ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

Graph-cut-assisted CNN training for pulmonary embolism segmentation

Nana Yang, Robin Verschuren and C. De Vleeschouwer *

ICTEAM - UCLouvain
Louvain-la-Neuve 1348 - Belgium

Abstract. We present a novel algorithm for pulmonary embolism segmentation, designed to alleviate the need for expert annotation. Our approach integrates deep learning with a conventional image segmentation techniques, operating in two distinct stages. Specifically, graph cut is used for initial segmentation, followed by manual refinement, to define the labels required to train a CNN. This CNN is then employed to generate pseudo-labels on a large dataset, enabling the training of an improved CNN*. Our findings demonstrate enhanced performance of CNN* over CNN. Overall, the CNN* builds on a very limited amount of manual intervention. Moreover, the injection of expert knowledge in the graph-cut avoids the need for expert knowledge in this manual intervention.

1 Introduction

Pulmonary embolism (PE) typically results from the obstruction of a pulmonary artery by a blood clot. These thrombi originate from deep veins in the lower extremities, dislodging and traversing through the bloodstream to occlude the pulmonary arteries. PE is the third most common cardiovascular condition, with its annual incidence rate showing an increasing trend. Timely diagnosis and intervention play pivotal roles in mitigating morbidity associated with PE.

Computed tomography pulmonary angiography (CTPA) is widely used in the diagnosis of PE[1]. However, each patient's CTPA typically comprises numerous images, rendering the manual review process by physicians laborious and susceptible to errors. Automatic computer vision methods have thus been considered to segment PE. A PE binary segmentation map identifies the scan voxels that correspond to arteries whose oxygenation is affected by the presence of a clot. Early approaches have implemented segmentation of blood vessels, followed by discrimination between PEs and non-PEs based on features such as blood vessel intensity and size[2]. Recently, [3] introduced a 2D CNN model for PE segmentation. However, the model's training necessitates a substantial volume of annotated data. Collecting a representative set of data is expensive in terms of human expertise, due to the diverse nature of PE and its low contrast with surrounding tissues, thereby rendering the annotation task particularly demanding. [4] proposed an automated PE segmentation approach eliminating the need for manual outlining. However, this method involves manual labeling of the central extra-pulmonary arteries and veins by experts.

*Nana Yang is funded by China Scholarship Council. C. De Vleeschouwer is funded by the Belgian F.N.R.S.

ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

In this paper, we propose to rely on anatomical knowledge to generate the PE annotations required to train a CNN. This is because PE is known to appear as darker voxels in the pulmonary artery. Hence, once pulmonary artery has been segmented, segmenting the PE becomes trivial, and even a non-expert becomes able to correct the segmentation errors resulting from simple intensity thresholding. In our work, the anatomical knowledge is encoded in the form of a set of 3D scan positions where the vascular system components are expected to be located. Those points are exploited to associate scores to each voxels, reflecting their likelihood to be part of one of each anatomical part of interest. A graph-cut algorithm is then considered to regularize those scores and assign anatomical labels to voxels. Optional manual refinement of the extracted PE is envisioned before using a limited number of (semi-)automatically extracted PE maps to train a CNN. The trained model is then applied to automatically define pseudo-labels on a larger, unlabeled dataset. These pseudo-labels are used to train a so-called data-enriched CNN*, which appears to improve the initial CNN.

2 Materials and methods

2.1 Dataset

The dataset utilized in this study originates from the RSNA/STR Pulmonary Embolism Detection Challenge hosted on Kaggle[5]. It encompasses 7,279 CTPA scans, manually annotated, with 401 scans identified as containing central PE. Among these 401 scans, 10 scans are kept as a test set, and PE segmentation maps are manually delineated for each test sample. The 391 remaining scans are considered to train a neural network, with minimal manual annotation effort.

2.2 Label generator based on graph cuts

Drawing inspiration from the methodology outlined in [6], we employ graph-cut[7] to segment the cava vena, aorta, pulmonary artery trunk, and background in the chest CT scans. Following the anatomical segmentation, a post-processing based on intensity thresholding is applied to facilitate PE candidate identification, which categorizes regions into two distinct classes: background and PE.

This procedure comprises five stages: (1) preprocessing to enhance the quality of raw CT scans, (2) employing quickshift[8] to generate superpixels, followed by (3) estimating class scores for each superpixel by aggregating data, considering intensities, and region characteristics, (4) constructing a graph based on the class scores to connect superpixels, then employing graph cut with α-expansions to turn pixel scores into labels, (5) finally, relying on pixel intensity to identify PE pixels among pulmonary arteries. Some regions that contain PE are sometimes erroneously classified as background. Therefore, we identify high intensity background regions that are adjacent to regions labeled as pulmonary artery as PE. Since the automatically generated PE maps are not always correct, they are manually refined when needed.

ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

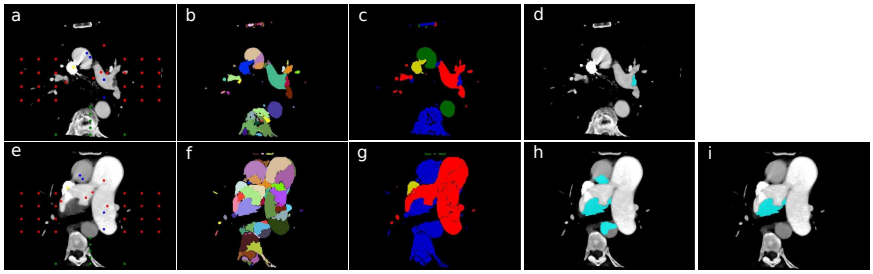

Fig. 1: Outcomes of the procedure of label generator.(a,e)The knowledge map(points on a 2D plane linked to spine(green), vena cava(yellow), aorta(blue) and pulmonary artery(red)) is applied to the preprocessed result. (b,f) Quickshift result. (c,g) Graph-cut result(background in blue, pulmonary arteries in red, vena cava in yellow, aorta in green). (d,h)Post processed result(PE in cyan). (i)Manually refined result.

The first stage comprises body and lung masking, and cropping procedures designed to exclude extraneous information beyond the lung boundaries, followed by three intensity-based operations(windowing, histogram equalization and Curvature Anisotropic Diffusion filter[9]) aimed at enhancing image quality and diminishing noise. In the second stage, we employ a superpixel approach to seamlessly cluster pixels into coherent regions. This method enhances computational efficiency while ensuring accurate boundary delineation. Within this study, we implement quickshift, a cutting-edge technique, recognized for its capacity to rapidly perform hierarchical segmentation across a spectrum of scales. Results are illustrated in Figure 1

In the third stage, our approach entails computing class scores sequentially, as follows:

$$s_b = max(\sigma(I_p); \max_{k \in K_{spine}} (Ker(p\_a, k))) \tag{1}$$

where s_b is the score of background, I_p is the intensity of the superpixel p, Ker() is the Gaussian Kernel function measuring similarity, p_a is p's coordinates in the axial plane, the first term is the alpha-beta-sigmoid function [10] of I_p , scoring regions with lower intensities as background. Background encompasses spine and other superpixels not linked to the other three semantic categories. The second is obtained using the knowledge map, a basic guide for anatomical segmentation, with points on a 2D plane (as is shown in Figure 1) linked to spine(green), vena cava(yellow), aorta(blue) and pulmonary artery(red). And K_{spine} represents the set of reference points in the knowledge map for the spine.

$$s_{vc} = I_p \cdot Ker(p\_a, VC) \cdot w(z_p) \tag{2}$$

In equation 2, VC is the vena cava's center in the axial plane. Its value at the scan's top is based on a predefined approximate position common to any scan.

ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

Then, other slices update the vena cava along its vertical structure. And $w(z_p)$ is a weight used to decrease the score in vertical regions where the vena cava is unlikely to be found. z_p denotes the vertical component of p

$$s_{aorta} = I_p \cdot w(z_p) \cdot \max_{k \in K_{aorta}} (Ker(p\_a, k)) \tag{3}$$

In this equation, K_{aorta} represents the set of reference points in the knowledge map for the aorta.

The scoring function for pulmonary artery is defined as:

$$s_{pa} = I_p \cdot w(z_p) \cdot (\max_{k \in K_{pa}} (Ker(p\_a, k)) + InLung(p)) \tag{4}$$

where InLung() provides a binary indication of whether the centroid of a given superpixel p lies within the lung mask, this term is to identify smaller pulmonary arteries distributed throughout the lungs.

Having obtained the score functions for each superpixel and anatomical-semantic class, we proceed to the fourth stage. In this stage, we construct a graph by connecting superpixels horizontally and vertically. Subsequently, we define the weight function for the edges.

Following[6], we define an energy function for graph-cut based image segmentation:

$$E(f) = \sum_{p \in P} D_p(f_p) + \sum_{(p,q) \in N} W(p,q)(1 - \delta(f_p, f_q)) \tag{5}$$

where p denotes the superpixel index, with $p \in \{1, 2, \ldots, \text{n\_superpix}\}$. Each superpixel p is assigned to a class $f_p \in$\{background, pulmonary artery, aorta, vena cava\}. The distance $D_p(f_p)$ of a superpixel p to the class f_p is defined as $1 - s_{f_p}(p)$. δ is the Kronecker delta, N includes all pairs of superpixels (p,q) such that p is in P and q is a neighbor of p. W(p,q) defines the energy penalty induced when assigning distinct labels to superpixel p and its neighbor q.

The weight of a vertical edge(between adjacent slices), aiming to assign higher weights to superpixels with larger intersections, is calculated as following:

$$W(p,q) = \frac{n(intersect(p,q))}{min\{n(p), n(q)\}} \tag{6}$$

For the horizontal edge weight(within a slice), it is calculated based on the semantic scores of each superpixel.

$$W(p,q) = 1 - \frac{1}{1 + \alpha e^{-\beta M(p,q)}} \tag{7}$$

where M(p, q) is defined by:

$$M(p,q) = \min_{k \in \{p,q\}} (s_{M_k}(k) - s_{M2_k}(k)) \tag{8}$$

ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

time(minutes)		20	40	80	120
CNN	foreground_iou	0.31	0.33	0.40	0.42
	precision	0.48	0.50	0.53	0.57
	sensitivity	0.47	0.50	0.61	0.63
CNN*	foreground_iou	0.32	0.37	0.42	0.44
	precision	0.48	0.53	0.55	0.58
	sensitivity	0.50	0.55	0.67	0.69

Table 1: Performance of CNN and CNN* under different manual refining time.

where α and β are parameters to fine-tune the model. M_k and $M2_k$ denote the class with highest and second highest score for superpixel k, respectively. Equation (8) induces a large M, and thus a small penalty which does not discourage a border, only when both pixels have a high score compared to the second best.

Then, we perform graph cut and identify PE. This is followed by manual refinement of the obtained results.

2.3 Pseudo-label generator

After obtaining annotated data, we proceed to train a CNN model using supervised learning methods. This model will be utilized to generate pseudo-labels for a large amount of unlabeled data, which will then be used for training CNN*. CNN* is trained on a large dataset(with 361 samples) with pseudo-labels.

3 Experiment results

We segment 10 scans from the dataset using graph cut and manually refine the segmentation results, as shown in Figure 1 . The refined results are treated as the test dataset. Subsequently, we manually refine a varying number of images from 30 scans, using 20 minutes(5 scans), 40 minutes(10 scans), 80 minutes(20 scans), and 120 minutes(30 scans) , respectively only the manually refined scans are used as training sets for CNN. And CNN* is trained on 361 scans with pseudo labels generated by CNN.

The network architecture employed in this study is based on the classic U-net[11]. We utilizes Adam as the optimizer and Log-Cosh Dice Loss[12] as the loss function. The batch size is set to 16, and images are resized to 256x256. During training, data augmentation techniques including rotation, flipping, transposition, brightness, and contrast adjustments are applied to enhance data diversity. The initial learning rate for CNN training is set to 0.001, while for CNN* training, it is initialized to 0.00001 with exponential decay learning rate. All models are trained on Nvidia GeForce GTX 321 1080 Ti 11Gb GPUs

Table 1 displays the metrics of foreground IoU, precision, and sensitivity on the test set for CNN and CNN* using different annotation time. Foreground IoU indicates the degree of overlap between the predicted PE and the actual PE. Precision denotes the proportion of pixels predicted as PE that are genuinely PE

ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

pixels, while sensitivity reflects the ratio of all true PE pixels that are accurately predicted.

From the experimental results, it is evident that the performance of CNN* has been enhanced compared to CNN. The improvement in precision and sensitivity indicates a reduction in error rates in identifying PE, thus enhancing the model's ability to capture true targets.

References

[1] Andetta R Hunsaker, Michael T Lu, Samuel Z Goldhaber, and Frank J Rybicki. Imaging in acute pulmonary embolism with special clinical scenarios. *Circulation: Cardiovascular Imaging*, 3(4):491–500, 2010.

[2] Haydar Özkan, Onur Osman, Sinan Şahin, and Ali Fuat Boz. A novel method for pulmonary embolism detection in cta images. *Computer methods and programs in biomedicine*, 113(3):757–766, 2014.

[3] Zhenhong Liu, Hongfang Yuan, and Huaqing Wang. Cam-wnet: An effective solution for accurate pulmonary embolism segmentation. *Medical Physics*, 49(8):5294–5303, 2022.

[4] Jiantao Pu, Naciye Sinem Gezer, Shangsi Ren, Aylin Ozgen Alpaydin, Emre Ruhat Avci, Michael G Risbano, Belinda Rivera-Lebron, Stephen Yu-Wah Chan, and Joseph K Leader. Automated detection and segmentation of pulmonary embolisms on computed tomography pulmonary angiography (ctpa) using deep learning but without manual outlining. *Medical Image Analysis*, 89:102882, 2023.

[5] Anouk Stein, Carol Wu, Chris Carr, Errol Colak, George Shih, Jeff Rudie, John Mongan, Julia Elliott, Luciano Prevedello, Marc Kohli, Phil Culliton, and Robyn Ball. RSNA STR pulmonary embolism detection. https://www.kaggle.com/competitions/rsna-str-pulmonary-embolism-detection, 2020.

[6] Arnaud Browet, Christophe De Vleeschouwer, Laurent Jacques, Navrita Mathiah, Bechara Saykali, and Isabelle Migeotte. Cell segmentation with random ferns and graph-cuts. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 4145–4149. IEEE, 2016.

[7] Andrew Delong, Anton Osokin, Hossam N Isack, and Yuri Boykov. Fast approximate energy minimization with label costs. *International journal of computer vision*, 96:1–27, 2012.

[8] Andrea Vedaldi and Stefano Soatto. Quick shift and kernel methods for mode seeking. In *Computer Vision–ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part IV 10*, pages 705–718. Springer, 2008.

[9] Joachim Weickert et al. *Anisotropic diffusion in image processing*, volume 1. Teubner Stuttgart, 1998.

[10] Qiang Huo and Bin Ma. Online adaptive learning of continuous-density hidden markov models based on multiple-stream prior evolution and posterior pooling. *IEEE transactions on speech and audio processing*, 9(4):388–398, 2001.

[11] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.

[12] Shruti Jadon. A survey of loss functions for semantic segmentation. In *2020 IEEE conference on computational intelligence in bioinformatics and computational biology (CIBCB)*, pages 1–7. IEEE, 2020.