PlacentaNet: Automatic Morphological Characterization of Placenta Photos with Deep Learning[⋆]

Yukun Chen[1], Chenyan Wu[1], Zhuomin Zhang[1], Jeffery A. Goldstein[2], Alison D. Gernand[1], and James Z. Wang[1⋆⋆]

[1] The Pennsylvania State University, University Park, Pennsylvania, USA
[2] Northwestern Memorial Hospital, Chicago, Illinois, USA

Abstract. Analysis of the placenta is extremely useful for evaluating health risks of the mother and baby after delivery. In this paper, we tackle the problem of automatic morphological characterization of placentas, including the tasks of placenta image segmentation, umbilical cord insertion point localization, and maternal/fetal side classification. We curated an existing dataset consisting of around 1,000 placenta images taken at Northwestern Memorial Hospital, together with their pixel-level segmentation map. We propose a novel pipeline, PlacentaNet, which consists of three encoder-decoder convolutional neural networks with a shared encoder, to address these morphological characterization tasks by employing a transfer learning training strategy. We evaluated its effectiveness using the curated dataset as well as the pathology reports in the medical record. The system produced accurate morphological characterization, which enabled subsequent feature analysis of placentas. In particular, we show promising results for detection of retained placenta (*i.e.*, incomplete placenta) and umbilical cord insertion type categorization, both of which may possess clinical impact.

Keywords: Placenta · Convolutional neural network · Segmentation · Transfer learning

1 Introduction

The placenta is a window into the events of a pregnancy and the health of the mother and baby [12]. Yet, a very small percentage of placentas around the world are ever examined by a pathologist. Even in developed countries like the U.S., placentas are examined and characterized by a pathologist only when it is considered necessary and resources are available. Full pathological examination

[⋆] This work was supported primarily by the Bill & Melinda Gates Foundation. The computation was support by the NVIDIA Corporations GPU Grant Program. Discussions with William Tony Parks have been helpful. Celeste Beck, Dolzodmaa Davaasuren, and Leigh A. Taylor assisted in dataset curation.

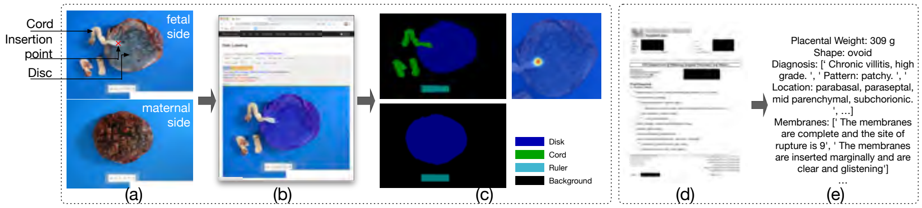
[⋆⋆] A. D. Gernand and J. Z. Wang have equal contributions.

is expensive and time consuming. In placenta examination, pathologists complete a report that contains various measurements (*e.g.*, the weight, the disc diameter) and diagnoses (*e.g.*, completeness or retained placenta, cord insertion type, shape category). These measurements and placental diagnoses are extremely useful for the short- and long-term clinical care of the mother and baby.

Automated placenta analysis based on photographic imaging can potentially allow more placentas to be examined, reduce the number of normal placentas sent for full pathological examination, and provide more accurate and timely morphological and pathological measurements or analyses. Typical photographs of the placentas capture the umbilical cord inserting into the fetal side of the disc, as well as the maternal side appearance. Two example images of placentas can be found later in Fig. 1(a). This paper focuses on a fully automated system for morphological characterization of placentas. Such systems will be the cornerstone for automated pathological analyses because segmentation of disc and cord, location of cord insertion point, and determination of fetal/maternal side are important first steps before further analyses can be done.

Related Work. Existing placenta imaging research can be roughly categorized into two types: those using microscopic images of slices of the placentas [15,6] and those using the macroscopic images of the placentas taken by cameras [17] or by MRI [1]. A comprehensive overview of both microscopic and macroscopic placenta pathology can be found in a book by Benirschke *et al.* [3]. To our knowledge, there has not been an automated approach to analyze placenta photographs. We believe such an approach has the potential to be adopted widely because it requires no specialized hardware beyond an ordinary camera or a camera phone.

In this paper, we propose a transfer learning (TL) approach to tackle the associated tasks of morphological characterization rather than employing one independent model for each task. TL promises performance gain and robustness enhancement through representation sharing for closely related tasks [10]. Specifically, we transfer the learned representation of the encoder from the segmentation task to the other two tasks, *i.e.* disc side classification and insertion point localization. Our network architecture design takes inspiration from the recent deep learning advances on classification [4], image segmentation [7,13], and key-point localization [9]. In particular, the design of our segmentation module follows the practice of concatenating feature maps in encoder with feature maps in decoder, such as performed in the U-Net [13]; and the design of our insertion point module follows the practice of regressing a Gaussian heat map, rather than using the coordinate values, as the ground truth, which has been shown to be successful in human key-point/joint localization tasks [16,3,9,11]. Tompson *et al.* first showed the importance of intermediate supervision to improving localization accuracy [9]. We take their idea in our design by considering two heat map predictions in the final loss — one from the final feature layer and one from the intermediate feature layer.



Placental Weight: 309 g
Shape: ovoid
Diagnosis: [' Chronic villitis, high grade. ', ' Pattern: patchy. ', ' Location: parabasal, paraseptal, mid parenchymal, subchorionic. ', ...]
Membranes: [' The membranes are complete and the site of rupture is 9', ' The membranes are inserted marginally and are clear and glistening']
...

Fig. 1: Data curation process. (a-c): collecting pixel-level segmentation map for cord, disc, and ruler, insertion point location, and classification of whether an image captures fetal or maternal side placenta through our web-based labeling tool. (d-e): extracting diagnoses and measurements from unidentified pathological report in PDF format.

2 The Dataset

We obtained a dataset consisting of 1, 003 placenta images, of which 430 are fetal-side images and 573 are maternal-side images[3], from Northwestern Memorial Hospital, a large urban academic medical center. We also have the complete pathology report for each placenta, written in natural language by the pathologist who originally examined the placenta. Pathology classification is standardized and pathologist are perinatal experts. Fig. 1 shows our data curation process. We developed a web-based tool (Fig. 1(b)) to collect i) the pixel-wise segmentation maps, ii) the side-type label as fetal side or maternal side, and iii) the cord insertion point (only for fetal side, visualized as a Gaussian heat map centered at the marked coordinate in (Fig. 1(c))) so that multiple trained labelers can annotate this dataset concurrently. We also extract diagnoses from the pathology reports.

We divide the dataset into training and testing sets with the ratio of $0.8 : 0.2$. Because the insertion point can only be observed from the fetal side, we only use the 430 fetal-side images for insertion point prediction, with the same training-testing ratio as aforementioned.

3 The Method

The proposed PlacentaNet model, as illustrated in Fig. 2, consists of an `Encoder` for feature pyramid extraction (blue), which is shared among all tasks, a fully convolutional `SegDecoder` for placenta image segmentation on both fetal- and maternal-side images (red), a `Classification Subnet` for fetal/maternal-side classification (purple), and a fully convolutional `IPDecoder` for insertion point localization.

[3] The numbers of fetal-side and maternal-side images are uneven because some of the collected images did not meet our image quality standard (*e.g.* disc occluded by irrelevant object such as scissors) and we had to discard them from the dataset. We plan to release our dataset in the future after substantial expansion.

`Encoder` as feature pyramid extractor. The `Encoder` takes a placenta image $\mathbf{x}$ (either the fetal side or the maternal side) as the input and outputs a pyramid of feature maps $\{\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3, \mathbf{f}_4, \mathbf{f}_5\}$ (represented as blue rectangles). Depending on the tasks, all or part of the feature maps are used by further task modules. Specifically, `SegDecoder` takes $\{\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3, \mathbf{f}_4, \mathbf{f}_5\}$ as input; `Classification Subnet` takes $\{\mathbf{f}_5\}$ as input; and `IPDecoder` takes $\{\mathbf{f}_3, \mathbf{f}_4, \mathbf{f}_5\}$ as input. The Conv-1 and Conv-2 blocks both consist of a Conv-BatchNorm-Relu layer. The difference, however, is that the Conv layer in Conv-1 block has stride 1, while the Conv layer in Conv-2 block has stride 2. The Res conv blocks are residual blocks with two convolutional layers with stride 2 and 1, respectively, and the same kernel size 3×3, each of which spatially downsamples the input feature maps to half of its size and doubles the number of feature channels. The residual structure has been shown especially helpful for training deep architectures by He *et al.* [4].

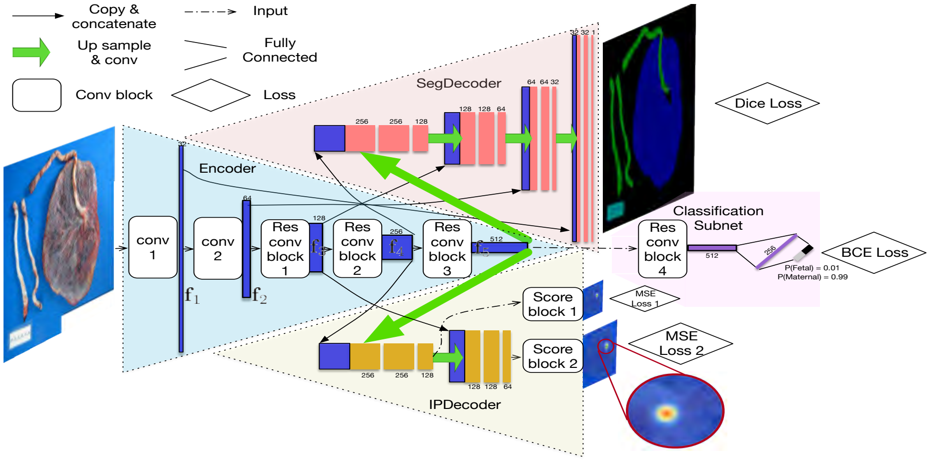


Fig. 2: The architecture of PlacentaNet: a multi-task convolutional neural network for placenta image segmentation, cord insertion point localization, and placenta disc side classification. “Up sample & Conv” is implemented by a transposed convolution layer. “Res conv blocks” are residual blocks with two convolutional layers with stride 2 and 1, respectively, and the same kernel size 3×3. “Score blocks” are convolutional layers with kernel size 1×1 and the number of output channel 1. The soft-max layers are omitted. We use dice loss, BCE loss and MSE loss for the segmentation, classification, and insertion point localization, respectively.

`SegDecoder` for segmentation. Our `SegDecoder` module consists of four expanding fully convolutional blocks, each of which takes the concatenation of a copy of the corresponding feature map $\mathbf{f}_i, i \in \{1, 2, 3, 4\}$, and transposes a convoluted (up-scaling factor 2) output feature map of the last layer. Finally, we apply soft-max to predict the probability of pixel (i, j) being of class k, denoted as $\mathbf{p}(i, j, k)$. To overcome the problem of highly imbalanced number of pixels for different categories, we use dice loss [8] instead of the common cross entropy loss. Since we have four classes rather than two classes in [8], we adjust the dice loss to suit the 4-class scenario:

$$L^{\text{seg}} = 1 - \frac{\sum_{i,j} \sum_{k=0}^{3} \mathbf{p}(i, j, k) \cdot \mathbf{g}(i, j, k)}{\sum_{i,j} \sum_{k=0}^{3} (\mathbf{p}(i, j, k) + \mathbf{g}(i, j, k))} \ , \tag{1}$$

where i, j run over the row and column indexes of an image, respectively; $\mathbf{p}(i, j, k)$ and $\mathbf{g}(i, j, k)$ denote the predicted probability of the pixel at location (i, j) and the $0/1$ ground truth of that pixel belonging to class k, respectively.

`Classification Subnet` **for fetal/maternal side classification.** Because the fetal/maternal side can be inferred from the "disc"region of a placenta alone, we crop the full placenta image $\mathbf{x}$ by a rectangle including the region of disc and resize the cropped image to 512 × 512 pixels as the input to the `Encoder`, which we denote as $\mathbf{x}_c$. The cropping is based on the ground truth segmentation map during training and on the predicted segmentation map at inference. Our `Classification Subnet` consists of a Res conv block, two fully connected layers, and a soft-max layer. At the end, a binary cross entropy (BCE) loss is applied to supervise the network.

`IPDecoder` **for insertion point localization.** Because the insertion point is always located within or adjacent to the "disc" region, we use cropped disc region image $\mathbf{x}_c$, by the same way as we perform cropping in `Classification Subnet`, as the input to the `Encoder`. Our `IPDecoder` is also fully convolutional and consists of two expanding fully convolutional blocks, the structure of which are the same as in the first two convolutional blocks in `SegDecoder`. The similarity of `IPDecoder`'s structure with `SegDecoder`'s helps us to ensure that the shared encoder representation could also be readily utilized here. Inspired by the success of intermediate supervision [9], we predict the insertion point localization heat map after each expanding convolutional block by a convolutional layer with kernel size 1 × 1 (denoted as "Score block" in Fig. 2) and use the MSE loss to measure the prediction error:

$$L_k^{\text{ip}} = \sum_{i,j} ||\mathbf{h}(i, j) - \hat{\mathbf{h}}(i, j)||^2, \ \ k \in \{1, 2\} \ , \tag{2}$$

where $\mathbf{h}(i, j)$ and $\hat{\mathbf{h}}(i, j)$ are the ground truth (Gaussian) heat map and the predicted heat map, respectively. And the final loss for insertion point is $L^{\text{ip}} = L_1^{\text{ip}} + L_2^{\text{ip}}$. During inference, the predicted insertion point location is determined by $(i, j) = \arg\max_{i,j} \hat{\mathbf{h}}(i, j)$.

Training and Testing. We use mini-batched stochastic gradient descent (SGD) with learning rate 0.1, momentum 0.9, and weight decay 0.0005 for all training. We use a batch size of 2 for all segmentation training and a batch size of 10 for all insertion point localization and fetal/maternal side classification training. The procedures of training are as follows. We first train the `SegDecoder` + `Encoder` from scratch with parameters initialized to zero. Next, we fix the learned weights for the Encoder and train Classification Subnet and IPDecoder subsequently (in other words, the Encoder only acts as a fixed feature pyramid extractor at this stage). The rationale for making such choices is that the training for segmentation task consumes all images we have gathered and makes use of pixel-wise dense supervision, which is much less likely to lead to an overfitting problem. In contrast, the training of Classification Subnet takes binary value as ground truth for each image and the training of IPDecoder only uses around half of the whole dataset (only fetal-side images). To alleviate the lack of labels and to make the model more robust, we use common augmentation techniques including random rotation (±30°), and horizontal and vertical flipping for all training images.

Implementation. We implemented the proposed pipeline in PyTorch and ran experiments on an NVIDIA TITAN Xp GPU. For segmentation training, all images are first resized to 768 × 1024, which is of the same aspect ratio as the original placenta images. For insertion point localization and fetal/maternal side classification training, we resize all cropped "disc" region images to 512 × 512, which is natural because the cropped "disc" regions often have a bounding box close to a square.

4 Experiments and Evaluation

Segmentation. We compared our approach with two fully convolutional encoder-decoder architectures, the U-Net [13] and the SegNet [2]. The results are shown in Fig. 3(a-d). We report the segmentation performance using standard segmentation metrics pixel accuracy, mean accuracy, and mean IoU. In Fig. 3 (b, c, and d), we compare pixel-wise prediction confusion matrices of our approach, U-Net, and Segnet, respectively, which reflects more detail about segmentation performance for different categories. We also show a few segmentation examples in Fig. 3(e) for qualitative comparison. Our approach yields the best segmentation results, especially for differentiating the cord and the ruler classes.

Fetal/Maternal Side Classification. We achieve an overall fetal/maternal side classification accuracy of 97.51% on our test set. Without the shared encoder representation, we can only achieve 95.52% by training Encoder + Classification Subnet from scratch. We also compare their confusion matrices in Fig. 1 in the supplementary material.

Insertion Point Localization. We use Percentage of Correct Keypoints (PCK) as the evaluation metric. PCK measures the percentage of the predictions falling within a circle of certain radius centered at the ground truth location. We compare our approach (both with and without shared encoder weights) to the Hourglass model (with number of stacks 1 and 2), which shows competitive results in human keypoint localization [9]. Fig. 3(f) shows the PCK curves, with the x-axis being the radius normalized by the diameter of the placenta. Each curve in Fig. 3(f) is the average of the results for five models trained with different seeds, and the light-colored band around each curve (viewable when the figure is enlarged) shows the standard deviation of the results. Our approach with shared

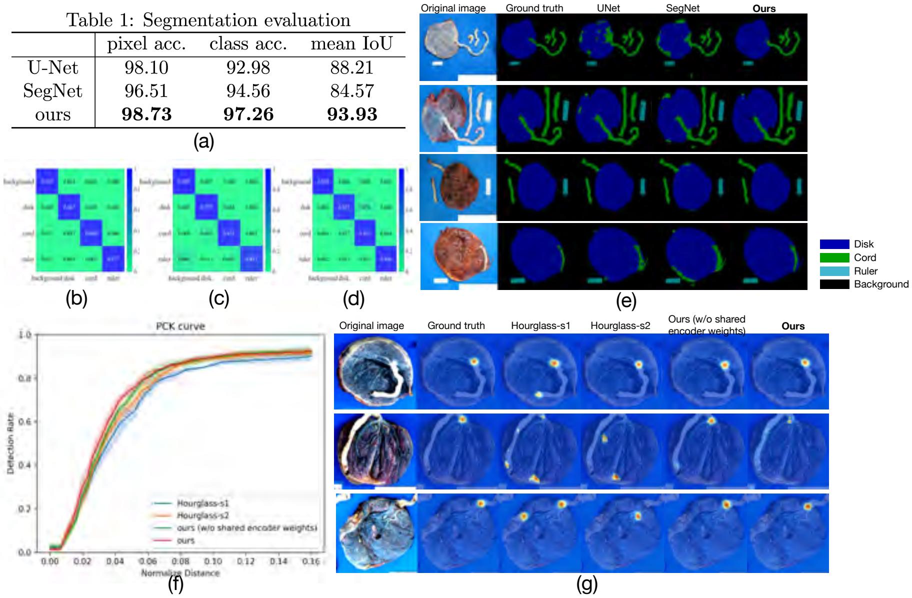


Table 1: Segmentation evaluation

	pixel acc.	class acc.	mean IoU
U-Net	98.10	92.98	88.21
SegNet	96.51	94.56	84.57
ours	**98.73**	**97.26**	**93.93**

(a)

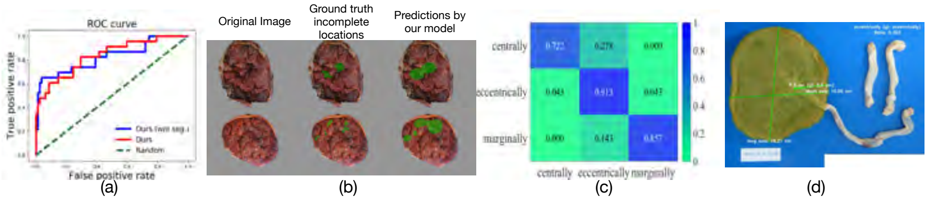
Fig. 3: Evaluation results. (a) Segmentation evaluation accuracy. (b-d) Confusion matrices of our approach, U-Net, and SegNet, respectively. (e) Example segmentation results. We show both fetal-side results (top two rows) and maternal-side results (bottom two rows). (f) Quantitative evaluation of insertion point localization with PCK curves. (g) Examples of insertion point heat map prediction.

`Encoder` consistently gives the best results, especially when the normalized distance is from 0.2 to 0.6. We show a few qualitative examples of the insertion point heat maps predicted by each model, along with the ground truth (Fig. 3(g)).

Placenta Feature Analysis. The predictions of PlacentaNet enable us to conduct automatic placenta feature analysis by subsequent models/procedures.

(1) Detection of retained placenta. Retained placenta is a cause of postpartum hemorrhage and, if prolonged, it can serve as a nidus for infection [14]. Pathologists judge if there could be retained placenta by carefully inspecting the maternal surface of a placenta's disc. We identified 119 out of 573 maternal side placenta images in our dataset with possible "retained placenta" based on the pathology reports and we asked a perinatal pathologist (coauthor) to annotate where the possible missing parts are for each of them. We trained two neural networks for this task, for classification and localization, respectively, and both achieved promising results. We show the ROC curve of the classification network in Fig. 4(a) and example localization results along with the ground truth in Fig. 4(b). (More localization results are in supplementary material Fig. 3).

(2) Umbilical cord insertion type categorization. Abnormal cord insertion is a feature of fetal vascular malperfusion [5]. Based on the segmentation, the predicted insertion point location, and the scale we extracted from the ruler, we



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

Fig. 4: (a) ROC curve for retained placenta classification. AUC for red(blue) curve is 0.836(0.827). (b) Example of retained placenta problem localization. (c) The confusion matrix for insertion type categorization. (d) Example of insertion point type prediction.

can measure the distance from the insertion point to the nearest margin of the disc, and the lengths of the long and short axes of the disc (all in centimeters). Further, we classify the cord insertion type into "centrally", "eccentrically", and "marginally", based on the ratio between *the distance from the insertion point to its closest disc margin* and *the average between the lengths of the long and short axes.* We achieve an overall 88% test accuracy. We show the classification confusion matrix in Fig. 4(c). One qualitative example of our prediction is shown in Fig. 4(d). Detailed procedures and more qualitative examples of measurement and classification are in supplementary material Figs. 1 and 2.

5 Conclusions and Future Work

We proposed a novel, compact multi-head encoder-decoder CNN to jointly solve placenta morphological characterization tasks. We showed that our approach can achieve better performance than competitive baselines for each task. We showed that the representation learned from segmentation task could benefit insertion point localization and fetal/maternal side classification task. In the future, it would be interesting to explore if these tasks could mutually benefit each other. The use of this method in automated prediction of pathological indicators is the next direction we will pursue.

References

1. Alansary, A., et al.: Fast fully automatic segmentation of the human placenta from motion corrupted mri. In: MICCAI. pp. 589–597. Springer (2016)
2. Badrinarayanan, V., et al.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE T-PAMI **39**(12), 2481–2495 (2017)
3. Benirschke, K., Burton, G.J., Baergen, R.N.: Pathology of the Human Placenta. Springer, 6 edn. (2012)
4. He, K., et al.: Deep residual learning for image recognition. In: IEEE CVPR. pp. 770–778 (2016)

5. Khong, T.Y., et al.: Sampling and definitions of placental lesions: Amsterdam placental workshop group consensus statement. Archives of Pathology & Laboratory Medicine **140**(7), 698–713 (2016)
6. Kidron, D., et al.: Automated image analysis of placental villi and syncytial knots in histological sections. Placenta **53**, 113–118 (2017)
7. Long, J., et al.: Fully convolutional networks for semantic segmentation. In: IEEE CVPR. pp. 3431–3440 (2015)
8. Milletari, F., et al.: V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: International Conf. on 3D Vision (3DV). pp. 565–571. IEEE (2016)
9. Newell, A., et al.: Stacked hourglass networks for human pose estimation. In: ECCV. pp. 483–499. Springer (2016)
10. Pan, J., et al.: A survey on transfer learning. IEEE TKDE **22**(10), 1345–1359 (2009)
11. Payer, C., et al.: Integrating spatial configuration into heatmap regression based cnns for landmark localization. Medical Image Analysis **54**, 207–219 (2019)
12. Roberts, D.J., et al.: Placental pathology, a survival guide. Archives of Pathology & Laboratory Medicine **132**(4), 641–651 (2008)
13. Ronneberger, O., et al.: U-Net: Convolutional networks for biomedical image segmentation. In: MICCAI. pp. 234–241. Springer (2015)
14. Silver, R.: Abnormal placentation: placenta previa, vasa previa, and placenta accreta. Obstetrics & Gynecology **126**(3), 654–668 (2015)
15. Thomas, K.A., et al.: Unsupervised segmentation for inflammation detection in histopathology images. In: International Conf. on Image and Signal Processing. pp. 541–549. Springer (2010)
16. Tompson, J., et al.: Joint training of a convolutional network and a graphical model for human pose estimation. In: NIPS. pp. 1799–1807 (2014)
17. Yampolsky, M., et al.: Centrality of the umbilical cord insertion in a human placenta influences the placental efficiency. Placenta **30**(12), 1058–1064 (2009)

Supplementary Materials for “PlacentaNet: Automatic Mophological Characterization of Placenta with Deep Learning”

Yukun Chen[1], Chenyan Wu[1], Zhuomin Zhang[1], Jeffery A. Goldstein[2], Alison D. Gernand[1], and James Z. Wang[1]*

[1] The Pennsylvania State University, University Park, Pennsylvania, USA
[2] Northwestern Memorial Hospital, Chicago, Illinois, USA

1 Definition of the Metrics for Evaluating Segmentation

Suppose we have counted how many pixels are predicted to class j but with their ground truth being class i (for every $i, j \in \{0, 1, \ldots, k-1\}$, k is the number of classes) and we store it as the term $\mathbf{C}_{i,j}$ in a $k \times k$ matrix $\mathbf{C}$. We also denote the (ground truth) total number of pixels for class i as T_i. It's easy to see that $T_i = \sum_{j=0}^{k-1} \mathbf{C}_{i,j}$. The pixel accuracy, mean class accuracy, and mean IoU are then defined as follows.

Pixel accuracy:

$$\frac{\sum_{i=0}^{k-1} \mathbf{C}_{i,i}}{\sum_{i=0}^{k-1} T_i}$$

Mean class accuracy:

$$\frac{1}{k} \frac{\sum_{i=0}^{k-1} \mathbf{C}_{i,i}}{T_i}$$

Mean IoU:

$$\frac{1}{k} \sum_{i=0}^{k-1} \frac{\mathbf{C}_{i,i}}{T_i + \sum_{j \neq i} \mathbf{C}_{i,j}}$$

2 Definition of Percentage of Correct Keypoints (PCK)

Suppose we are making predictions for n keypoints $\{\mathbf{p}_i\}_{i=1}^{n}$. And we denote the prediction for keypoint $\mathbf{p}$ as $\hat{\mathbf{p}}$. And we use $||.||_2$, *i.e.* the L-2 Euclidean distance, to measure the error of the prediction $\hat{\mathbf{p}}$ from the ground truth $\mathbf{p}$. Then the formal definition for PCK at normalized distance x ($x \in [0, 1]$) is:

$$PCK@x = \frac{|\{\mathbf{p} : \frac{\sqrt{||\hat{\mathbf{p}} - \mathbf{p}||_2}}{d} < x \wedge \mathbf{p} \in \{\mathbf{p}_i\}_{i=1}^{n}\}|}{n} .$$

In our paper, we choose the diameter of the disc as the normalizing factor d.

* A. D. Gernand and J. Z. Wang have equal contributions.

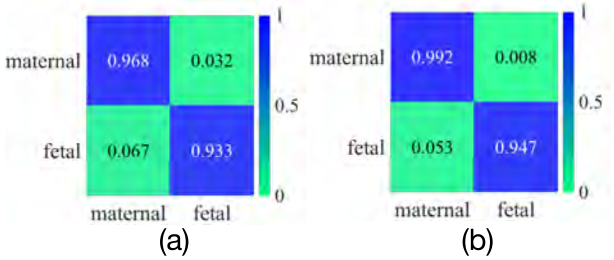
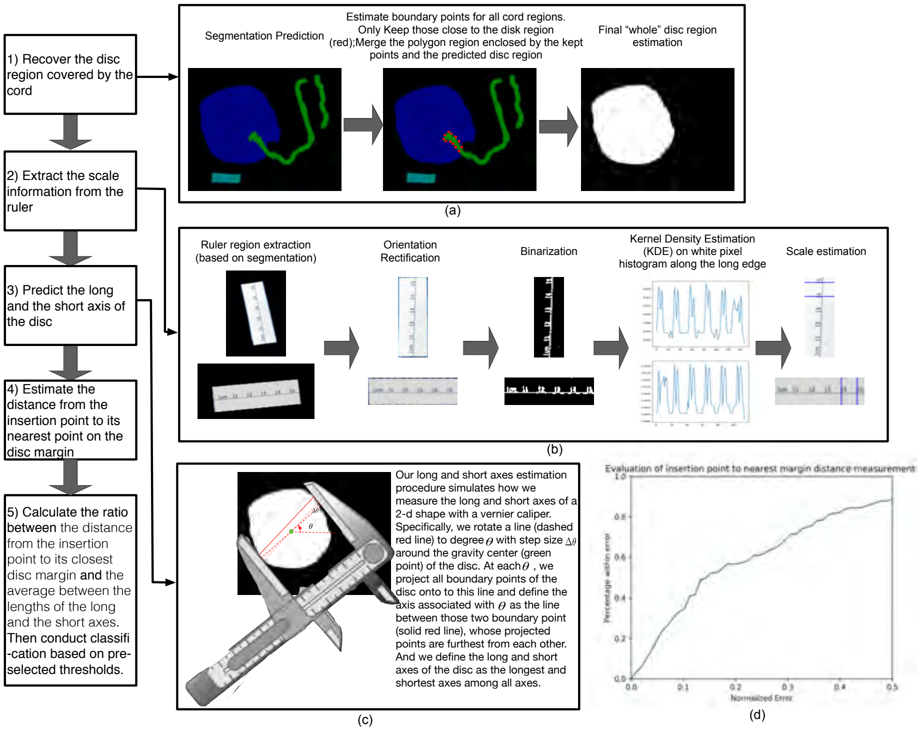


Fig. 1: Fetal/maternal side classification confusion matrices comparison. (a) Without shared encoder weights. (b) Ours.


1) Recover the disc region covered by the cord

2) Extract the scale information from the ruler

3) Predict the long and the short axis of the disc

4) Estimate the distance from the insertion point to its nearest point on the disc margin

5) Calculate the ratio between the distance from the insertion point to its closest disc margin and the average between the lengths of the long and the short axes. Then conduct classifi -cation based on pre-selected thresholds.

Estimate boundary points for all cord regions. Only Keep those close to the disk region (red);Merge the polygon region enclosed by the kept points and the predicted disc region

Our long and short axes estimation procedure simulates how we measure the long and short axes of a 2-d shape with a vernier caliper. Specifically, we rotate a line (dashed red line) to degree θ with step size $\Delta\theta$ around the gravity center (green point) of the disc. At each θ , we project all boundary points of the disc onto to this line and define the axis associated with θ as the line between those two boundary point (solid red line), whose projected points are furthest from each other. And we define the long and short axes of the disc as the longest and shortest axes among all axes.

Segmentation Prediction
Final “whole” disc region estimation
(a)
Ruler region extraction (based on segmentation)
Orientation Rectification
Binarization
Kernel Density Estimation (KDE) on white pixel histogram along the long edge
Scale estimation
(b)
$\Delta\theta$
θ
(c)
Evaluation of insertion point to nearest margin distance measurement
1.0
0.8
0.6
0.4
0.2
0.0
Percentage within error
0.0
0.1
0.2
0.3
0.4
0.5
Normalized Error
(d)

Fig. 2: The insertion type categorization process consists of steps 1) to 5). (a), (b), and (c) illustrate the detailed procedure for steps 1), 2), and 3), respectively. (d) shows the evaluation for our estimation of *the distance from the insertion point to its nearest point on the disc margin* on the test set. The x-axis represents the threshold of the normalized error (absolute error normalized by the ground truth) and the y-axis shows the percentage of our estimation, the error of which is below such threshold. The ground truth are extracted from the pathology reports. It can be seen that we have a 60% prediction accuracy if we set the threshold to 0.2.

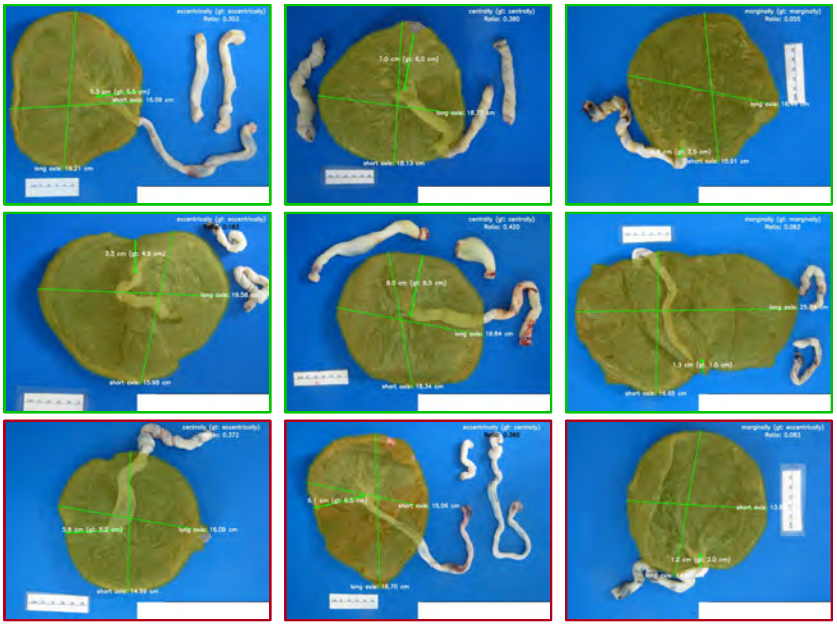


Fig. 3: Qualitative examples of insertion point type categorization. Insertion type predictions are displayed in the upper right corner of each image, along with the ground truth in brackets. The success cases are green boxed and the failed cases are red boxed. For each image, the predicted insertion point location are marked with a green dot; a transparent green mask is overlaid on the image representing the predicted "whole" disc region; a (green) line is drawn between the insertion point and its nearest point on the disc margin. The predicted length of such line is displayed next to it, along with the ground truth length extracted from the pathology report (in brackets). The predicted long and short axes are also displayed, along with their predicted length in centimeters.

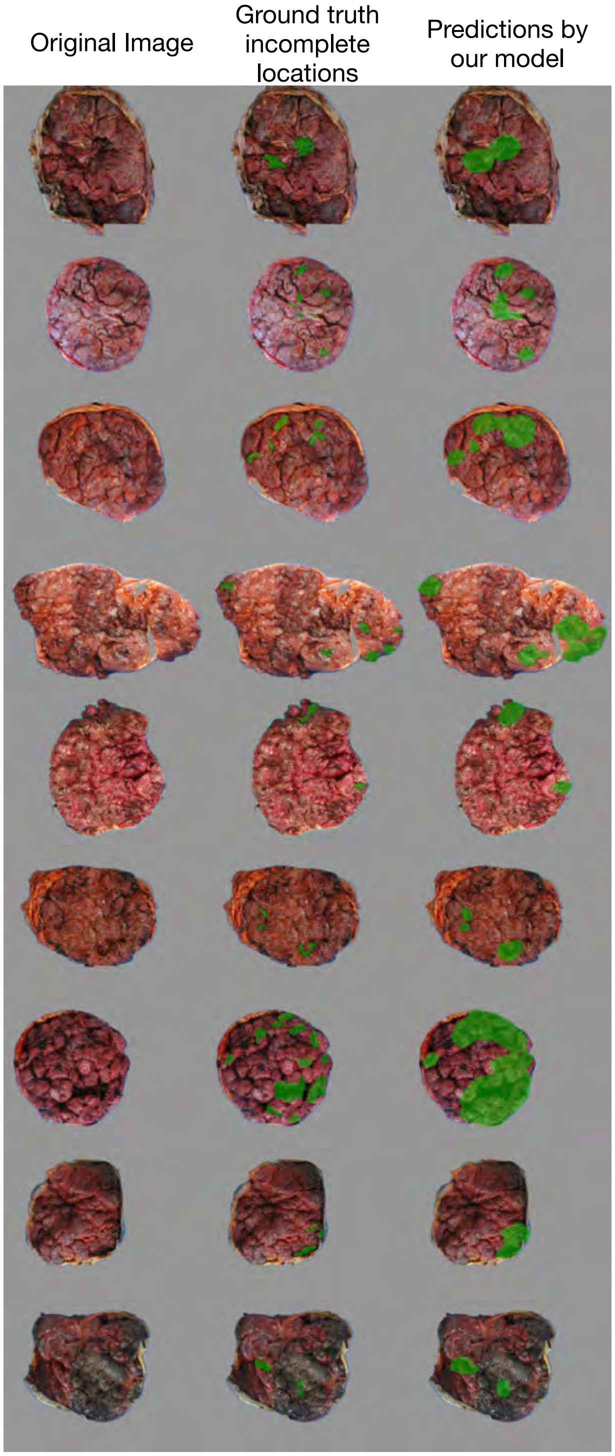


Fig. 4: Qualitative examples of our incomplete part localization predictions produced by our localization network. The localization network assumes that the input has already been predicted as having "retained placenta" by our classification network. The results are promising, but further improvement is likely when substantially more labeled training data become available.