CSC577 Final Report: Vertebrae Segmentation in Pathological Spine CT via Fully Convolutional Neural Network

Botao Deng
University of Rochester
601 Elmwood Ave., Rochester,
bdeng3@UR.Rochester.edu

Abstract

Accurate segmentation of vertebrae in spine CT scan is very important to localize the centroids of the vertebrae, which has the potential to assist clinical task of diagnosis and surgical planning.

Our contribution is 1) applying different preprocessing schemes to Microsoft Spine dataset to enhance the visibility of sagittal spine CT scan, 2) marrying recently popular Deep Learning framework Fully Convolutional Neural Network with a semi-automatic labeling strategy, and show the potential for this marriage to produce accurate segmentation of vertebrae.

1. Introduction

Spine imaging plays an important role in evaluation of spinal injury, diagnosis, surgical planning and post-operative assessment. At the same time, wrong-level surgery is a unique problem in the spine surgery, and the consequence of wrong-level spine surgery not only generates another surgery of the intended level, but also usually associated with lawsuit ranging from $62,000 to $1,500,000 [3]. Reliable vertebrae identification on computed tomography (CT) could greatly reduce the risk of wrong-level surgery. Among all the methods to prevent wrong-level spine surgery, computer-aided approach, which support quantitative analysis on the CT scans, are of great importance. In this study, segmentation of each individual vertebra is explored. That is a crucial step toward vertebrae localization and identification, which benefits applications include fracture detection and statistical shape analysis. [2]

This task is challenging for a number of reasons, including: 1) the similarity of shape of vertebrae, 2) different scanning protocols result in CT scans with different field-of-views and resolutions, and 3) large anatomical and pathological variation. Examples of challenging cases in the dataset are highlighted in Figure 1.

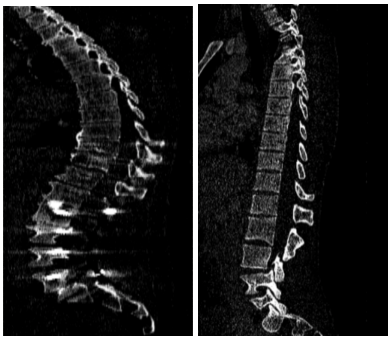

Figure 1. Two examples from the dataset. The spine on the left suffers from kyphosis, a disease denoting a forward rounding of the back. Spine on the right has fracture at the top. These two CT scans also have different field-of-view, meaning their number of vertebrates are different. All of these issues pose greater challenge for accurately segment the vertebrae within the CT scan.

2. Related Work

Automatic vertebrae localization has become a upraising research topic in medical image analysis community. And successful localization relies on acuurate vertebrae segmentation. Ma J *et al*. [6] used coarse-to-fine deformable model and learning-based edge detection to segment vertebrae, but their model was confined to Thoracic vertebra. Similarly, Oktay [7] located vertebra with PHOG based SVM and a probabilistic graphical model, but only focusing on Lumbar vertebrae. Some methods [4, 8] rely on a prior knowledge about which part of the vertebrae is visible, which makes them less generative and robust to general spine CT scans. Method [1] that rely on statistical models of shape and appearance do not work well for pathological and abnormal cases. Kindker *et al*. [5] proposed an method for automatic localization and identification of vertebrae in arbitrary field-of-view CT scans, but this method struggled when implants or fractures appeared in the Spine.

3. Method

To overcome the limitations mentioned above, Glocker *et al*. [2] proposed a vertebrae locate-and-name approach based on classification forests avoiding explicit parametric modeling of appearance. For this project, we replicated their annotation scheme in a 2-D manner. This scheme avoid the tedious manual annotation of each CT scan, and it transforms sparse centroid annotations into dense probabilistic labels easily. After obtaining probabilistic labels, instead of using extracting features using local and short-range contextual features like Glocker *et al*. [2], we employ a recent Deep Learning framework - Fully Convolutional Neural Network to classify each pixel from the input scan either as background, or one of the 26 kinds of vertebrae.

3.1. Data

The dataset we use in this study is obtained from Microsoft Research. It contains 242 annotated spine CT scans from 125 patients. Those patients suffered from high grade scoliosis and kyphosis, fractures, and numerous post-operative cases where surgical implants are causing severe image artifacts. Those images also have a varying field of view such that different images capture different parts of the spine depending on the pathology. In a few scans the whole spine is visible, while in most scans the views are limited to 5-15 vertebrae. Image data is provided in MetaImage file format (mhd/raw).

3.2. Preprocessing

The CT scans in this dataset were taken from axial prospective, so the 2D sagittal slices that we use in the classification task do not have the highest resolution and they need to be preprocessed. One of the original sagittal scan was shown in Figure 2. There are lots of noise both inside and outside the spine, some of the noise was generated when warping the spline slide for visibility purposes. To remove other noise around the Spine, we applied the following strategies.

The first scheme is converting the data type of the original image slices from floating point to unsigned byte, with value in (0, 255). We can see from Figure 2 that most of the gray areas are removed, that's because the majority of gray area have negative value. And during conversion those areas are all thresholded to zero. But there are still lots of gray dots lying in the background.

The second preprocessing scheme is adaptive histogram equalization. We've tried global histogram equalization but the result turned out to be very bad. Adaptive Histogram Equalization differs from global equalization in the sense that it computes various histograms, and each corresponds to a distinct area of the image. After redistributing the lightness of those histograms, the local contrast and the definition of the edges are enhanced. This technique suits well to spine CT images because the high lightness spots are concentrated around the Spine, and the small dots away from the spine can become invisible after redistributing its lightness with it local surrounding background.

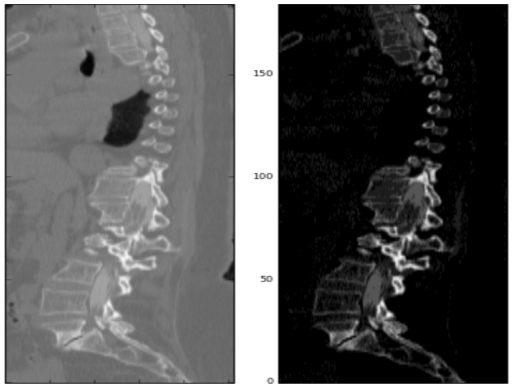

Figure 2. Left: Original sagittal CT scan obtained from the dataset. Right: After converting date type to unsigned byte.

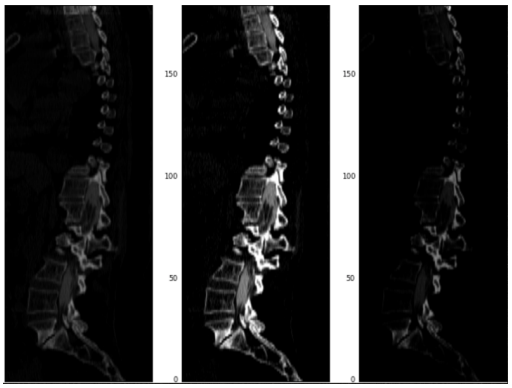

Figure 3. Left: Image after adaptive histogram equalization. Middle: Image after contrast stretching. Right: Element-wise multiplication of the previous two image.

The third scheme is Contrast Stretching, we prefer this scheme over the former one because the background constitute the majority of the spine CT scan, and even though Adaptive Histogram Equalization remove some artifacts in the background, it darkens the spine in a certain degree. As the first step on Contrast Stretching, we selected the 65^{th} and 99^{th} percentage from the global histogram, then linearly stretched the intensity range to (0, 255). The number 65 and 99 were manually chosen to prevent outliers that might lead to unrepresentative scaling, meaning the spine's intensity in the CT scan should approximately lies between 65^{th} and 99^{th} of the maximum intensity of the whole image. The result is shown in the middle of Figure 3.

The last scheme is element-wise multiplication between the image after Adaptive Histogram Equalization and the image after Contrast Stretching. The motivation behind this

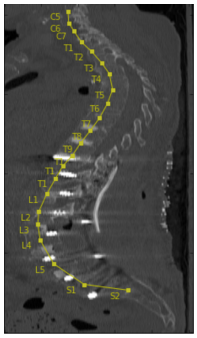

Figure 4. Plotted the vertebrae centroid in the CT scan using the voxel coordinates after conversion.

is that, some surgical implant may survive after Contrast Stretching, as long as their intensities lie between 65^{th} and 99^{th}. And their intensities are reduced after Adaptive Histogram Equalization. Therefore, the intensity of the implants in the image after Contrast Stretching can be reduced with an element-wise multiplication between these two. Example was shown on the right of Figure 3

3.3. Dense Annotation

To fine-tune a fully convolutional neural network, pixel-wise labeled training data is required. Manually annotated original CT scans can be very tedious and time-consuming. The labeling strategy proposed by Glocker *et al.* [2] suggested to transform sparse centroid annotations into dense probabilistic labels. The centroid likelihood function for each vertebrae is shown as

$$\psi_v(x) = exp(-\frac{||c_v - x||^2}{h_v})$$

But to create dense label using the defined function, the first step is to locate the centroid of the vertebrae. The provided data only contains physical coordinate of the centroids, the formula for the conversion is

$$voxel\,coordinate = \frac{physical\,coordinate}{element\,spacing}$$

The resulting localization of vertebrae centroids is shown in Figure 4.

Unlike Glocker *et al.* [2] who perform voxel classification on a 3D CT scan volume, in this study we use 2D sagittal slices to conduct the classification task. Therefore, we modified the centroid likelihood function into

$$\psi_v(x) = exp(-(\frac{||c_v(y) - x(y)||^2}{8 * h_v} + \frac{||c_v(z) - x(z)||^2}{h_v}))$$

This is done because of the difference in resolution from different prospectives. The original CT was obtained from

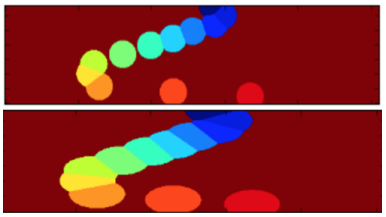

Figure 5. Comparison of the label map using different centroid likelihood function.

axial prospective, therefore the spacing between axial slices only ranges from $0.2 - 0.3$ millimeter. But for the sagittal slices, the spacing is 2.5 millimeter. Therefore, the sagittal images are longer in y axis than in the z axis. That's the most important reason for us adjust the likelihood function. This way, the vertebrae labels have a better coverage on the vertebrae. Comparison between original label mask and the adjusted label map was shown in Figure 5.

We also define a likelihood function for background label B as

$$\psi_B(x) = 1 - max_v \psi_v(x)$$

The intuition behind this function is, the closer a pixel to a centroid, the more likely that it would be labeled as centroid. If a pixel is far away from all centroids, then that pixel is going to have a high value on function ψ_B. Based on the likelihood definition, we further normalized the functions to obtain a labeling distribution

$$p(l|x) = \frac{\psi_l(x)}{\sum_{m \in L} \psi_m(x)}$$

Here l corresponds to any of the vertebrae centroid, L is a set of all the vertebrae and x refers to any pixel. In this case, we have 26 kinds of vertebrae plus one background, therefore by the end of this section we end up with 27 label maps, each of this map shows the probability distribution of one label in all the pixels.

3.4. Fully Convolutional Neural Network

Unlike Glocker *et al.* [2] who extract features based on local and contextual intensity and localize vertebrae with supervised classification random forest, we employ a Fully Convolutional Network (FCN) to classify all pixels as one of the fixed classes all at once.

There are three sets of input to a pre-trained FCN. In the first set of input, we copy the original image to R, G, B channel. This way, no information is lost, but at the same time noise was kept in the input. In the second set, we transform the original CT scan, image after Contrast Stretching, image after Adaptive Histogram Equalization into R, G, B channel. In this case, original image is kept in the R channel to ensure no information is lost. For the third set of input,

Network	batch-size	Optimization	20iters	3-epoch	learning-rate
FCN-8s	64	SGD	>30 mins	-	$10^{(-10)}$
FCN-8s	16	SGD	>30 mins	-	$10^{(-10)}$
FCN-8s	3	SGD	20 mins	4 days	$10^{(-10)}$
FCN-32s	3	SGD	5 mins	25 hours	$10^{(-10)}$
FCN-32s	3	Adam	4-6 mins	25 hours	$10^{(-12)}$

Table 1. Different fine-tuning protocols. Column "20iters" shows the time needed to complete 20 iterations; "3-epoch" shows the time needed to complete training of 3 epochs

original CT scan, image after Contrast Stretching, image after multiplication of Contrast Stretching and Adaptive Histogram Equalization are transformed into R, G, B, aiming at providing a more spine-focused CT scan to the FCN.

We consider the FCN-32s as well as FCN-8s, which gives the optimal performance over the other. But we pick FCN-32s because of the high computational cost of FCN-8s. FCN-32s is FCN-VGG16, and the output prediction is computed on top of the last deconvolutional layer.

4. Implementation Detail

The following experiments were done in an iMac with 2 Quad-Core Intel Xeon CPU.

We experimented different hyper-parameters during fine-tuning, shown in the table above. At first we tried to use FCN-8s with a relatively large batch-zise. But it turned out even with batch size 16, it would take forever to train even 3 epochs. And changing the batch size to 3 still take 4 days to run through 3 epochs. We choose 3 here because each scan has 30 sagittal slices, and 30 is a multiple of 3 so that the input slices of the FCN will have the same size. We eventually switched to FCN-32s given the high computation cost of FCN-8s.

After deciding on the hyper-parameters of fine-tuning, we changed the number of output of the last layer from 21 to 2 and set the learning rate of all previous layers to zero. But after 25 hours running, the FCN predicted every pixel in the test image as background.

To solve that problem, we ignored the background so that the prediction wouldn't bias toward predicting background. We also added Xavier initialization on the deconvolutional layers. But the result after 3 epochs looks like random noise, shown in Figure 6

To analyze the underlying reasons, we plot the learning curve during fine-tuning and observed that it was not converging at all. Therefore we tried a smaller learning rate for a better convergence, and we also tried switching to another optimization type. But neither of these solve the problem. By a closer look at Figure 6, we assumed that it might be possible that the network didn't learn anything from the very beginning and predicting random noise all the time. So we change the initialization scheme from Xavier to Bilinear.

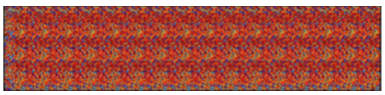

Figure 6. Prediction of FCN-32s by ignoring background and Xavier initialization.

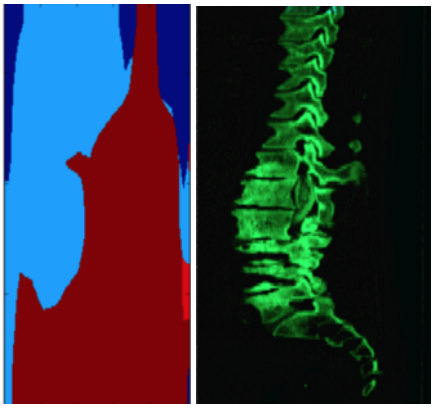

Figure 7. Prediction of FCN-32s after initializing deconvolutional layers with bilinear kernels

And we plot the result in Figure 7.

5. Conclusion

Noticing the the result shown in 7 only fine-tuned for 3 epochs, and it's becoming aware of the difference between vertebrae and background. Therefore, it's fair to say that with longer training time and a GPU to accelerate the training process, the FCN can provide a better prediction on the vertebrae.

We must also look at the down sides of this approach, the most significant problem with this approach is that it's not robust toward diseases like Scoliosis. Because by extracting sagittal slices from a 3D volume, we lost information along the x axis. Therefore the ground truth annotations of sagittal slices are not very accurate.

References

[1] B. Glocker, J. Feulner, A. Criminisi, D. R. Haynor, and E. Konukoglu. Automatic localization and identification of vertebrae in arbitrary field-of-view ct scans. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 590–598. Springer, 2012.

[2] B. Glocker, D. Zikic, E. Konukoglu, D. R. Haynor, and A. Criminisi. Vertebrae localization in pathological spine ct via dense classification from sparse annotations. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 262–270. Springer, 2013.

[3] J. Hsiang. Wrong-level surgery: a unique problem in spine surgery. *Surgical neurology international*, 2, 2011.

[4] S.-H. Huang, Y.-H. Chu, S.-H. Lai, and C. L. Novak. Learning-based vertebra detection and iterative normalized-cut segmentation for spinal mri. *IEEE transactions on medical imaging*, 28(10):1595–1605, 2009.

[5] T. Klinder, J. Ostermann, M. Ehm, A. Franz, R. Kneser, and C. Lorenz. Automated model-based vertebra detection, identification, and segmentation in ct images. *Medical image analysis*, 13(3):471–482, 2009.

[6] J. Ma, L. Lu, Y. Zhan, X. Zhou, M. Salganicoff, and A. Krishnan. Hierarchical segmentation and identification of thoracic vertebra using learning-based edge detection and coarse-to-fine deformable model. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 19–27. Springer, 2010.

[7] A. B. Oktay and Y. S. Akgul. Localization of the lumbar discs using machine learning and exact probabilistic inference. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 158–165. Springer, 2011.

[8] S. Schmidt, J. Kappes, M. Bergtholdt, V. Pekar, S. Dries, D. Bystrov, and C. Schnörr. Spine detection and labeling using a parts-based graphical model. In *Biennial International Conference on Information Processing in Medical Imaging*, pages 122–133. Springer, 2007.