Spiorama: Automatic Spine Detection and Labelling Using Panoramic Images

Amir Yaacobi

April 18, 2022

1 Introduction

Automatic spine vertebrae localisation and labelling are important steps in diagnosing spine patholigies, as the similarity between neighboring vertebrae makes it hard for physician to identify the specific vertebra from a single 2D image and thus imcreases the reading time. Deep learning approaches to the problem have improved significantly since the publication of the VerSe 2019 \& 2020 challenge benchmarks [1]. Additional improvements such as Spine-Transformers [2] were later published. However, all these methods run in 3D and therefore are more memory and compute intensive which makes them less ideal to run on an end-user's machine.

In this paper we introduce a novel and efficient method, coined Spinorama, to localise and label the vertebrae accurately for Computed Tomography (CT) scans, by repeating the detection stage twice: initial detection is run on original 2D saggital slices; the predicted bounding boxes are used to construct a 3D panormic image along the spine centerline; then, a second detection stage is executed on the panoramic image, using the same detector in both stages. The panoramic image simplifies the problem by separating and aligning the vertebrae, thus improving detection results. Challenging cases with missing anchor vertebae are labelled via a 2D classifier on coronal Maximum Intensity Projections (MIPs) of each detected vertebra. Spinorama achieves State Of The Art (SOTA) results in both labelling and localisation tasks of the VerSe 2019 challange and competative results to the SOTA in the 2020 challange.

2 Method

For the sake of runtime efficienty, 2D models were prefered. The advantage of the 2D approach over 3D here is two-fold: it takes up less memory and runs faster, which is a major advantage when integrated into a diagnostic workstation workflow; it's easy to annotate ground truth and thus provide a large training set leading to better and more robust results. The full pipeline of Spinorama, illustrated in figure 1, consists of the following steps, detailed in the following subsections:

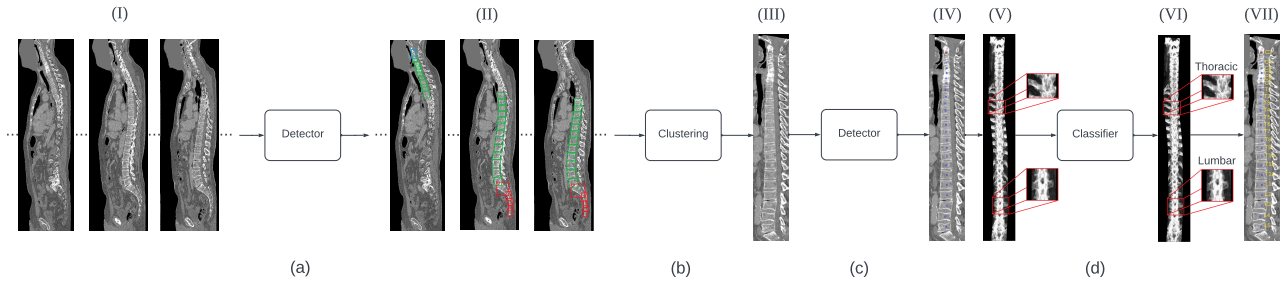

Figure 1: Spinorama pipeline: (a) Initial detection on input Sagittal images (I) producing bounding boxes (II) classified as C2 (blue), S1 (red) and Other (green). (b) Bounding boxes are clustered and a panoramic image (III) is created along their centerline. (c) detection on the panoramic image (IV). (d) On CT images, for each detected vertebra (V) a patch surrounding its center on a panoramic coronal MIP image is created and fed into the Classifier, producing the vertebra labels (VI) which are used to label the vertebrae correctly (VII).

1. Data preparation – prepare input images for the neural network to run on.
2. Initial detection – run a 2D detection neaural network in order to detect a bounding box for each vertebra on each image.
3. Clustering – remove suspected false positives and cluster the 2D bounding boxes calculated on all images in order to calculate a 3D bounding box for each vertebra.
4. Panoramic image detection – create a streightened panoramic image using the vertebra centers calculated in the previous step and run 2D detection on it using the same neural network. Use the results to create a 3D bounding box and center for each vertebra
5. Classification – The neural network can only recognize C2 and S and use them as anchors to label the vertebrae. Use a classification neural network to distinguish between C, T, L and S vertebrae and thus be able to label the vertebrae correctly in most cases.

2.1 Data Preparation

Spinorama's detector runs on sagittal images with a resolution of 416x416 and a pixel spacing of 1mm and with triple windowing. Therefore such images need to be produced.

CT images are not sagittal, and the input series is therefore resampled in a sagittal orientation and a pixel spacing of 1mm. There are 50 resampled images around the middle of the input volume with a spacing of 2mm between them. If the data is taller than 416 pixels it is divided into patches covering the data, each of them 416x416 pixels, with an overlap of 32 pixels. 3 copies of each image are created, each with a different windowing. The windowing values (min/max) are: $[600\text{HU}, 1500\text{HU}]$, $[1000\text{HU}, 1400\text{HU}]$, $[450\text{HU}, 1950\text{HU}]$.

2.2 Initial Detection

First, a YOLOv3 [3] detector is run on the 2D sagittal slices. The detector classifies vertebrae into 3 classes -- C2, S and Other (figure 1a). Since the initial detection is used to generate the panoramic

image, the threshold for the detector confidence is set at 0.55 to exclude false positives outside the spine. The inference starts on the middle slice, and then works its way out in each direction. It stops running on slices once the last slice in each direction has been reached or when 3 consecutive slices have produced zero bounding boxes. If there are multiple patches per slice the output for all patches is saved and passed to the next step.

2.3 Clustering

The initial detection step produces multiple bounding boxes per image, which now need to be combined in order to create a 3D bounding box for each vertebra. The 2D bounding boxes are clustered into a list of 3D bounding boxes using DBSCAN [4] with $\epsilon = 10\text{mm}$ the maximum distance between boxes to be considered neighbors within a cluster, and a minimum cluster size of 4. A custom distance metric (equation (1)) is used to take into account the significantly smaller size of the cervical vertebrae.

$$d = \begin{cases} h > 80, & \Delta \\ h \leq 80, & \Delta * (80/h)^2 \end{cases} \qquad (1)$$
$$h = h_1 + h_2,\ \Delta = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Here, d represents the distance between bounding boxes, h_1 and h_2 are the bounding boxes heights and (x_1, y_1) and (x_2, y_2) are the bounding boxes centers. The z coordinate, where x and y are on the sagittal plane, is ignored when measuring the distance between bounding boxes. Each cluster defines one vertebra, and the 3D bounding box of all the 2D bounding boxes in the cluster is the vertebra's bounding box. Then, detected vertebrae with center outside of a predefined range with respect to neighboring vertebrae are suspected to be false positives, and are therefore removed.

2.4 Panoramic Image Detection

A stretched panoramic image is generated along a spline built from the 3D bounding boxes centers (figure 1III). The spline is extrapolated up and down to include any possible missed vertebrae at the top or bottom of the spine. Then, The same YOLOv3 detector is run on the stretched panoramic image (figure 1c). This time a confidence threshold of 0.5 is used to ensure no vertebra is missed. False positives are unlikely due to the nature of the panoramic image where all the vertebrae are aligned one on top of the other. The depth of the bounding boxes is taken to be the same as their height.

2.5 Classification

When a C2 or an S1 vertebra is detected, it is used as an anchor to label the remaining vertebrae. When they are missing on a scan, an EfficientNet-b0 [5] classifier runs on a coronal MIP image of each vertebra with a resolution of 224x224, classifying into 4 classes -- Cervical, Thoracic, Lumbar and Sacrum. The coronal MIP projection highlights the interfaces between the different parts of the spine,

as demonstrated in figure 1VI, and increases the classifier's accuracy. The results are used to label the vertebrae correctly (figure 1d).

3 Training

For the purpose of evaluation on the VerSe 2019 & 2020 benchmarks, training was performed on the VerSe 2019 training set and VerSe 2020 training set + VerSe 2019 whole dataset respectively.

3.1 YOLOv3 Detector

The detector is a YOLOv3 model which is trained on both sagittal slices and stretched panoramic images sampled from the input volume with a ratio of 3:1. The input images have a resolution of 416x416 with a pixel spacing of 1mm. Whenever an image does not fit within this resolution it is divided into patches. The images have triple windowing. The windowing values for CT (min/max) are: $[600\text{HU}, 1500\text{HU}]$, $[1000\text{HU}, 1400\text{HU}]$, $[450\text{HU}, 1950\text{HU}]$.

The model was trained for 60 epochs using the Adam optimizer, where for the first 5 epochs, the backbone is frozen and only the YOLO head is trained. The learning rate goes from 10^{-6} to 10^{-4} linearly for the first 2 warmup epochs, then goes back to 10^{-6} for the rest of the epochs using a cosine function. The network is trained with 3 classes – C2, Sacrum and Other.

In the standard implementation of YoloV3 for each input box in the ground truth only one anchor per scale, which is placed in the center of the input box, is considered as a positive example for training. In order to improve efficiency and increase robustness we changed that so that for each input box, any anchor with an IOU > 0.4, no matter where its located and what others satisfy that condition, is considered as a positive example. This provides many more positive examples for the training, and makes the network less liable to miss vertebrae on inference.

3.2 EfficientNet Classifier

For the EfficientNet-b0 classifier training, panoramic images are created using the ground truth vertebra centers in each annotated slice. For each vertebra on the panoramic image, a coronal MIP image perpendicular to the panoramic image is created, with a resolution of 224x224 and a pixel spacing of 1mm. These images are created with triple windowing (min/max): $[1000\text{HU}, 2000\text{HU}]$, $[1000\text{HU}, 1600\text{HU}]$, $[1100\text{HU}, 1400\text{HU}]$. The vertebra centers are randomly shifted in the y and z axes to mimic detection inaccuracies.

4 Results

Spinorama was tested on the VerSe 2019 and 2020 benchmarks [1] for spine vertebrae labelling and segmentation in order to compare with the existing SOTA. The models were trained on the VerSe 2019

Method	**2019**			
	Public		Hidden	
	$id.rate$	d_{mean}	$id.rate$	d_{mean}
Payer C.	95.65 (100)	4.27 (3.29)	94.25 (100)	4.80 (3.37)
Chen M.	96.94 (100)	4.43 (3.7)	86.73 (100)	7.13 (3.81)
Tao R.	97.22	4.33	96.74	5.31
Spinorama	**98.15 (100)**	**3.00 (2.62)**	**99.07 (100)**	**2.78 (2.66)**
Method	**2020**			
	Public		Hidden	
	$id.rate$	d_{mean}	$id.rate$	d_{mean}
Payer C.	95.06 (100)	2.90 (1.62)	92.82 (100)	2.91 (1.54)
Chen M.	95.61 (100)	**1.98 (0.65)**	**96.58 (100)**	**1.38 (0.59)**
Spinorama	**97.35 (100)**	2.94 (2.91)	95.80 (100)	2.92 (2.82)

Table 1: Mean and median (in brackets) results over the Verse 2019 and 2020 datasets. $id.rate$ is reported in % and d_{mean} is reported in mm.

and 2020 datasets respectively without using any additional private data. Only the labelling metrics were compared, i.e. $id.rate$, which measures the percentage of vertebrae detected and labelled correctly, and d_{mean}, which measures the distance between the detected vertebrae centers and the ground truth vertebrae centers. It reached best in class results on the 2019 data, and on par with the SOTA on the 2020 data. The results are detailed in table 1.

References

[1] Anjany Sekuboyina. Verse: A vertebrae labelling and segmentation benchmark for multi-detector ct images. *Medical Image Analysis*, 73:102166, 2021.

[2] Rong Tao and Guoyan Zheng. Spine-transformers: Vertebra detection and localization in arbitrary field-of-view spine ct with transformers. In *Medical Image Computing and Computer Assisted Intervention - MICCAI 2021*, pages 93–103. 09 2021.

[3] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *CoRR*, abs/1804.02767, 2018.

[4] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, pages 226–231. AAAI Press, 1996.

[5] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *CoRR*, abs/1905.11946, 2019.