy, in 2007.

He was an Associate Professor, from June 2016 to June 2018, and has been an Assistant Professor, Ricercatore, since February 2007, with the Università del Salento, Lecce, Italy. He has been a Visiting Scholar with the University of Stuttgart; the University of California at Santa Barbara, Santa Barbara, CA, USA; and the University of Colorado Boulder, Boulder, CO, USA. He is currently a Professor with the Department of Electrical, Electronic, and Information Engineering G. Marconi, Alma Mater Studiorum, University of Bologna, Bologna, Italy. His research interests include distributed optimization, cooperative control in complex networks, applied nonlinear optimal control, and trajectory optimization and maneuvering of aerial and car vehicles.

Dr. Notarstefano was a recipient of an ERC Starting Grant, in 2014. He serves as an Associate Editor for IEEE TRANSACTIONS ON AUTOMATIC CONTROL, IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, and IEEE CONTROL SYSTEMS LETTERS. He has been also part of the Conference Editorial Board of the IEEE Control Systems Society and EUCA.

Open Access funding provided by 'Alma Mater Studiorum - Università di Bologna' within the CRUI CARE Agreement