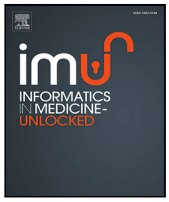
Informatics in Medicine Unlocked 38 (2023) 101238

Contents lists available at ScienceDirect

Informatics in Medicine Unlocked

journal homepage: www.elsevier.com/locate/imu



Vertebral compression fracture detection using imitation learning, patch based convolutional neural networks and majority voting

Sankaran Iyer [a,*], Alan Blair [a], Christopher White [b], Laughlin Dawes [c], Daniel Moses [c], Arcot Sowmya [a]

[a] School of Computer Science and Engineering, The University of New South Wales, Australia
[b] Department of Endocrinology and Metabolism, Prince of Wales Hospital, NSW, Australia
[c] Department of Medical Imaging, Prince of Wales Hospital, NSW, Australia

ARTICLE INFO

Keywords:
3D localisation
VCF detection
Deep reinforcement learning
Imitation learning
Convolutional neural networks
Majority voting
Transfer learning

ABSTRACT

Vertebral compression fractures often go clinically undetected and consequently untreated, resulting in severe secondary fractures due to osteoporosis, and potentially leading to permanent disability or even death. Automated detection of vertebral compression fractures (VCF) could assist in routine screening and followup of incidentally scanned patients, thereby mitigating secondary fractures later. A novel fully automated method for the detection of VCF in 3D computed tomography (CT) of the chest or abdomen is presented in this work. It starts with 3D localisation of thoracic and lumbar spine regions using deep reinforcement learning (DRL) and imitation learning (IL). Six different 3D bounding boxes are generated by the localisation step, achieving an average Jaccard Index (JI)/ Dice Coefficient (DC) of 74.21%/84.71%, and detection accuracy of 97.16 % using 3 different CNN architectures. The localised region is then split into 2D sagittal slices around the coronal centre. Each slice is further divided into patches, on which convolutional neural networks (CNNs) are trained to detect VCF. Four different CNN architectures, namely 3 layered, 6 layered and transfer learning (TL) using VGG16 and ResNet50, were experimented with. The best performing architecture turned out to be the 6 layered CNN. Aggregation is performed on the VCF detection in the 2D Patches extracted from individual bounding boxes, followed by majority voting to arrive at the final decision on the status of VCF for a given patient. An average three-fold cross validation accuracy of 85.95%, sensitivity of 88.10%, specificity of 84.20% and F1 score of 85.94% were achieved on chest images using 6 layered CNN on chest images from 308 patients. An average five-fold cross validation accuracy of 86.67%, sensitivity of 88.13%, specificity of 85.02% and F1 Score of 87.04% were achieved on abdomen images from 168 patients with the 6 layered CNN.

1. Introduction

Osteoporosis is a skeletal disorder resulting from reduced bone mineral density, with the affected person becoming susceptible to bone fractures, particularly vertebral compression Fractures (VCF). It is a major cause of morbidity and mortality in the elderly population in many parts of the world. Unfortunately, these fractures may go undetected clinically due to various reasons:

(a) the asymptomatic nature of the fractures
(b) the radiologist focussing on other areas suggested by clinicians
(c) the challenge of distinguishing between normal and pathologically deformed vertebral bodies.

Left untreated, they can lead to secondary fractures causing permanent disability and even death. In general women over 50 are vulnerable to such fractures, with the rate of occurrence varying across regions between 9% (Indonesia) and 26% (Scandinavia) [1]. Early detection of VCF is of paramount importance in preventing secondary fractures with severe consequences. An automatic computer aided diagnostic system can help mitigate the effects of non-detection of VCFs.

Methods based on machine learning have been used for VCF detection. Majority voting was applied among multiple classifiers on segmented images based on geometric and intensity features [2]. A committee of Support Vector Machines was used to distinguish the origin of VCF between osteoporotic and neoplastic causes on segmented images using adaptive thresholding, watershed and directed graph [3]. Other work [4] segmented the vertebrae using the watershed algorithm and then extracted the vertebral height, height relative to neighbouring vertebrae and bone density in various sectors of a height compass.

* Corresponding author.
E-mail address: sankaran.iyer@unsw.edu.au (S. Iyer).

https://doi.org/10.1016/j.imu.2023.101238
Received 20 January 2023; Received in revised form 23 March 2023; Accepted 2 April 2023
Available online 5 April 2023

These features were then used to train a Support Vector Machine to grade the compression fractures. The authors achieved a sensitivity of 95.7% and a false positive rate of 0.29 per patient. However they restricted studies to no more than 2 contiguous fracture levels to avoid scaling inaccuracies due to height loss.

Deep learning has gained popularity during the last decade ever since AlexNet [5] won the ImageNet challenge in 2012. Since then, deep learning architectures, especially CNNs, have been commonly used for medical image analysis. A detailed study has been carried out by a number of authors and CNNs have the proven ability to outperform human experts and other machine learning methods [6–8]. One of the advantages of using CNN is their ability to automatically extract features, unlike other classical machine learning approaches which rely on hand crafted features from medical images.

The strength of CNNs lies in their ability to detect patterns in different locations of an image using weight sharing of convolution operations, which drastically reduces the number of parameters that need to be learned, compared to a fully connected artificial neural network architecture like multilayer perceptron. This results in a network equivariant to input translations. The main mathematical operations performed are "convolutions" which result in merging of the information in the input image with kernels, also known as filters or feature detectors, to filter out feature maps. Starting from the input image, feature maps are generated at each layer, which in turn are convolved with the kernels to generate feature maps in subsequent layers. It is typical of convolution layers to be followed by pooling layers for aggregating the pixel values in the neighbourhood using a permutation invariant function. Typically, CNNs are several layers deep and in general deeper models seem to perform better with large datasets. The end of a stream of convolution layers are a few fully connected layers followed by classification layers, where the weights are not shared. CNNs have been utilised for a number of tasks in computer vision in the areas of detection, segmentation and classification. This work involves experiments with CNNs. Three different architectures were used in localisation and 4 different ones for VCF detection.

Due to their proven performance in outperforming other machine learning methods and the ability to extract hierarchical features and generalise across all datasets, recent works on VCF detection have made use of deep learning methods involving CNNs. Many works use Computed Tomography (CT) images as they provide more detailed information of bone tissues. Bar et al. [9] extracted the sagittal slices along the coronal centre of CT images, divided them into patches and trained a CNN with them. A recurrent neural network (RNN) was then trained using the output of the CNN on a sequence of patches to detect the VCF. They used a balanced dataset of 1673 CT studies containing nearly equal numbers of positive and negative samples, achieving an accuracy of 89.1%. Tomita et al. [10] built a model using the ResNet34 architecture with 5% of the slices around the centre of each CT image, without any localisation or segmentation. The CNN output for each slice for a specific vertebra was fed to a long short term memory (LSTM) [11] to detect VCF in a CT image. They used 1432 CT scans for training and achieved an accuracy of 89.2%. They also tried rule based approaches to aggregate the detection in the slices, however these did not perform as well as the LSTM. Husseini et al. [12] pretrained the networks with their proposed novel method "gradient loss" before classification on a public VerSe dataset consisting of 157 CT scans, achieving an F1 score of 82%. Nicolaes et al. [13] proposed a 3D model for detection of vertebral fractures using 3D CNN to classify first at voxel level achieving an AUC of 95%, which was then aggregated at the patient level, achieving an AUC of 93% using five-fold cross validation. They used a dataset consisting of 90 CT scans, with 90% belonging to abdomen level.

There are also other works focussed on non CT images. Murata et al. [14] achieved an accuracy of 86.0%, sensitivity of 84.7% and specificity of 87.3% on 300 plain thoracolumbar radiographs (PTLR) using IBM Visual recognition V3 deep convolutional neural network (DCNN) architectures. Chen et al. [15] employed TL using ResNet50 pretrained on Imagenet to classify 1458 plain frontal radiographs to achieve an accuracy of 73.59%, sensitivity of 73.81%, specificity of 73.02% and AUC of 0.72.

This work proposes a totally novel method of detecting VCFs. The regions of interest (ROI), namely the thoracic spine in chest scans and the lumbar spine in abdomen scans, were extracted using a localisation algorithm. This step is intended to result in better performance by focussing on the region of interest alone. The localisation algorithm generates six bounding boxes, from each of which 2D sagittal slices were extracted around the coronal centre. The slices were further divided into patches for training CNNs to predict the VCFs. Four different CNN architectures were experimented with to arrive at the right model. The results of the patch based CNN were then aggregated at the bounding box level. Finally majority voting is performed on the results of the six bounding boxes to decide on the VCF status of a patient. There are not many similarities between this approach and the state-of-the-art methods. The work of Bar et al. [9] involved segmentation to extract the spine region, which was then divided into patches for training a CNN, and to that extent is similar to the method proposed in this work. However, their aggregation method to combine the detection of the CNNs involved an RNN. The method used by Tomita et al. [10] involved extraction of 5% of slices around the centre of the CT scans for training a ResNet34 architecture to extract the features. For aggregation they experimented with three rule based methods, namely average, maximum and polling on the prediction of the slices. They however achieved best performance by aggregation using an LSTM. The method proposed in this work involves aggregation using polling, similar to one of their methods, but on the results of bounding boxes rather than slices. This coupled with the fact that the focus is on the ROI rather than the entire slice is likely to ensure better performance. There is no LSTM involved in the proposed method.

2. Material and methods

This paper proposes a novel method of automatic detection of vertebral compression fractures in CT chest and abdominal images. Preliminary work on a limited dataset was presented earlier [16], reporting on VCF detection in selected localised regions in chest scans of 3 mm slice thickness. In further work reported here, multiple classifiers are trained on all localised regions in a larger dataset and the outputs combined by majority voting. The new dataset consists of both chest and abdomen scans of 3 mm and 5 mm slice thickness extracted from a hospital PACS system. An improvement in accuracy of nearly 6% is achieved over the earlier model for chest scans. Additionally, the results of analysis on abdomen scans are also provided. The results show that majority voting improves accuracy by at least 4% on both chest and abdomen scans compared to simple averaging.

Most known methods focus on the whole image. The present work is based on the idea that better results can be achieved if the focus is narrowed down to the region of interest (ROI). Hence a novel 3D localisation step is added to extract the thoracic and lumbar spine regions from the whole image. A selective number of sagittal slices are then extracted around the coronal centre from the ROI and split into patches. In the training phase, the correct ROI is identified by a clinician, the sliced patches are annotated using radiology reports as well as Genant's criteria as containing either VCF or non VCF, and these annotated patches are collected into a database for training a deep convolutional neural network (CNN) model. In the testing phase, six different bounding boxes are generated using the localisation step which present six different views of the ROI. The trained CNN is then used to classify patches from these six ROI's. The results of model prediction are aggregated for a specific ROI. As there are six bounding boxes, it was found that better results can be achieved, and false positives and false negatives reduced, by performing majority voting on the bounding boxes to decide on the presence of VCF for a specific patient.

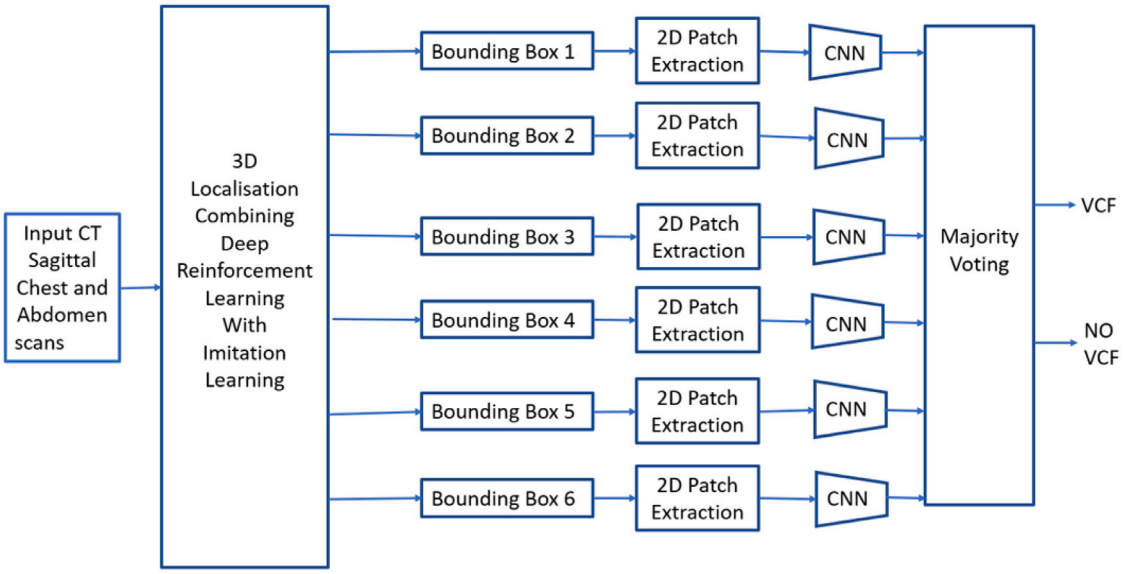


Fig. 1. Process diagram of the proposed method. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

The following is a summary of the proposed method:

(i) 3D localisation of thoracic and lumbar vertebrae that combines DRL with IL. Using a novel approach that involves three different architectures of 3D-CNNs, six different 3D bounding boxes surrounding the region of interest (ROI) are generated.
(ii) Cropping of localised ROIs within the bounding boxes, followed by extraction of a selective number of sagittal slices around the coronal centre.
(iii) Splitting of sagittal slices into patches for training CNNs to detect the presence or absence of VCFs. Four different CNN architectures namely: 3 layered, 6 layered, VGG16 and ResNet50 were trained. TL using pretrained ImageNet weights was applied for VGG16 and ResNet50.
(iv) Aggregation of the results of the individual patches in all the slices to determine the presence of VCF in a bounding box.
(v) Majority voting on the results of detection from the 6 bounding boxes to improve the accuracy of detection of VCF for a given patient.

The overall workflow is shown in Fig. 1.

2.1. Datasets

The dataset for vertebral analysis was provided by the Prince of Wales Hospital, Randwick, NSW, Australia in an anonymised form after ethics approval. The CT datasets were acquired for both chest and abdominal regions. Abdominal datasets are required for lumbar spine analysis and chest datasets for thoracic spine analysis.

Initially the focus was on localisation and therefore CT scans were collected only for training an algorithm for 3D localisation around thoracic and lumbar scan regions. Upon analysis, it was found that many of the archived images were either chest or abdomen scans. The slice thickness of the stored images was one of 3 mm, 5 mm or 7 mm, with the majority being 5 mm thick. There were very few full body scans. It was decided to build models only with 3 mm and 5 mm slice thickness images for chest and abdomen scans in this study, however the proposed method may be extended to 7 mm slice thickness with slight modifications.

A summary of chest and abdomen datasets is provided in Tables 1 and 2 respectively. The slices themselves were mostly of dimension 512×512. The number of slices per patient depended on slice thickness and was about 120 for 3 mm and 70 for 5 mm.

Table 1
Summary of datasets – chest scans.

Name	Slice thickness 3 mm	Slice thickness 5 mm	Purpose
Chest1	144		Localisation
Chest2	127		VCF Detection
Chest3		126	VCF Detection
Chest4		142	VCF Detection
Total	**271**	**268**	

Table 2
Summary of datasets – abdomen scans.

Name	Slice thickness 3 mm	Slice thickness 5 mm	Purpose
Abd1	132		Localisation
Abd2	84		VCF Detection
Abd3		86	VCF Detection
Abd4		96	VCF Detection
Total	**216**	**182**	

2.2. 3D localisation using DRL and IL

Many methods have been proposed for automatic organ localisation, which rely on multi-atlas registration or machine learning using hand-crafted features [17–20]. These methods can be computationally intensive or highly dependent on feature selection. Recent methods make use of deep learning with its ability to automatically extract features. Some methods perform landmark detection by combining the detected organs in 2D slices in the axial, coronal and sagittal planes to estimate 3D bounding boxes [21–23]. There are however limitations to this approach of processing in 2D and aggregating in 3D:

(a) annotations are required for each of three orthogonal planes
(b) CNNs need to be run for each slice in three orthogonal planes, which can be redundant as many of the slices may have identical information

(c) 3D contextual information is not available, therefore the resulting localisation may not be accurate.

Region-proposal based localisation [24] has shown a lot of promise in object detection and has been extended to multi organ detection [25]. One of the salient aspects of region proposal based methods is the generation of multiple candidate bounding boxes that may overlap with each other.

DRL is an area that has seen major successes in recent times [26], combining the representation power of CNNs with reinforcement learning. Using a Markov Decision Process (MDP), an artificial agent can be trained to achieve an intended goal. At any given time, an agent in a state s_t selects an action from action space $\mathcal{A}$ based on policy $\pi(a_t|s_t)$ which represents the agent's behaviour. The agent is taken to state s_{t+1} and receives a reward r_t. In an episodic problem, this process continues until a terminal state is reached. The expected return at the end of the episode is the discounted accumulated reward:

$$R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}, \quad \gamma \in (0, 1] \tag{1}$$

The goal is to maximise this reward. The expected future discounted rewards for a given action a in state s is known as the Q value and is given by:

$$Q_\pi(s, a) = \mathbb{E}\left[R_t \mid s_t = s, a_t = a\right] \tag{2}$$

The optimal value function at any given state s for an action a is Q^*. Q-Learning involves updating the action value as follows:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\left[r_t + \text{argmax}_{a_{t+1}}(Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t))\right] \tag{3}$$

where α is the learning rate.

The agent has two choices at each time step:

(i) explore by selecting a random action (with probability ε), or
(ii) exploit using already gained knowledge by choosing an action with the maximum Q value (with probability $1 - \varepsilon$).

This is known as an ε-*greedy* policy. After each episode, the state is reset to the initial state s_0 and the process is repeated until the Q value converges. The parameter ε continuously decays as the training progresses. At each time step, the current value of ε is compared with a randomly generated value between 0 and 1. If the latter is less then ε, the action chosen is to explore, otherwise the action chosen is to exploit. The starting value of ε is high because the model is not yet trained and exploration is to be encouraged. As training progresses, ε is gradually decayed and the probability of the exploit action increases. Towards the end of training, ε is decayed to a stage that results in more exploitation than exploration.

DRL has been used in bounding box object localisation in 2D datasets [27,28]. However, bounding box localisation in 3D has remained a challenge due to high computational resource requirements. DRL has been used for detection of anatomical landmarks in 3D CT datasets [29,30] by training an artificial agent to navigate from a random starting point towards the landmark and learning to move in the correct direction in the three coordinates, and has recently been used also for organ localisation using bounding boxes [31].

2.2.1. Method overview

The proposed algorithm in this work is motivated by a direct search to the ROI using DRL [30] and is an extension of work reported earlier (see [16]). In Fig. 2 the navigation strategy of a pre-selected bounding box towards the target landmark is shown. The bounding box can be navigated in three coordinate directions in both positive and negative directions thereby giving rise to six possible movements to a new state from the present one. To identify an optimal navigation path to the target using DRL, random exploration in three coordinate directions to a new position from each state is required. The reward function for each movement is the reduction in the resulting effective distance to the ground truth centroid [30]. DRL generally assumes no prior knowledge, instead it bootstraps from an initially random strategy and is therefore better suited for applications such as video games where the true target location might not be known during training. In the current application on the other hand, where the goal location is known, it may be more efficient and less complex for the agent to be trained in a guided manner. A simple strategy of navigating in the coordinate direction that is at maximum distance from the current location to the centre of the ground truth should suffice. Hence instead of random explorations, the direction of movement of the pre-defined bounding box is decided by a function which performs the role of an expert who guides the agent to the target, thereby converting into an imitation learning paradigm [32]. Unlike DRL, where the task of associating states to actions is learned over several iterations, IL associates states with actions chosen by the expert. This converts the task to one of supervised learning of a mapping from the states to expert actions.

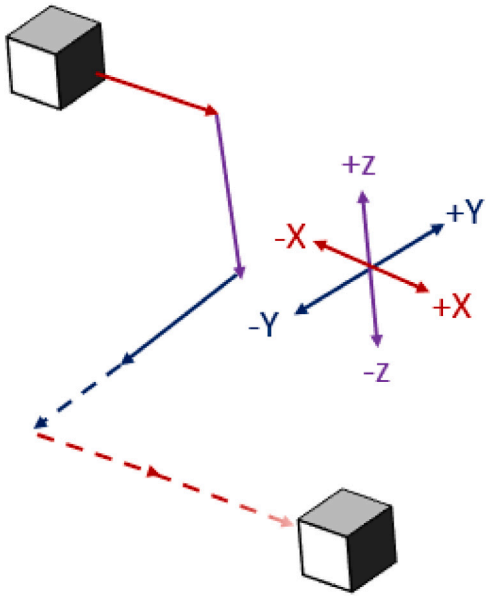


Fig. 2. Navigation strategy of navigating a pre-selected bounding box on top towards a target as shown by the box in the bottom. The bounding box is navigated in three coordinate directions shown in **red for X**, **blue for Y** and **purple for Z**.

In this work, localisation involves identification of a 3D bounding box around the lumbar/thoracic vertebrae. The proposed approach combines the deep Q learning algorithm [26] with IL when searching for an ROI from a predefined starting point in the image. The algorithm is presented in Section 2.2.3.

2.2.2. Annotation

As seen in Tables 1 and 2, there are in all 271 of 3 mm slice thickness and 268 of 5 mm slice thickness chest scans and 216 of 3 mm slice thickness and 182 of 5 mm slice thickness abdomen scans. These were manually annotated and verified by a radiologist with over 11 years of experience, to identify the two diagonally opposite corner points of a tight 3D bounding box around the thoracic spine for chest images and lumbar spine for abdomen images. The annotation process using ITK-SNAP in the three planes is illustrated in Fig. 3 for creating a tightly fitting 3D bounding box around the lumbar spine region in axial, coronal and sagittal planes.

2.2.3. Algorithm for localisation training

The localisation algorithm involves training of two networks:

(i) the first network navigates a preselected bounding box to the centre of the ROI

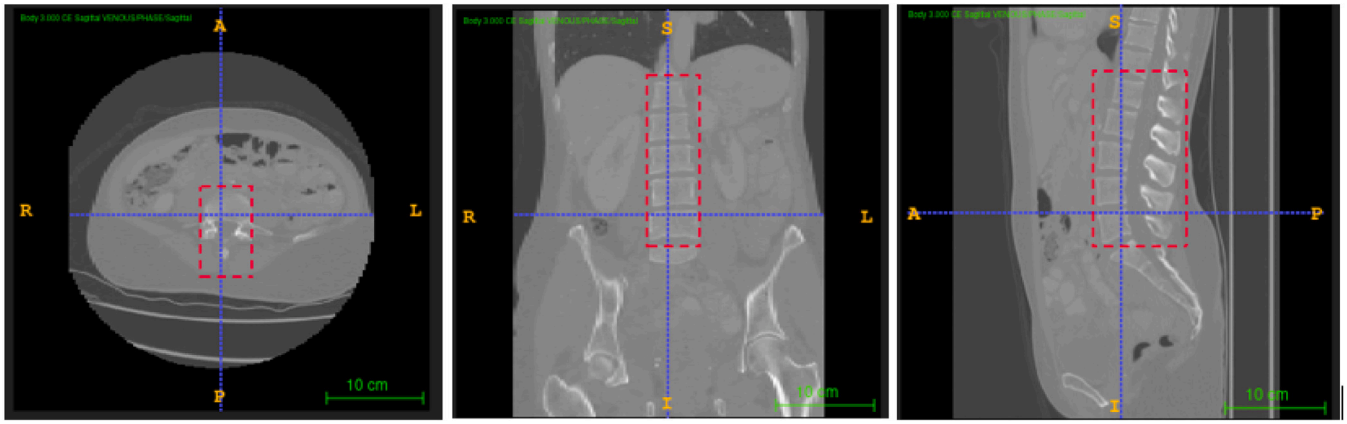


Fig. 3. Annotation using ITK snap to surround lumbar spine with tightly fitting bounding box around axial (left), coronal (middle) and sagittal (right).

(ii) the second network predicts the actual size of the bounding box surrounding the ROI.

Algorithm 1 Localisation by combining DRL with IL for ROI detection

Input: CT chest and abdominal 3D datasets.
Output: *Policy function* for navigating bounding Box.
Bounding Box function that predicts the actual bounding box coordinate sizes
initialize Policy replay memory D
initialize Bounding Box replay memory B
initialize action-value function Q with random weights
for cycles from 1 to M
  for each range of starting points
    for each randomly selected scan
      set a pre-selected bounding box at a predefined starting point = s_1
        for steps from 1 to N
          following ϵ-*greedy* policy select an action

$$a_t = \begin{cases} \textit{Imitation action with probability } \epsilon \\ \textit{argmax}_a Q(s_t, a) \textit{ otherwise} \\ \textit{correction is applied if the predicted action is away from target} \end{cases}$$

          execute action a_i to shift to image to s_{t+1}
          store transition s_t, a_t in D
          calculate the IOU of s_t with the ground truth
          if IOU >= threshold
            store s_t, *ground truth bounding box coordinate* sizes in B
          set s_t=s_{t+1}
          if bounding box centre = ground truth centre
            set a_t = "Terminate"
            store resulting transitions in D and B
          break
        end for
        train Policy network with random samples from D using mean square error loss
        train Bounding Box network with random samples from B using mean square error loss
      end for
    end for
  end for

The localisation process is illustrated in Fig. 4 and the pseudo code is in Algorithm 1. The upper network in Fig. 4 is the Policy network that is trained to predict the coordinate direction of shift (action) for an image region bounded by an initial pre-selected bounding box.

In each coordinate direction, 3 levels of movement namely 25 voxels, 10 voxels and 1 voxel of the bounding box in both positive and negative directions require 6 actions. For the three coordinate directions, therefore, 18 actions are possible.

The imitation function in Algorithm 1 returns an action, which is the coordinate direction at maximum distance from the ground truth centre. It also corrects predictions deviating from the intended course. The appropriate level of coarse, fine or very fine movement is selected based on the distance between the current centre and the ground truth centre. The starting point for the first navigation trajectory is set at 40% of the coordinate sizes to eliminate margins and extract meaningful information from the datasets. Thereafter the network is trained by shifting the initial starting point by 25 voxels in the three coordinate directions until 80% of the coordinate sizes is reached, to help the model recover from unfamiliar locations.

A final action called "Terminate" is used to indicate that the ground truth centre has been reached. Thus, the network should predict 19 possible actions in all.

The Policy network is made up of three 3D convolution layers together with batch normalisation and ReLU activation. The kernel sizes of the first, second and third Convolution layers are $7 \times 7 \times 7$, $5 \times 5 \times 5$ and $3 \times 3 \times 3$ respectively. The network takes as input the data within the bounding box shrunk by half. The convolution layers are followed first by a fully connected layer and then by a softmax layer for 19 possible actions.

To evaluate localisation, Jaccard Index (JI) also known as IOU (defined in Section 3.1.1) of the predicted bounding box with the ground truth is used. A 50% threshold level for JI is used for detection, although the generally accepted standard in computer vision for 3D object detection is lower. For example Song et al. [33] set the threshold to 25% and Xu et al. [25] set it to 33%. It is to be noted that these are the threshold levels only and as can be seen in Section 3.4, the average JI achieved was much higher (74.21%). The Dice Coefficient (DC) is also reported, which is the ratio of twice the intersection over sum of the volumes of ground truth and predicted bounding boxes.

The lower network in Fig. 4 is the Bounding Box network, which is trained to predict the three coordinate sizes of the ROI. As the pre-selected bounding box is navigated, those regions whose IOUs exceed a threshold level are stored, along with the ground truth sizes for training the Bounding Box network. The latter is made up of three 3D convolution layers together with batch normalisation and ReLU activation. The kernel size of the first, second and third convolution layers are $7 \times 7 \times 7$, $5 \times 5 \times 5$ and $3 \times 3 \times 3$ respectively. The convolution layers are followed first by a fully connected layer and then by a ReLU layer for 3 coordinate sizes.

1	3D Convolution + RELU + BN 32 Feature Maps 7x7x7 kernel + Max Pooling
2	3D Convolution + RELU + BN 64 Feature Maps 5x5x5 kernel + Max Pooling
3	3D Convolution + RELU + BN 128 Feature Maps 3x3x3 kernel
4	Fully connected Softmax for 19 actions
5	Fully connected RELU for 3 bounding box coordinate sizes and predicted IOU

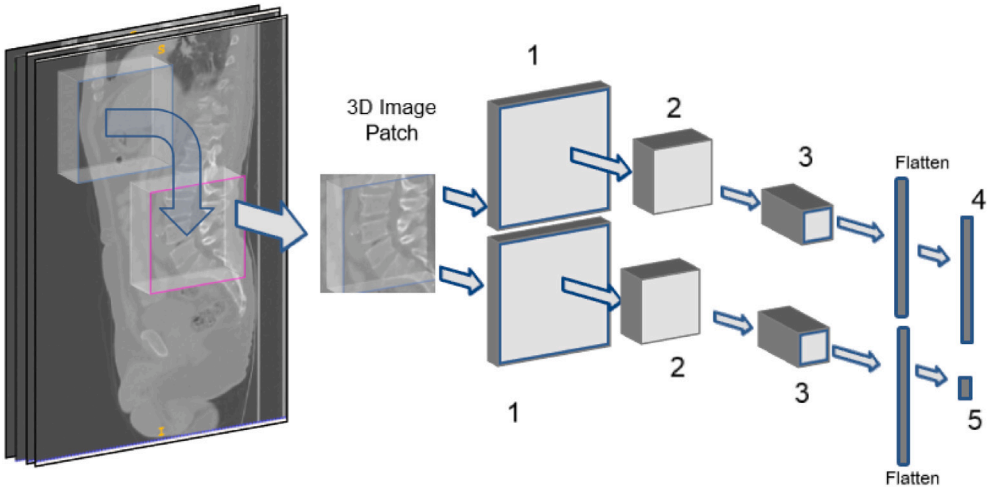


Fig. 4. Navigation of the bounding box to the Region of Interest (ROI) for a 3 layered CNN.

To improve overall performance of the Bounding Box network, two other CNN architectures were trained besides the one described above, and the predicted bounding boxes using all three models are provided to the next stage for analysis. The architecture of the second model consists of 6 convolution layers. The first 2 layers have kernel size $7 \times 7 \times 7$, followed by 2 convolution layers with kernel size $5 \times 5 \times 5$ and the final 2 convolution layers having kernel size $3 \times 3 \times 3$. Each convolution layer is followed by batch normalisation. Max pooling is added after the second and fourth layer. The third model has a convolution layer with $9 \times 9 \times 9$ kernel and a batch normalisation preceding the architecture in the 3 layered CNN with the intention that the larger kernel would provide better response. The architectures of the 6 layered and 4 layered CNNs respectively are shown in Figs. 5 and 6.

2.2.4. Testing

In the testing mode there is no IL involved. Each test image is simply run for 25 steps, which was found to be sufficient to reach the ROI. The search terminates when a "Terminate" action is triggered or when a loop is detected between the states. The bounding box prediction was run on all the steps and two different methods were used to predict the bounding box size:

(a) the predicted size in the terminating state, and
(b) the mean size of the predicted bounding boxes in the last 10 states.

This gives rise to 2 bounding box predictions for each CNN model, therefore 6 predictions for the three CNN models used.

The average performance of the two different methods of predicting the bounding boxes per model was similar, and it was difficult to choose one over the other, as there were advantages in individual performances. Therefore, both the methods were retained.

2.2.5. Time complexity analysis

Time complexity analysis of the Q learning algorithm has been analysed [34] and was found to have an upper bound complexity of $O(n^3)$. This is due to search for maximum reward from a starting point to a goal. This complexity is drastically reduced in the proposed method by guided search using imitation learning, which results in 'm' (constant) episodes of searching through 'n' states to $O(mn)$ which is effectively $O(n)$. The time complexity analysis of CNN processing 2D inputs has been performed [35] and is shown in Eq. (4).

$$O\left(\sum_{i=1}^{d} n_{l-1}.s_l^2.n_l.m_i^2\right) \tag{4}$$

where l is the index of the convolution layer, d is the depth (number of convolution layers) n_l is the number of filters also known as the width in the lth layer, s_l is the spatial size (length) of the filter and m_l is the spatial size of the output feature map. This does not take into account pooling and fully connected layers which take 5%–10% computational time.

The localisation algorithm used 3D inputs and the filter sizes are 3D and involves 3 CNNs. Therefore the complexity becomes

$$O\left(n.\sum_{cnn=1}^{cnn=3}\sum_{i_{cnn}=1}^{d_{cnn}} n_{l_{cnn}-1}.s_{l_{cnn}}^3.n_{l_{cnn}}.m_{i_{cnn}}^2\right) \tag{5}$$

where n is the number states involved in between the starting point and the goal, $l_{cnn}, d_{cnn}, n_{l_{cnn}}, s_{l_{cnn}}, m_{l_{cnn}}$ are respectively the index of the convolution layer, the depth (number of convolution layers), number of filters also known as the width in the l_{cnn}th layer, the spatial size (length) of the filter and the spatial size of the output feature map of the concerned CNN.

2.2.6. Localisation strategy

The localisation strategy for all the datasets is summarised in Table 3. The models trained on 3 mm slice thickness are not compatible

1,2	3D Convolution + RELU + BN 32 Feature Maps 7x7x7 kernel + Max Pooling
3,4	3D Convolution + RELU + BN 64 Feature Maps 5x5x5 kernel + Max Pooling
5,6	3D Convolution + RELU + BN 128 Feature Maps 3x3x3 kernel
7	Fully connected Softmax for 19 actions
8	Fully connected RELU for 3 bounding box coordinate sizes and predicted IOU

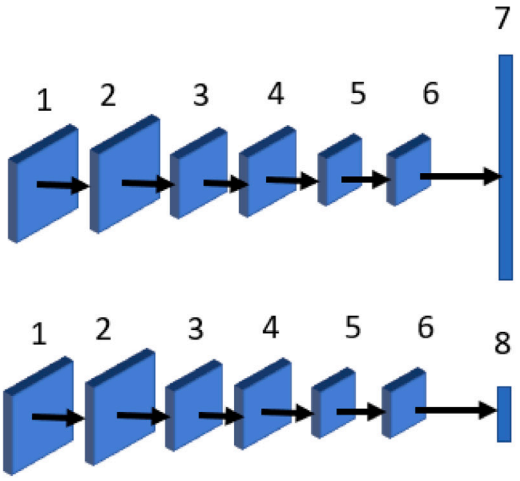


Fig. 5. 6 layered architecture for localisation.

1	3D Convolution + RELU + BN 32 Feature Maps 9x9x9 kernel + Max Pooling
2	3D Convolution + RELU + BN 32 Feature Maps 7x7x7 kernel + Max Pooling
3	3D Convolution + RELU + BN 64 Feature Maps 5x5x5 kernel + Max Pooling
4	3D Convolution + RELU + BN 128 Feature Maps 3x3x3 kernel
5	Fully connected Softmax for 19 actions
6	Fully connected RELU for 3 bounding box coordinate sizes and predicted IOU

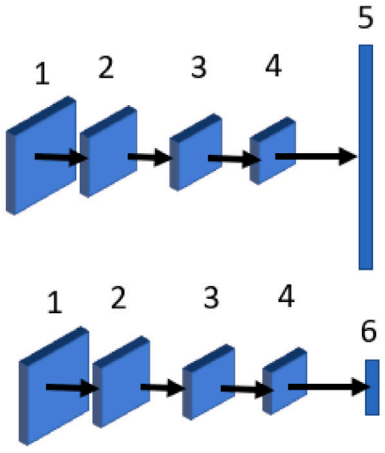


Fig. 6. 4 layered architecture for localisation.

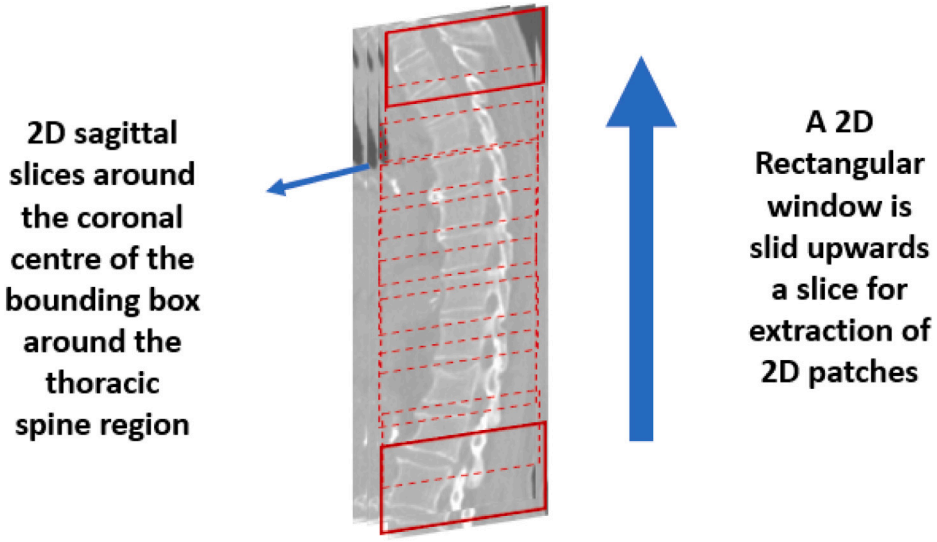


Fig. 7. Extraction of 2D Patches by sliding a red rectangular window from the selected number of slices around the coronal centre.

Table 3
Localisation strategy.

Name	Slice thickness	No. of scans	Localised by	Purpose
Chest1	3 mm	144	–	Localise Chest2
Abd1	3 mm	132	–	Localise Abd2
Chest2	3 mm	127	Chest1	Localisation prior to VCF detection
Abd2	3 mm	84	Abd1	Localisation prior to VCF detection
Chest3	5 mm	126	five-fold cross validation[a]	Localisation prior to VCF detection
Abd3	5 mm	86	five-fold cross validation[a]	Localisation prior to VCF detection
Chest4	5 mm	142	Chest3	Localisation prior to VCF detection
Abd4	5 mm	96	Abd3	Localisation prior to VCF detection

[a]The test set for each five-fold run was localised for VCF Detection.

with 5 mm slice thickness, therefore separate models were built with Chest3 and Abd3. Five-fold cross validation was performed on Chest3 and Abd3 and the localised output of the test set for each five-fold run was used in VCF detection.

2.3. VCF detection

The patch extraction and annotation strategy has been previously reported [16], and is briefly summarised below. After localisation, zeroing of the negative values of the DICOM NumPy arrays of the patches was performed as a pre-processing step to reduce the background noise. This led to removal of background noise and better representation of the vertebrae contours for subsequent processing.

2.3.1. 2D patch extraction

For building the VCF detection model, ROIs from the ground truth used for localisation as well as the 6 bounding boxes generated from localisation were extracted and split into 2D patches, as shown in Fig. 7. By splitting into 2D patches, a single model suffices to detect VCF in slices from CT images of different slice thicknesses after localisation. Visual analysis of scans showed that slices around the coronal centre carry sufficient information on the vertebrae. Consequently, slices in the middle 30% of the coronal width were selected and split into patches, in order to assess the vertebrae condition. While the primary objective is only to detect VCFs within the CT scan of a patient, it is also possible to identify their relative positions within a group of vertebrae in the spine (e.g. T11–T12, T5–T6, L1–L2).

Table 4
Summary of VCF and non VCF cases in the dataset.

Name	Number of scans	VCF cases	Non VCF cases
Chest2	127	58	69
Chest3 and Chest4	268	183	85
Abd2	84	53	31
Abd3 and Abd4	182	129	53

2.3.2. Annotation

Radiology reports usually provide the location of the positive VCF cases, with which the patches were annotated where available. However, as pointed out in Section 1, VCFs go undetected in radiology reports due to various reasons. Genant's criteria [36] provide a method of qualitative assessment of vertebral fractures based on height loss of the anterior, posterior and middle portions of a vertebra. The criteria help grade a fracture into mild, moderate or severe, depending on the degree of height loss. Manual annotations carried out using this criterion resulted in 20%–25% more VCF cases than were originally identified by radiology reports. A summary of the VCF and non-VCF cases after annotation with Genant's criteria together with radiology reports appears in Table 4.

2.3.3. Model architectures

The annotated patches extracted from 2D slices were used to build a CNN to detect VCF in a patch. The results of detection were aggregated for a bounding box.

Four different architectures were experimented with:

(i) 3 layered CNN
(ii) 6 layered CNN

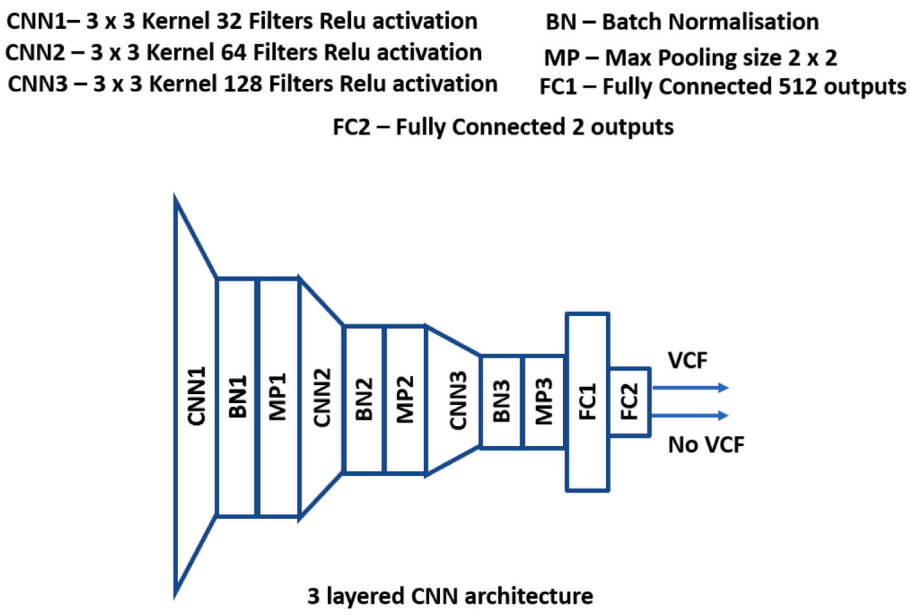


Fig. 8. 3 layered CNN architecture.

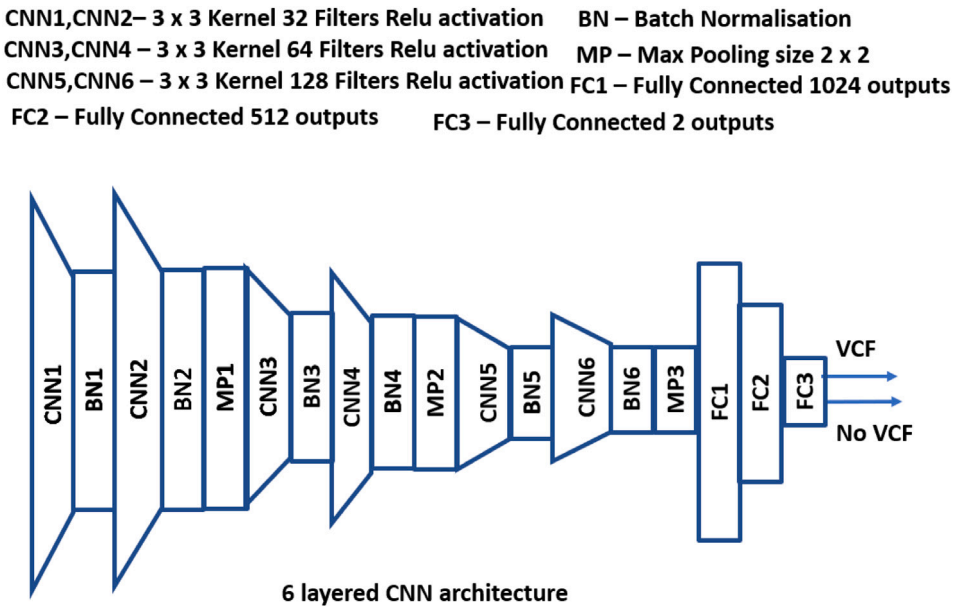


Fig. 9. 6 layered CNN architecture.

(iii) TL on pre-trained VGG16
(iv) TL on pre-trained ResNet50

The architecture of the 3 layered CNN is shown in Fig. 8, while that of the 6 layered architecture is shown in Fig. 9.

VGG16 was one of the best performing CNN architectures in ILSVRC (Imagenet) competition in 2014 [37]. It is quite popular for image recognition tasks. The architecture focusses on repetition of 3 × 3 filters with a stride of 1 followed by max pooling of 2 × 2 with a stride of 2. It has 16 layers that have weights and has approximately 138 million parameters. At the end, there are 2 fully connected layers which are followed by a softmax layer for output, and Fig. 10 shows the architecture of VGG16. The output layer is replaced by a fully connected layer of 512 followed by a final layer having 2 outputs. The TL process involved retaining the weights when pre-trained on Imagenet, for detecting the lower level features, and retraining only the last two layers FC3 and FC4.

ResNet is another popular model that won the ILSVRC (Imagenet) challenge in 2015. ResNet has a number of variants depending on the number of layers, and Fig. 11 shows the architecture of ResNet50 with 50 layers. The ResNet architecture employs skip connections to improve overall accuracy. Further details can be found elsewhere [38]. For TL, the output is replaced by a fully connected layer of 512 neurons followed by an output of 2.

2.3.4. Time complexity analysis of models

As stated in Section 2.2.5, the time complexity of the three layered and six layered CNNs are according to Eq. (4). The time complexity of VGG16 and ResNet50 has been reported by a number of researchers in terms of the FLOPs (Floating Point Operations) and the number of parameters used. VGG16 performs 1.55×10^{10} FLOPs using 134.2×10^{6} parameters, while ResNet50 performs 3.80×10^{9} FLOPs using 23.5×10^{6} parameters [39].

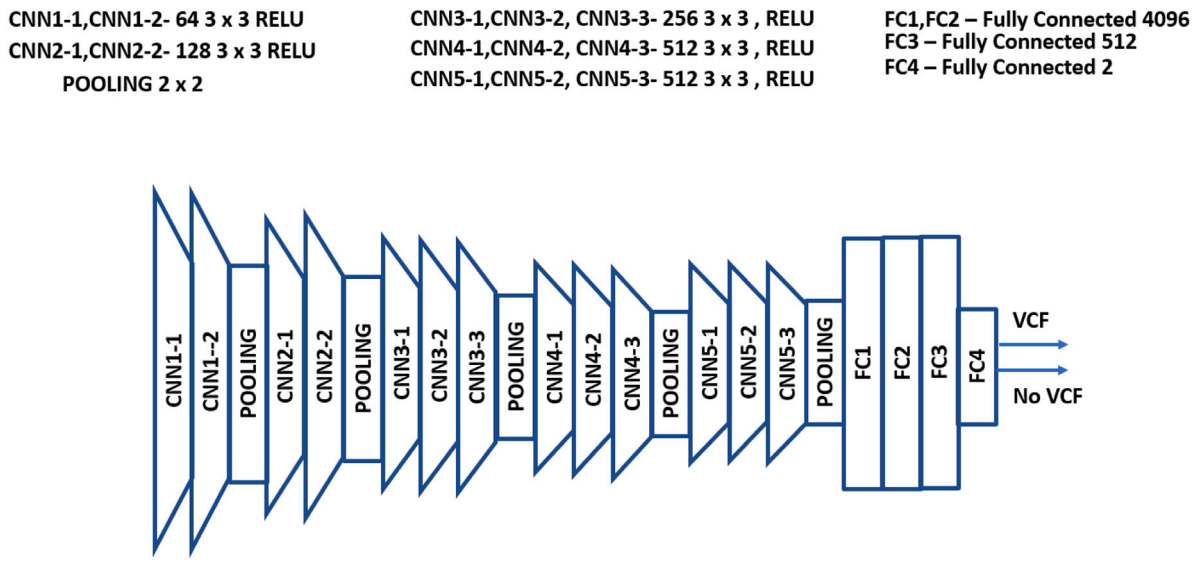


Fig. 10. Pre-trained VGG16.

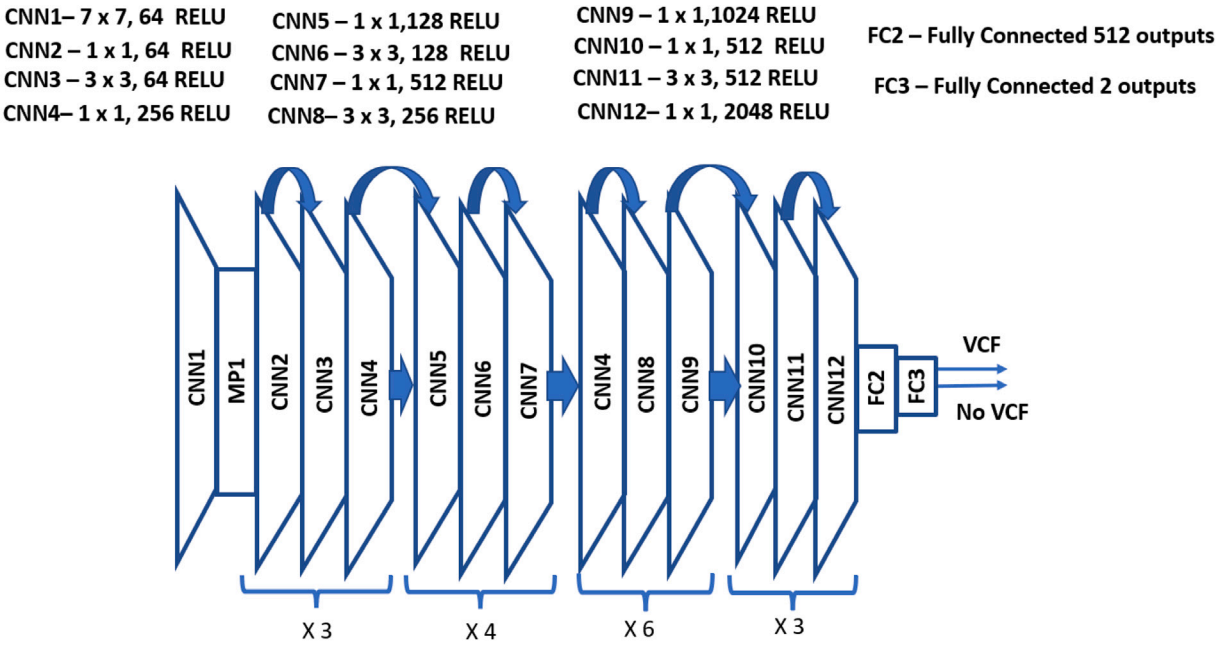


Fig. 11. Pre-trained ResNet50.

2.4. Majority voting

The localisation algorithm results in 6 bounding boxes for each scan, as discussed in Section 2.2.4. As also discussed in Section 2.2, popular algorithms such as Faster RCNN generate multiple bounding boxes. They then employ "non maximum suppression" to select the best bounding boxes. In this work instead eliminating some bounding boxes, detection is performed in each one of them. After that there are two options available:

(i) compute the average performance of all bounding boxes, or
(ii) perform voting using the prediction from each bounding box and take the consensus.

This work takes the second option.

Each bounding box is split into 2D patches as described in Section 2.3.1. It was found that performance can be improved significantly by majority voting. This involved evaluation by varying the criteria for detection from at least one to all 6 bounding boxes predicting VCF. Majority voting was performed for consensus from at least half of them. Best performance was achieved when there was agreement by either 3 or 4 bounding boxes.

3. Results

The proposed method requires the results of localisation and VCF detection in each bounding box before computing the final consensus using majority voting.

3.1. Performance metrics

The following metrics were used to evaluate performance on localisation and VCF detection.

Table 5
Balanced dataset.

Name	Available scans	VCF cases	Selected VCF cases	Non VCF cases
Chest2, Chest3 and Chest4	395	241	154	154
Abd2, Abd3 and Abd4	266	182	84	84

3.1.1. Localisation metrics

Two metrics were used:

(i) Jaccard Index, also known as intersection over union (IOU), is a measure of the overlap between the predicted and ground truth bounding boxes. If A is the predicted bounding box and B is the ground truth bounding box:

$$\text{Jaccard Index (JI)} = \frac{A \cap B}{A \cup B} \tag{6}$$

(ii) Dice coefficient of predicted bounding box A and ground truth bounding box B is given by

$$\text{Dice Coefficient (DC)} = 2 * \frac{A \cap B}{A + B} \tag{7}$$

3.1.2. VCF detection metrics

Five different metrics were used to measure detection performance, requiring definitions of True Positive (TP), False Positive (FP), True Negative (TN) and False Negative (FN).

(i) A detection outcome is true positive (TP) if it correctly identifies a positive case. False positive (FP) is an incorrectly identified positive outcome. Similarly, true negative (TN) is a correct negative detection, while false negative (FN) is an incorrect negative detection.

(ii) Accuracy is the percentage of correctly identified cases over a target set of test images:

$$\text{Accuracy} = \frac{100 * (\text{TP} + \text{TN})}{(\text{TP} + \text{FP} + \text{TN} + \text{FN})} \tag{8}$$

(iii) Sensitivity, also known as recall, is the percentage of correctly identified positive cases over all positives:

$$\text{Sensitivity} = 100 * \frac{\text{TP}}{(\text{TP} + \text{FN})} \tag{9}$$

(iv) Specificity is the percentage of correctly identified negative cases over all negatives:

$$\text{Specificity} = 100 * \frac{\text{TN}}{(\text{TN} + \text{FP})} \tag{10}$$

(v) Precision is the percentage of true positives over all detections identified as positive:

$$\text{Precision} = 100 * \frac{\text{TP}}{(\text{TP} + \text{FP})} \tag{11}$$

(vi) F1 Score is the harmonic mean of precision and recall, and is defined as:

$$\begin{aligned} \text{F1 Score} &= 100 * 2 * \frac{(\text{Precision} * \text{Recall})}{(\text{Precision} + \text{Recall})} \\ &= 100 * 2 * \frac{\text{TP}}{(2*\text{TP} + \text{FP} + \text{FN})} \end{aligned} \tag{12}$$

3.2. Dataset balancing

After annotation it was found that the datasets were not balanced, as there were more VCF cases than non VCFs. To perform the experiments, a balanced dataset with an equal number of VCF and non VCF cases was created by randomly selecting VCF cases for final analysis. The 3 mm and 5 mm scans were combined, as shown in Table 5, to create a balanced dataset containing 154 scans each for VCF and Non VCF respectively for chest, and 84 scans each for VCF and non VCF for abdomen.

3.3. Evaluation method

Many authors choose to present their results on a hold-out test set. The evaluation method used in this work is K-fold cross validation. This is a technique where the data is split into K equal-sized subsets, and each subset is used as a validation set while the other K-1 subsets are used for training. The process is repeated K times, with each subset used exactly once as the validation set. The performance is then averaged over the K iterations.

K-fold cross validation has several advantages over evaluation on a single test set. It presents a more accurate estimation of model performance, as the performance is averaged over multiple iterations and different subsets of the data. This reduces the variance of the estimated performance and makes it more reliable. Secondly, K-fold cross validation allows for more efficient use of the data, as each data point is used for both training and validation. K-fold cross validation also provides a more realistic estimate of the model's performance on new data. Finally, K-fold cross validation provides a mechanism for selecting and fine-tuning the hyper parameters for model performance.

Other work [40] clearly shows that k-fold cross validation is a preferred approach over hold-out test set validation.

3.4. Localisation results

All the localisation experiments were performed on a Keras/Tensorflow platform with Titan XP GPU. The training was performed for 10 episodes for each CNN architecture. The learning rate was set to 0.00001. The loss function used was mean square error. The best model was captured during the 10 episodes. The results of localisation are shown in Table 6.

3.5. Average bounding box results for VCF detection

The individual patches extracted following the process in Section 2.3.1 were then resized. Three different patch sizes of 64×48, 128×96 and 48×32 were experimented with before deciding in favour of 64×48 due to better VCF detection performance. Data augmentation was performed to shift the width and height by ±20%, rotation by ±20° and to flip horizontally. The training was performed for 80 epochs using keras/tensorflow and the best model was chosen. The learning rate was set at 0.00001 and the loss used was categorical cross entropy. The dataset was balanced at patient level i.e. equal number of VCF and non VCF cases. However, the spinal column (thoracic or lumbar as the case may be) is split into slices and slices into patches. Therefore, at the patch level the number of VCF and non-VCF cases could be imbalanced and balancing with Keras was needed while training. The class weight parameter of Keras was used to deal with this imbalance between the number of VCF and non VCF cases at patch level. Three-fold cross validation was performed on the balanced chest datasets (row 1 of Table 5). Five-fold cross validation was used on balanced abdomen datasets (row 2 of Table 5). Cross validation was performed separately for each of the 6 bounding boxes that resulted from localisation for each CT image. The bounding boxes from both training and test folds were split into patches.

3.5.1. Model performance

In Table 7 are shown the comparative average three-fold cross validation performance on 308 thoracic scans of the four architectures discussed in Section 2.3.3. The best performing architecture is the six layered CNN and its performance is shown in **bold**. In Table 8 the average five-fold cross validation performance on 168 lumbar scans is shown. Again the best performance is achieved using the six layered CNN architecture,

Table 6
Localisation results.

Dataset/ Slice thickness	Scans	Method	JI *mean* ± *SD*	DC *mean* ± *SD*	Detection Accuracy *mean* ± *SD* (for Average)
Chest1/3 mm	144	3 sets of experiments training on 115 and testing on 29	74.39 ± 1.27	85 ± 0.69	100
Chest2/3 mm	127	Using model built with Chest1	73.28 ± 9.67	84.21 ± 6.6	96.85
Chest3/5 mm	126	Five-fold Cross validation	72.56 ± 9.73	83.73 ± 6.57	96.83
Chest4/5 mm	142	Using model built with Chest3	73.82 ± 9.64	84.58 ± 6.51	97.18
Abd1/3 mm	132	3 sets of experiments training on 105 and testing on 27	76.96 ± 4.58	85.92 ± 3.93	96.5
Abd2/3 mm	84	Using model built with Abd1	74.54 ± 11.07	84.95 ± 7.49	96.43
Abd3/5 mm	86	Five-fold cross validation	74.19 ± 10.22	84.79 ± 6.87	97.67
Abd4/5 mm	96	Using model built with Abd3	73.95 ± 11.31	84.53 ± 7.65	95.83
Average			**74.21 ± 1.28**	**84.71 ± 0.64**	**97.16 ± 1.27**

Table 7
Average three-fold cross validation results using 308 chest scans for 4 architectures.

Model	Accuracy % *mean* ± *SD*	Sensitivity % *mean* ± *SD*	Specificity % *mean* ± *SD*	Precision % *mean* ± *SD*	F1 score % *mean* ± *SD*
3 layered CNN	74.46 ± 5.18	77.97 ± 1.76	71.39 ± 7.88	72.88 ± 9.84	75.01 ± 6.00
6 layered CNN	**81.64 ± 0.01**	**85.27 ± 0.02**	**77.96 ± 0.03**	**78.92 ± 0.02**	**81.84 ± 0.01**
Pre-trained VGG16	75.05 ± 3.30	73.81 ± 4.97	75.16 ± 9.06	75.50 ± 4.12	74.38 ± 4.10
Pre-trained ResNet50	75.65 ± 1.14	71.06 ± 3.91	79.81 ± 3.08	77.66 ± 5.68	74.04 ± 4.31

3.6. VCF detection majority voting results

After the results of VCF detection in the patches were aggregated for a given bounding box, voting was performed on the individual bounding box predictions for a given CT scan. The mean ± SD results of three-fold cross validation of chest scans using the best performing 6 layered CNN is shown in Table 9. Each row shows the results of voting from the results of a different number of bounding boxes that detect VCF on a scan. As can be seen, sensitivity is high when any one model detects VCF. However, the corresponding accuracy, F1 score and specificity are low, as false positives are picked up as well.

Progressively higher consensus from multiple models, as shown in rows 2 through to 6, leads to a decrease in sensitivity while specificity increases. The accuracy and F1 Score keep increasing and reach an optimal value usually when 3 or 4 models agree. Five-fold cross validation for 168 abdomen scans using the best performing 6 layered CNN is shown in Table 10, with an identical pattern and optimal values attained also when 3 or 4 models agree.

The results of comparison between Tables 7 and 9 show that majority voting improves the accuracy/F1 score of the 6 layered CNN architecture by 4%. Similarly as can be seen from Tables 8 and 10, majority voting improves the performance of the 6 layered CNN architecture by 4%.

3.6.1. Majority voting best fold results

In Tables 11 and 12 the best results achieved during 3 fold cross validation of thoracic and 5 fold cross validation of lumbar spine respectively are shown, using majority voting on 6 layered CNN architecture.

3.6.2. Majority voting using other models

The performance of other models for VCF detection were not as good as the six layered CNN. The majority voting results of the 3 layered CNN, pre-trained VGG16 and pretrained ResNet50 are shown in Appendices A–C respectively. For thoracic spine, the best results were obtained when at least 4 bounding boxes agree with accuracy/F1 score of 78.43/77.83 for 3 layered CNN, 80.07/78.75 for pre-trained VGG16 and 80.72/78.30 for pre-trained ResNet50. For lumbar spine, the best results were obtained were accuracies/F1 scores of 79.39/79.82, 81.21/83.46 and 83.64/82.32 for 3 layered CNN, pre-trained VGG16 and pre-trained ResNet50 respectively.

3.7. VCF detection example

The localisation process is explained in Section 2.2 using three different models. As explained in Section 2.2.4, each of the models uses two different methods for predicting the localised ROI:

(a) the predicted size in the terminating state, and
(b) the mean size of the predicted bounding boxes in the last 10 states.

Thus six bounding boxes are generated and the proposed method involves predicting VCF using the contents of each of the bounding boxes. Majority voting is then performed on the six predictions. The localisation performance in general met the criteria for subsequent VCF detection, and the average JI (IOU) was 74.21%. However, there were a few cases for which some bounding boxes did not meet the criteria for successful ROI detection. An example is shown, where two bounding boxes of model 1 (Figs. 12, 13) and model 3 (Figs. 16, 17) were successful in localisation, while the two from model 2 (Figs. 14, 15) missed localisation in an unusual manner. Majority voting helps to iron out the differences in performance on individual bounding boxes, and also helps to eliminate false positives and false negatives. 2D Patches were extracted from the individual bounding boxes. The figures illustrate the leftmost patch from the bounding boxes as explained in the captions. Four of the six bounding boxes predicted VCF, resulting in a successful overall prediction of VCF. The captions of the individual figures illustrate the locations of VCF in the models where localisation was successful.

Table 8
Average five-fold cross validation results using 168 abdomen scans for 4 architectures.

Model	Accuracy % $mean \pm SD$	Sensitivity % $mean \pm SD$	Specificity % $mean \pm SD$	Precision % $mean \pm SD$	F1 score % $mean \pm SD$
3 layered CNN	71.72 ± 1.56	82.16 ± 6.93	59.95 ± 6.96	67.81 ± 4.72	74.03 ± 4.71
6 layered CNN	**82.22 ± 0.01**	**84.85 ± 0.03**	**78.98 ± 0.03**	**81.11 ± 0.02**	**82.67 ± 0.01**
Pre-trained VGG16	71.62 ± 3.08	76.69 ± 11.82	64.95 ± 11.88	70.91 ± 8.74	72.09 ± 6.55
Pre-trained ResNet50	74.55 ± 3.16	69.38 ± 7.62	79.11 ± 7.02	78.67 ± 5.53	72.92 ± 4.29

Table 9
Majority voting of average ($mean \pm SD$) three-fold cross validation results using 6 layered CNN from 308 chest scans.

Description	Accuracy % $mean \pm SD$	Sensitivity % $mean \pm SD$	Specificity % $mean \pm SD$	Precision % $mean \pm SD$	F1 score % $mean \pm SD$
Any bounding box detects VCF	76.14 ± 5.74	96 ± 0.55	56.64 ± 7.9	68.52 ± 8.13	79.79 ± 5.86
Any 2 bounding boxes detect VCF	80.39 ± 5.96	93.91 ± 2.36	67.27 ± 8.23	73.68 ± 9.24	82.38 ± 6.38
Any 3 bounding boxes detect VCF	**83.01 ± 5.66**	**92.58 ± 2.02**	**73.76 ± 7.7**	**77.38 ± 9.3**	**84.14 ± 6.32**
4 bounding boxes detect VCF	**85.95 ± 3.96**	**88.1 ± 1.89**	**84.2 ± 7.5**	**84.27 ± 9.26**	**85.94 ± 4.88**
5 bounding boxes detect VCF	83.33 ± 4.9	78.01 ± 4.77	88.54 ± 5.74	86.52 ± 9.07	82.02 ± 6.73
All bounding boxes detect VCF	79.08 ± 4.08	66.23 ± 10.33	90.84 ± 2.98	87.23 ± 6.39	75.18 ± 8.97

Table 10
Majority voting of average ($mean \pm SD$) five-fold cross validation results using the 6 layered CNN from 168 abdomen scans.

Description	Accuracy % $mean \pm SD$	Sensitivity % $mean \pm SD$	Specificity % $mean \pm SD$	Precision % $mean \pm SD$	F1 score % $mean \pm SD$
Any bounding box detects VCF	78.18 ± 5.42	98.82 ± 2.63	56.65 ± 11.23	70.14 ± 6.59	81.87 ± 4.46
Any 2 bounding boxes detect VCF	84.24 ± 3.95	97.65 ± 5.26	70.46 ± 6.97	77.02 ± 5.74	85.94 ± 3.93
Any 3 bounding boxes detect VCF	**83.64 ± 6.28**	**90.26 ± 5.27**	**75.98 ± 12.06**	**80.45 ± 8.96**	**84.82 ± 5.78**
4 bounding boxes detect VCF	**86.67 ± 6.28**	**88.13 ± 6.91**	**85.02 ± 9.47**	**86.54 ± 9.36**	**87.04 ± 6.28**
5 bounding boxes detect VCF	81.82 ± 4.79	71.1 ± 6.4	92.3 ± 7.76	91.92 ± 8.68	79.8 ± 3.81
All bounding boxes detect VCF	78.79 ± 3.71	63.15 ± 9.49	93.48 ± 6.16	92.56 ± 7.81	74.47 ± 5.74

Table 11
Majority voting of best fold results from three-fold cross validation results from 308 chest scans using 6 layered CNN.

Description	Accuracy %	Sensitivity %	Specificity %	Precision %	F1 score %
Any bounding box detects VCF	80.39	96	65.38	72.73	82.76
Any 2 bounding boxes detect VCF	83.33	92	75	77.97	84.4
Any 3 bounding boxes detect VCF	**86.27**	**92**	**80.77**	**82.14**	**86.79**
4 bounding boxes detect VCF	**88.24**	**86**	**90.38**	**89.58**	**87.76**
5 bounding boxes detect VCF	88.24	82	94.23	93.18	87.23
All bounding boxes detect VCF	82.35	70	94.23	92.11	79.55

4. Discussion

Computed Tomography images are the best suited for VCF detection as they provide better visualisation of the bone tissues. However, they require 3D data analysis and therefore multistage processing to narrow the focus to the vertebrae. Typically good localisation and/or segmentation of vertebrae is recommended, as VCF detection depends on them. The approach used in this work is to perform localisation followed by simple preprocessing for better visualisation of the vertebrae.

A novel approach to localisation is proposed that combines DRL and IL. This approach to localisation was motivated by directed search to the ROI. Only a few works [41,42] have performed 3D bounding box localisation prior to segmentation. Most others [43–46] have focussed on locating the centre of ROI and reported the mean distance between

Table 12
Majority voting of best fold results from five-fold cross validation results from 168 abdomen scans using 6 layered CNN.

Description	Accuracy %	Sensitivity %	Specificity %	Precision %	F1 score %
Any bounding box detects VCF	84.85	100	73.68	73.68	84.85
Any 2 bunding boxes detect VCF	87.88	100	78.95	77.78	87.5
Any 3 bounding boxes detect VCF	**93.94**	**92.86**	**94.74**	**92.86**	**92.86**
4 bounding boxes detect VCF	**96.97**	**92.86**	**100**	**100**	**96.3**
5 bounding boxes detect VCF	87.88	71.43	100	100	83.33
All bounding boxes detect VCF	81.82	57.14	100	100	72.73

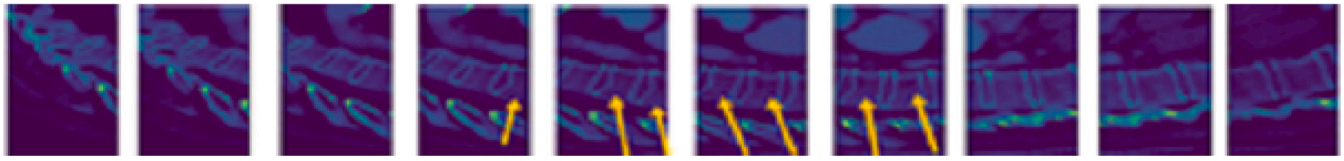


Fig. 12. Patches extracted from the leftmost slice around the coronal centre of the terminating step bounding box of model 1. The arrow marks identify the VCFs as per Genant's criteria. The patches cover the thoracic region with the rightmost patch covering T11 and T12.

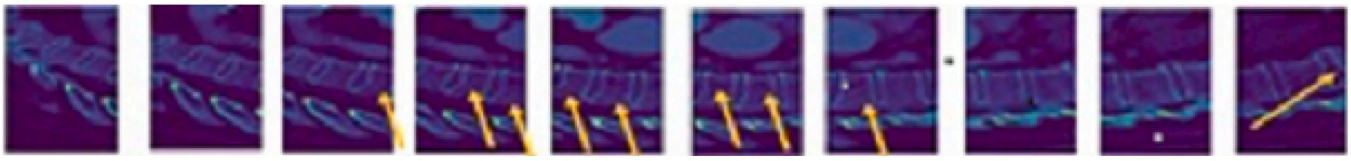


Fig. 13. Patches extracted from the leftmost slice around the coronal centre of the bounding box whose dimensions are the average of the last 10 steps of model 1. The arrow marks identify the VCFs as per Genant's criteria. The patches cover the entire thoracic region with the rightmost patch covering T11 and T12 with a part of L1 having VCF.

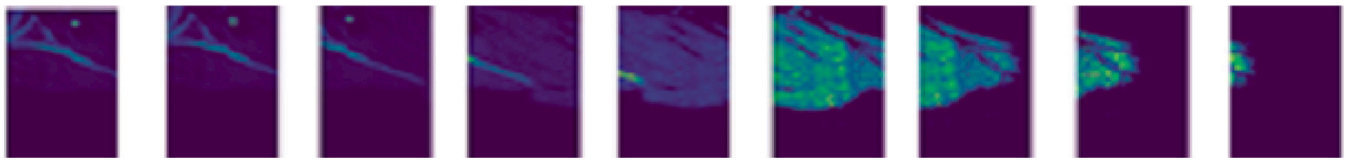


Fig. 14. Patches extracted from the leftmost slice around the coronal centre of terminating step bounding box of model 2. Model 2 has missed the mark. This is an extreme case of failure.

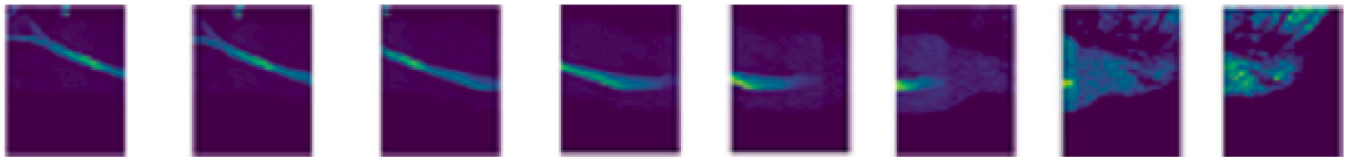


Fig. 15. Patches extracted from the leftmost slice around the coronal centre of the bounding box whose dimensions are the average of the last 10 steps of model 2. Model 2 has missed the mark. This is an extreme case of failure.

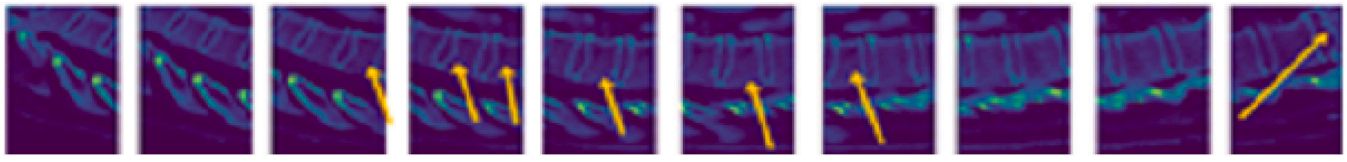


Fig. 16. Patches extracted from the leftmost slice around the coronal centre of terminating step bounding box of model 3. The arrow marks identify the VCFs as per Genant's criteria. The patches cover the entire thoracic region with the rightmost patch covering T11 and T12 with a part of L1 having VCF.

the predicted and ground truth centres. The requirement in this work was to determine the proximity of the predicted bounding box to the ground truth, in order to extract vertebrae regions. Localisation achieved an average JI of 74.21% and DC of 84.71%, which contributed to the overall higher accuracy of VCF detection. It is important to note that these figures are for 3D localisation only. Quite often

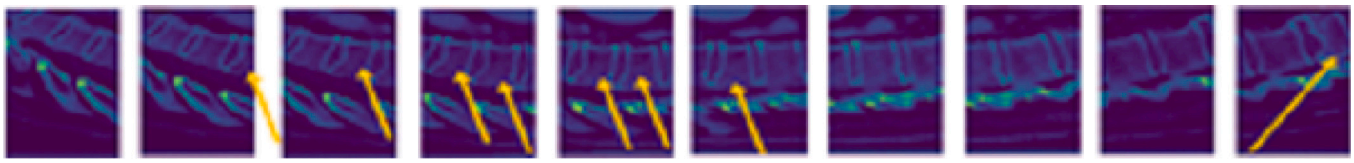


Fig. 17. Patches extracted from the leftmost slice around the coronal centre of the bounding box whose dimensions are the average of the last 10 steps of model 3. The arrow marks identify the VCFs as per Genant's criteria. The patches cover the entire thoracic region with the rightmost patch covering T11 and T12 with a part of L1 having VCF.

Table 13
Comparison with other works on VCF detection.

Work	Method	Dataset size	Result
Tomita et al.	Full CT Chest, Abdomen and Pelvic scans processed by Resnet followed by LSTM	1432 scans separate validation set 15%	Accuracy 89.2% F1 score 90.8%
Bar et al.	CT Chest or Abdomen Segmentation followed by Patch based CNN, the output of which is fed to an RNN	1673	89.10% on a of dataset separate validation set 15%
Husseini et al.	grading loss representational learning	157 scans 966 vertebrae for training, 312 for testing	F1 score 81.5%
Nicolaes et al.	3D CNN	90 scans with 90% abdomen five-fold cross validation	AUC 95% patient level AUC 93% vertebrae level
Murata et al.	DCNN	300 thoracolumbar radiographs	Accuracy 86.0%
Chen et al.	pre-trained ResNet 50	1458 frontal radiographs	Accuracy 73.59%
Proposed method	Localisation followed by patch-based 6 layered CNN, majority voting	308 chest scans	Accuracy 85.95%, F1-score 85.94% three-fold cross validation
Proposed method	Localisation followed by patch-based 6 layered CNN, majority voting	168 abdomen scans	Accuracy 86.67%, F1-score 87.04% five-fold cross validation
Proposed method	Localisation followed by patch-based pre-trained ResNet50, majority voting	308 chest scans	Accuracy 80.72%, F1-score 78.30% three-fold cross validation
Proposed method	Localisation followed by patch-based pre-trained ResNet50, majority voting	168 abdomen scans	Accuracy 83.64%, F1-score 82.32% five-fold cross validation
Proposed method	Localisation followed by patch-based pre-trained VGG16, majority voting	308 chest scans	Accuracy 80.07%, F1-score 78.75% three-fold cross validation
Proposed method	Localisation followed by patch-based pre-trained VGG16, majority voting	168 abdomen scans	Accuracy 81.21%, F1-score 83.46% five-fold cross validation
Proposed method	Localisation followed by patch-based 3 layered CNN, majority voting	308 chest scans	Accuracy 78.43%, F1-score 77.83% three-fold cross validation
Proposed method	Localisation followed by patch-based 3 layered CNN, majority voting	168 abdomen scans	Accuracy 79.39%, F1-score 79.82% five-fold cross validation

there is a tendency to compare localisation results with segmentation results, wherein the same metrics are used and the reported values are much higher. However, localisation and segmentation are quite different problems and must be evaluated separately. Localisation resulted in six bounding boxes. Generating multiple bounding boxes is similar to multiple region proposals generated by Faster RCNN. While Faster RCNN focusses on reducing redundant bounding boxes using non maximum suppression, in this work every bounding box was retained and processed independently, and a consensus was computed from individual bounding box predictions.

Patches were extracted from the individual bounding boxes after splitting them into slices. The individual bounding boxes introduce variations in the "views" to the ROI. The prediction from each patch is aggregated to the corresponding bounding box to decide on the presence of VCF for a given patient. Four different CNN architectures were experimented with, starting from a shallow 3 layered CNN, followed by a 6 layered CNN. The remaining two models employed TL using VGG16 and ResNet50, both pre-trained on Imagenet. The architecture that was best suited for the data was the 6 layered CNN. No CNN architecture is ever perfect. The sensitivity and specificity measure the false positives and negatives, which can be effectively reduced by majority voting as shown in this work.

Most of the known works on VCF detection were trained and tested on different datasets that are not available publicly, and it is not possible to make a direct comparison. However, their results may be taken as an indication of the capability of the different methods. In Table 13 a comparison with other state of the art methods is provided, together with the results of the four models presented in this work. It is to be pointed out that the best performers used hold-out test sets. In comparison, the proposed methods use three-fold cross validation for

Table A.1
Majority voting of average ($mean \pm SD$) three-fold cross validation results using 3 layered CNN from 308 chest scans.

Description	Accuracy % $mean \pm SD$	Sensitivity % $mean \pm SD$	Specificity % $mean \pm SD$	Precision % $mean \pm SD$	F1 score % $mean \pm SD$
Any bounding box detects VCF	68.63 ± 7.07	98.67 ± 1.19	39.42 ± 5.78	61.48 ± 8.58	75.5 ± 6.70
Any 2 bounding boxes detect VCF	74.18 ± 7.36	92.28 ± 2.84	57.26 ± 12.61	68.10 ± 10.59	77.92 ± 6.47
Any 3 bounding boxes detect VCF	**76.47 ± 5.46**	**83.71 ± 2.50**	**70.00 ± 8.62**	**73.17 ± 9.99**	**77.80 ± 5.81**
4 bounding boxes detect VCF	**78.43 ± 7.66**	**76.56 ± 6.79**	**80.78 ± 9.05**	**79.57 ± 11.21**	**77.83 ± 7.99**
5 bounding boxes detect VCF	76.80 ± 5.99	65.54 ± 6.45	87.97 ± 6.62	83.65 ± 11.75	73.45 ± 8.49
All bounding boxes detect VCF	72.22 ± 2.04	51.06 ± 3.70	92.95 ± 6.29	88.25 ± 12.08	64.41 ± 3.93

Table A.2
Majority voting of average ($mean \pm SD$) five-fold cross validation results using the 3 layered CNN from 168 abdomen scans.

Description	Accuracy % $mean \pm SD$	Sensitivity % $mean \pm SD$	Specificity % $mean \pm SD$	Precision % $mean \pm SD$	F1 score % $mean \pm SD$
Any bounding box detects VCF	58.18 ± 6.91	98.67 ± 2.98	16.78 ± 8.31	54.54 ± 7.77	70.01 ± 6.38
Any 2 bounding boxes detect VCF	66.67 ± 4.79	98.67 ± 2.98	33.45 ± 9.62	60.12 ± 6.63	74.52 ± 5.13
Any 3 bounding boxes detect VCF	**76.97 ± 2.71**	**97.24 ± 3.79**	**55.15 ± 7.74**	**68.79 ± 4.88**	**80.55 ± 4.51**
4 bounding boxes detect VCF	**79.39 ± 5.83**	**83.81 ± 13.48**	**72.97 ± 11.94**	**76.92 ± 3.82**	**79.82 ± 7.88**
5 bounding boxes detect VCF	78.18 ± 3.95	65.30 ± 11.25	89.23 ± 8.39	87.73 ± 3.47	74.35 ± 7.88
All bounding boxes detect VCF	70.91 ± 6.28	49.31 ± 11.90	92.15 ± 6.45	88.03 ± 7.07	62.33 ± 9.66

the chest scans and five-fold cross validation for the abdomen scans. The results of the fold which performed the best for chest and abdomen are shown in Tables 11 and 12 respectively. The results show that the best fold performance accuracy of chest scans (88.24%) is nearly comparable to the top performance accuracy of 89.2% and 89.1%. The best fold performance of abdomen scans (96.97%) outperformed the top performers. The average of three-fold and five-fold cross validation accuracies of 85.95% for chest and 86.67% for abdomen are nearly comparable even when using a relatively smaller dataset, and the top two performers required multistage processing and much larger datasets.

A fully automated method for VCF detection has been presented. As mentioned in Section 1, VCFs are often missed by radiologists for various reasons. It is therefore useful to have a method that can run retrospectively on archived images and prospectively as a background task for the analysis of new scans. One disadvantage of this method is however the amount of time needed to annotate the patches. Other approaches using weakly supervised learning such as multiple instance learning are being investigated for feasibility and performance comparison.

5. Future work

The proposed methods achieved good performance with a relatively smaller dataset. However, it would be interesting to see how the models perform with a much larger dataset. Technology and models keep evolving, and it would be useful to evaluate and explore ways to improve performance using transformers [47,48] and other state of the art architectures.

The presented work flow involved 3D localisation followed by VCF detection on 2D slices extracted from the localised bounding boxes. More detailed studies need to be performed by directly processing 3D localised images, as detection accuracy may be improved with the availability of 3D contextual information.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: This project was funded by Prince of Wales Hospital Foundation, Sydney. The three clinicians on the author list are employed at Prince of Wales Hospital, Sydney. Datasets were provided by Prince of Wales Hospital, Sydney after due ethics approvals HREC ref no. 17/350 (LNR 17/POWH/701) from South East Sydney Local Health District (SESLHD) and HC180049 from the University of New South Wales.

Acknowledgement

This project was funded by the Prince of Wales Hospital Foundation, NSW, Australia.

Appendix A. Majority voting results using 3 layered CNN

In Tables A.1 and A.2, the performance of majority voting using 3 layered CNN is shown.

Appendix B. Majority voting results using pre-trained VGG16

In Tables B.1 and B.2, the results of TL using pre-trained VGG16 are shown.

Appendix C. Majority voting results using pretrained ResNet50

In Tables C.1 and C.2 the corresponding results of TL using pre-trained ResNet50 are shown. After the 6 layered CNN, ResNet50 performed best.

Table B.1
Majority voting of average ($mean \pm SD$) three-fold cross validation results using pre-trained VGG16 from 308 chest scans.

Description	Accuracy % $mean \pm SD$	Sensitivity % $mean \pm SD$	Specificity % $mean \pm SD$	Precision % $mean \pm SD$	F1 score % $mean \pm SD$
Any bounding box detects VCF	68.63 ± 1.70	94.00 ± 0.82	42.72 ± 9.44	61.99 ± 3.72	74.68 ± 2.97
Any 2 bounding boxes detect VCF	72.88 ± 3.96	83.92 ± 3.48	60.53 ± 15.15	68.59 ± 4.09	75.40 ± 2.40
Any 3 bounding boxes detect VCF	**77.78 ± 5.91**	**78.59 ± 5.10**	**75.49 ± 14.23**	**77.39 ± 5.78**	**77.89 ± 4.39**
4 bounding boxes detect VCF	**80.07 ± 4.42**	**74.59 ± 5.55**	**84.26 ± 9.99**	**83.64 ± 4.85**	**78.75 ± 4.01**
5 bounding boxes detect VCF	79.08 ± 4.93	64.99 ± 11.03	92.06 ± 3.80	88.67 ± 6.31	74.85 ± 9.61
All bounding boxes detect VCF	71.90 ± 1.13	46.79 ± 6.31	95.94 ± 2.50	92.34 ± 3.08	61.93 ± 5.57

Table B.2
Majority voting of average ($mean \pm SD$) five-fold cross validation results using the pre-trained VGG16 from 168 abdomen scans.

Description	Accuracy % $mean \pm SD$	Sensitivity % $mean \pm SD$	Specificity % $mean \pm SD$	Precision % $mean \pm SD$	F1 score % $mean \pm SD$
Any bounding box detects VCF	60.00 ± 7.23	100.00 ± 0.00	19.68 ± 4.80	54.63 ± 8.39	71.21 ± 6.58
Any 2 bounding boxes detect VCF	71.52 ± 11.66	95.89 ± 6.30	45.81 ± 22.66	65.482 ± 13.00	77.23 ± 9.54
Any 3 bounding boxes detect VCF	**81.21 ± 11.22**	**92.26 ± 9.20**	**68.58 ± 25.71**	**77.88 ± 14.05**	**83.46 ± 8.51**
4 bounding boxes detect VCF	**80.00 ± 5.07**	**77.84 ± 6.07**	**80.34 ± 19.36**	**84.37 ± 12.11**	**79.21 ± 5.78**
5 bounding boxes detect VCF	73.33 ± 3.95	58.03 ± 21.91	86.53 ± 15.92	88.42 ± 12.91	66.48 ± 12.07
All bounding boxes detect VCF	63.64 ± 55.67	36.14 ± 20.09	88.75 ± 14.10	86.81 ± 12.94	46.87 ± 18.14

Table C.1
Majority voting of average ($mean \pm SD$) three-fold cross validation results using pre-trained ResNet50 from 308 chest scans.

Description	Accuracy % $mean \pm SD$	Sensitivity % $mean \pm SD$	Specificity % $mean \pm SD$	Precision % $mean \pm SD$	F1 score % $mean \pm SD$
Any bounding box detects VCF	68.95 ± 3.44	90.67 ± 1.79	47.39 ± 1.10	62.81 ± 6.63	74.11 ± 5.04
Any 2 bounding boxes detect VCF	79.08 ± 2.26	87.82 ± 3.65	69.89 ± 3.65	74.11 ± 4.95	80.37 ± 4.41
Any 3 bounding boxes detect VCF	**80.07 ± 0.57**	**76.47 ± 2.82**	**84.07 ± 4.12**	**82.19 ± 6.96**	**79.04 ± 2.30**
4 bounding boxes detect VCF	**80.72 ± 2.26**	**70.77 ± 3.05**	**90.15 ± 5.26**	**87.81 ± 6.76**	**78.30 ± 3.66**
5 bounding boxes detect VCF	74.84 ± 1.50	57.26 ± 7.20	91.37 ± 6.04	87.37 ± 8.60	68.92 ± 5.75
All bounding boxes detect VCF	70.26 ± 2.26	43.36 ± 10.80	96.00 ± 3.56	92.38 ± 7.19	58.32 ± 9.39

Table C.2
Majority voting of average ($mean \pm SD$) five-fold cross validation results using the pre-trained ResNet50 from 168 abdomen scans.

Description	Accuracy % $mean \pm SD$	Sensitivity % $mean \pm SD$	Specificity % $mean \pm SD$	Precision % $mean \pm SD$	F1 score % $mean \pm SD$
Any bounding box detects VCF	66.67 ± 6.78	93.68 ± 5.15	38.84 ± 12.81	60.87 ± 8.24	73.53 ± 6.32
Any 2 bounding boxes detect VCF	79.39 ± 3.32	89.97 ± 8.05	67.09 ± 14.36	75.082 ± 6.93	81.35 ± 2.15
Any 3 bounding boxes detect VCF	**83.64 ± 4.60**	**79.52 ± 10.83**	**86.13 ± 8.71**	**86.75 ± 9.54**	**82.32 ± 6.56**
4 bounding boxes detect VCF	**76.36 ± 6.57**	**61.22 ± 9.72**	**91.27 ± 8.83**	**89.32 ± 10.68**	**71.96 ± 7.07**
5 bounding boxes detect VCF	72.73 ± 6.06	51.54 ± 11.37	93.62 ± 4.31	90.32 ± 6.44	64.87 ± 8.66
All bounding boxes detect VCF	68.48 ± 11.85	40.35 ± 17.94	97.71 ± 3.13	93 ± 10.95	54.67 ± 18.79

References

[1] Ballane G, Cauley JA, Luckey MM, Fuleihan GEl-Hajj. Worldwide prevalence and incidence of osteoporotic vertebral fractures. Osteoporos Int 2017;28(5):1531–42.

[2] Ghosh S, Alomari RS, Chaudhary V, Dhillon G. Automatic lumbar vertebra segmentation from clinical CT for wedge compression fracture diagnosis. Proc SPIE 2011;7963.

[3] Y.Wang Y, Yao J, Burns JE, Summers R. Osteoporotic and neoplastic compression fracture classification on longitudinal CT. In: 2016 IEEE 13th international symposium on biomedical imaging. 2016.

[4] Burns JE, Yao J, Summers RM. Vertebral Body Compression Fractures and Bone Density: Automated Detection and Classification on CT Images Radiology, volume 284. 2017, p. 788–97.

[5] Krizhevsky A, Sutskever I, Hinton G. ImageNet classification with deep convolutional neural networks. Commun ACM 2017;60:84–90.

[6] Litjens G, Kooi T, Bejnordi BE, Setio AAA, Ciompi F, Ghafoorian M, et al. A survey on deep learning in medical image analysis. Med Image Anal 2017;42:60–88.

[7] Sarvamangala DR, Kulkarni RV. Convolutional neural networks in medical image understanding: A survey. Evol Intell 2022;15:1–22.

[8] Anwar SM, Majid M, Qayyum A, Awais M, Alnowami M, Khan MK. Medical image analysis using convolutional neural networks: A review. J Med Syst 2018;42(226).

[9] Bar A, Wolf L, Bergman AO, Toledano E, Elnekave E. Compression fractures detection on CT. Proc SPIE 2017;10134.

[10] Tomita N, Cheung YY, Hassanpour S. Deep neural networks for automatic detection of osteoporotic vertebral fractures on CT scan. Comput Biol Med 2018;98:8–15.

[11] Sepp Hochreiter, Jürgen Schmidhuber. Long short-term memory. Neural Comput 1997;1735–80. http://dx.doi.org/10.1162/neco.1997.9.8.1735.

[12] Husseini M, Sekuboyina A, Loeffler M, Navarro F, Menze B H, Kirschke JS. Grading loss: A fracture grade-based metric loss for vertebral fracture detection. In: Medical image computing and computer assisted intervention – MICCAI 2020. 2020, p. 733–42.

[13] Nicolaes J, Raeymaeckers S, Robben D, Wilms G, Vandermeulen D, Libanati C, et al. Detection of vertebral fractures in CT using 3D convolutional neural networks. Lecture Notes in Comput Sci 2020;11963:3–14.

[14] Murata K, Endo K, Aihara T, Suzuki H, Sawaji Y, Matsuoka Y, et al. Artificial intelligence for the detection of vertebral fractures on plain spinal radiography. Sci Rep 2020;10:20031.

[15] Chen H, Hsu BW, Yin Y, Lin F, Yang T, Yang R, et al. Application of deep learning algorithm to detect and visualize vertebral fractures on plain frontal radiographs. PLoS One 2021;16:e0245992.

[16] Iyer S, Sowmya A, Blair A, White C, Dawes L, Moses. A novel approach to vertebral compression fracture detection using imitation learning and patch based convolutional neural network. In: 2020 IEEE 17th international symposium on biomedical imaging. 2020.

[17] Criminisi A, Robertson D, Konukoglu E, Shotton J, Pathak S, White S, et al. Regression forests for efficient anatomy detection and localization in computed tomography scans. Med Image Anal 2013;17:1293–303.

[18] Jimenez-Del-Toro O, Muller H, Krenn M, Gruenberg K, Taha AA, Winterstein M, et al. Cloud-based evaluation of anatomical structure segmentation and landmark detection algorithms: VISCERAL anatomy benchmarks. IEEE Trans Med Imaging 2016;35:2459–75.

[19] Zhou X, Wang S, Chen H, Hara T, Yokoyama R, Kanematsu M, et al. Automatic localization of solid organs on 3D CT images by a collaborative majority voting decision based on ensemble learning. Comput Med Imaging Graph 2012;36(4):304–13.

[20] Zheng Y, Georgescu B, Comaniciu D. Marginal space learning for efficient detection of 2D/3D anatomical structures in medical images. Inf Process Med Imaging 2009;21:411–22.

[21] Hussain MA, Alborz A, Ghassan H, Rafeef A. Segmentation-free kidney localization and volume estimation using aggregated orthogonal decision CNNs. In: Medical image computing and computer assisted intervention - MICCAI 2017. 2017, p. 612–20.

[22] de Vos BD, Wolterink JM, de Jong PA, Leiner T, Viergever MA, Isgum I. ConvNet-based localization of anatomical structures in 3-D medical images. IEEE Trans Med Imaging 2017;36:1470–81.

[23] Humpire-Mamani GE, Setio AAA, Ginneken Bvan, Jacobs C. Efficient organ localization using multi-label convolutional neural networks in thorax-abdomen CT scans. Phys Med Biol 2018;63:085003.

[24] Shaoqing R, Kaiming H, Girshick R, Jian S. Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Trans Pattern Anal Mach Intell 2017;39:1137–49.

[25] Xu X, Zhou F, Liu B, Fu D, Bai X. Efficient multiple organ localization in CT image using 3D region proposal network. IEEE Trans Med Imaging 2019;38:1885–98.

[26] Li Y. Deep reinforcement learning: An overview. 2017, Available: arXiv:1701.07274.

[27] Caicedo JC, Lazebnik S. Active object localization with deep reinforcement learning. In: 2015 IEEE international conference on computer vision, 2015. Volume 2015. 2015, p. 2488–96.

[28] Kong X, Xin B, Wang Y, Hua G. Collaborative deep reinforcement learning for joint object search. In: 2017 IEEE conference on computer vision and pattern recognition, volume 2017. 2017, p. 7072–81.

[29] Ghesu FC, Georgescu B, Mansi T, Neumann D, Hornegger J, Comaniciu D. An artificial agent for anatomical landmark detection in medical images. In: Medical image computing and computer-assisted intervention, Vol. 9902, 2016, p. 229–37.

[30] Ghesu FC, B. Georgescu, Zheng Y, Grbic S, Maier A, Hornegger J, et al. Multi-scale deep reinforcement learning for real-time 3D-landmark detection in CT scans. IEEE Trans Pattern Anal Mach Intell 2019;41(1):176–89.

[31] Navarro F, Sekuboyina A, Waldmannstetter D, Peeken JC, Combs SE, Menze BH. Deep reinforcement learning for organ localization in CT. Medical Imaging with Deep Learning (MIDL); 2020.

[32] Hussein A, Gaber MM, Elyan E, Jayne C. Imitation learning: A survey of learning methods. ACM Comput Surv 2017;50:1–35.

[33] Song S, Lichtenberg SP, Xiao J. SUN RGB-D: A RGB-D scene understanding benchmark suite. In: 2015 IEEE conference on computer vision and pattern recognition. 2015, p. 567–76.

[34] Koenig S, Simmons RG. Complexity analysis of real-time reinforcement learning. In: Proceedings of the eleventh national conference on artificial intelligence. 1993, p. 99–105.

[35] He K, Sun J. Convolutional neural networks at constrained time cost. 2014, http://dx.doi.org/10.48550/arXiv.1412.1710.

[36] Grigoryan M, Guermazi A, Roemer FW, Delmas PD, Genant HK. Recognizing and reporting osteoporotic vertebral fractures. Eur Spine J 2003;12:S104–12.

[37] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. 2014, http://dx.doi.org/10.48550/arXiv.1409.1556.

[38] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: 2016 IEEE conference on computer vision and pattern recognition. 2016, p. 770–8.

[39] Li W, Zhu X, Gong S. Harmonious attention network for person re-identification. 2018, http://dx.doi.org/10.48550/arxiv.1802.08122.

[40] Yadav S, Shukla S. Analysis of k-fold cross-validation over hold-out validation on colossal datasets for quality classification. In: 2016 IEEE 6th international conference on advanced computing. 2016, p. 78–83.

[41] Sekuboyina A, Valentinitsch A, Kirschke JS, Menze BH. A localisation-segmentation approach for multi-label annotation of lumbar vertebrae using deep nets. 2017, http://dx.doi.org/10.48550/arXiv.1703.04347.

[42] Janssens R, Zeng G, Zheng G. Fully automatic segmentation of lumbar vertebrae from CT images using cascaded 3D fully convolutional networks. In: 2018 IEEE 15th international symposium on biomedical imaging. 2018, p. 893–7.

[43] Suzani A, Seitel A, Liu Y, Fels S, Rohling RN, Abolmaesumi P. Fast automatic vertebrae detection and localization in pathological CT scans - A deep learning approach. In: Medical image computing and computer-assisted intervention. 2015, p. 678–86.

[44] Suzani A, Rasoulian A, Seitel A, Fels S, Rohling RN, Abolmaesumi P. Deep learning for automatic localization, identification, and segmentation of vertebral bodies in volumetric MR images. In: Medical imaging 2015: Image-guided procedures, robotic interventions, and modeling. 2015.

[45] Chen H, Shen C, Qin J, Ni D, Shi L, Cheng JCY, et al. Automatic localization and identification of vertebrae in spine CT via a joint learning model with deep neural networks. In: Medical image computing and computer-assisted intervention. 2015, p. 515–22.

[46] Shen W, Yang F, Mu W, Yang C, Yang X, Tian J. Automatic localization of vertebrae based on convolutional neural networks. In: Medical imaging 2015: Image processing. 2015.

[47] Shamshad F, Khan SH, Zamir SW, Khan MH, Hayat M, Khan FS, et al. Transformers in medical imaging: A survey. 2022, http://dx.doi.org/10.48550/arXiv.2201.09873.

[48] He K, Gan C, Li Z, Rekik I, Yin Z, Ji W, et al. Transformers in medical image analysis. Intell Med 2023;3:59–78.