3D Binary Lesion Mask Parsing

YI-QING WANG, IBM Watson Health Imaging, France; yi-qing.wang@ibm.com
GIOVANNI PALMA, IBM Watson Health Imaging, France; giovanni.palma@ibm.com

Liver lesion segmentation is a key module for an automated liver disease diagnosis system. Numerous methods have been developed recently to produce accurate 3D binary lesion masks for CT scans. From the clinical perspective, it is thus important to be able to correctly parse these masks into separate lesion instances in order to enable downstream applications such as lesion tracking and characterization. For the lack of a better alternative, 3D connected component analysis is often used for this task, though it does not always work, especially in the presence of confluent lesions. In this paper, we propose a new method for parsing 3D binary lesion masks and an approach to evaluating its performance. We show that our method outperforms 3D connected component analysis on a large collection of annotated portal-venous phase studies.

CCS Concepts: • **Applied computing → Imaging**; **Health care information systems**.

Additional Key Words and Phrases: liver lesion segmentation, 3D binary lesion mask parsing, performance evaluation

ACM Reference Format:
Yi-Qing Wang and Giovanni Palma. 2018. 3D Binary Lesion Mask Parsing. In . ACM, New York, NY, USA, 9 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Computed Tomography (CT) scans with contrast are a common medical imaging technique for liver disease diagnosis. Automating their analysis thus helps facilitate clinical workflow. To achieve this goal, it is important to be able to segment lesion instances accurately because its performance directly impacts downstream applications such as lesion tracking and characterization. In recent years, many lesion segmentation methods have been developed. Most of them [2, 3, 6, 7, 10–14] cast the problem as a binary classification task. Specifically, for a CT volume, they determine whether its individual voxels belong to a liver lesion and produce a 3D binary lesion mask as a result. Despite their impressive performances, accurate binary masks by themselves are often not enough for the clinical purposes, because in general, they cannot answer such important questions as: How many lesions have been found? How big are they?

To bridge this gap, a post-processing step is needed to parse a 3D binary lesion mask into separate lesion instances. Thanks to its simplicity, 3D connected component analysis (CCA) is usually employed to this end [3]. Unfortunately, it does not always work, especially in the presence of confluent lesions. It is because, due to their proximity, the segmentation masks of confluent lesions generally form a single connected component. As a consequence, even when these lesions are perfectly segmented, CCA is unable to distinguish them.

This motivates us to propose a new post-processing method to parse 3D binary lesion masks. To the best of our knowledge, it represents the first attempt at substituting CCA for this task. Our method runs in two stages. First, it splits a binary lesion mask slice-wise into separate 2D lesion

DMIP, November 10–13, 2022, Kyoto, Japan

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

cross-sections. Then it generates 3D lesion instances by grouping the obtained lesion cross-sections across slices. To evaluate its performance, we introduce a new metric that allows us to demonstrate that our method can better distinguish confluent lesions than CCA on a large collection of annotated liver lesion data.

The rest of this paper is organized as follows. First, we describe our data and method. Next we present the performance evaluation protocol and validate our approach with experimental results.

2 DATA AND METHOD

2.1 Data

Our data is a private set of 2511 portal-venous liver studies of adult patients collected from several sources, ranging from broadly distributed tertiary care hospitals to teleradiology services of a number of smaller hospitals. They were split into two mutually exclusive sets for training and testing, leading to a total of 1934 and 577 CT volumes respectively. All of them were annotated by a team of radiologists, who reviewed radiology reports and drew contours for each focal liver finding present in the images. It resulted in 10356 and 3175 ground truth 3D lesions in the two datasets respectively.

2.2 Method

Consider a CT volume and an accompanying 3D binary lesion mask of the same size. We are interested in the following question: *how many distinct 3D lesions are there covered by the mask and what are their spatial extents?* Clearly, the mask can help locate lesions only if it is of good quality. Thanks to the recent works [3, 6, 7, 10–14], we may assume the availability of a binary segmentation algorithm able to consistently produce accurate masks that align closely with the ground truth lesion contours.

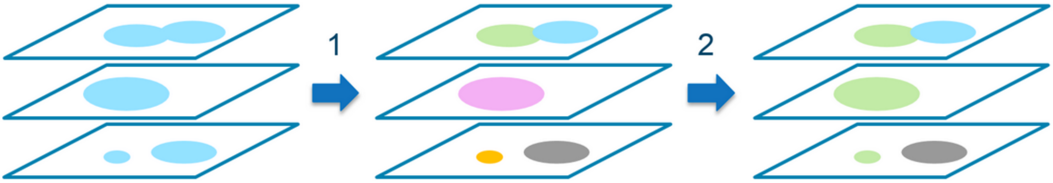


Fig. 1. Illustration of the proposed post-processing algorithm. Step 1, the input binary lesion mask is processed slice-wise to split up partially merging lesion bubbles, as shown in the top slice on the left-most figure. Step 2, the resulting 2D lesion cross-sections are grouped across slices to form 3D lesion instances. For visualization, we assigned different colors to different 2D and 3D lesions.

Because of its voxel-to-voxel connectivity, CCA is sensitive to noise. As a matter of fact, for some binary masks, it is possible to merge multiple separate connected components into one single connected component by flipping the label of just a few voxels. For this very reason, CCA can successfully parse a binary lesion mask only if 1) confluent lesions are absent and 2) the employed binary segmentation algorithm creates one 3D connected component per detected lesion. Unfortunately, these conditions are too restrictive and do not hold for real life studies in general.

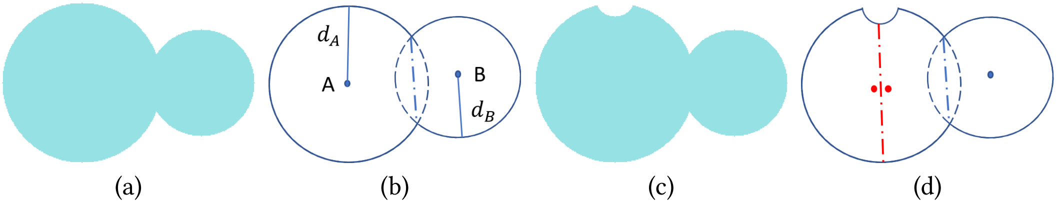
In order to translate a binary lesion mask into separate 3D lesions, rather than relying on the voxel-to-voxel connectivity, we take 2D lesion cross-sections as the basic building blocks. Specifically, our algorithm proceeds in two steps: 1) slice-wise lesion mask splitting and 2) z-wise cross-section connection. The first step partitions a 3D binary lesion mask into a set of 2D lesion cross-sections located across slices, which are then grouped by the second step to result in 3D lesion instances. See Fig.1 for an illustration of its operations on a toy example.

Let us now describe the two steps in details.

2.2.1 Slice-wise lesion mask splitting. This step aims to identify distinct 2D lesion cross-sections present in each slice of a binary lesion mask. To this end, we use the popular watershed algorithm [1]. Since this algorithm is sensitive to noise and may lead to over-segmentation [5], we constrain its behaviour by taking into account some lesion shape prior. Specifically, based on the observation that most biological lesions are of a bubble shape and that an abdominal CT scan typically has the same spatial resolution along its x and y axes, we developed a scheme to coarsen the watershed partition to result in roughly circular sub-regions.

To simplify the presentation, let us consider the synthetic binary lesion mask in Fig.2(a). To extract all the 2D lesions covered under this mask, ideally we should set one center point per 2D lesion to seed the watershed algorithm. It would then partition the mask into sub-regions, each of which has exactly one seed point sitting inside. The resulting sub-regions could be interpreted as the estimated 2D lesions.

Although we do not know what the 2D lesions are, it is possible to locate their center points approximately. To do so, we compute the 2D mask's distance transform and take its local maxima because the 2D lesion centers generally correspond to the points the most distant from their segmented boundaries. Unfortunately, in this way, we may accidentally create more approximate center points than there are 2D lesions. In fact, as shown in Fig.2(d), as soon as a mask presents some geometric irregularities, clusters of local maxima tend to emerge from its distance transform, which causes the watershed algorithm to over-segment the 2D lesions.



(a) (b) (c) (d)

Fig. 2. Direct application of the watershed algorithm on (a) a synthetic mask in shape of two partially merging circles and (c) a slightly altered mask. For (a), it results in (b) two local maxima and their corresponding sub-regions separated by the vertical dashed line segment. For (c), three local maxima (two red and one blue) emerge, leading to an over-segmentation (d) of the larger 2D lesion on the left.

To correct this behavior, we introduce a hypothesis testing step to coarsen a watershed partition when needed. To explain how it works, consider the local maxima A and B in Fig.2(b). We posit the hypothesis that they are the centers of two *distinct* circular 2D lesions with radius d_A and d_B respectively. This simplified setup allows us to derive an analytical measure $m : (A, d_A, B, d_B) \in (\mathbb{R} \times \mathbb{R}_+)^2 \mapsto \mathbb{R}_+$ to assess how much the two 2D lesions overlap. We accept the hypothesis if their overlap is small, i.e. $m(A, d_A, B, d_B) < t$ for some prefixed threshold t. Otherwise, we consider the local maxima as two approximate center points of the same 2D lesion and merge their corresponding watershed sub-regions. Such is the case for the two red local maxima in Fig.2(d).

For this scheme to work, the measure m needs to reflect the closeness of two circles. In this work, we set it to $\frac{|S_A \cap S_B|}{\min(|S_A|,|S_B|)}$ where S_x denotes the 2D ball centered on x with radius d_x and the operator $|\cdot|$ returns the size of its argument. For the threshold t, we ran empirical evaluations on the training data for its determination (see Section 3.2).

2.2.2 Z-wise cross-section connection. A 2D lesion represents a cross-section of a 3D lesion. In this step, we are interested in recovering the 3D lesions which best explain the obtained 2D lesions. To do so, let us consider two arbitrary 2D lesions located on two adjacent slices. We denote them L_A and L_B. *Do they belong to the same 3D lesion?* This question is of great importance here because the ability to answer it correctly for all the 2D lesion pairs allows us to group them into separate 3D lesion instances using e.g. the Union-Find algorithm [4]. Moreover, such a formulation essentially casts the problem in the binary classification setting so that we can make explicit use of our annotated data.

A biological lesion is typically smooth in shape. As a result, the event that L_A and L_B are the cross-sections of the same 3D lesion is more likely if they 1) intersect significantly when projected onto the same xy-plane and 2) are sufficiently similar in size. Therefore, in order to assess its conditional probability, we created a logistic model with the following features:

$$r_0(L_A, L_B) = \frac{|\pi(L_A) \cap \pi(L_B)|}{\min(|L_A|, |L_B|)}, \tag{1}$$

$$r_1(L_A, L_B) = \frac{|\pi(L_A) \cap \pi(L_B)|}{\max(|L_A|, |L_B|)} \tag{2}$$

where π maps a set L of 3D points to $\{(x, y, 0), (x, y, z) \in L\}$. These features help measure the relative size of the overlap of the two projected 2D lesions. Both are by design symmetric in their arguments and therefore invariant under reversal of the slice order.

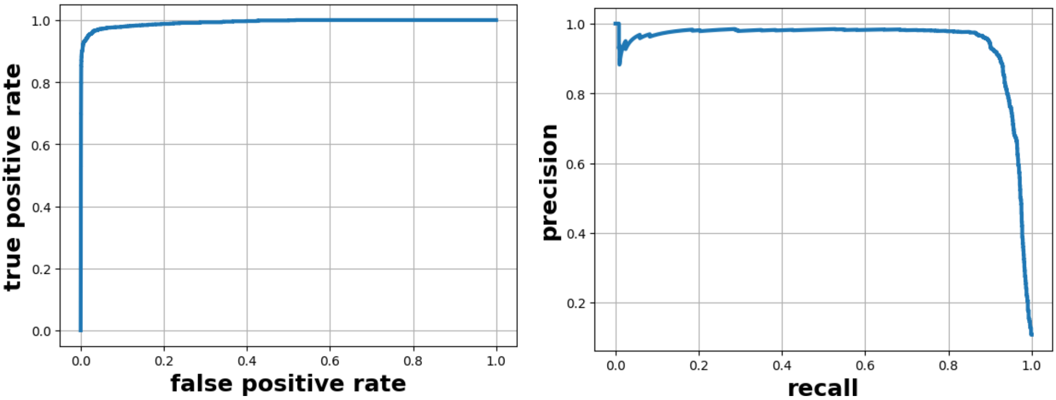


Fig. 3. The ROC (left) and Precision-Recall (right) curves of the trained logistic model for z-wise cross-section connection on our test data.

From our training dataset, we derived a total of 34578 2D lesion pairs. A pair is said to be negative if its elements belong to the same 3D lesion. It is positive otherwise. Positive pairs accounted for 5.1 percent of the training pairs. Fig.3 plots the Receiver Operating Characteristic (ROC) and Precision-Recall curves of our trained model on 12450 2D lesion pairs extracted from the test dataset with a similar imbalance ratio.

3 EXPERIMENTS

3.1 Performance evaluation protocol

To explain our evaluation protocol, we need some notations. Let $1_L(x)$ be the indicator function of a 3D lesion L. For a CT volume with n separate lesions $\mathcal{L} = \{L_i \mid i = 1, \ldots, n\}$, we represent its ground truth lesion mask by $\sum_{i=1}^{n} i \times 1_{L_i}(x)$. If we apply a permutation σ to the n lesion indices, the mask becomes $\sum_{i=1}^{n} \sigma(i) \times 1_{L_{\sigma(i)}}(x)$. Clearly, when *binarized*, both ground truth masks lead to the same $\sum_{i=1}^{n} 1_{L_i}(x)$. As a result, from the binarized ground truth mask, the best one can do is to recover one of the $n!$ permuted ground truth lesion masks. The lesion index order is not relevant for our purposes, since all the permuted lesion masks distinguish the same set of 3D lesions.

To study the binary mask parsing quality in a general setting, consider the CT volume with ground truth $\mathcal{L}$. After analyzing this volume, a lesion segmentation algorithm produces a binary lesion mask. Let C be one of its 3D connected components. We denote the set of ground truth lesions which intersects with C by

$$\mathcal{L}^C = \{L_i \mid L_i \cap C \neq \emptyset, 1 \leq i \leq n\}. \tag{3}$$

We call C a *confluent detection* if $|\mathcal{L}^C| > 1$. It is an *isolated detection* if $|\mathcal{L}^C| = 1$. Otherwise i.e. $|\mathcal{L}^C| = 0$, it is a *false detection.* We now introduce our parsing quality metric.

Definition 1. *Consider a fixed pair $(C, \mathcal{L}^C)$. Let $\mathcal{P} = \{P_j \mid j = 1, \ldots, m\}$ be a partition of C. We measure its quality by*

$$q(\mathcal{P}) = \max_{\phi} \sum_{(L,P) \in \mathcal{L}^C \times \mathcal{P}} d(L, P)\phi(L, P) \tag{4}$$

where $d(L, P)$ denotes the dice coefficient $2|L \cap P|/(|L| + |P|)$ and ϕ a one-to-one mapping such that $\phi(L, P)$ equals 1 if ϕ matches the pair (L, P) and 0 otherwise.

Clearly, the metric is positive valued. A higher metric value indicates a better parsing quality. Specifically, for an isolated detection, there can be one and only one matched pair. As a result, the metric simply reduces to the pair's dice coefficient. For a confluent detection C, a partition $\mathcal{P}$ has a higher metric value if it allows to match more ground truth lesions in $\mathcal{L}^C$ with greater pairwise dice coefficients. Finally, the metric value is zero if and only if C is a false detection. Since it is the lesion segmentation algorithm that is responsible for false detections, we ignore them for the parsing quality evaluation.

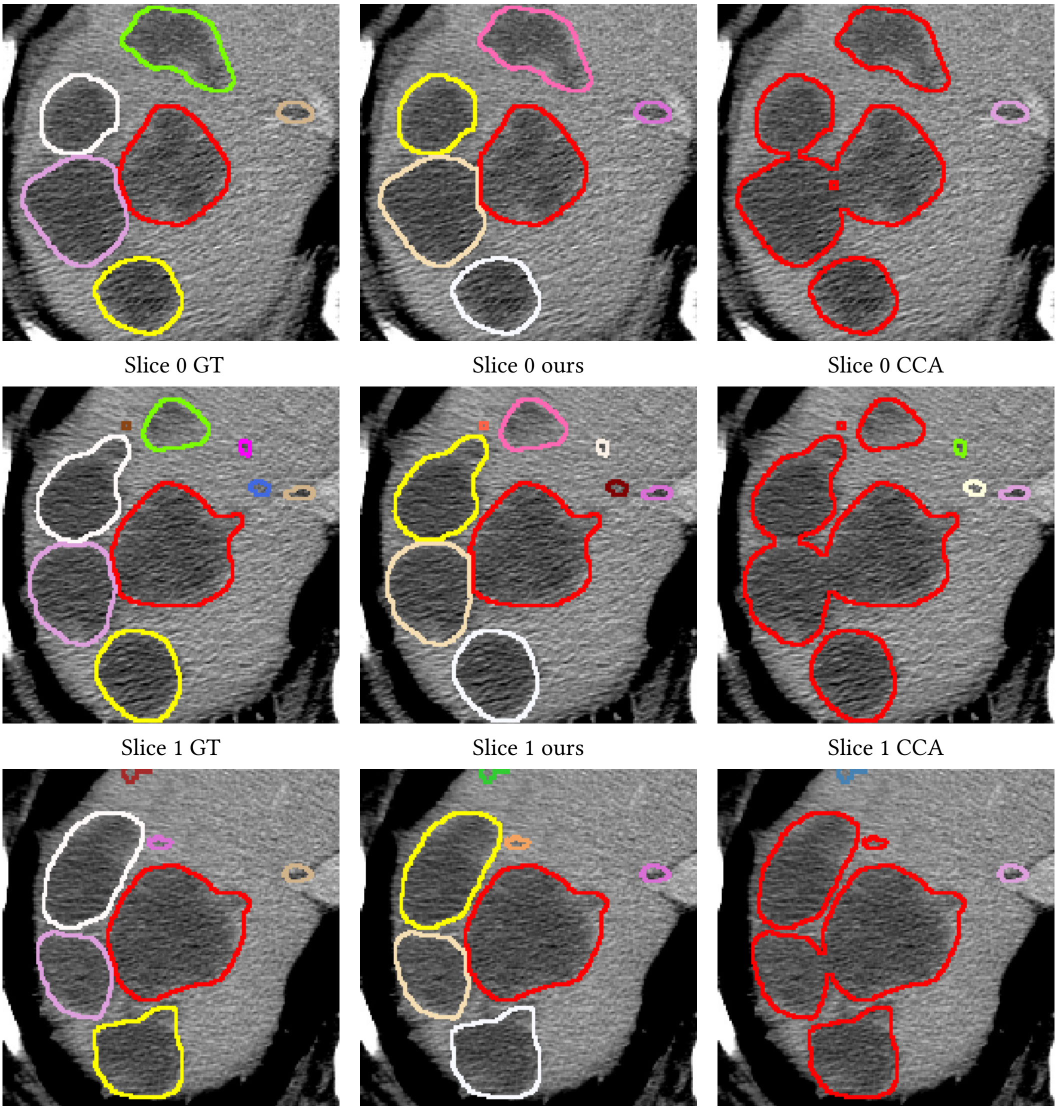
The metric can be computed numerically because Eq.(4) is a linear sum assignment problem and has efficient solutions [8, 9]. Note also that, due to its search for an optimal one-to-one mapping, the metric value is invariant to the lesion index permutation in both $\mathcal{L}$ and $\mathcal{P}$.

To compare the performance of our algorithm against that of CCA, we propose to separate isolated and confluent detections. It is because CCA generally does well on isolated detections. However, it invariably makes mistakes on confluent detections. To quantify their relative performance on a given detection, we compute the quality metric for both our algorithm and CCA and refer to their ratio as the *parsing quality improvement.* Our algorithm outperforms CCA when the improvement is greater than 1.

3.2 Results

We call a lesion segmentation model the oracle if it always outputs binarized ground truths. Since a real lesion segmentation model is trained to imitate the oracle, we first used our algorithm to parse the binarized ground truths. With the oracle, an isolated detection is the same as an isolated lesion whereas a confluent detection is a connected component consisting of more than one confluent lesions.

In our test data, in addition to 2410 isolated lesions, there are 198 oracle confluent detections which collectively account for 765 confluent lesions. We processed the confluent detections and computed their parsing quality improvement. According to their cumulative distribution function (CDF) in Fig.5(a), around 79 percent are strictly greater than 1, which means that our algorithm strictly outperformed CCA on the corresponding confluent detections. We show one such example in Fig.4. For the remaining confluent detections, our algorithm erroneously agreed with CCA because it either failed to distinguish 2D lesions in its slice-wise mask splitting step (i.e. Section 2.2.1) or



Slice 0 GT

Slice 0 ours

Slice 0 CCA

Slice 1 GT

Slice 1 ours

Slice 1 CCA

Slice 2 GT

Slice 2 ours

Slice 2 CCA

Fig. 4. Three consecutive cross-sections of a few ground truth (GT) 3D lesions in different colors are shown in the left column. We parsed the corresponding binarized GT lesion mask with our algorithm (middle column) and CCA (right column). Our algorithm succeeded in distinguishing all the confluent lesions. CCA failed to do so. Specifically, CCA lumped seven GT lesions into a single connected component because they have voxel-to-voxel connections either in the same slice or across adjacent slices.

mistakenly grouped separate lesion cross-sections on adjacent slices in the z-wise connection step (i.e. Section 2.2.2). See Fig.6(e)(f) for an example.

For oracle isolated detections, the parsing quality improvement is the same as our algorithm's parsing quality. It is because for such detections, CCA's parsing quality is always equal to 1 as CCA keeps them as they are. We computed the improvements for all the isolated lesions. From their CDF in Fig.5(b), we deduce that our algorithm made mistakes on 3.7 percent of these lesions. Most of them possess irregular cross-sectional shapes, thereby inducing erroneous in-plane splits by our algorithm. See Fig.6(a)(b)(c)(d) for an illustration.

In order to see whether these results carry over to the binary masks from a real lesion segmentation model, we repeated the same experiment with a binary lesion segmentation denseNet [7] trained on our training data. For the isolated detections, our algorithm and CCA again yielded similar parsing performance. See the CDF of their parsing quality improvements in Fig.5(d). Unlike with the oracle, here the isolated detections generally do not coincide with their matched ground truth lesions, which allowed the parsing quality improvement to exceed 1. For the confluent detections, the segmentation model did not always produce tight masks. For example, for the two ground truth confluent lesions marked in Fig.6(g), it created a loose segmentation mask in the shape of an ellipse i.e. Fig.6(h). As a result, our algorithm failed to distinguish them. Nonetheless, as shown by the CDF of the quality improvements in Fig.5(c), it still did better than CCA on about 70 percent of the confluent detections.

4 CONCLUSION

In this paper, we have presented a novel scheme for efficiently parsing a 3D binary lesion segmentation mask into 3D lesion instances. We have also introduced a parsing quality metric to demonstrate that our algorithm is better at parsing confluent detections than 3D connected component analysis.

REFERENCES

[1] Serge Beucher and Fernand Meyer. 1993. The morphological approach to segmentation: the watershed transformation. *Mathematical morphology in image processing* 34 (1993), 433–481.

[2] Patrick Bilic, Patrick Ferdinand Christ, Eugene Vorontsov, Grzegorz Chlebus, Hao Chen, Qi Dou, Chi-Wing Fu, Xiao Han, Pheng-Ann Heng, Jürgen Hesser, Samuel Kadoury, Tomasz K. Konopczynski, Miao Le, Chunming Li, Xiaomeng Li, Jana Lipková, John S. Lowengrub, Hans Meine, Jan Hendrik Moltz, Chris Pal, Marie Piraud, Xiaojuan Qi, Jin Qi, Markus Rempfler, Karsten Roth, Andrea Schenk, Anjany Sekuboyina, Ping Zhou, Christian Hülsemeyer, Marcel Beetz, Florian Ettlinger, Felix Grün, Georgios Kaissis, Fabian Lohöfer, Rickmer Braren, Julian Holch, Felix Hofmann, Wieland H. Sommer, Volker Heinemann, Colin Jacobs, Gabriel Efrain Humpire Mamani, Bram van Ginneken, Gabriel Chartrand, An Tang, Michal Drozdzal, Avi Ben-Cohen, Eyal Klang, Michal Marianne Amitai, Eli Konen, Hayit Greenspan, Johan Moreau, Alexandre Hostettler, Luc Soler, Refael Vivanti, Adi Szeskin, Naama Lev-Cohain, Jacob Sosna, Leo Joskowicz, and Bjoern H. Menze. 2019. The Liver Tumor Segmentation Benchmark (LiTS). *CoRR* abs/1901.04056 (2019). arXiv:1901.04056 http://arxiv.org/abs/1901.04056

[3] Grzegorz Chlebus, Andrea Schenk, Jan Moltz, Bram Ginneken, Hans Meine, and Horst Hahn. 2018. Automatic liver tumor segmentation in CT with fully convolutional neural networks and object-based postprocessing. *Scientific Reports* 8 (10 2018). https://doi.org/10.1038/s41598-018-33860-7

[4] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2009. *Introduction to algorithms.* MIT press.

[5] Vicente Grau, AUJ Mewes, M Alcaniz, Ron Kikinis, and Simon K Warfield. 2004. Improved watershed transform for medical image segmentation using prior information. *IEEE transactions on medical imaging* 23, 4 (2004), 447–458.

[6] Qiangguo Jin, Zhaopeng Meng, Changming Sun, Hui Cui, and Ran Su. 2020. RA-UNet: A hybrid deep attention-aware network to extract liver and tumor in CT scans. *Frontiers in Bioengineering and Biotechnology* (2020), 1471.

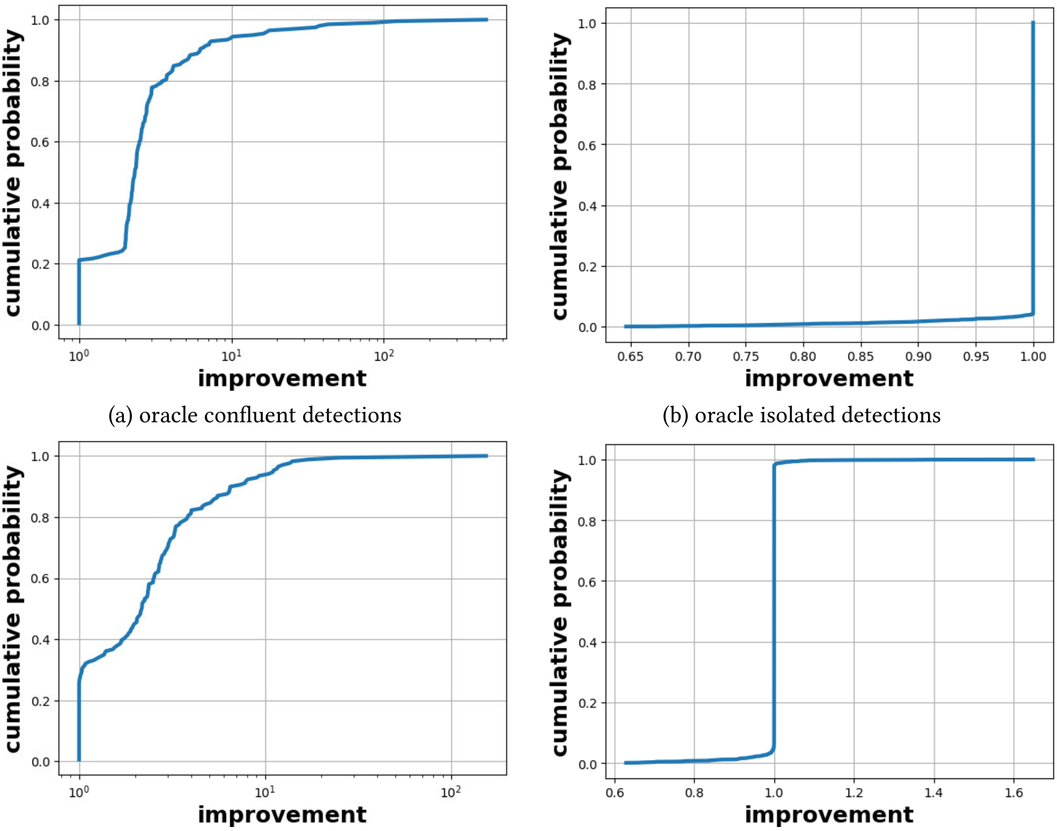
[7] Xiaomeng Li, Hao Chen, Xiaojuan Qi, Qi Dou, Chi-Wing Fu, and Pheng-Ann Heng. 2018. H-DenseUNet: hybrid densely connected UNet for liver and tumor segmentation from CT volumes. *IEEE transactions on medical imaging* 37, 12 (2018), 2663–2674.

[8] Jiri Matousek and Bernd Gärtner. 2006. *Understanding and using linear programming.* Springer Science & Business Media.

[9] James Munkres. 1957. Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics* 5, 1 (1957), 32–38.

[10] Lei Song, Haoqian Wang, and Z Jane Wang. 2021. Bridging the gap between 2D and 3D contexts in CT volume for liver and tumor segmentation. *IEEE Journal of Biomedical and Health Informatics* 25, 9 (2021), 3450–3459.

[11] Youbao Tang, Jinzheng Cai, Ke Yan, Lingyun Huang, Guotong Xie, Jing Xiao, Jingjing Lu, Gigin Lin, and Le Lu. 2021. Weakly-supervised universal lesion segmentation with regional level set loss. In *International Conference on Medical*



(a) oracle confluent detections

(b) oracle isolated detections

(c) algorithm confluent detections

(d) algorithm isolated detections

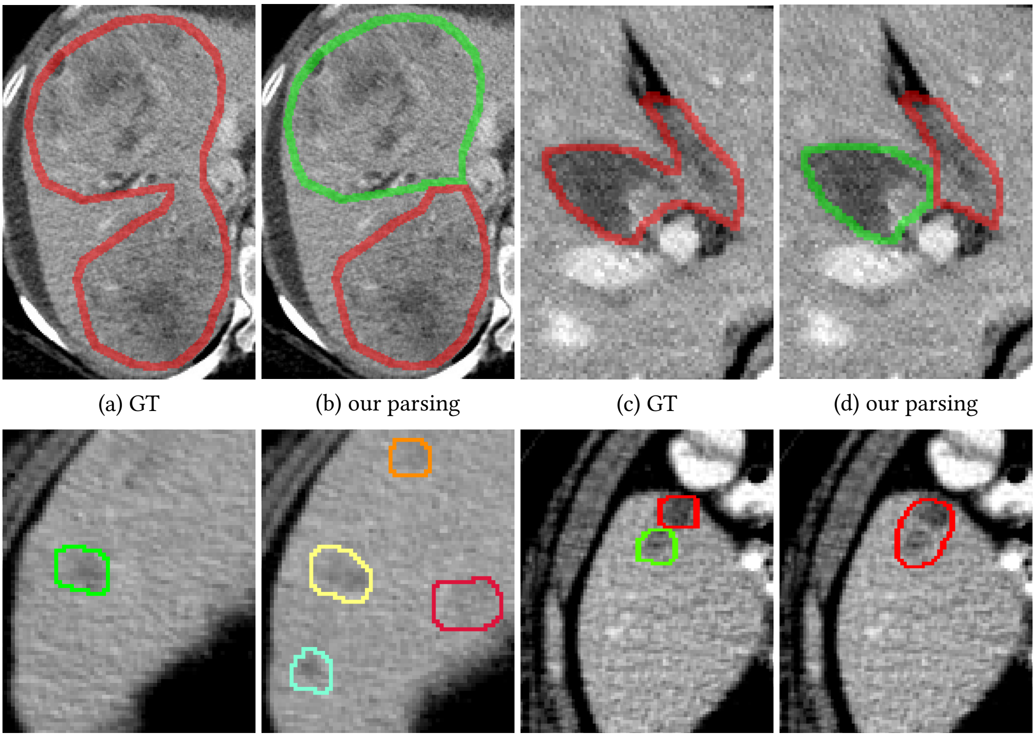
Fig. 5. Performance comparison of our algorithm and CCA on isolated and confluent detections. (a) For the oracle confluent detections, our algorithm is strictly better than CCA on around 79 percent of them. (c) On the binary masks produced by a real lesion segmentation algorithm, this percentage remains at about 70 percent. On the other hand, for the isolated detections by both (b) the oracle and (d) a real segmentation algorithm, our algorithm and CCA yield similar performances since most of the quality improvements are equal to 1.

Image Computing and Computer-Assisted Intervention. Springer, 515–525.

[12] Youbao Tang, Yuxing Tang, Yingying Zhu, Jing Xiao, and Ronald M Summers. 2020. E2Net: An Edge Enhanced Network for Accurate Liver and Tumor Segmentation on CT Scans. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 512–522.

[13] Jianpeng Zhang, Yutong Xie, Yong Xia, and Chunhua Shen. 2021. DoDNet: Learning to segment multi-organ and tumors from multiple partially labeled datasets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1195–1204.

[14] Jianpeng Zhang, Yutong Xie, Pingping Zhang, Hao Chen, Yong Xia, and Chunhua Shen. 2019. Light-Weight Hybrid Convolutional Network for Liver Tumor Segmentation.. In *IJCAI*, Vol. 19. 4271–4277.



(a) GT (b) our parsing (c) GT (d) our parsing

(e) GT (f) GT (g) GT (h) loose mask

Fig. 6. Failure cases. (a)(c) are the ground truth lesion cross-sections. CCA did not modify them. Our algorithm divided them to produce circular sub-regions, leading respectively to (b)(d). In another scenario, across two adjacent slices (e)(f), our algorithm connected two lesion cross-sections (green and yellow) because they are located at similar in-plane positions and have similar area. However they belong to separate lesions according to the ground truth. Finally, for the two ground truth lesions in (g), a real segmentation algorithm created a loose segmentation mask in the shape of an ellipse (h). Our parsing algorithm did not split the mask, resulting in an error.