



Combining Deep Learning and Multi-atlas Label Fusion for Automated Placenta Segmentation from 3DUS

Baris U. Oguz¹, Jiancong Wang¹, Natalie Yushkevich¹, Alison Pouch¹,
James Gee¹, Paul A. Yushkevich¹, Nadav Schwartz², and Ipek Oguz¹(✉)

¹ Penn Image Computing and Science Laboratory (PICSL),
Department of Radiology, University of Pennsylvania,
Philadelphia, PA, USA
ipek@cs.unc.edu

² Maternal and Child Health Research Program, Department of OBGYN,
University of Pennsylvania, Philadelphia, PA, USA

Abstract. In recent years there is growing interest in studying the placenta in vivo. However, 3D ultrasound images (3DUS) are typically very noisy, and the placenta shape and position are highly variable. As such, placental segmentation efforts to date have focused on interactive methods that require considerable user input, or automated methods with relatively low performance and various limitations. We propose a novel algorithm using a combination of deep learning and multi-atlas joint label fusion (JLF) methods for automated segmentation of the placenta in 3DUS images. We extract 2D cross-sections of the ultrasound cone beam with a variety of orientations from the 3DUS images and train a convolutional neural network (CNN) on these slices. We use the prediction by the CNN to initialize the multi-atlas JLF algorithm. The posteriors obtained by the CNN and JLF models are combined to enhance the overall segmentation performance. The method is evaluated on a dataset of 47 patients in the first trimester. We perform 4-fold cross-validation and achieve a mean Dice coefficient of 86.3 ± 5.3 for the test folds. This is a substantial increase in accuracy compared to existing automated methods and is comparable to the performance of semi-automated methods currently considered the bronze standard in placenta segmentation.

1 Introduction

Adverse pregnancy outcomes such as preeclampsia and intrauterine growth restriction contribute to perinatal morbidity and mortality. Given mounting evidence [1–3] that placental abnormalities are related to such outcomes, placental morphometry has recently become major foci of study. Ultrasound imaging is the most common modality to study the placenta in clinical settings due to its low cost, wide availability and ease of acquisition. 2D and 3D ultrasound have

been used to study early, in utero placental size and morphology in relation to birthweight and preeclampsia [14, 16]. While placental biometry may be helpful in clinical care [15], there is currently no clinical tool to evaluate placental morphology due to the lack of automated segmentation methods.

Automated quantification of the placenta from 3D ultrasound images (3DUS) is challenging. 3DUS images are prone to high levels of speckle noise, and the contrast between the placenta and uterine tissue is especially weak in early pregnancy. Additionally, the position of the placenta with respect to the amniotic sac is highly variable which makes it difficult to even detect the placenta automatically, much less determine its precise boundaries. In particular, placentas can be positioned anteriorly or posteriorly to the amniotic sac. Maternal habitus and fetal shadowing artifacts can further obscure the placenta boundary, especially in posterior placentas. The size and shape of the placenta are variable, and uterine contractions can dramatically affect the shape of the placenta.

Many current techniques for placental segmentation from 3DUS rely on user input to overcome these challenges. These include the commercial VOCAL software (GE Healthcare) and a random-walker (RW) algorithm [4, 17]. Such interactive methods are time-consuming, subjective and prone to intra- and inter-observer variability, which makes automated methods more attractive. Only 4 fully automated approaches have been proposed to date.

1. A recurrent neural network [19]. While this is a more general-purpose segmentation framework that aims to segment not just the placenta but also the gestational sac and the fetus, the placenta segmentation accuracy achieved with this method is rather limited (average Dice of 0.64 is reported in [19]).
2. A multi-atlas label fusion algorithm [10]. Since this is a registration-based method, it suffers from registration initialization robustness issues. The fully automated version of the algorithm relies on initialization based on the ultrasound cone beam, but is limited to anterior placentas (mean Dice of 0.83 in anteriors reported in [10]). For generalization to non-anterior placentas, manual initialization on a 2D slice is required to guide the registration [11].
3. A deep convolutional neural network (CNN), DeepMedic [8]. The “ground truth” segmentations are produced via the RW algorithm [17], as opposed to fully manual expert annotations used in other studies. While this algorithm is readily applicable to anterior and posterior placentas, the performance leaves room for improvement (median Dice of 0.73 reported in [8]).
4. An improved version of [8], with a 3D fully convolutional network (OxNNNet) [9]. Perhaps the most distinguishing aspect of this paper is the significantly larger amount of training data. They utilize 2400 labelled 3DUS images in a 2-fold cross validation setup. They report a mean Dice score of 0.81. Like [8], this used the results of the random walker algorithm as “truth”.

We propose a novel technique that combines the strengths of the deep learning and multi-atlas label fusion (MALF) approaches for fully automated segmentation of the placenta from 3DUS images. MALF methods require good registration initialization and have limited performance in hard-to-register patches such

as thin features and low-contrast boundaries. CNN’s [6] are more robust to such problems but tend to be noisier, and are hard to train in 3D given the sparsity of training data in medical image analysis; as such, their results can lack 3D shape context. Triplanar CNN’s were studied in [12] as a workaround. In our proposed algorithm, we begin with a 2D CNN to construct an initial prediction and we leverage the rotational ambiguity of the ultrasound cone beam for an innovative data augmentation strategy. Next, we deploy a MALF algorithm using the CNN results to initialize the registrations. We use a second-tier model to combine the posterior maps from the CNN model and the MALF method.

2 Methods

Our algorithm begins with pre-processing and extracting 2D slices from the 3DUS images and trains a convolutional neural network (CNN) with online random augmentation to create an initial prediction in 3D. Next, multi-atlas joint label fusion (JLF) is applied on the CNN output to construct an alternative prediction. The predictions from CNN and JLF are combined via a second tier random forest (RF) model to obtain the final segmentation result (Fig. 1a).

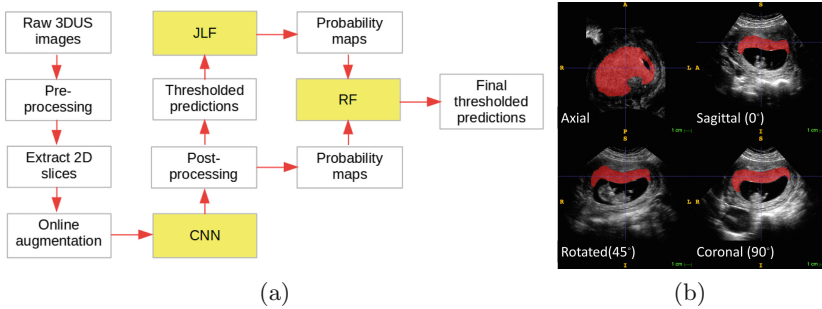


Fig. 1. (a) Workflow of our algorithm. (b) Placenta cross-sections from various views.

2.1 Dataset, Pre-processing and Augmentation

Our dataset consists of first trimester 3D US images from 47 subjects with singleton pregnancies, acquired with GE Voluson E8 ultrasound machines. 28 subjects had anterior placentas, and 19 had posterior placentas. Each image has isotropic resolution (mean: 0.47 mm, min: 0.35 mm, max: 0.57 mm). Each image was manually segmented by N.Y. under the supervision of N.S., who has over 10 years of experience in prenatal ultrasound imaging and who has segmented 100’s of placentas for other research endeavors. The ITK-SNAP software¹ was used for segmentation. The main metric for evaluating the pipeline is the Dice overlap between automated results and these manual segmentations.

¹ www.itksnap.org.

26 of the 47 subjects were imaged twice within the same session. Patients were allowed to move around between the two acquisitions. These secondary images were used in a reproducibility experiment, described in Sect. 2.6 below.

The images in our dataset have various dimensions and may contain sizable blank spaces around the edges. Thus, the images are cropped to a 3D bounding box of the ultrasound cone beam (automatically, by simply thresholding at 0), and downsampled to standard dimensions (128^3).

We extract 2D slices from the original 3D images and train a CNN with 2D convolutions. Our experiments showed that the axial view presents little information to the CNN regarding the placenta. Therefore, we extract slices only in the sagittal (0°) and coronal (90°) planes, and a 45° plane between them. This leads to a stack of $128 \times 3 = 384$ slices per subject, along 3 different orientations. Given the rotational ambiguity of the ultrasound probe, these 3 orientations contain similar cross-sections of the placenta (Fig. 1b).

To further mitigate the impact of a small training dataset, we use online random augmentation. Various transforms are applied to each 2D slice before it is seen by the CNN. These transforms consist of horizontal/vertical flipping, 2D translation, 2D rotations (in-plane), and scaling. Whether each transformation is applied, and if relevant its magnitude, is determined randomly. Translation magnitudes are between $-5/+5$ pixels (image size is 128×128). In-plane rotation amount is between $-15/+15^\circ$. Scaling is performed by zooming in 0–10%.

2.2 Convolutional Neural Network

Typical CNN’s for classification tasks [6] contain 3 main types of layers: (1) Convolutional layers extract local features by sliding a kernel (aka filter) over the input image to compute a feature map. (2) Pooling layers downsample the feature maps obtained by the convolutional layer, commonly applying a function like max, mean, sum, etc. (3) Fully connected layers consolidate the features from convolutional and pooling layers, and output probabilities for each class.

The strength of convolutional layers and CNN’s comes from their ability to preserve spatial relationships in the input images. The main parameters that affect the performance of a convolutional layer are the kernel size, the strides, and the number of feature maps. Kernel size and strides are often chosen empirically in relation to the input image dimensions. The number of feature maps determines the number of trainable parameters in the layer and must be chosen with consideration of under/over-fitting.

Initial convolutional layers usually extract low-level features such as edges, corners, etc. Successive layers build on top of previously extracted feature maps and discover higher-level features significant for the task. Pooling layers condense the information from convolutional layers, by retaining whether a feature is present in the active kernel window, but dismissing the exact location within the kernel. While this quality is favorable in many vision tasks where the goal is to detect a higher level entity (cat, car, face) in the image, in our preliminary experiments we found that using pooling layers degrades the mean Dice results of CNN predictions. This might be because the task of segmentation requires a

decision for each voxel in the input. However, further investigation remains for future work; in the work presented here, pooling layers were not used.

U-net [13] proposes a multi-channel approach to the problem of retaining location information. They use max-pooling layers for downsampling but still present the unpooled feature maps to the final convolution by side channels. In our approach, we utilize convolutional layers with larger strides instead of pooling, to retain the benefit of condensing feature information, while keeping the location information intact. This effectively downsamples the feature maps from the convolutional layers, while preventing systematic loss of information.

Fully connected layers are used when the dimensions of the output are much smaller relative to the input (classification, bounding box locations, etc.). In the segmentation task, the dimensions of the output are identical to that of the input. To reach the required output dimensions from the downsampled feature maps, we apply upsampling via transposed convolution (aka deconvolution) layers.

Our CNN architecture consists of 23 layers in total: 1 batch normalization, 15 ordinary convolutions (1×1 strides), 3 downsampling convolutions (2×2 strides), 3 upsampling deconvolutions (2×2 strides), and 1 sigmoid output (Fig. 2). All the kernels we use in the convolutional layers have the dimensions of 3×3 , and ReLU was used as activation function in all intermediate layers.

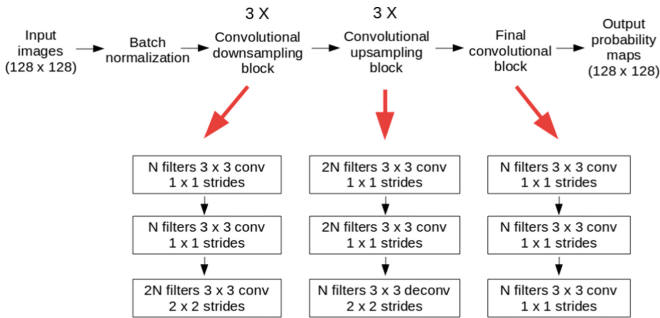


Fig. 2. Our CNN architecture contains 23 layers in total: 1 normalization + 18 convolutional + 3 deconvolutional + 1 sigmoid. There are 2,730,627 trainable parameters.

In the training phase, we use 384 2D slices for each subject (128 each from 0° , 45° , and 90° orientations), and train a single CNN. In the prediction phase, we produce 4 probability maps for each 2D slice, by flipping the slice horizontally and vertically. The mean of the 4 orientations is used as the prediction for that slice. For each subject, we predict for the three orientations above, obtaining three 3D probability maps. To provide an initialization for the JLF step, we take the mean of these three maps and binarize by a threshold.

The CNN code is implemented on the Python/Numpy stack. We used TensorFlow for the CNN backend and Keras as a higher level frontend. The CNN training is run on a machine with 4x Nvidia GTX-1080Ti GPUs. The online

random augmentation is performed on the CPU, in parallel to the CNN working on the GPU. One epoch of training takes around 2–3 min, the final model at 35 epochs lasts around 90 min. Due to our cross-validation setup, we trained 4 such models, one for each fold. The total time for the CNN part of the pipeline takes around 8 h including pre/post processing. The prediction phase is much faster and completes within minutes for the entire dataset.

2.3 Joint Label Fusion (JLF)

Registration-based segmentation methods are popular in medical image analysis. In the simplest form, an atlas image is created via manual annotation. This atlas image is registered deformably to the target image, and the segmentation of the target is calculated by applying the same deformation to the atlas annotation. This single atlas method does not generalize well, due to the large variability in shape, size, and location of anatomical structures. The next evolution of this method utilizes multiple atlases instead of a single one [5]. Each atlas produces a solution to the target image, and these solutions are combined in a label fusion step. The fusion is typically based on a weighted voting scheme, usually taking into consideration the similarity of each atlas to the target image.

Specifically, we use the publicly available JLF approach [18] for label fusion, which jointly estimates the weights for all available atlases, minimizing the redundancy from correlated atlases. These weights are given by $w_x = \frac{M_x^{-1}1_n}{1_n^t M_x^{-1}1_n}$, where $1_n = [1; 1; \dots; 1]$ is a vector of size n and t stands for transpose. The dependency matrix M is estimated by $M_x(i, j) = [|A^i(N(x)) - T(N(x))| \cdot |A^j(N(x)) - T(N(x))|]^\beta$, where $|A^i(N(x)) - T(N(x))|$ is the vector of absolute intensity difference between a warped atlas A^i and the target image over a local patch $N(x)$ centered at x and \cdot is the dot product. β is a model parameter. Default values were used for all parameters in the JLF implementation [18].

All registrations were done using the open source greedy package². We use the binary prediction from the CNN as a mask to guide the affine registrations, which use the SSD metric. The subsequent deformable registrations do not use a mask as the affine registration provides a stable enough initialization; the normalized cross correlation (NCC) metric is used with a $4 \times 4 \times 4$ patch size.

2.4 Combining CNN and JLF Results

Our CNN step produces 3 probability maps for each subject (for 0° , 45° , and 90° orientations). We average and threshold these to provide input for the JLF step. We obtain another probability map from the JLF, ending up with 4 maps. Simply averaging these maps offers limited Dice improvement over the individual probability maps. To extract more information from the 4 maps, we utilize a second tier model. This model takes the 4 probabilities for each pixel and additional statistical features (min, mean, max, stdev, max-min) to produce a final probability. We use a random forest model with 50 trees and max depth of 6.

² <https://sites.google.com/view/greedyreg/home>.

2.5 Experimental Methods for Cross-Validation

Given the relatively small dataset, we opted for a two-tiered 4-fold cross-validation (cv) scheme. At each cv step of the 1st tier, we used 3 folds for training a CNN, and the last fold for testing, ending up with 4 CNN’s. Each of these CNN’s use identical architecture and parameters, to prevent overfitting to a single test fold.

These unseen test fold predictions are then used as a registration mask in the JLF part of the pipeline. For each subject, the manual segmentations from all training subjects are used as atlases (i.e., training), using the same train/test split as the CNN folds. Thus, for each of the test folds, we end up with 3 probabilities (for 0° , 45° , and 90° orientations) from a CNN and the JLF probability.

To train the 2nd tier RF, we apply an inner 4-fold cv to each test fold from the CNN/JLF tier, for a total of 16 RF models with identical parameters. Finally, we approximate the performance on an independent validation set by taking the mean of the results from the 16 unseen test folds of the 2nd tier inner cv.

2.6 Reproducibility Experiment

During the recording of 3DUS images, movements of the subject and the fetus, specific position of the ultrasound probe, and other factors may result in significant variations in image appearance. 26 of the 47 subjects in our dataset have secondary images taken within the same session, which we use to test the reproducibility of our algorithm. In this experiment, we trained our CNN model on the 21 subjects with only one image. We performed the JLF step with using only these 21 subjects as atlases. We trained the RF models via an inner 4-fold on these 21 subjects. Using this pipeline trained on the disjoint set of 21 subjects, we segmented the placenta for the 26 test subjects, 2 segmentations for each subject (one per image). We calculated the volume of the segmentations, and evaluated the correlation of volume between the pairs of images.

3 Results

Our results are summarized in Fig. 3. The proposed combination of CNN and JLF results via a second tier RF model outperforms the individual methods.

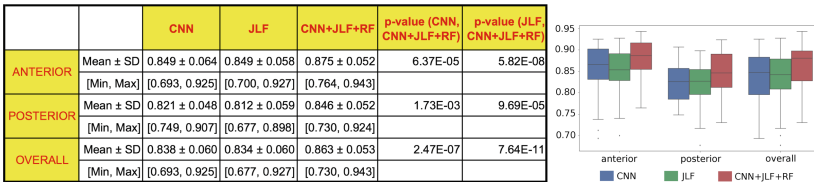


Fig. 3. Dice scores for test folds. P-values from paired two-tailed t-tests are reported.

These differences were found to be highly significant ($p < 0.001$) in paired two-tailed t -tests. We note that all compared methods have lower performance in the posterior placentas where fetal shadowing artifacts are common. Qualitative results are shown in Fig. 4. The reproducibility results are shown in Fig. 5-a.

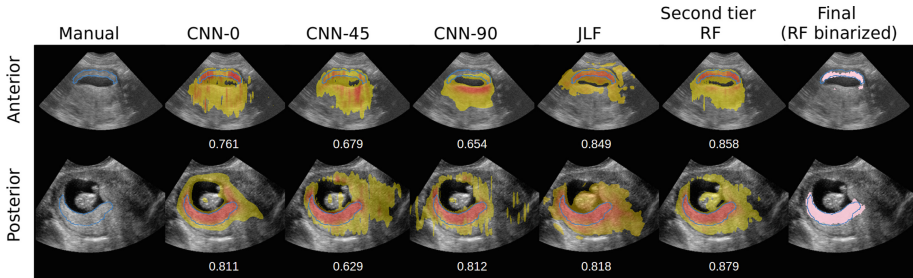


Fig. 4. Qualitative results from an anterior and a posterior subject. In the first subject, CNN models are erroneously drawn to bright regions, but the JLF accurately captures the placenta. In the second subject, JLF oversegments the placenta due to weak boundaries, but CNN models capture the correct result. In both subjects, the second-tier RF model effectively combines the two approaches.

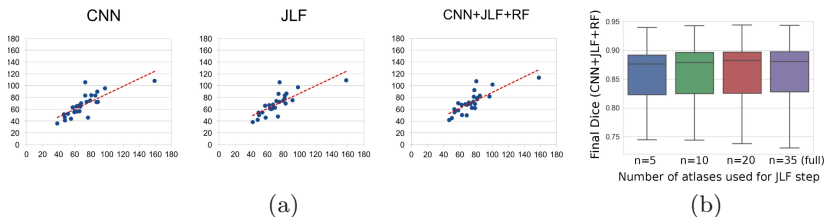


Fig. 5. (a) Test-retest volumes (measured in ml) at each stage of our pipeline are shown for 26 pairs of images. The Pearson correlation coefficient between volume test-retest measurements was 0.786 for CNN, 0.787 for CNN+JLF, and 0.797 for the final (CNN+JLF+RF) method. Final correlation was 0.848 when one outlier was removed, and 0.874 when 3 outliers were removed. (b) Final Dice scores with various number of atlases for the JLF step. The differences between using the entire training fold ($n = 35$) and only 5 randomly selected subjects were minimal (0.859 vs. 0.863).

4 Discussion

Our Findings. Our combined approach utilizes both the automated power of CNNs and the 3D context of multi-atlas label fusion. Both methods have their strengths and weaknesses, and our second tier RF model effectively blends their results into a more accurate and robust final prediction. Our results (0.863 mean

Dice overall) provide a substantial improvement over existing automated methods (mean Dice of 0.81 reported in [9], median Dice of 0.73 reported in [8], mean Dice of 0.64 reported in [19], mean Dice of 0.83 reported for anterior-only placentas in [10]), and are comparable to the performance of semi-automated methods (mean Dice scores of 0.80 reported in [11] and 0.86 reported in [17]), which require manual input and may have reproducibility issues.

Comparison to Other Network Architectures. A common task in computer vision is classification, including binary, multi-class and multi-label problems. In these settings, the output is one or more ordinal labels for each input image. Common architectures for this task contain convolution and downsampling layers in the first half of the network. The second half contains fully connected dense layers, producing predicted labels from the inner representation of the first half. Image segmentation differs from classification, requiring an output for each input voxel. Thus, most common architectures are variations of the fully convolutional network [7]. In this setup, the second half contains upsampling layers, producing a full size prediction from the inner representation. Our CNN is also based on this architecture. In the downsampling layers, we used strided convolutions, which produced better results in our experiments than the pooling approach. We also experimented with U-net [13] style side channels, but that also did not provide much improvement, and extended training time. A formal comparison of these alternative architectures remains as future work.

2D vs. 3D CNNs. In recent years, CNN’s have consistently shown high performance in many visual learning tasks, especially thriving on large amounts of training data. The medical imaging field, in contrast, typically has much less data available. Our annotated 3DUS dataset consists of 47 subjects, which is significantly low in quantity when compared to general image datasets containing multi-million labelled images used for more general CNN tasks. It is difficult to obtain large amounts of labeled placenta images, since the segmentation needs to be manually created by expert annotators in a time-consuming process.

While it is possible to train a CNN on 3D images using 3D convolutions, our dataset size is too small to fully take advantage of such an approach. Training 3D CNN’s also requires much larger computational resources compared to 2D CNN’s. Therefore, we opted for using 2D CNNs on slices from the 3DUS images. Evidently, this leads to a loss of 3D context. We mitigated this shortcoming by extracting 2D slices from three different axes. We also applied random online augmentation during training to further increase the variance in the dataset.

Number of Atlases Needed for MALF. MALF is computationally intensive. The runtime grows linearly with each atlas and each test subject, since each pair needs to be registered. In our dataset consisting of 47 subjects, using all train-fold subjects ($n = 35$) as atlases took around 1000 CPU-hours on a computer cluster, utilizing up to 25 CPUs in parallel. This is longer than the CNN training time. More importantly, in a real-life application, the CNN only needs to be trained once whereas the atlas registrations are needed for each

new test subject. This causes a bottleneck for the practical applicability of our method.

We hypothesized that the main benefit from the JLF step in this application is the access to 3D context, which can be gained from just a handful of atlases. This is unlike other applications where a large set of atlases are needed to adequately represent the underlying distribution of image appearance. To test this hypothesis, we experimented with using a small random subset of subjects in the train fold as atlases, instead of all of them. The results for utilizing 5-10-20 atlases are given in Fig. 5-b. While using the full train fold as atlases gives the best results, the mean Dice when using only 5 randomly chosen subjects is still very closely comparable (0.8588 vs. 0.8631). This finding supports our hypothesis. It also reinforces the main idea behind our approach, which is combining two methodologically different approaches to produce a more robust segmentation. Running the JLF step with even just 5 atlases provides considerable improvement in the Dice scores while reducing the runtime substantially.

Acknowledgments. This work was funded by the NICHD Human Placenta Project (U01 HD087180) and NIH grants R01 EB017255, R01 NS094456 and F32 HL119010.

References

1. Baptiste-Roberts, K., Salafia, C.M., Nicholson, W.K., Duggan, A., Wang, N.Y., Brancati, F.L.: Gross placental measures and childhood growth. *J. Matern.-Fetal Neonatal Med.* **22**(1), 13–23 (2009)
2. Barker, D.J., Bull, A.R., Osmond, C., Simmonds, S.J.: Fetal and placental size and risk of hypertension in adult life. *BMJ* **301**(6746), 259–262 (1990)
3. Biswas, S., Ghosh, S.K.: Gross morphological changes of placentas associated with intrauterine growth restriction of fetuses: a case control study. *Early Hum. Dev.* **84**(6), 357–362 (2008)
4. Collins, S.L., Stevenson, G.N., Noble, J.A., Impey, L.: Rapid calculation of standardized placental volume at 11 to 13 weeks and the prediction of small for gestational age babies. *Ultrasound Med. Biol.* **39**(2), 253–260 (2013)
5. Iglesias, J.E., Sabuncu, M.R.: Multi-atlas segmentation of biomedical images: a survey. *Med. Image Anal.* **24**(1), 205–219 (2015)
6. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: *NIPS*, pp. 1097–1105 (2012)
7. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: *IEEE CVPR*, pp. 3431–3440 (2015)
8. Looney, P., et al.: Automatic 3D ultrasound segmentation of the first trimester placenta using deep learning. In: *IEEE ISBI*, pp. 279–282 (2017)
9. Looney, P., et al.: Fully automated, real-time 3D ultrasound segmentation to estimate first trimester placental volume using deep learning. *JCI Insight* (2018)
10. Oguz, I., et al.: Fully automated placenta segmentation from 3D ultrasound images. In: *Perinatal, Preterm and Paediatric Image Analysis, PIPPI Workshop, MICCAI* (2016)
11. Oguz, I., et al.: Semi-automated 3DUS placental volume measurements with minimal user interaction. *The American Institute of Ultrasound in Medicine* (2018)

12. Prasoon, A., Petersen, K., Igel, C., Lauze, F., Dam, E., Nielsen, M.: Deep feature learning for knee cartilage segmentation using a triplanar convolutional neural network. In: Mori, K., Sakuma, I., Sato, Y., Barillot, C., Navab, N. (eds.) MICCAI 2013. LNCS, vol. 8150, pp. 246–253. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40763-5_31
13. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
14. Schwartz, N., Quant, H.S., Sammel, M.D., Parry, S.: Macrosomia has its roots in early placental development. *Placenta* **35**(9), 684–690 (2014)
15. Schwartz, N., Wang, E., Parry, S.: Two-dimensional sonographic placental measurements in the prediction of small-for-gestational-age infants. *Ultrasound Obstet. Gynecol.* **40**(6), 674–679 (2012)
16. Schwartz, N., et al.: Placental volume measurements early in pregnancy predict adverse perinatal outcomes. *Am. J. Obstet. Gynecol.* **201**(6), S142–S143 (2009)
17. Stevenson, G.N., Collins, S.L., Ding, J., Impey, L., Noble, J.A.: 3-D ultrasound segmentation of the placenta using the random walker algorithm: reliability and agreement. *Ultrasound Med. Biol.* **41**(12), 3182–3193 (2015)
18. Wang, H., Yushkevich, P.: Multi-atlas segmentation with joint label fusion and corrective learning—an open source implementation. *Front. in Neuroinf.* **7**, 27 (2013)
19. Yang, X., et al.: Towards automatic semantic segmentation in volumetric ultrasound. In: Descoteaux, M., Maier-Hein, L., Franz, A., Jannin, P., Collins, D.L., Duchesne, S. (eds.) MICCAI 2017. LNCS, vol. 10433, pp. 711–719. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66182-7_81