Medical Image Analysis 47 (2018) 127–139

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

VP-Nets : Efficient automatic localization of key brain structures in 3D fetal neurosonography

Ruobing Huang*, Weidi Xie, J. Alison Noble

Institute of Biomedical Engineering, Department of Engineering Science, University of Oxford, UK

ARTICLE INFO

Article history:
Received 27 July 2017
Revised 8 April 2018
Accepted 14 April 2018
Available online 23 April 2018

Keywords:
Ultrasound
Fetal brain volume
3D Structure detection
Convolutional neural networks

ABSTRACT

Three-dimensional (3D) fetal neurosonography is used clinically to detect cerebral abnormalities and to assess growth in the developing brain. However, manual identification of key brain structures in 3D ultrasound images requires expertise to perform and even then is tedious. Inspired by how sonographers view and interact with volumes during real-time clinical scanning, we propose an efficient automatic method to simultaneously localize multiple brain structures in 3D fetal neurosonography. The proposed *View-based Projection Networks (VP-Nets)*, uses three view-based Convolutional Neural Networks (CNNs), to simplify 3D localizations by directly predicting 2D projections of the key structures onto three anatomical views.

While designed for efficient use of data and GPU memory, the proposed VP-Nets allows for full-resolution 3D prediction. We investigated parameters that influence the performance of VP-Nets, e.g. depth and number of feature channels. Moreover, we demonstrate that the model can pinpoint the structure in 3D space by visualizing the trained VP-Nets, despite only 2D supervision being provided for a single stream during training. For comparison, we implemented two other baseline solutions based on Random Forest and 3D U-Nets. In the reported experiments, VP-Nets consistently outperformed other methods on localization. To test the importance of loss function, two identical models are trained with binary corss-entropy and dice coefficient loss respectively. Our best VP-Net model achieved prediction center deviation: 1.8 ± 1.4 mm, size difference: 1.9 ± 1.5 mm, and 3D Intersection Over Union (IOU): $63.2 \pm 14.7\%$ when compared to the ground truth. To make the whole pipeline intervention free, we also implement a skull-stripping tool using 3D CNN, which achieves high segmentation accuracy. As a result, the proposed processing pipeline takes a raw ultrasound brain image as input, and output a skull-stripped image with five detected key brain structures.

1. Introduction

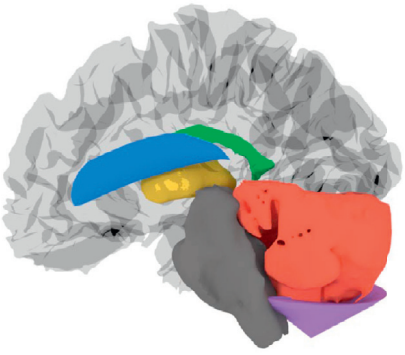
Ultrasound (US) has been a predominant imaging modality employed to examine fetal health for decades. Recent advances in 3D US technology have prompted investigation of its use in assessment of healthy and at-risk pregnancies. A standard fetal neurosonography examination involves identification and evaluation of several key brain anatomies (ISUOG et al., 2007), such as the lateral ventricles (LV), cavum septi pellucidi (CSP), thalami (Tha), cerebellum (CE), and cisterna magna (CM) (see Fig. 1a). Localizing these structures is non-trivial as: (i) US image quality is greatly affected by the presence of speckle; (ii) skull calcification reduces intracranial visibility; (iii) variations in position of US probe with respect to the fetal brain give different image appearance; (iv) and, fetal brain structures change in size and shape continuously over gestation. As a result, an automatic method to localize brain structures across a large range of gestational age (GA) is desirable to shorten the time to interpret 3D scans, e.g. assist the diagnosis of neurological conditions (Carroll et al., 2000; Archibald et al., 2001; Krain and Castellanos, 2006), assessment of fetal growth (Chang et al., 1993; Hadlock et al., 1985), and gestational age estimation (Hadlock et al., 1982; Namburete et al., 2015).

In this paper, we propose **View-based Projection Networks (VP-Nets)** an original framework to detect multiple brain structures simultaneously in 3D fetal neurosonography. Fig. 2 presents a schematic pipeline of our solution. It uses a multi-stream Convolutional Neural Networks (CNNs) to incorporate information from different anatomical views. Our main contributions are:

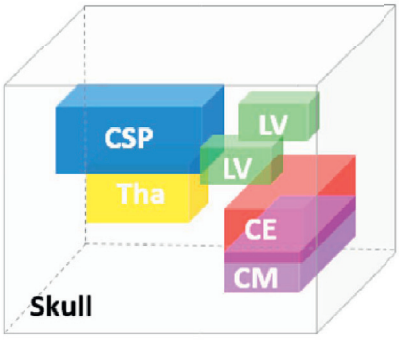
- An end-to-end approach to detect and localize key structures in 3D fetal neurosonography. The networks are trained with bounding box masks. To our knowledge, this is the first work to achieve this goal with Fully Convolutional Networks (FCNs).

* Corresponding author.
E-mail address: ruobing.huang@eng.ox.ac.uk (R. Huang).

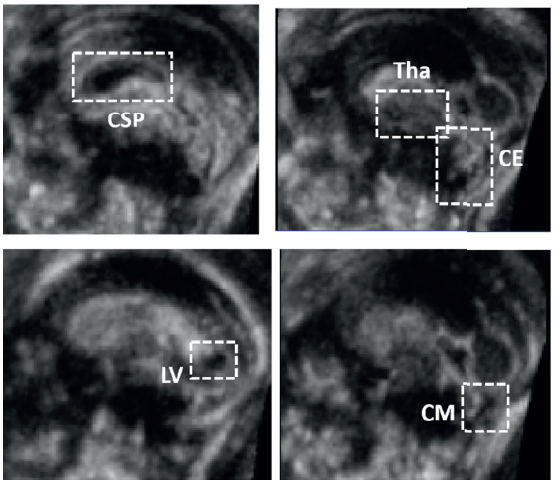
https://doi.org/10.1016/j.media.2018.04.004
1361-8415/



(a) 3D schematic of target structures



(b) Bounding boxes of the structures in 3D



(c) Structures shown in the US images

Fig. 1. (a) Schematic of the detected structures in the right-hemisphere of the brain in 3D. Each structure is plotted as: cavum septi pellucidi (CSP) in blue, thalami (Tha) in yellow, lateral ventricles (LV) in green, cerebellum (CE) in red and cisterna magna (CM) in purple. (b) 3D spatial configuration of the bounding boxes of the target brain structures. Each box is shown as a 3D cube whose colour is in accord with that in sub-figure a. Notice that there is geometric overlap between CE and CM, as well as CSP and Tha. (c) Targeted structures in sagittal planes of a US volume. The structures is bounded by white dashed boxes. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

- Economical time and memory consumption. Inference from a 3D input often imposes time and memory issues on CNN models. Our approach, however, uses projection to reduce the dimensionality and complexity of the task, thus enable 3D outputs with higher resolution than existing 3D CNN models.
- Multi-class prediction of challenging 3D volumes. In 3D fetal neurosonography, the target structures often have different local appearance (Fig. 1c) and exhibit complex spatial configurations (Fig. 1b). Some structures are closely adjacent, with potentially overlapping bounding boxes (e.g. CE and CM in Fig. 1). This further complicates the process of identifying the correct label of each pixel. VP-Nets predicts different structures separately in the last layer but sharing features in all previous layers. In this way, the prediction of different structures are not competing with each other, while interactions between them are still kept at the same time.

The structure of the paper is as follows. In Section 2, we review the related approaches for 3D localization and segmentation in medical image analysis. The next section explains the core design of our multi-stream CNN model. In Section 3, as a preprocessing, the skull stripping tool is first introduced, and the detailed design of the individual stream of VP-Nets is given, Section 3.3 reinterprets the localization task as a dense segmentation problem and explains in detail how we get the final 3D bounding boxes from the predicted masks. In Section 3.4, we visualize a 3D saliency volume to show that the model has learnt the 3D locations of structures from 2D projections. Comparison experiments with Random Forest (RF), 3D U-Nets, and variants of VP-Nets are presented in Section 4. Section 4 also investigate the effects of different loss functions and data partition. Results are reported and discussed in Section 5. Finally, we conclude the paper in Section 6.

2. Related work

Machine learning techniques have become popular for detecting and localizing anatomical structures in 3D medical image analysis. Object detection, as a high-level abstraction task itself(from a whole 3D volume to several digits), researchers have designed specialized features and complex processing systems. Criminisi et al. (2010) described a method that localizes multiple structures based on regression random forests (RF). In that work, hand-crafted features were extracted to regress the coordinates of the bounding boxes using RFs in full-body CT images. Sofka et al. (2014) presented a 3D structural detection framework based on graphical models. A sequential estimation technique was used to capture the interdependence of different structures in 3D space, while still relying on traditional features (e.g. Haar). Ghesu et al. (2016) proposed a framework based on fully-connected neural networks and cascaded filtering. That method fulfilled the detection in three consecutive stages: finding location only, finding location and orientation, and finding location, orientation and scale. The method was evaluated on the task of localizing the aortic heart valve in 3D ultrasound data. That work avoided the need to design specific features but was limited to localization of one anatomical structure at a time. Localizing multiple structures requires training of separate networks, thus scaling with time and memory usage. These three works Criminisi et al. (2010), Sofka et al. (2014) and Ghesu et al. (2016), though differ in terms of feature types and learning techniques, all aimed to learn the mapping from the feature representations to raw coordinates of the targeted bounding boxes.

Other papers have proposed to solve structure localiztion in lower dimensions (Zhou et al., 2013; Lu et al., 2016; de Vos et al., 2017; Baumgartner et al., 2017). Gao et al. (2016) and Baumgartner et al. (2017) trained CNNs to classify 2D frames in ultrasound video sequence, and the latter further used the pre-trained models to localize the structures using saliency maps obtained through back propagation. Zhou et al. (2013), Lu et al. (2016) and de Vos et al. (2017) localized organs in 3D scans by initially detecting bounding boxes in *2D image slices* from three orthogonal planes. Subsequently, the 3D bounding box was reconstructed by combining the outputs for all axial, coronal and sagittal slices with a voting mechanism. Specifically, Zhou et al. (2013) applied AdaBoost and ensemble of stump classifiers with Haar-like features, while

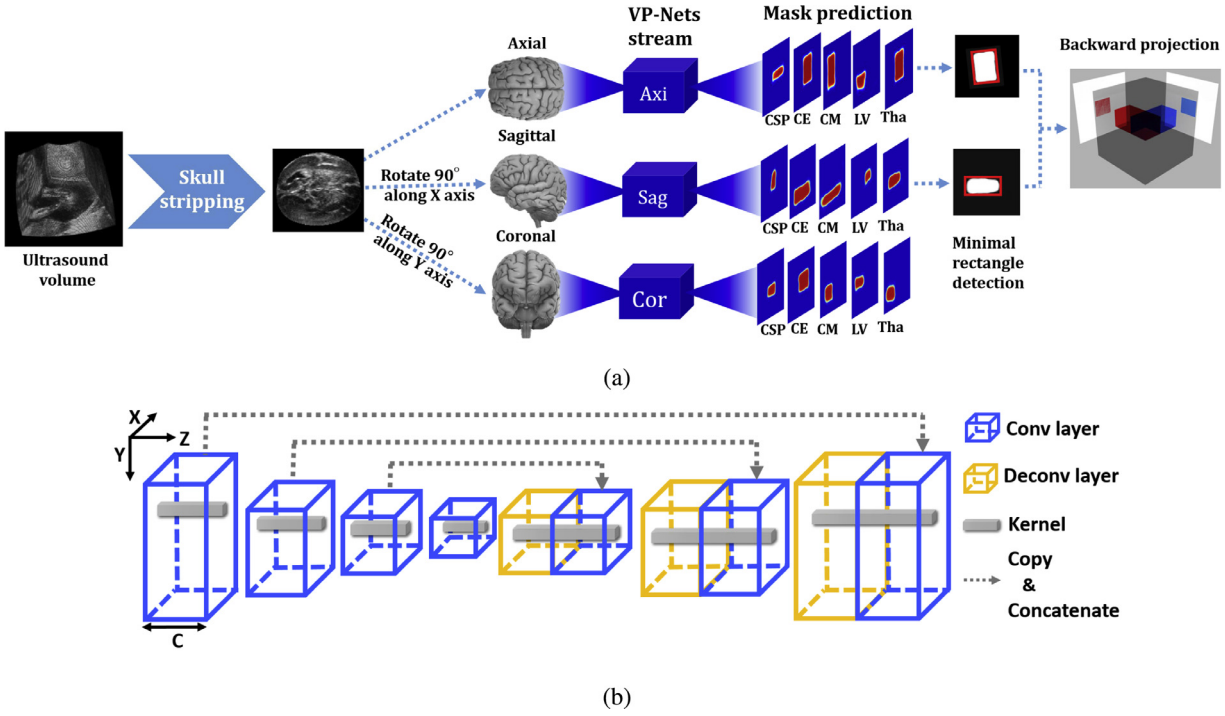


Fig. 2. Networks design details: (a) An overview of the CNN-based image analysis pipeline. A raw ultrasound volume is preprocessed using our skull-stripping tool. The masked brain image and two of its transformed (90° rotation) volume are passed, in parallel, to three independent VP-Nets. Each VP-Net outputs the projection mask of five structures (15 projected masks for 5 structures). The minimal rectangle that encloses the prediction silhouette is detected for each structure in each view, to generate the bounding box parameters. The 3D bounding box for a structure can be reconstructed with two obtained rectangle masks with backward projection. **A graphical demo for the VP-Net model is available at: https://youtu.be/KVxkbqWYWxc.** (b) Each individual VP-Net shares the same spirit as 2D U-Net (Ronneberger et al., 2015) for full-resolution prediction. In VP-Net, convolution filters in all layers only scan along the 2D X-Y plane, while always penetrating along the Z-axis to capture contextual information along that dimension.

Lu et al. (2016) and de Vos et al. (2017) trained ConvNets to determine the presence of anatomical target structures, and roughly localized the structures in the image slices. These methods demonstrate fast inference purely based on 2D image slices, thereby losing the contextual information from the third dimension.

Volumetric FCNs (Çiçek et al., 2016; Milletari et al., 2016) were proposed for 3D **segmentation** in order to capture contextual information in 3D space. Their architectures consist of a down-sampling path (canonical classification CNN), followed by an up-sampling path, where spatial resolution is recovered by performing up-sampling with skip layers. These networks have achieved good performance on the segmentation of MRI volumes (Çiçek et al., 2016; Milletari et al., 2016). However, these models can have large memory footprints which makes it infeasible to train a deconvolution decoder for high-resolution 3D outputs (Tatarchenko et al., 2017). In practice, we have found resolutions higher than 64^3 voxels bring memory issues even on a current state-of-the-art Titan X GPU (12GB).

Aiming to incorporate merits from previous methods, a key idea in the proposed VP-Nets is to **use 2D projections to reduce dimensionality**. This reduces the problem dimensionality from cubic to quadratic, while also incorporating 3D contextual information. We have chosen to use CNNs to avoid sophisticated feature engineering while maintaining affordable memory footprints.

3. Methods

The task we consider is to automatically detect and localize five key structures in a 3D fetal US brain volumes (Fig. 1a). The volumes were collected following a standard US clinical acquisition protocol. As a preprocessing step, the fetal brain is extracted using an automatic skull stripping tool described in Section 3.1. Then, the whole VP-Nets are trained with the target structures labelled with 3D bounding boxes by human experts (Sections 3.2 and 3.3). During testing, the whole pipeline does not require human intervention and it can process input of different sizes as the model is fully-convolutional. Design details of the architecture are explained in subsequent sections.

3.1. Preprocessing

To remove the extra-cranial tissues, we build an automatic skull-stripping tool using a 3D U-Net model. Many algorithms dedicated to skull segmentation in ultrasound images have been proposed in the literature (Lu et al., 2005; Namburete and Noble, 2013; Huang et al., 2015). Here we use a 3D U-Net model, which showed great performance on similar tasks (Çiçek et al., 2016)(Network architecture is described in the Appendix). Detailed comparison of the this tool and other methods is beyond the scope of this paper. The model consists of contracting networks (convolution layers interwoven with pooling layers) and expansive networks (convolution layers coupled with up-sampling layers). As a result, the model accept raw fetal brain ultrasound images as input and predict the skull segmentation masks of the same resolution.

3.2. View-based Projection Network (VP-Net)

The design of VP-Nets is inspired by two observations. Firstly, in clinical practice, detecting a structure in a fetal 3D US image is typically achieved by scrolling through the slices to find the largest cross-section of the structure in one of the three standard views: axial, coronal and sagittal (see Fig. 3a). Then the smallest rectangle that encloses the contour of the structure in that view is manually labelled. The same procedure is repeated for another orthogonal

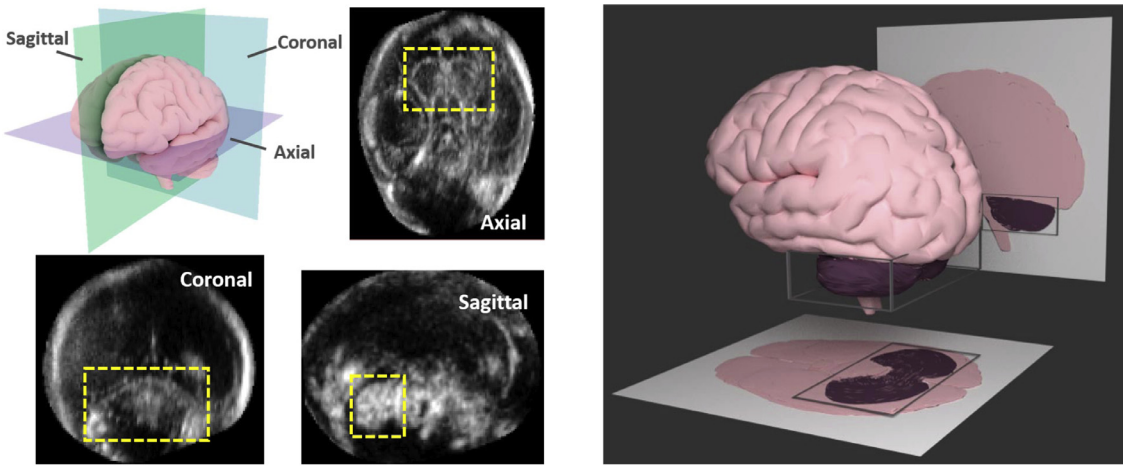


Fig. 3. View-based Projections. (a) Cerebellum (CE) in different views. Yellow boxes refer to the cross section of 3D human expert annotation. (b) Schematic of orthogonal projection. The cerebellum is highlighted in purple and bounded by a gray cube. Its orthogonal projections on axial plane (horizontal) and sagittal plane (vertical) are shown respectively. The combination of the two rectangles defines the 3D bounding box of the object. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

view. The two 2D rectangles define a cuboid which envelopes the object in 3D space with a 3D bounding box. Secondly, the orthogonal projection of a 3D object in a given direction is identical to the union set of all cross-sections along that direction (Fig. 3b). In our case, as the considered anatomies do not possess lobes, a projection of a 3D structure is equal to its largest cross-section in the projection direction. With these two principles, we solve detection as finding the orthogonal projection of the object.

Following these observations, the proposed framework accepts input volumes of standard anatomical views (axial, sagittal and coronal). The three volumes are passed in parallel to three identical modules. The 3D bounding box is therefore reconstructed from 2D mask predictions in the different views.

It should be noted that the proposed framework is different from other 2D or 3D CNN architectures (Long et al., 2015; Ronneberger et al., 2015; Çiçek et al., 2016; Milletari et al., 2016). 2D models treat an input volume as 2D image slices: $\mathrm{I} \in \mathbb{R}^{M\times N\times C}$, where *M, N* refer to spatial image dimensions (X-Y), and *C* denotes the number of channels (RGB or grayscale). A common way of using 2D architecture to analyse a 3D input is to take slices from a 3D volume and process them separately. The long-range dependencies between these 2D slices is modelled by post-processing. 3D CNN takes input volumes of $\mathrm{I} \in \mathbb{R}^{M\times N\times D\times C}$, with the additional notation: *D* represents the third Z dimension of 3D data. In this case, each convolution kernel needs to scan over the whole 3D space in every layer. In contrast, each individual VP-Net (Fig. 2b) shares a similar configuration with the 2D models, whereas the *C* are set to be the same size as the Z dimension, e.g. $C = 128$. Therefore, its deep convolution filters penetrate the whole volume along the Z-axis and only scan along the 2D X-Y plane. In this way, the model can capture 3D spatial dependencies without adding the Z dimension as used in 3D CNN. Therefore, it avoids cubically increasing memory requirements. Note that due to the penetration behaviour of the convolutional kernels, the deep kernels learn useful spatial features along the third dimension but are not expected to determine the exact location of the object on this axis. This information is later recovered by incorporating information from the other two streams (as shown in Fig. 2).

3.3. Detection as segmentation

Instead of regressing raw coordinates (Criminisi et al., 2010; Sofka et al., 2014; Ghesu et al., 2016), each VP-Net projects the desired structures to its corresponding anatomical views and **segments** the bounding-box masks of structures.

The loss function is therefore defined as :

$$E(\theta_{vp_1}, \theta_{vp_2}, \theta_{vp_3}) = \sum_{i=1}^{3}\sum_{s=1}^{S}\sum_{MN}\mathcal{L}_{MN}^{s}(\theta_{vp_i}) \tag{1}$$

where θ_{vp_i} refers to the parameters in the *i*th VP-Net ($i \in 1 \rightarrow 3$), *S* corresponds to the number of target structures (in our case $S = 5$) and *M, N* refer to volume dimensions: X and Y respectively. Sigmoid units are used for each pixel on the mask, indicating the probability of the target structure. Different loss functions are investigated in Section 4, e.g. traditional binary cross-entropy, dice coefficient loss.

This detection-as-segmentation design avoids fitting a highly abstract mapping function (feature representations to bounding box coordinates), and increases the foreground/background ratio (i.e. alleviates label sparsity) without further annotations. In practise, the approach increases the foreground/background ratio from 0.6% to 5.0% (3D to 2D). Since the receptive fields of different pixels correspond to different image regions, each of them captures a different variation on the appearance of the input. The dense, pixel-wise mask supervision can therefore be regarded as a sliding-window scheme, which serves as a natural way of data augmentation (same as random cropping of the input).

Finally, the mapping from the predicted silhouettes (in different views) to the final 3D bounding boxes takes two steps. First, the rectangle *R* that encloses the silhouette mask (denote as *T*) is calculated as follows:

1. The centroid of *R*: $p(x_c, y_c)$ is obtained by finding the mean of pixel position x_i, y_i along the x and y axis for all pixel $p(x_i, y_i) \in T$.
2. The contour of T is extracted as ∂T by simply calculating the gradient of T.
3. The ellipse that best fits the contour ∂T is detected using least square fitting. The orientation ϕ of the ellipse (the angle between the major axis and the image axis) is recorded.
4. For every point $p(x_j, y_j) \in \partial T$, the distance between it and the centroid $p(x_c, y_c)$ along the direction of ϕ and that in the perpendicular direction are recorded as da_j and db_j respectively.

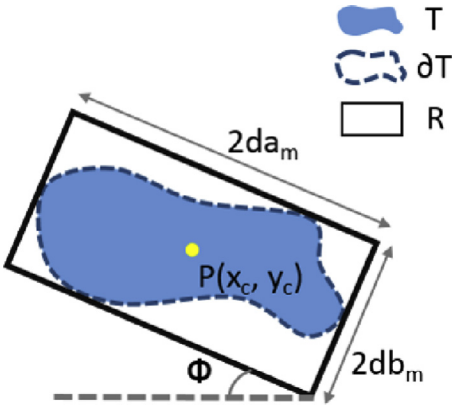


Fig. 4. Rectangle detection denotation. The prediction mask T is plot in solid blue, and its contour is plotted in dashed blue line. The centroid $p(x_c, y_c)$, the orientation ϕ, the width $2da_m$ and the height $2db_m$ are labelled respectively. The black box denotes the detected minimal rectangle. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

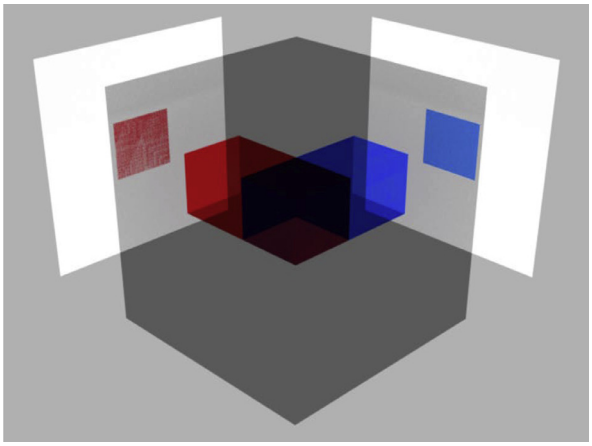


Fig. 5. Back-ward projection. The red and blue are rectangle masks detected from two different views. The 3D bounding box can be obtained by intersection of their projection (dark purple). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

5. The maximal value of $da_j \in da$ and $db_j \in db$ are selected as da_m, db_m.
6. The rectangle R is defined by $p(x_c, y_c)$, ϕ, $2da_m$, and $2db_m$ which correspond to the centroid, orientation, the width and height of the R respectively (as shown in Fig. 4).

A unique rectangle mask can be created with the obtained parameters and the process is repeated for three anatomical views. A 3D bounding box is defined by $\mathbf{B} = \{C_x, C_y, C_z, \theta_x, \theta_y, \theta_z, S_x, S_y, S_z, \}$, where the subscript x, y, z corresponds to three axis, C_i represents the centre coordinate, θ_i denotes the box orientation, and S_i represents the box size in view i. The centroid, orientation and size of each calculated 2D rectangle define a subset of **B**. Specifically, the rectangle in axis view defines C_x, θ_x, S_x (X axis), that of the coronal view provides C_y, θ_y, S_y (Y axis), and that of the sagittal view defines C_z, θ_z, S_z (Z axis). The accuracy result reported in Section 5 is calculated based on this approach.

A 3D cuboid mask can be created based on parameter set **B**. This process involves calculating all six facets of the cuboid and finding voxels located within them. The whole process takes approximately 5.6 s to generate each 3D cuboid therefore in total costs around $0.5min$ to generate the visualization for all five structures.

To achieve real-time visualization, we propose an approximation of the 3D bounding box thorough back-ward projection using the predicted silhouettes from any two views (see Fig. 5). This could be simply obtained by: a) stacking each mask N_v times along the corresponding view to create a projection volume P_v, where N_v is the depth of 3D volume in the view $v \in V$. b) The intersection volume In of two different P_v (e.g. P_a from axial view and P_c from coronal view) can be extracted by a simple summation followed by thresholding. These two steps only involve basic matrix manipulations thus the backward-projection can be efficiently implemented with a few lines of code and costs less than 0.02 s to reconstruct each 3D cuboid-like mask. In average, the results overlap with the more rigorously obtained cuboid by $88.9 \pm 5.7\%$ (intersection over union). This fast implementation might be of clinical interest in practice.

3.4. Visualizing the 3D saliency volume

The visualization of a 3D saliency volume allows one to see what deep filters have learnt along the third dimension, with only 2D supervision (X-Y axis masks) during training.

Formally, each stream of one VP-Net can be formulated as:

$$\mathrm{P} = f(\mathrm{I}; \theta) \tag{2}$$

where I is the input volume, and $\mathrm{P} \in R^{128 \times 128 \times S}$ is the output predicted masks of the target structures, θ represents the learned weights, S is the number of target structures and f is a non-linear function mapping from input volume to prediction. Therefore, by perturbing the predicted score of a **single pixel** p_i ($p_i \in \mathrm{P}$), $\frac{\partial p_i}{\partial \mathrm{I}}$ is calculated as the saliency volume for p_i (adapted from Simonyan et al., 2014). Intuitively, the resulting saliency volume indicates the voxels in the input volume with the largest influence on output prediction. The experiment result is reported and discussed in Section 5.3.

3.5. Implementation details

We applied the same training strategy for all tested networks (except for batch size). The training was done end-to-end simultaneously via RMSProp (Tieleman and Hinton, 2012) from scratch. The initial learning rate was initialized as 10^{-3}, and further decayed by a factor of 10 every 15 epochs. The weights were initialized randomly, batch-normalization was used after each convolutional layer, followed by ReLU as non-linearity. Max-pooling was used to down-sample the feature map. Data-augmentation included: (1) random cropping and rescaling with a factor of $s = 0.7$, (2) rotation of an angle randomly selected within $\theta = \pm 20^{\circ}$, (3) random volume-wise flip i.e. reflection of the slices with a probability of $p = 0.5$.

4. Experiments

Datasets. We used a dataset of 285 3D US fetal brain images acquired following standard clinical protocol of collecting transabodominal fetal neurosonograms (Papageorghiou et al., 2014). The volumes were randomly selected from scans of different healthy fetuses age range between 20 and 29 gestational weeks. All the scans were collected using Philips HD-9 ultrasound machines with curvilinear abdominal transducers C5-2, C6-3, V7-3. Due to the purpose of anonymity, the authors were not informed about the

Table 1
Skull segmentation accuracy. The predicted skull mask and its corresponding ground truth are compared in: 1) the distance between their centre , 2) the difference between their scales (average across three dimensions), and 3) the 3D Intersection over Union (IOU) between the two.

Skull size(mm)	Cen Dev(mm)	Sca Dev(mm)	3D IOU (%)
51.2 ± 12.5	1.6 ± 1.0	2.1 ± 1.8	86.5 ± 5.1

relationship between each scan and its corresponding sonographer. The typical size of each volume was $180 \times 200 \times 170$ voxels, with isotropic voxels measuring ($0.6 \times 0.6 \times 0.6$ mm^3) in size. Volumes were divided into a training (200 volumes), validation (40 volumes), and testing set (45 volumes). Within the dataset, 285 CSP, 263 LV, 285 Tha, 266 CE, and 249 CM were visible based on visual assessment of the 3D volumes. Note that some structures are not visible due to the presence of speckles or inferior image quality.

Pre-processing. The only pre-processing step the proposed model requires is skull-stripping. We trained a 3D U-Net model based on the manual delineated skull masks. Table 1 reports the skull segmentation accuracy. The results shows that the skull-stripping tool localize and segment the skull accurately from the raw input image. The tool achieved 3D IOU of $86.5 \pm 5.1\%$, which indicates good overlap between the prediction and the ground truth in all dimensions. During test time, it takes 0.23 s to process one single volume. As part of the whole pipeline, this automatic tool provides a fast and accurate way to prepare the raw images for the main task: structure detection.

Proposed VP-Nets architectures. We experimented with 6 VP-Nets architectures in total; they varied in two ways: depth $l =$ 7 or 9 layers, and number of convolution kernels in every layer ($C = 64$, 128 or 256). Each stream of VP-Nets starts with several convolution and batch-norm blocks which coupled with max-pooling layer to reduce feature map resolution and obtained high-level features. Then a sequence of up-sampling (de-convolution) layers are added subsequently, each of which is concatenated with the corresponds features from the contracting path to retain high-resolution information. For simplicity, we used the same architecture for each of the three individual streams. Note that the different streams do not share parameters, thus different view-based representations can be learned. We kept the number of filters unchanged in every layer throughout the whole network. Our most sophisticated model Network F has 9 ConvBlocks and 256 convolutional filters for each convolutional layer, in total it costs 1.2G memory. Details of the network architecture and memory consumption can be found in Table A.8 in the Appendix. Comparing with the total memory consumption of the 3D U-Net, Network F only takes a quarter of the former.

Biometric measurements. The diameter of CE and the width of CM is of particular clinical interest to help evaluate fetal maturation and to diagnose congenital diseases (Liu et al., 2011; Serhatlioglu et al., 2003; Arısoy and Yayla, 2010). Using the predicted masks, automatic measurement of biometry can be easily derived. We also investigated the accuracy of these clinically-relevant measurements by comparing them to results obtained from manual measurements by a human expert.

Baseline 1: regression random forest. We implemented a 3D detection framework based on the regression random forest algorithm (Criminisi et al., 2010). In this approach, each 3D bounding box is defined by a 6 component vector (namely, $\mathbf{b} = \{b^L, b^R, b^A, b^P, b^H, b^F\}$), where each component represents the corresponding axis-aligned bounding box. Its superscripts follow the radiological convention as: L = left, R = right, A = anterior, P = posterior, H = head and F = foot. Note that this method was not designed for different orientations. The model aimed to find the function mapping $\mathrm{b} = f(\mathrm{Q})$, where $\mathrm{Q} \in R^{D \times 1}$ is the extracted Haar-like features, D refers to its dimensionality (2048 in our experiment), $\mathrm{B} \in R^{6N \times 1}$ corresponds to the coordinates of the bounding boxes for N structures. In our experiments, we used 25 trees, with a tree depth of 8. For consistency, similar data-augmentation technique was used in this experiment.

Baseline 2: 3D U-Net. We implemented a 3D U-Net with input $\mathrm{I} \in R^{128 \times 128 \times 128 \times 1}$, and a 3D bounding box prediction: $\mathrm{P} \in R^{64 \times 64 \times 64 \times 5}$ (the largest allowed resolution by our GPU). Overall the 3D U-Net consists of four convolution $\wedge$ pooling layers and four four up-sampling $\wedge$ concatenated layers. Details of the network architecture and memory consumption can be found in the Appendix (Table A.7). Although only 64 convolutional kernels are applied for all layers, the memory cost for a single input volume during feed-forward process amounts to 2.1 G (without considering the memory consumption for parameters). Theoretically, during training, a naive back-propagation process (without special optimization) should take the same amount of memory consumption as feed-forward propagation (Vanhoucke, 2016), i.e. 4.2G in total for one 3D volume. Thus, even to predict at half the resolution of the original input, an state-of-the-art GPU (Titan X, 12G) can have problems in training networks with batch size larger than 2.

Evaluation metrics. For all experiments, structure detection was evaluated using Intersection Over Union (Jaccard index) to evaluate the overlap of bounding boxes between ground truth and prediction:

$$J_b(\mathbf{C}_g, \mathbf{C}_p) = \frac{\|\mathbf{C}_g \cap \mathbf{C}_p\|}{\|\mathbf{C}_g \cup \mathbf{C}_p\|} \tag{3}$$

where $\mathbf{C}_g$ and $\mathbf{C}_p$ refer to the ground-truth and prediction respectively, and $\|\mathbf{C}_g \cup \mathbf{C}_p\| = \|\mathbf{C}_g\| + \|\mathbf{C}_p\| - \|\mathbf{C}_g \cap \mathbf{C}_p\|$.

Cross-validation. To further estimate how the proposed model will generalize to the dataset, we use 5-fold cross-validation on our most sophistic model: VP-Net F. The dataset is partitioned equally into 5 sub-samples (each contains 57 subjects). Each cross-validation model is trained using the same hyper-parameter and data augmentation techniques as described in Section 3.5.

Loss function. Apart from the traditional binary-crossentropy as baseline experiment, we also investigate the network training based on a different loss function. In medical image processing, it is common that the anatomy of interest only occupies a small portion of the whole image. As a result, the loss is trapped in local minima and the networks tend to yield negative predictions. The VP-Nets model has already increase the foreground/ background ratio 8 times thorough projection. However, the value of the ratio (approximately 5%) is still relatively low. To tackle the class imbalance problem, several works use customized loss functions based on dice coefficient or IOU metric (Milletari et al., 2016; Sudre et al., 2017). Here, we explore whether the detection accuracy can be further improved with dice loss function, which can be defined as:

$$\mathcal{L} = 1 - \frac{2\sum_{i=1}^{N} \mathbf{C}_{gi}\mathbf{C}_{pi} + \epsilon}{\sum_{i=1}^{N} \mathbf{C}_{gi}^2 + \sum_{i=1}^{N} \mathbf{C}_{pi}^2 + \epsilon} \tag{4}$$

, where N is the number of samples, $\mathbf{C}_{gi} \in \mathbf{C}_g$ denotes the ith ground truth sample, $\mathbf{C}_{gi} \in \mathbf{C}_p$ denotes the ith prediction sample and ϵ is a

small number to avoid division of zero. The gradient of the loss can be compute w.r.t. $\mathbf{C}_{pj}$:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{C}_{pj}} = 2\frac{2\mathbf{C}_{pj}\sum_{i=1}^{N}\mathbf{C}_{gi}\mathbf{C}_{pi} - \mathbf{C}_{gj}(\sum_{i=1}^{N}\mathbf{C}_{gi}^2 + \sum_{i=1}^{N}\mathbf{C}_{pi}^2)}{(\sum_{i=1}^{N}\mathbf{C}_{gi}^2 + \sum_{i=1}^{N}\mathbf{C}_{pi}^2)^2} \tag{5}$$

, without considering the effects of ϵ. To compare the impact of loss function, another model with the same VP-Nets F architecture using the dice coefficient loss is trained. The same dataset, optimizer and data-augmentation techniques were employed.

5. Results and discussion

In this section, we report accuracy results on targeted structures such as center error, size error, biometry error and 2D & 3D Intersection Over Union (IOU), where 2D IOU is computed for a cross-sectional plane only, and 3D IOU is calculated over a volume.

5.1. Qualitative evaluation

Qualitative results are shown in Fig. 6, where 2D bounding boxes for individual structures are drawn on cross-sections of different views. The ground truth is shown in a yellow dashed box while the prediction is shown in a red dashed box. The figure shows examples of challenging anatomical configurations present in fetal neurosonography, which display varying contrast and orientation. During data augmentation, part of the brain is cropped. All images are resized to the same size for visualization purposes. It can be seen that the VP-Nets and 3D U-Nets outperform RFs in both finding the location and determining the scale of structures. It should be pointed out that the same data-augmentation technique was used for both RF and CNN experiments. This shows that the intrinstic limitation of features presented in Criminisi et al. (2010), which are not orientation invariant, as they were designed for CT scans where subjects present in the same orientation. On the contrary, although rotation-invariant is not built-in characteristics of CNN, it can learn hierarchical representations to cope with this variation given enough training images or suitable data-augmentation. It can reduce the necessity to design bespoke rotation-invariant features for ultrasound images. Fig. 6d shows that the LV comprises of bright and dark blobs in the axial plane. Both VP-Nets and 3D U-Nets picked out this unique pattern and were able to identify the LV quite well. However, as its boundary is ambiguous in the sagittal view, both models correlate well with the ground truth. This phenomenon coincides with typical human observation. A human expert usually finds one structure easier to identify in one or two views. This phenomenon is affected by the way in which the ultrasound beam interacts with different tissues within the fetal brain. It suggests that the imaging protocol may need to be adapted if a specific structure is of particular interest. That is, the image may be required from the sagittal view to diagnose congenital disease related to CSP.

Although these examples are drawn from different test subjects, one can still identify the overlap between CE, CM and Tha (Fig. 6a, b and e). This means some voxels cannot be assigned with a single label which is an additional challenge posed by our task. This is similar to attribute learning in computer vision, where multiple visual nouns and adjectives are learned to describe one object (Russakovsky and Fei-Fei, 2010). By predicting different structures separately, the proposed network avoids the ambiguity of the label of each pixel/voxel. Meanwhile, this strategy ensures that a large number of features can be shared within one anatomical view which simplifies the whole system as well.

5.2. Quantitative evaluations

5.2.1. Detection accuracy of VP-Net-F

We first report the accuracy result of our best VP-Net model: VP-Net-F (Table 2). From the reported average length of each structure, it can be seen that different structures vary in size and each structure has large intra-class variation. As shown in Table 2, our model copes well with this challenge, as the size prediction error is only 2.2 $\pm$ 1.8 mm (average across different structures). We also report 3D Intersection over Union (IoU) between prediction and annotation bounding boxes. The model achieved a 3D IoU larger than 62% in average. The CE detection task achieved the highest accuracy. We conjecture that it results from its distinct shape and relatively large size. In contrast, the LV scored the lowest accuracy among the five structures. Our interpretation is that the size is considerably smaller compared with other structures leading to a more imbalanced class.

As a side-product, the VP-net provides automatic fetal brain structure biometry. On average, the automatic measurements correlated well with the manual ones, the reported deviation is less than 2.0 mm.

5.2.2. Comparison of different models

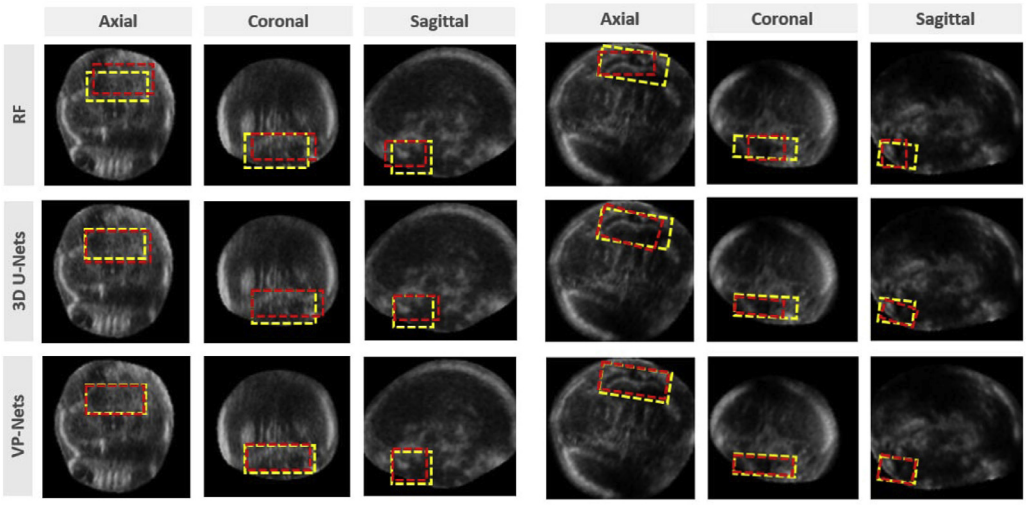
Table 3 reports qualitative results for 3D U-Net and the 6 VP-Nets (VP-Net-A to VP-Net-F). The results show that 3D U-Net and VP-Nets all outperform RF and are capable of dealing with multiple structures with varying shapes and scales. The results show the benefit of the end-to-end trainable model over hand-crafted features. Interestingly, the RF model is accurate in predicting the scale of the bounding boxes while it scores poorly in the 2D and 3D IOU metrics.

VP-Net-F achieved the highest accuracy among all calculated metrics. It may benefit from its large model capacity, i.e. largest number of trainable parameters compare to its counterparts. We speculate that its performance can be further enhanced by increasing the model size given enough training data. VP-Nets scores higher accuracy given a model with similar number of parameters. For example, VP-Net-E shows higher accuracy than 3D U-Nets in both center and scale detection (Table 3). It may suggest that by decreasing label sparsity using projection, the VP-Nets deal better with the over-fitting.

All detection methods achieve the best performance on the axial view. This observation echoes with the clinical observation that structures are easier to identify on axial views.

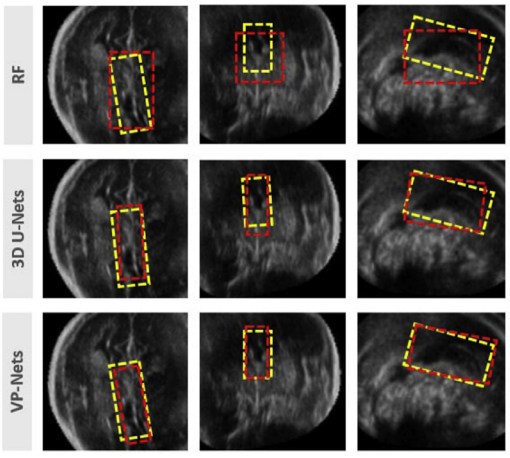
When comparing variants of VP-Nets, deeper networks usually lead to better performance, e.g. VP-Net-D *versus* VP-Net-A. Moreover, the number of channels plays an important role as shown by VP-Nets A, B and C. However, 3D U-Nets cannot afford a large channel size with current standard GPU capacity, this limits the model performance in practice. Interestingly, 3D U-Nets perform similarly to VP-Nets C in the mean 2D IOU metrics (Fig. 7d), while its performance is lower than almost all VP-Nets in terms of 3D IOU measurements (Fig. 7e). This indicates that 3D IOU is a more discriminating metric than the mean of 2D IOU metric as the former is the product of accuracy of the prediction along each orthogonal axis. In other words, a high 3D IOU score requires that the prediction is consistent with the ground truth in all three axis, while the mean of 2D IOU does not.

We further investigated this point by reporting the 2D IOU on different views separately (as shown in Fig. 7a–c). Fig. 7b shows 3D U-Nets achieve comparable performance in coronal view, while it perform inferiorly in sagittal view (Fig. 7c). On the contrary, all the VP-Nets perform similarly across different views. This may be explained by the fact that our design simplifies the task for VP-Nets as each stream only needs to deal with one single anatomical view, thus leading to overall superior performance. Furthermore,

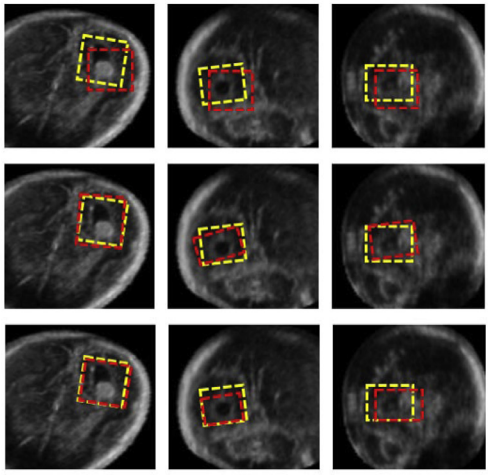


(a) Qualitive result of **CE** detection

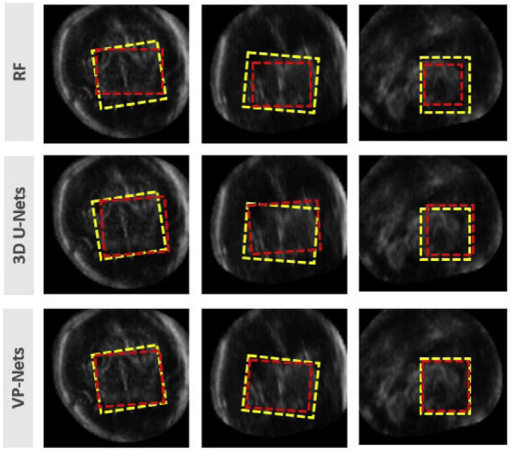
(b) Qualitive result of **CM** detection



(c) Qualitive result of **CSP** detection



(d) Qualitive result of **LV** detection



(e) Qualitive result of **Tha** detection

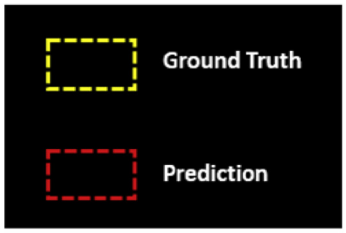


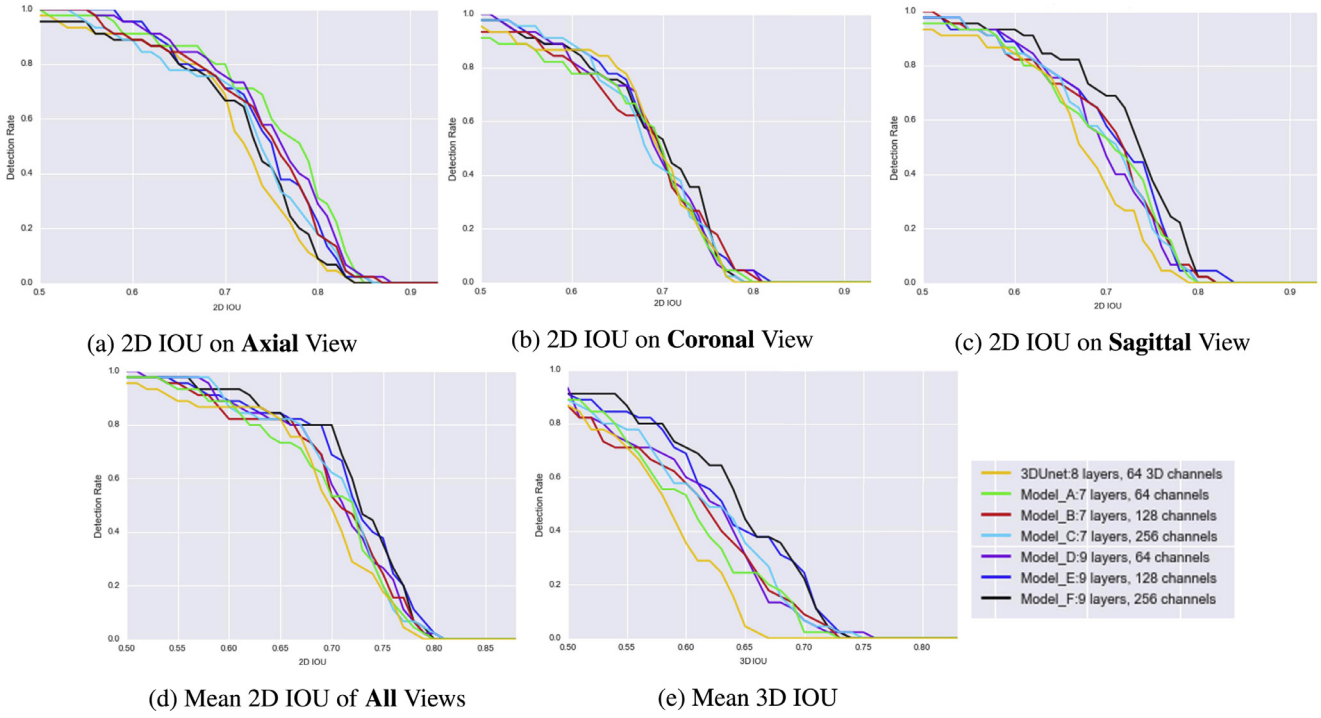
Fig. 6. Example of predicted bounding boxes of targeted structure in different anatomical views. Yellow box show the ground truth, and Red boxes are predictions.
First to last row: Results from Random Forest, 3D U-Net, VP-Nets.
First to last column: Results from Axial, Coronal, Sagital views.
Note that there are overlap between CE and CM in all the views (as shown in a and b). Tha coincides with CE and CM in coronal view as well. The results show the prediction of VP-Nets coincided better with ground truth than that of 3D U-nets and random forest. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 2
Results of **VP-Nets F**. The largest diameter of each structures is given in the second column for comparison. The prediction and ground truth (GT) are compared in terms of the center deviation (Cen Dev), scale deviation (Sca Dev) and 3D IOU. The automatically generated measurements of CE and CM is also reported.

Structure	GT (mm)	Cen Dev(mm)	Sca Dev(mm)	3D IoU (%)	Biometry Err(mm)
CSP	31.4 ± 5.8	1.8 ± 1.5	2.2 ± 1.8	63.2 ± 15.2	–
LV	9.0 ± 2.5	2.4 ± 2.0	1.5 ± 1.4	48.7 ± 19.1	–
Tha	25.8 ± 3.8	1.9 ± 1.7	2.1 ± 1.9	68.1 ± 14.7	–
CE	26.9 ± 14.0	2.0 ± 1.9	2.4 ± 2.2	66.5 ± 14.6	1.5 ± 0.7
CM	16.2 ± 5.9	2.1 ± 1.9	2.3 ± 2.1	65.9 ± 15.3	2.0 ± 0.9

Table 3
Quantitative results of different models. Column 2,3,4 list the channels, depths and the number of trainable parameters (Par) of each model. The prediction and ground truth (GT) are compared in terms of the center deviation (Cen Dev), scale deviation (Sca Dev), 2D IOU and 3D IOU. RF has the lowest 3D IOU accuracy as the predicted center locations have the largest error margin. The VP-Nets F achieves the highest accuracy in all metrics as the model has the largest model capacity (trainable parameters numbers). 3D U-Nets shows similar accuracy in mean 2D IOU deviation, but lags behind in 3D IOU. Refer to the text to see detail analysis. The last column report the runtime of different CNN models. The VP-Nets models use considerably less time than the 3D U-Nets.

Model	Channels	Depths	Par (10^6)	Cen Dev (mm)	Sca Dev (mm)	2D IOU (%)	3D IOU (%)	Run time
RF	–	–	–	6.6 ± 4.8	3.9 ± 3.6	55.6 ± 16.7	15.1 ± 14.5	
3D U-Nets	64	8	9	2.9 ± 2.7	3.1 ± 2.3	68.5 ± 9.2	56.5 ± 19.6	0.36 s
VP-Nets A	64	7	2	3.0 ± 2.8	3.2 ± 2.2	67.8 ± 9.1	57.1 ± 17.1	**0.04 s**
VP-Nets B	128	7	3	2.8 ± 2.2	3.3 ± 2.0	69.1 ± 7.9	58.6 ± 18.1	0.07 s
VP-Nets C	256	7	29	2.6 ± 2.3	3.1 ± 1.8	70.0 ± 8.3	57.2 ± 17.0	0.07 s
VP-Nets D	64	9	7	2.7 ± 1.9	2.9 ± 2.0	69.2 ± 8.6	59.3 ± 16.9	**0.04 s**
VP-Nets E	128	9	9	2.5 ± 1.7	2.5 ± 1.8	70.2 ± 7.5	60.8 ± 17.5	0.08 s
VP-Nets F	256	9	38	**2.0 ± 1.9**	**2.2 ± 1.8**	**71.7 ± 8.1**	**62.0 ± 15.8**	0.08 s



(a) 2D IOU on **Axial** View (b) 2D IOU on **Coronal** View (c) 2D IOU on **Sagittal** View

(d) Mean 2D IOU of **All** Views (e) Mean 3D IOU

Fig. 7. Mean 2D & 3D IOU curves of 3D U-Nets (yellow) and VP-Nets (other colours). b.shows 3D U-Nets perform similarly with other VP-Nets on coronal view in 2D IOU. All VP-Nets achieves higher accuracy when combine the three views together (e). Comparing different variants of VP-Nets, Model F(black) achieves the highest accuracy. It shows that larger feature channel and deeper layers increase the network performance. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

as the memory burden is reduced from cubic to quadratic, VP-Nets support a larger batch size and prediction with higher resolution masks (equal to the input image).

The last column of Table 3 lists the running time of different methods (per volume). All the VP-Nets models need substantially less time than the 3D U-Nets, which proves the previous analyse that the proposed VP-Nets model can cut down the time consumption. The fastest, VP-Nets A, only need 0.04 s to process an input volume. The VP-Nets F uses 0.08 s, a quarter of that of the 3D U-Nets, and achieves the best localization accuracy. This indicates the applicability of the proposed method on large datasets in the future.

5.2.3. Cross-validation

As our dataset is randomly sampled from a large clinical database, it is interesting to see whether the partition of training

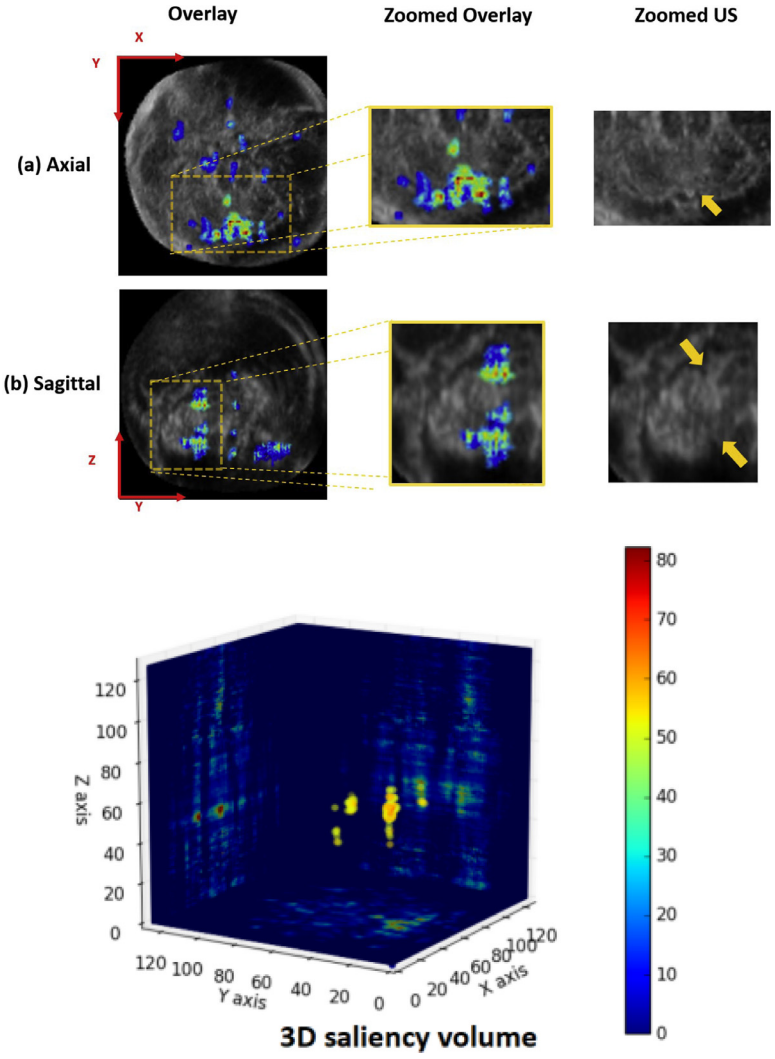


Fig. 8. Visualization of a 3D saliency volume derived from CE detection in the axial view. The right-hand figure shows the points with top 2% gradient values in 3D. The colour red represents higher gradient values, indicating a higher influence on the prediction outcome. Sub-figure (a)(b) shows a axial and a sagittal US slice with 3D saliency points projected on each view respectively. The images and the original US are zoomed in for detail visualization. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 4
Cross-validation results. The prediction and ground truth (GT) are compared in terms of the center deviation (Cen Dev), scale deviation (Sca Dev), 2D IOU and 3D IOU. The model performed similarly across different folds.

Fold no.	Cen Dev (mm)	Sca Dev (mm)	2D IOU (%)	3D IOU (%)
Fold 1	1.9 ± 1.7	2.1 ± 1.7	72.6 ± 7.2	62.6 ± 16.0
Fold 2	1.9 ± 1.8	2.3 ± 1.9	71.0 ± 7.6	61.3 ± 15.9
Fold 3	2.2 ± 1.9	2.5 ± 1.9	70.9 ± 8.3	60.6 ± 16.8
Fold 4	1.9 ± 1.7	2.2 ± 1.9	71.4 ± 7.9	62.3 ± 16.0
Fold 5	2.1 ± 2.0	2.4 ± 1.8	71.2 ± 8.6	61.7 ± 16.5
Average	2.0 ± 1.8	2.3 ± 1.9	71.3 ± 7.7	60.9 ± 16.1

and testing data will affect the model performance. A 5-fold cross-validation is carried out using the same training paradigm of previous experiment. Table 4 reports the results of each fold and the average based on VP-Net-F, showing that the partition of dataset has minor effect on the model performance and the proposed method generalize well on the dataset.

5.2.4. Dice coefficient Loss

Table 5 compares the results of models trained by different loss function, while other training hyper parameters were set to be identical. The results show that model trained with dice coefficient loss outperformed the previous model in finding the locations and the scales of the targets. It shows the dice loss handles the class imbalance problem superiorly, therefore can boost the prediction accuracy by a small margin.

Table 5
Comparison of models trained with binary cross-entropy and dice coefficient loss. The model performance is reported as: the center deviation (Cen Dev), scale deviation (Sca Dev), 2D IOU and 3D IOU. The model trained with dice coeffcient loss outperformed the model trained with binary cross-entropy.

Loss	Cen Dev (mm)	Sca Dev (mm)	2D IOU (%)	3D IOU (%)
Cross-entropy	2.0 ± 1.9	2.2 ± 1.8	71.7 ± 8.1	62.0 ± 15.8
Dice	1.8 ± 1.4	1.9 ± 1.5	74.6 ± 7.1	63.2 ± 14.7

5.3. Saliency volume

An example of a 3D saliency volume derived from the CE prediction (from the test set) in the axial stream (X-Y dimension) is shown in Fig. 8. Salient points scattered far away from the CE indicate the incorporated context information, while those clusters around the structure itself represents regional characteristics captured by the model. For example, the bow-shape edge of the CE is of particular interest (Fig. 8a): it shows these salient points are

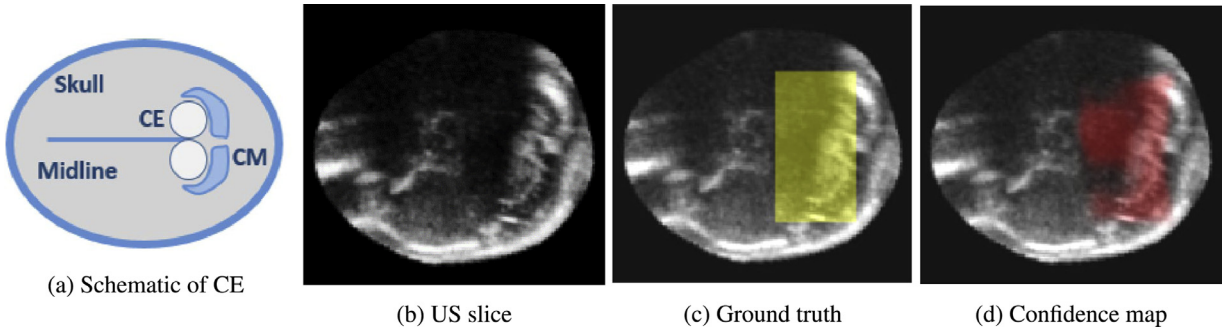


Fig. 9. Challenging case of CE in test sets. (a) shows the schematic of anatomical configuration of CE in axial plane. The white eyeglass-shaped structure is the CE. (b) is the US slice of the discussed case. Half of the CE is blocked by acoustic shadows. (c) shows the ground truth bounding box of CE (overlaid in yellow). The confidence map predicted by VP-Nets F is shown as red in (d). In this case, IoU of the raw prediction map and the ground truth is 68%. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

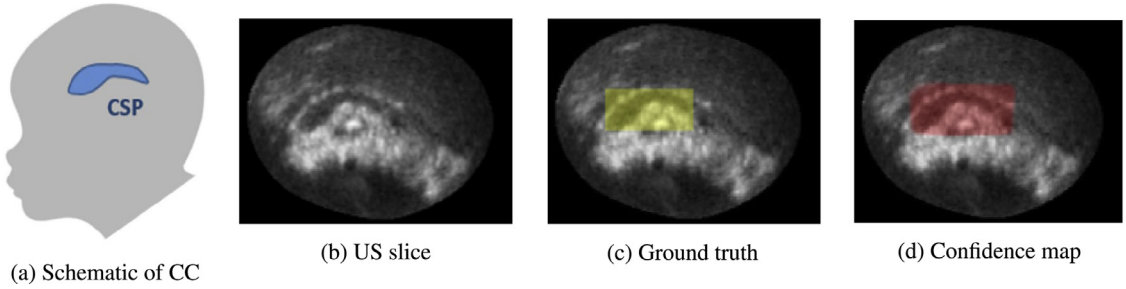


Fig. 10. Challenging case of CC in test sets. (a) is the schematic of CC in mid-sagittal plane. The CC is shown as the blue comma-shape structure. The corresponding mid-sagittal slice of the US volume is given in (b). The ground truth is plotted as transparent yellow rectangle in (c). Similarly, the predicted confidence map is shown as red rectangle in (d). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

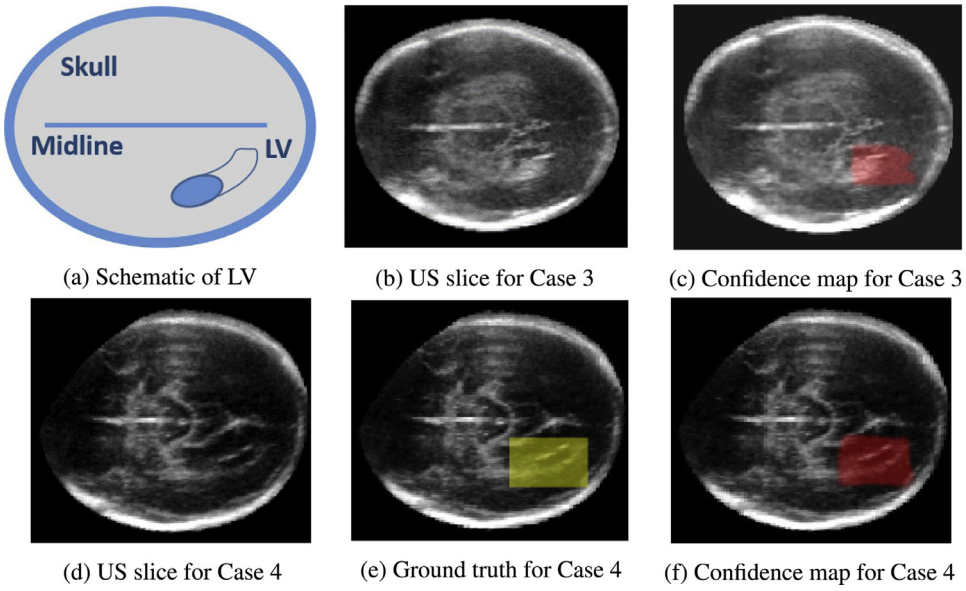


Fig. 11. Challenging case of LV in test sets. (a) is the schematic of LV in axial plane. (b), (d) show the corresponding axial image of the third and the forth case respectively. No ground truth is displayed for (b) as the human expert reckoned LV is not visible in this case. (e) show the ground truth label of the LV for the forth case. The predicted confidence map is superimposed on the corresponding ultrasound image in (c) and (f).

consistently correlated with salient features of the target structure. This visualization indicates that the trained model is able to capture the most discriminant features of the target structure. *More interestingly*, the salient points (red spots) closely correlate with the target structure along the Z axis (see Fig. 8b). It suggests that the model can pinpoint the location of the object in the third dimension Z, despite supervision given only in the X-Y plane,

5.4. Challenging cases

Here we investigate the model further by discussing three particular challenging test cases. In the first case, the original US image quality is relatively poor (Fig. 9). Fig. 9a is a schematic of the CE in the axial plane (a trans-cerebellum plane). The CE is the white eyeglass-shape structure, while CM is the dark 'M'-shape structure .[1] However, as the US image is corrupted by acoustic shadows, half of the CE is invisible (Fig. 9b). The human expert had to label the structure based on subjective assumptions (overlaid as yellow box in Fig. 9c). Fig. 9d shows the output confidence map of the VP-Nets model in red, where a higher intensity indi-

[1] The reader can also refer to the upper-right figure in Fig. 3a for a clearer visualization of the CE in the trans-cerebellum plane.

cates higher confidence. The prediction is more confident around the region where the CE is visible.

The second case is the one where prediction of the CSP bounding box has the largest scale deviation from the ground truth in the test set. Similar to the above example, the schematic of CSP in the mid-sagittal plane is displayed in Fig. 10a. Compare the ground truth (Fig. 10c) and the prediction (Fig. 10d). They disagree mostly at the rear region of the CSP. Refer to the original US slice (Fig. 10b), this may be caused by the poor visibility of the tail of the CSP, which is connected with a small dark structure. The third case is where the human expert indicated that the structure is not present while the proposed model detected a structure (Fig. 11b). Referring to the location pointed out by Fig. 11c, one can argue that part of the LV is visible. It is possible that the VP-Nets has captured some unique ultrasonic patterns that may be hard to interpret by human eye.

The fourth case shows the detection results where 2D IOU with the ground truth is low (45.6%, Fig. 11f). Comparing Fig. 11e and Fig. 11f, one can see the results have a high correlation. Nevertheless, as the area of the LV is relatively small, a small deviation in the scale or center prediction causes a significant decrease in IOU estimation.

6. Conclusions

This paper has presented a new efficient approach to 3D detection of multiple brain structures in fetal neurosonography through a FCN-based model: VP-Nets. The model does not require a large amount of training data and learns from coarsely annotated volumetric images (bounding-box masks). Its application can be easily extended to other medical imaging modalities where detection is desirable. Models with different depth and feature channels were compared and the prediction accuracy was reported. Cross-validation experiment demonstrated that the proposed model generalize well on the dataset. Different loss functions were also tested, where the model trained with the dice coefficient loss out performed the one trained using binary cross-entropy. Visualization of the 3D saliency volume demonstrated that the network is capable of localizing key structure in 3D space without any supervision in the third dimension. The paper ended with discussion of several challenging cases which revealed how the model performed when the structure visibility is limited.

Conflicts of Interest

The authors whose names are listed immediately below certify that they have NO affiliations with or involvement in any organization or entity with any financial interest (such as honoraria; educational grants; participation in speakers' bureaus; membership, employment, consultancies, stock ownership, or other equity interest; and expert testimony or patent-licensing arrangements), or non-financial interest (such as personal or professional relationships, affiliations, knowledge or beliefs) in the subject matter or materials discussed in this manuscript.

Acknowledgement

We acknowledge the UK site of the *Intergrowth-21st study* (Papageorghiou et al., 2014) for the fetal neurosonography datasets and Ana Namburete for helpful discussions. This work was supported by the EPSRC Programme Grant Seebibyte (EP/M013774/1) and the National Institutes of Health (NIH) through National Institute on Alcohol Abuse and Alcoholism (NIAAA) (2 U01 AA014809-14). W. Xie is supported by a Google DeepMind Scholarship.

Appendix A

Table A1
(a). Convolution Block Details. (b) Architecture of the skull segmentation network. The whole structure is similar to 3D U-Net and has 9 layers. For simplicity, each layer has 64 convolutional filters. Due to the memory consumption of 3D U-Net, the resolution of input images is reduced to $96 \times 96 \times 96$. The network output skull segmentation result in the same resolution.

Layer type	Conv	BNorm	Conv	BNorm
Filter size	$3 \times 3 \times 3 \times C$	–	$3 \times 3 \times 3 \times C$	–
(a) ConvBlock Detail				
Layer Type			Output size	
Input:			$96 \times 96 \times 96 \times 1$	
ConvBlock 1:			$96 \times 96 \times 96 \times 64$	
MaxPooling 1:			$48 \times 48 \times 48 \times 64$	
ConvBlock 2:			$48 \times 48 \times 48 \times 64$	
MaxPooling 2:			$24 \times 24 \times 24 \times 64$	
ConvBlock 3:			$24 \times 24 \times 24 \times 64$	
MaxPooling 3:			$12 \times 12 \times 12 \times 64$	
ConvBlock 4:			$12 \times 12 \times 12 \times 64$	
MaxPooling 4:			$6 \times 6 \times 6 \times 64$	
ConvBlock 5:			$6 \times 6 \times 6 \times 64$	
Deconv 5:			$12 \times 12 \times 12 \times 64$	
Merge 5:			$12 \times 12 \times 12 \times 128$	
ConvBlock 6:			$12 \times 12 \times 12 \times 64$	
Deconv 6:			$24 \times 24 \times 24 \times 64$	
Merge 6:			$24 \times 24 \times 24 \times 128$	
ConvBlock 7:			$24 \times 24 \times 24 \times 64$	
Deconv 7:			$48 \times 48 \times 48 \times 64$	
Merge 7:			$48 \times 48 \times 48 \times 128$	
ConvBlock 8:			$48 \times 48 \times 48 \times 64$	
Deconv 8:			$96 \times 96 \times 96 \times 64$	
Merge 8:			$96 \times 96 \times 96 \times 128$	
ConvBlock 9:			$96 \times 96 \times 96 \times 64$	
Output:			$96 \times 96 \times 96 \times 1$	
(b) Skull segmentation network				

Table A2
(a). Convolution Block Details. BNorm refers to Batch Normalization (Ioffe and Szegedy, 2015). ReLU is used after each BNorm. (b). Architecture of 3D U-Net and memory cost for single input volume during feed-forward process. To compute gradients in the backward pass, the activation of all layers should be stored. Therefore, each back-propagation step takes approximately twice the memory and the compute time than the forward propagation step (Vanhoucke, 2016). For all layers, we simply applied 64 convolutional filters. Thus, while training on single input volume, the sum of memory consumption for the whole 3D-UNet should be **4.2G**.

Layer type	Conv	BNorm	Conv	BNorm
Filter size	$3 \times 3 \times 3 \times C$	–	$3 \times 3 \times 3 \times C$	–
(a) ConvBlock Detail				
Layer type	Output size	Memory cost		
Input:	$128 \times 128 \times 128 \times 1$			$\approx$0.008G
ConvBlock 1:	$128 \times 128 \times 128 \times 64$			$\approx$1.073G
MaxPooling 1:	$64 \times 64 \times 64 \times 64$			$\approx$0.067G
ConvBlock 2:	$64 \times 64 \times 64 \times 64$			$\approx$0.134G
MaxPooling 2:	$32 \times 32 \times 32 \times 64$			$\approx$0.008G
ConvBlock 3:	$32 \times 32 \times 32 \times 64$			$\approx$0.016G
MaxPooling 3:	$16 \times 16 \times 16 \times 64$			$\approx$0.001G
ConvBlock 4:	$16 \times 16 \times 16 \times 64$			$\approx$0.002G
MaxPooling 4:	$8 \times 8 \times 8 \times 64$			$\approx$0.0001G
ConvBlock 5:	$8 \times 8 \times 8 \times 64$			$\approx$0.0002G
Deconv 5:	$16 \times 16 \times 16 \times 64$			$\approx$0.002G
Merge 5:	$16 \times 16 \times 16 \times 128$			$\approx$0.004G
ConvBlock 6:	$16 \times 16 \times 16 \times 64$			$\approx$0.002G
Deconv 6:	$32 \times 32 \times 32 \times 64$			$\approx$0.016G
Merge 6:	$32 \times 32 \times 32 \times 128$			$\approx$0.032G
ConvBlock 7:	$32 \times 32 \times 32 \times 64$			$\approx$0.064G
Deconv 7:	$64 \times 64 \times 64 \times 64$			$\approx$0.134G
Merge 7:	$64 \times 64 \times 64 \times 128$			$\approx$0.268G
ConvBlock 8:	$64 \times 64 \times 64 \times 64$			$\approx$0.268G
Output(CE):	$64 \times 64 \times 64 \times 1$			$\approx$0.001G
Output(CM):	$64 \times 64 \times 64 \times 1$			$\approx$0.001G
Output(CSP):	$64 \times 64 \times 64 \times 1$			$\approx$0.001G
Output(LV):	$64 \times 64 \times 64 \times 1$			$\approx$0.001G
Output(THA):	$64 \times 64 \times 64 \times 1$			$\approx$0.001G
–	–			Sum $\approx$ 2.10G
(b) 3D U-Net				

Table A3
(a). Convolution Block Details. BNorm refers to Batch Normalization (Ioffe and Szegedy, 2015), ReLU is used after each BNorm. (b). Architecture of **Network F** (9 ConvBlocks) and memory cost for single stream during feed-forward process. Theoretically, back-propagation should take the same amount of memory. For all layers, we simply applied 256 convolutional filters. Thus, while training on single input volume, the sum of memory consumption for the whole VP-Nets (three streams) should be only **1.2G**.

Layer type	Conv	BNorm	Conv	BNorm
Filter size	$3 \times 3 \times C$	–	$3 \times 3 \times C$	–
(a) ConvBlock Detail				
Layer type	Output size			Memory Cost
Input:	$128 \times 128 \times 128$			$\approx$0.032G
ConvBlock 1:	$128 \times 128 \times 256$			$\approx$0.032G
MaxPooling 1:	$64 \times 64 \times 256$			$\approx$0.004G
ConvBlock 2:	$64 \times 64 \times 256$			$\approx$0.008G
MaxPooling 2:	$32 \times 32 \times 256$			$\approx$0.0012G
ConvBlock 3:	$32 \times 32 \times 256$			$\approx$0.0024G
MaxPooling 3:	$16 \times 16 \times 256$			$\approx 2.6e-4$G
ConvBlock 4:	$16 \times 16 \times 256$			$\approx 5.2e-4$G
MaxPooling 4:	$8 \times 8 \times 256$			$\approx 6.4e-5$G
ConvBlock 5:	$8 \times 8 \times 256$			$\approx 1.3e-4$G
Deconv 5:	$16 \times 16 \times 256$			$\approx 5.2e-4$G
Merge 5:	$16 \times 16 \times 512$			$\approx 1.0e-3$G
ConvBlock 6:	$16 \times 16 \times 256$			$\approx 1.0e-3$G
Deconv 6:	$32 \times 32 \times 256$			$\approx$0.0024G
Merge 6:	$32 \times 32 \times 512$			$\approx$0.0048G
ConvBlock 7:	$32 \times 32 \times 256$			$\approx$0.0048G
Deconv 7:	$64 \times 64 \times 256$			$\approx$0.004G
Merge 7:	$64 \times 64 \times 512$			$\approx$0.008G
ConvBlock 8:	$64 \times 64 \times 256$			$\approx$0.008G
Deconv 8:	$128 \times 128 \times 256$			$\approx$0.016G
Merge 8:	$128 \times 128 \times 512$			$\approx$0.032G
ConvBlock 9:	$128 \times 128 \times 256$			$\approx$0.032G
Output(CE):	$128 \times 128 \times 1$			$\approx 2.6e-4$G
Output(CM):	$128 \times 128 \times 1$			$\approx 2.6e-4$G
Output(CC):	$128 \times 128 \times 1$			$\approx 2.6e-4$G
Output(LV):	$128 \times 128 \times 1$			$\approx 2.6e-4$G
Output(THA):	$128 \times 128 \times 1$			$\approx 2.6e-4$G
–	–			Sum $\approx$ 0.2G
(b) Single Stream VP-Net				

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.media.2018.04.004.

References

Archibald, S.L., Fennema-Notestine, C., Gamst, A., Riley, E.P., Mattson, S.N., Jernigan, T.L., 2001. Brain dysmorphology in individuals with severe prenatal alcohol exposure. Dev. Med. Child Neurol. 43 (03), 148–154.

Arısoy, R., Yayla, M., 2010. Nomogram of fetal cisterna magna width at 15-24th gestational weeks. Perinatal J.

Baumgartner, C.F., Kamnitsas, K., Matthew, J., Fletcher, T.P., Smith, S., Koch, L.M., Kainz, B., Rueckert, D., 2017. Sononet: real-time detection and localisation of fetal standard scan planes in freehand ultrasound. IEEE Trans. Med. Imaging 36 (11), 2204–2215. doi:10.1109/TMI.2017.2712367.

Carroll, S., Porter, H., Abdel-Fattah, S., Kyle, P., Soothill, P., 2000. Correlation of prenatal ultrasound diagnosis and pathologic findings in fetal brain abnormalities. Ultrasound Obstet. Gynecol. 16 (2), 149–153.

Chang, T., Robson, S.C., Spencer, J.A., Gallivan, S., 1993. Ultrasonic fetal weight estimation: analysis of inter-and intra-observer variability. J. Clin. Ultrasound 21 (8), 515–519.

Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O., 2016. 3d u-net: Learning dense volumetric segmentation from sparse annotation. In: MICCAI. MICCAI, p. MICCAI.

Criminisi, A., Shotton, J., Robertson, D., Konukoglu, E., 2010. Regression forests for efficient anatomy detection and localization in ct studies. In: International MICCAI Workshop on Medical Computer Vision. Springer, pp. 106–117.

Gao, Y., Maraci, M.A., Noble, J.A., 2016. Describing ultrasound video content using deep convolutional neural networks. In: Proc. IEEE 13th Int. Symp. Biomedical Imaging (ISBI), pp. 787–790. doi:10.1109/ISBI.2016.7493384.

Ghesu, F.C., Krubasik, E., Georgescu, B., Singh, V., Zheng, Y., Hornegger, J., Comaniciu, D., 2016. Marginal space deep learning: efficient architecture for volumetric image parsing. IEEE Trans. Med. Imaging 35 (5), 1217–1228.

Hadlock, F., Deter, R., Harrist, R., Park, S., 1982. Fetal head circumference: relation to menstrual age. Am. J. Roentgenol. 138 (4), 649–653.

Hadlock, F.P., Harrist, R., Sharman, R.S., Deter, R.L., Park, S.K., 1985. Estimation of fetal weight with the use of head, body, and femur measurementsa prospective study. Am. J. Obstet. Gynecol. 151 (3), 333–337.

Huang, R., Namburete, A.I., Yaqub, M., Noble, J.A., 2015. Automated mid-sagittal plane selection for corpus callosum visualization in 3d fetal ultrasound images. MIUA.

Ioffe, S., Szegedy, C., 2015. Batch normalization: accelerating deep network training by reducing internal covariate shift. In: International Conference of Machine Learning.

ISUOG, 2007. Sonographic examination of the fetal central nervous system: guidelines for performing the 'basic examination' and the 'fetal neurosonogram'.. Ultrasound Obstet. Gynecol. 29 (1), 109.

Krain, A.L., Castellanos, F.X., 2006. Brain development and adhd. Clin. Psychol. Rev. 26 (4), 433–444.

Liu, F., Zhang, Z., Lin, X., Teng, G., Meng, H., Yu, T., Fang, F., Zang, F., Li, Z., Liu, S., 2011. Development of the human fetal cerebellum in the second trimester: a post mortem magnetic resonance imaging evaluation. J. Anat. 219 (5), 582–588.

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440.

Lu, W., Tan, J., Floyd, R., 2005. Automated fetal head detection and measurement in ultrasound images by iterative randomized hough transform. Ultrasound Med. Biol. 31 (7), 929–936.

Lu, X., Xu, D., Liu, D., 2016. Robust 3d organ localization with dual learning architectures and fusion. In: International Workshop on Large-Scale Annotation of Biomedical Data and Expert Label Synthesis. Springer, pp. 12–20.

Milletari, F., Navab, N., Ahmadi, S.-A., 2016. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: 2016 Fourth International Conference on 3D Vision (3DV). IEEE, pp. 565–571.

Namburete, A.I., Noble, J.A., 2013. Fetal cranial segmentation in 2d ultrasound images using shape properties of pixel clusters. In: Biomedical Imaging (ISBI), 2013 IEEE 10th International Symposium on. IEEE, pp. 720–723.

Namburete, A.I., Stebbing, R.V., Kemp, B., Yaqub, M., Papageorghiou, A.T., Noble, J.A., 2015. Learning-based prediction of gestational age from ultrasound images of the fetal brain. Med. Image Anal. 21 (1), 72–86.

Papageorghiou, A.T., Ohuma, E.O., Altman, D.G., Todros, T., Ismail, L.C., Lambert, A., Jaffer, Y.A., Bertino, E., Gravett, M.G., Purwar, M., et al., 2014. International standards for fetal growth based on serial ultrasound measurements: the fetal growth longitudinal study of the intergrowth-21 st project. The Lancet 384 (9946), 869–879.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. In: MICCAI. Springer, pp. 234–241.

Russakovsky, O., Fei-Fei, L., 2010. Attribute learning in large-scale datasets. In: European Conference on Computer Vision. Springer, pp. 1–14.

Serhatlioglu, S., Kocakoc, E., Kiris, A., Sapmaz, E., Boztosun, Y., Bozgeyik, Z., 2003. Sonographic measurement of the fetal cerebellum, cisterna magna, and cavum septum pellucidum in normal fetuses in the second and third trimesters of pregnancy. J. Clin. Ultrasound 31 (4), 194–200.

Simonyan, K., Vedaldi, A., Zisserman, A., 2014. Deep inside convolutional networks: visualising image classification models and saliency maps. Workshop at ICLR.

Sofka, M., et al., 2014. Automatic detection and measurement of structures in fetal head ultrasound volumes using sequential estimation and integrated detection network (idn). IEEE TMI 33 (5), 1054–1070.

Sudre, C.H., Li, W., Vercauteren, T., Ourselin, S., Cardoso, M.J., 2017. Generalised Dice Overlap as a Deep Learning Loss Function for Highly Unbalanced Segmentations. In: Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support. Springer, pp. 240–248.

Tatarchenko, M., Dosovitskiy, A., Brox, T., 2017. Octree generating networks: efficient convolutional architectures for high-resolution 3d outputs. arXiv preprint arXiv:1703.09438.

Tieleman, T., Hinton, G., 2012. Lecture 6.5-rmsprop, coursera: neural networks for machine learning. Tech. Rep. University of Toronto.

Vanhoucke, V., 2016. Lecture notes in deep learning by google.

de Vos, B., Wolterink, J., de Jong, P., Leiner, T., Viergever, M., Isgum, I., 2017. Convnet-based localization of anatomical structures in 3d medical images. IEEE Trans. Med. Imaging.

Zhou, X., Yamaguchi, S., Zhou, X., Chen, H., Hara, T., Yokoyama, R., Kanematsu, M., Fujita, H., 2013. Automatic organ localizations on 3d ct images by using majority-voting of multiple 2d detections based on local binary patterns and haar-like features. SPIE Medical Imaging. International Society for Optics and Photonics. 86703A–86703A