Detection of Fetal Anatomies from Ultrasound Images using a Constrained Probabilistic Boosting Tree

Gustavo Carneiro, Bogdan Georgescu, Sara Good, Dorin Comaniciu *Senior Member, IEEE*.

***Abstract*—We propose a novel method for the automatic detection and measurement of fetal anatomical structures in ultrasound images. This problem offers a myriad of challenges, including: difficulty of modeling the appearance variations of the visual object of interest; robustness to speckle noise and signal drop-out; and large search space of the detection procedure. Previous solutions typically rely on the explicit encoding of prior knowledge and formulation of the problem as a perceptual grouping task solved through clustering or variational approaches. These methods are constrained by the validity of the underlying assumptions and usually are not enough to capture the complex appearances of fetal anatomies. We propose a novel system for fast automatic detection and measurement of fetal anatomies that directly exploits a large database of expert annotated fetal anatomical structures in ultrasound images. Our method learns automatically to distinguish between the appearance of the object of interest and background by training a constrained probabilistic boosting tree classifier. This system is able to produce the automatic segmentation of several fetal anatomies using the same basic detection algorithm. We show results on fully automatic measurement of biparietal diameter (BPD), head circumference (HC), abdominal circumference (AC), femur length (FL), humerus length (HL), and crown rump length (CRL). Notice that our approach is the first in the literature to deal with the HL and CRL measurements. Extensive experiments (with clinical validation) show that our system is, on average, close to the accuracy of experts in terms of segmentation and obstetric measurements. Finally this system runs under half second on a standard dual-core PC computer.**

***Index Terms*—Medical Image Analysis, Supervised Learning, Top-down Image Segmentation, Visual Object Recognition, Discriminative Classifier.**

I. Introduction

Accurate fetal ultrasound measurements are one of the most important factors for high quality obstetrics health care. Common fetal ultrasound measurements include: bi-parietal diameter (BDP), head circumference (HC), abdominal circumference (AC), femur length (FL), humerus length (HL), and crown rump length (CRL). In this paper we use the American Institute of Ultrasound in Medicine (AIUM) guidelines [1] to perform such measurements. These measures are used to estimate both the gestational age (GA) of the fetus (i.e., the length of pregnancy in weeks and days [33]), and also as an important diagnostic auxiliary tool. Accurate estimation of GA is important to estimate the date of confinement and the expected delivery date, to assess the fetal size, and to monitor the fetal growth. The current workflow requires expert users to perform those measurements manually, resulting in the following issues: 1) the quality of the measurements are user-dependent; 2) exams can take more than 30 minutes; and 3) expert users can suffer from Repetitive Stress Injury (RSI) due to the multiple keystrokes needed to perform the measurements. Therefore, the automation of ultrasound measurements has the potential of: 1) improving everyday workflow; 2) increasing patient throughput; 3) improving accuracy and consistency of measurements, bringing *expert-like consistency* to every exam; and 4) reducing the risk of RSI to specialists.

We focus on a method that targets the *automatic on-line detection and segmentation* of fetal head, abdomen, femur, humerus, and body length in typical ultrasound images, which are then used to compute BDP and HC for head, AC for abdomen, FL for femur, HL for humerus, and CRL for the body length (see Fig. 5). We concentrate on the following goals for our method: 1) efficiency (the process should last less than one second); 2) robustness to the appearance variations of the visual object of interest; 3) robustness to speckle noise and signal drop-out typical in ultrasound images; and 4) segmentation accuracy. Moreover, we require the basic algorithm to be the same for the segmentation of the different anatomies aforementioned in order to facilitate the extension of this system to other fetal anatomies.

To achieve these goals, we exploit the database-guided segmentation paradigm [13] in the domain of fetal ultrasound images. Our approach directly exploits the expert annotation of fetal anatomical structures in large databases of ultrasound images in order to train a sequence of discriminative classifiers. The classifier used in this work is based on a constrained version of the probabilistic boosting tree [36].

Our system is capable of handling a previously issue in the domain of fetal ultrasound image analysis, which are: the automatic measurements of HL and CRL, and the fact that our approach is designed to be completely automatic. This means that the user does not need to provide any type of initial guess. The only inputs to the system are the image and the measurement to be performed (BPD, HC, AC, FL, HL, or CRL). Extensive experiments show that, on average, the measurement produced by our system is close to the accuracy of the annotation made by experts for the fetal measurements

Manuscript received May 12, 2007; revised March 7, 2008; accepted March 12, 2008. G. Carneiro, B. Georgescu, and D. Comaniciu are with the Integrated Data Systems Department at Siemens Corporate Research, Princeton, NJ, USA. S. Good is with the Innovations Department, Ultrasound Division, Siemens Medical Solutions, Mountain View, CA, USA.

mentioned above. Moreover, the algorithm runs under half second on a standard dual core PC computer[1].

A. Paper Organization

This paper is organized as follows. Section II presents a literature review, Section III defines the problem, and in Section IV we explain our method. Finally, Section V shows the experiments, and we conclude the paper in Section VI.

II. Literature Review

In this literature review we survey papers that aim at the same goals as ours, which are: precise segmentation, robustness to noise and to the visual class intra variability, and fast processing. First, we focus on the papers that describe approaches for detecting and segmenting fetal anatomies in ultrasound images. Then, we survey methods designed to work on the segmentation of anatomical structures from ultrasound images that, in principle, could also be applied to our problem. We also discuss relevant computer vision techniques for detection and segmentation since our method is closely related to these computer vision methods. Finally, we explain the main novelties of our approach compared to the state-of-the-art in the fields of computer vision, machine learning, and medical image analysis.

There is relatively little work in area of automatic segmentation of fetal anatomies in ultrasound images [6], [7], [14], [17], [23], [28], [35]. One possible reason for this, as mentioned by Jardim [17], is the low quality of fetal ultrasound images, which can be caused by low signal-to-noise ratio, markedly different ways of image acquisition, large intra class variation because of differences in the fetus age and the dynamics of the fetal body (e.g., the stomach in the abdomen images can be completely full or visually absent, and the shape of the fetal body changes significantly in terms of the gestational age - see Fig. 7), strong shadows produced by the skull (in head images), spine and ribs (in abdomen images), femur, and humerus. A noticeable commonality among the papers cited above is their focus on the detection and segmentation of only fetal heads and femurs, but not fetal abdomen (except for [7]), humerus, or body. Among these anatomies, the fetal head segmentation is the least complicated due to the clear boundaries provided by the skull bones, and the similar texture among different subjects (see Fig. 7-(a)). The problem of femur and humerus segmentation is somewhat more complicated because of the absence of internal texture (see Fig. 7-(c,d)), but the presence of clear edges produced by the imaging of the bones facilitates the problem. Finally, the segmentation of the fetal abdomen and fetal body are the hardest among these anatomies. The fetal abdomen presents a lack of clear boundaries and inconsistent imaging of the internal structures among different subjects (see Fig. 7-(b)), while the fetal body changes its shape considerably as a function of the fetal age (see Fig. 7-(e)).

The initial approaches for automatic fetal anatomical segmentation in ultrasound images were mostly based on morphological operators [14], [23], [35]. These methods involve a series of steps, such as edge detection, edge linking, Hough transform, among other standard computer vision techniques, to provide head and femur segmentation. When compared to the measurements provided by experts, the segmentation results showed correlation coefficients bigger than 0.97 (see Eq. 21). However, a different method had to be implemented for each anatomy, showing the lack of generalization of such algorithms. Also, the segmentation of abdomen has not been addressed. Finally, the implemented systems needed a few minutes to run segmentation process.

Chalana et al. [6], [7], [28] describe a method for fetal head and abdomen segmentation in ultrasound images based on the active contour model. This method can get stuck at local minima, which might require manual correction. Also, the algorithm does not model the texture inside the fetal head, which means that no appearance information is used to improve the accuracy and robustness of the approach. Experiments on 30 cases for BPD, HC, and AC, show that the algorithm performs as well as five sonographers, and that it runs in real time. Finally, another issue is that the user needs to provide an initial guess for the algorithm, which makes the system semi-automatic.

Jardim and Figueiredo [17] present a method for the segmentation of fetal ultrasound images based on the evolution of a parametric deformable shape. Their approach segments the input image into two regions, so that pixels within each region have similar texture statistics according to a parametric model defined by the Rayleigh distribution. A drawback of this method is that there is no guarantee that the algorithm will always find the optimal solution, which is a fact noted by the authors. Another limitation is that the appearance model based on the Rayleigh distribution cannot take into account the spatial structure of textural patterns present inside the cranial cross-section. This method also needs an initial guess from the user, which makes the system semi-automatic. The authors use this approach for the segmentation of fetal heads and femurs in 50 ultrasound images with good results.

The segmentation of other anatomies from ultrasound images has also produced relevant solutions that can be applied to the problem of segmentation of fetal anatomical structures. Thus, in this section we focus on methods designed to work on problems involving similar challenges, which are: low quality of ultrasound images, large intra class variation, and strong shadows produced by the anatomical structure. Several techniques have been proposed [29], but we shall focus this review on the following promising techniques: pixel-wise and region-wise classifier models, low-level models, Markov random field models, machine learning based models, and deformable models.

The most promising techniques in this area are based on a combination of region-wise classifier models and deformable models, where an evolving contour defines a partition of the image into two regions. Assuming a parametric distribution for each region, one can have a term of appearance coherence for each region in the optimization algorithm for the deformable model [5], [40]. This is a similar approach to the paper above by Jardim [17], and consequently shares the same problems that makes it not ideal for our goals. Level set representations

[1]Intel Core 2 CPU 6600 at 2.4GHz, 2GB of RAM

that integrates boundary-driven flows with regional information [25], [34] can handle arbitrary initial conditions, which makes these approaches completely automatic, but they are sensitive to noise and incomplete data. The latter problem has been dealt with by adding a shape influence term [19], [26]. The most prominent similarity among these techniques is the under utilization of the appearance model of the anatomical structure being detected. The parameter estimation of the probability distributions for the foreground and background regions is clearly insufficient to model the complex appearance patterns for several reasons. First, the parametric distribution might not provide a reasonable representation for the appearance statistics. Second, the parameters may not be correctly estimated using only the image being processed. Third, the spatial structure of the texture cannot be captured with such representation. In general, these techniques tend to work well whenever image gradients separate the sought anatomical structure, but recall that for abdomens, this assumption may not always be true, so one has to fully rely on its internal appearance for proper segmentation.

The use of deformable models alone has also been exploited [2], but the lack of a learning scheme for the appearance term restricts their applicability to our problem. Moreover, the priors assumed for the anatomical structure and imaging process does not generalize well for fetal anatomical structures in ultrasound images, and even though Akgul et al. [2] work on the local minima issues of such approaches, their design only alleviates the problem. Deformable models can also be used with machine learning techniques to learn shape and motion patterns of anatomical structures [16]. However, the lack of a term representing appearance characteristics of the anatomical structure in [16] restricts the applicability of this method to our problem. Typically, the issue of low signal-to-noise ratio has been solved with the utilization of a sequence of low-level models [22], [27]. However, it is not clear whether these methods can generalize to all possible different imaging conditions that we have to deal with. Finally, an interesting area of research is the use of pixel-wise posterior probability term using a Markov random field prior model [38]. The main problems affecting such approaches are the difficulty in determining the parameters for spatial interaction [29], and the high computational costs that limits its applicability for on-line methods.

More generally, in the fields of computer vision and machine learning there has been a great interest in the problem of accurate and robust detection and segmentation of visual classes. Active appearance models [9] use registration to infer the shape associated with the current image. However, modeling assumes a Gaussian distribution of the joint shape-texture space and requires initialization close to the final solution. Alternatively, characteristic points can be detected in the input image [10] by learning a classifier through boosting [10], [37]. The most accurate segmentation results have been presented by recently proposed techniques that are based on strongly supervised training, and the representation is based on parts, where both the part appearance and the relation between parts, is modeled as a Markov random field or conditional random field [4], [15], [18], [20], [21]. Although the segmentation results presented by such approaches are excellent, these algorithms are computationally intensive, which makes on-line detection a hard goal to be achieved. Also, the use of parts is based on the assumption that the visual object of interest may suffer severe non-rigid deformations or articulation, which is not true in the domain of fetal anatomical structure segmentation.

The method we propose in this paper is more aligned with the state-of-the-art detection and top-down segmentation methods proposed in computer vision and machine learning. Specifically, we exploit the database-guided segmentation paradigm [13] in the domain of fetal ultrasound images. In addition to the challenges present in echocardiography [13], our method has to handle new challenges present in fetal ultrasound images, such as the extreme appearance variability of the fetal abdomen and fetal body imaging, generalization to the same basic detection algorithm to all anatomical structures, and extreme efficiency. In order to cope with these new challenges, we constrain the recently proposed probabilistic boosting tree classifier [36] to limit the number of nodes present in the binary tree, and also to divide the original classification into hierarchical stages of increasing complexity.

III. Automatic Measurement of Fetal Anatomy

Our method is based on a learning process that implicitly encodes the knowledge embedded in expert annotated databases. This learning process produces models that are used in the segmentation procedure. The segmentation is then posed as a task of *structure detection*, where the system automatically segments an image region containing the sought structure. Finally, the fetal measurements can be derived from this region.

A. Problem Definition

The ultimate goal of our system is to provide a segmentation of the most likely rectangular image region containing the anatomical structure of interest. From this rectangular region, it is possible to determine the measurements of interest (i.e., BPD, HC, AC, FL, HL, and CRL), as shown below. We adopt the following definition of segmentation: assume that the image domain is defined by $I : \Re^{N \times M} \rightarrow \Re$ with N denoting the number of rows and M the number of columns, then the segmentation task determines the sets $S, B \subset I$, where S represents the foreground region (i.e., the structure of interest), and B means the background. The sets satisfy the constraint $S \bigcup B = I$, where $S \cap B = \emptyset$. The foreground image region S is determined by the following vector:

$$\theta = [x, y, \alpha, \sigma_x, \sigma_y], \tag{1}$$

where the parameters (x, y) represent the top left region position in the image, α denotes orientation, and (σ_x, σ_y), the region scale (see Fig. 1).

The appearance of the image region is represented with features derived from the Haar wavelets [30], [37]. The decision for the use of such feature set is based on two main reasons: 1) good modeling power for the different types of visual patterns, such as pedestrians [30], faces [37], and left ventricles in

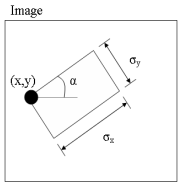


Fig. 1. Foreground (rectangular) image region with five parameters.

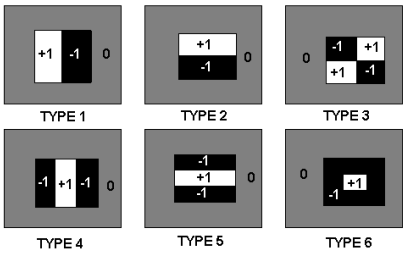


Fig. 2. Image feature types used. Notice that the gray area represents the foreground region S.

ultrasound images [13]; and 2) computation efficiency with the use of integral images. All the feature types used in this work are displayed in Fig. 2, and each feature is denoted by the following feature vector:

$$\theta_f = [t, x_f, y_f, d_x, d_y, s], \tag{2}$$

where $t \in \{1, ..., 6\}$ denotes the feature type, (x_f, y_f) is the top-left coordinate of the feature location within S defined by θ in Eq. 1 (i.e., $x_f \in [1, 1 + (\sigma_x - d_x)]$ and $y_f \in [1, 1 + (\sigma_y - d_y)]$), d_x, d_y are the length and width of the spatial support of the feature with $d_x \in [1, \sigma_x]$ and $d_y \in [1, \sigma_y]$ (note that $\sigma_{\{x,y\}}$ is defined in Eq. 1), and $s \in \{+1, -1\}$ represents the two versions of each feature with its original or inverted signs. Note that the feature has the same orientation α as the image region.

The output value of each feature is the difference between the image pixels lying in the white section (in Fig. 2, the region denoted by +1) and the image pixels in the black section (in Fig. 2, the region denoted by -1). This feature value can be efficiently computed using integral images [30]. The integral image is computed as follows:

$$\mathcal{T}(x, y) = \sum_{i=0}^{x} \sum_{j=0}^{y} I(x, y), \tag{3}$$

where $\mathcal{T} : \Re^{N \times M} \rightarrow \Re$ denotes the integral image. Then the feature value is computed efficiently through a small number of additions and subtractions. For example, the feature value of feature type 1 in Fig. 2 can be computed as

$$f(\theta_f) = \mathcal{T}_f^+ - \mathcal{T}_f^-,$$

where

$$\begin{aligned} \mathcal{T}_f^+ = \; & \mathcal{T}(x_f + \tfrac{d_x}{2}, y_f + d_y) + \mathcal{T}(x_f, y_f) - \\ & \mathcal{T}(x_f + \tfrac{d_x}{2}, y_f) - \mathcal{T}(x_f, y_f + d_y) \\ \mathcal{T}_f^- = \; & \mathcal{T}(x_f + d_x, y_f + d_y) + \mathcal{T}(x_f + \tfrac{d_x}{2}, y_f) - \\ & \mathcal{T}(x_f + d_x, y_f) - \mathcal{T}(x_f + \tfrac{d_x}{2}, y_f + d_y). \end{aligned}$$

This means that the integral image is computed once and each feature value involves the addition and subtraction of six values from the integral image. It is important to mention that the original image is rotated in intervals of δ_α (in this work, $\delta_\alpha = 10^o$) and an integral image is computed for each rotated image. These rotations and integral image computations comprise the pre-processing part of our method. Taking into account all possible feature types, locations, and sizes, there can be in the order of 10^5 possible features within a region.

A classifier then defines the following function: $P(y|S)$, where $y \in \{-1, +1\}$ with $P(y = +1|S)$ representing the probability that the image region S contains the structure of interest (i.e., a positive sample), and $P(y = -1|S)$, the probability that the image region S contains background information (i.e., a negative sample). Notice that the main goal of the system is to determine

$$\theta^* = \arg\max_{\theta} P(y|S), \tag{4}$$

where S is the foreground image region defined by θ in Eq. 1. Therefore, our task is to train a discriminative classifier that minimizes the following probability of mis-classification:

$$P(\text{error}) = \int_{\theta} P(\text{error}|\theta) P(\theta) d\theta,$$

where

$$P(\text{error}|\theta) = \begin{cases} +1 & , \text{if } y \neq \tilde{y} \\ 0 & , \text{otherwise} \end{cases},$$

with $y = \arg\max_{y \in \{-1, +1\}} P(y|S)$ and $\tilde{y}$ being the correct response for the parameter value θ.

IV. Region Classification Process

In this section, we discuss the classifier used in this work and the strategy to improve the efficiency and efficacy of the classification problem. We also show the training and detection algorithms along with the training results.

A. Probabilistic Boosting Tree

The classifier used for the anatomical structure detection is derived from the probabilistic boosting tree classifier (PBT) [36]. The PBT classifier is a boosting classifier [11], [32], where the strong classifiers are represented by the nodes of a binary tree. Tu [36] demonstrates that the PBT is able to cluster the data automatically, allowing for a binary classification of data sets presenting multi-modal distributions, which is typically the case studied in this paper. Another attractive property of the PBT classifier is that after training, the posterior probability can be used as a threshold to balance between precision and recall, which is an important advantage over the cascade method [37] that needs to train different classifiers based on different precision requirements.

Training the PBT involves the recursive construction of a binary tree, where each of its nodes represents a strong classifier. Each node is trained with the AdaBoost algorithm [12],

which automatically learns a strong classifier by combining a set of weak classifiers $H(S) = \sum_{t=1}^{T} \omega_t h_t(S)$, where S is an image region determined by θ in (1), $h_t(S)$ is the response of a weak classifier, and ω_t is the weight associated with each weak classifier. By minimizing the probability of error, the Adaboost classifier automatically selects the weak classifiers and their respective weights. The probabilities computed by each strong classifier is then denoted as follows [36]:

$$q(+1|S) = \frac{e^{2H(S)}}{1+e^{2H(S)}}, \text{ and } q(-1|S) = \frac{e^{-2H(S)}}{1+e^{-2H(S)}}. \quad (5)$$

The posterior probability that a region S is foreground ($y = +1$), or background ($y = -1$) is computed as in [36]:

$$P(y|S) = \sum_{l_1,l_2,...,l_n} P(y|l_n,...,l_1,S)...q(l_2|l_1,S)q(l_1|S), \quad (6)$$

where n is the total number of nodes of the tree (see Fig. 3), and $l \in \{-1,+1\}$. The probability at each tree node is computed as:

$$P(y|l_i,...,l_1,S) = \sum_{l_{i+1}} \delta(y = l_{i+1}) q(l_{i+1}|l_i,...,l_1,S),$$

where $q(.|.)$ is defined in (5)[2], and

$$\delta(x) = \begin{cases} 1, \text{ if } x = true \\ 0, \text{ otherwise} \end{cases}$$

The original PBT classifier presents a problem: if the classification is too hard (i.e., it is difficult to find a function that robustly separates positive from negative samples, which is the case being dealt with in this paper), the tree can become overly complex, which can cause: a) overfit of the training data in the nodes close to the leaves, b) long training procedure, and c) long detection procedure. The overfit of the data in the leaf nodes happens because of the limited number of training samples remaining to train those classifiers. The number of strong classifiers to train grows exponentially with the number of tree levels, which in turn grows with the complexity of the classification problem; hence the training process can take quite a long time for complex classification problems. Finally, note that for each sample θ (Eq. 1) to evaluate during detection, it is necessary to compute the probability over all the nodes of the classification tree. As a result, it is necessary to compute $P(y|S)$ for $N_\theta = N_x \times N_y \times N_\alpha \times N_{\sigma_x} \times N_{\sigma_y}$ times, where N_θ denotes the number of sampling points to evaluate. Usually, N_θ is in the order of 10^8, which can have a severe impact in the running time of the algorithm (in a standard dual-core computer the probability computation of 10^8 samples using a full binary PBT classifier of height five can take around 10 seconds, which is substantially above our target of less than one second).

B. Constrained Probabilistic Boosting Tree

We propose a two-part solution to the problems mentioned in Sec. IV-A. The first part is based on dividing the parameter space into subspaces, simplifying both the training and testing procedures. The second part consists of constraining the growth of the tree by limiting the height and number of nodes. This solution decreases learning and detection times and improves the generalization of the classifier, as shown below.

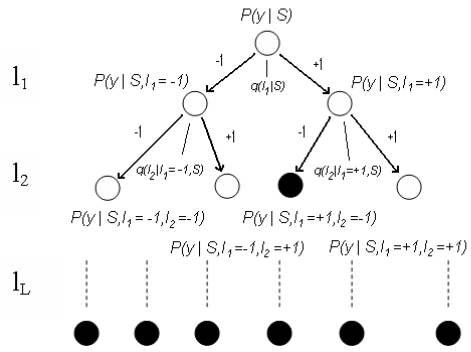


Fig. 3. PBT binary tree structure.

Motivated by the argument that "visual processing in the cortex is classically modeled as a hierarchy of increasingly sophisticated representations" [31], we design a simple-to-complex classification scheme. Assuming that the parameter space is represented by Θ, the idea is subdivide this initial space into subspaces $\Theta_1 \subseteq \Theta_2 \subseteq ... \subseteq \Theta_T \subseteq \Theta$, where the classification problem grows in terms of complexity from Θ_1 to Θ_T. This idea is derived from the works on marginal space learning [39] and sequential sampling [24], where the authors study the trade-off between accuracy and efficiency of such strategy, and the main conclusion is that by implementing such strategy, the training and detection algorithms are several orders of magnitude more efficient without damaging the accuracy of the approach. In Fig. 4, we show a visual example of this idea. Notice that the idea is to train different classifiers, where the first stages tend to be robust and less accurate, and the last stages are more accurate and more complex. The main difference between this approach and the cascade scheme is that the first stages are trained with a *subset* of the initial set of parameters instead of a *subspace* of the full parameter space. We only train classifiers using a subspace of the full parameter space in the last stages.

Each subset and subspace is designed to have in the order of 10^4 to 10^5 parameter space samples to be evaluated, which results in a reduction of three orders of magnitude compared to the initial number of samples mentioned above. Moreover, the initial classifiers are presented with relatively simple classification problems that produces classification trees of low complexity, and consequently the probability computation in these trees are faster than in sub-sequent trees. Finally, given that the classification problem of each classifier is less complex than the original problem, the height and the number of tree nodes can be constrained. These implementations significantly reduce the training and detection times, and improve the generalization ability of the classifier. We call the resulting classifier the Constrained PBT (CPBT).

[2]The value $q(l_{i+1}|l_i,...,l_1,S)$ is obtained by computing the value of $q(l_{i+1}|S)$ at PBT node reached following the path $l_1 - > l_2 - >, ..., l_i$, with l_1 representing the root node and $l \in \{-1,+1\}$ (see Fig. 3).

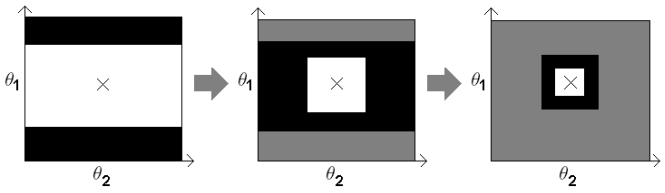
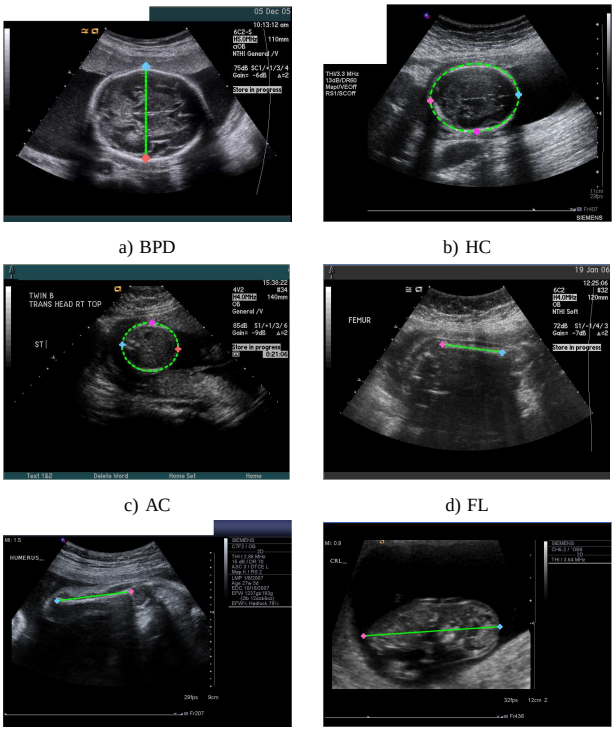


Fig. 4. Simple to complex strategy using a 2-dimensional parameter space, where the target parameter values are represented by the position X. From left to right, the first graph shows two regions in the parameter space: the black area containing the negative samples, and the white area with the positive samples. Notice that in this first graph, the training and detection happen only for the parameter θ_1. The second graph shows a training and detection using both parameters, where the positive samples are acquired from the center of the white circle around position X, and negatives are the samples in the black region. The gray area is a no sampling zone. The last graph shows another classification problem in the parameter space, with positive and negatives samples closer to the position X. In Sec. IV-D those three graphs can be related to the region of interest (ROI) classifier, coarse classifier, and fine classifier, respectively.



a) BPD b) HC

c) AC d) FL

e) HL f) CRL

Fig. 5. Expert annotation of BPD, HC, AC, FL, HL, and CRL.

C. Annotation Protocol

We explore the representation used by sonographers and clinicians for the BPD, HC, AC, FL, HL, and CRL measures. That is, HC and AC are represented with an ellipse, and BPD, FL, HL, and CRL, with a line. Figure 5 shows expert annotations of each measurement. This annotation explicitly defines the parameter θ in (1) for the positive sample of the training image as follows:

- For the ellipsoidal measurements, the user defines three points: $\mathbf{x}_1$ and $\mathbf{x}_2$, defining the major axis, and $\mathbf{x}_3$, defining one point of the minor axis (see Fig. 6-a). With $\mathbf{x}_1$ and $\mathbf{x}_2$, we can compute the center of the ellipse $\mathbf{x}_c = \frac{\mathbf{x}_1+\mathbf{x}_2}{2}$, then the region parameters of (1) are computed as follows:

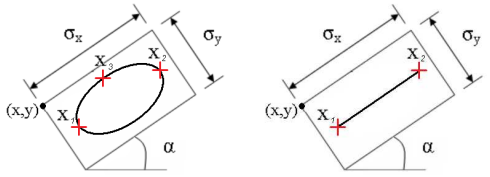
$$\begin{aligned} \sigma_x &= 2\kappa \times \|\mathbf{x}_1 - \mathbf{x}_c\|, \\ \sigma_y &= 2\kappa \times \|\mathbf{x}_3 - \mathbf{x}_c\|, \\ \alpha &= \cos^{-1}\left(\frac{(\mathbf{x}_1-\mathbf{x}_c)\bullet(1,0)}{\|\mathbf{x}_1-\mathbf{x}_c\|}\right), \\ x &= x_c - \frac{\sigma_x}{2}\cos(\alpha), \\ y &= y_c - \frac{\sigma_y}{2}\sin(\alpha), \end{aligned} \tag{7}$$

  where $\mathbf{x}$ represents a two-dimensional vector, $\bullet$ represent vector dot product, $\kappa > 1$ such that a region comprises the anatomy plus some margin, $(1, 0)$ denotes the horizontal unit vector, and $\mathbf{x}_c = (x_c, y_c)$.
- For the line measurements, the user defines two points: $\mathbf{x}_1$ and $\mathbf{x}_2$ (see Fig. 6-b). With $\mathbf{x}_1$ and $\mathbf{x}_2$, we can compute the center $\mathbf{x}_c = \frac{\mathbf{x}_1+\mathbf{x}_2}{2}$, then the region parameters of (1) are computed as follows:

$$\begin{aligned} \sigma_x &= 2\kappa \times \|\mathbf{x}_1 - \mathbf{x}_c\|, \\ \sigma_y &= \eta\sigma_x, \\ \alpha &= \cos^{-1}\left(\frac{(\mathbf{x}_1-\mathbf{x}_c)\bullet(1,0)}{\|\mathbf{x}_1-\mathbf{x}_c\|}\right), \\ x &= x_c - \frac{\sigma_x}{2}\cos(\alpha), \\ y &= y_c - \frac{\sigma_y}{2}\sin(\alpha), \end{aligned} \tag{8}$$

  where $\mathbf{x}$ represents a two-dimensional vector, $\bullet$ represent vector dot product, $\kappa > 1$ such that a region comprises the anatomy plus some margin, $(1, 0)$ denotes the horizontal unit vector, $\mathbf{x}_c = (x_c, y_c)$, and $\eta \in (0, 1]$.

The manual annotation is used to provide aligned images of anatomies normalized in terms of orientation, position, scale, and aspect ratio. These images will be used for training the classifier. There are five classifiers to be trained: 1) head, 2) abdomen, 3) femur, 4) humerus, and 5) fetal body. The head classifier is used to provide the HC and BPD measurements (note that even though the BPD is a line measurement it is derived from the HC measurement through the use of its minor axis), the abdomen classifier allows for the AC, femur classifier is used to produce the FL, humerus classifier produces HL, and fetal body is used to compute the CRL measurement. Figure 5(b) shows the head annotation, where caliper $\mathbf{x}_1$ (red) is located at the back of the head, caliper $\mathbf{x}_2$ (blue) is at the front of the head, and caliper $\mathbf{x}_3$ (pink) defines the minor axis of the ellipse and is located at the side of the head (moving from $\mathbf{x}_1$ to $\mathbf{x}_2$ in counter-clockwise direction). Figure 5(c) shows the abdomen annotation, where caliper $\mathbf{x}_1$ (red) is located at the umbilical vein region, caliper $\mathbf{x}_2$ (blue) is at the spinal chord, and caliper $\mathbf{x}_3$ (pink) defines the minor axis of the ellipse and is located close to the



a) Ellipse b) Line

Fig. 6. Ellipse and line annotations.

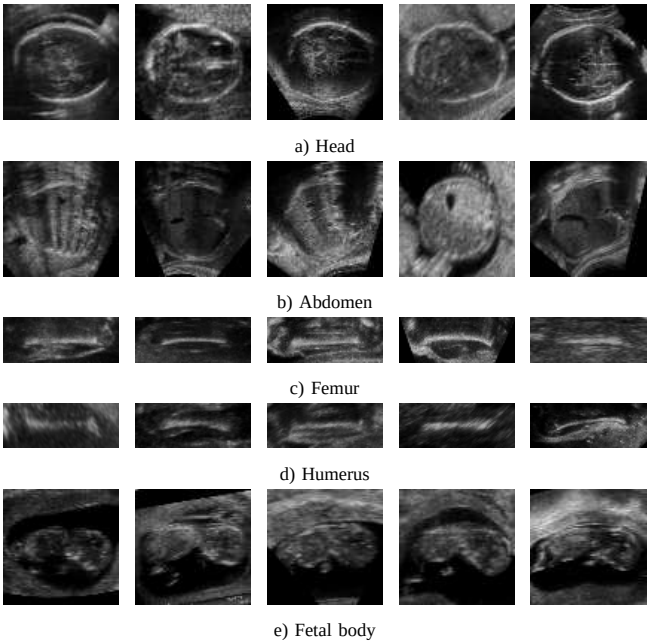


Fig. 7. Examples of the training set for BPD and HC (a), AC (b), FL (c), HL (d), and CRL (e).

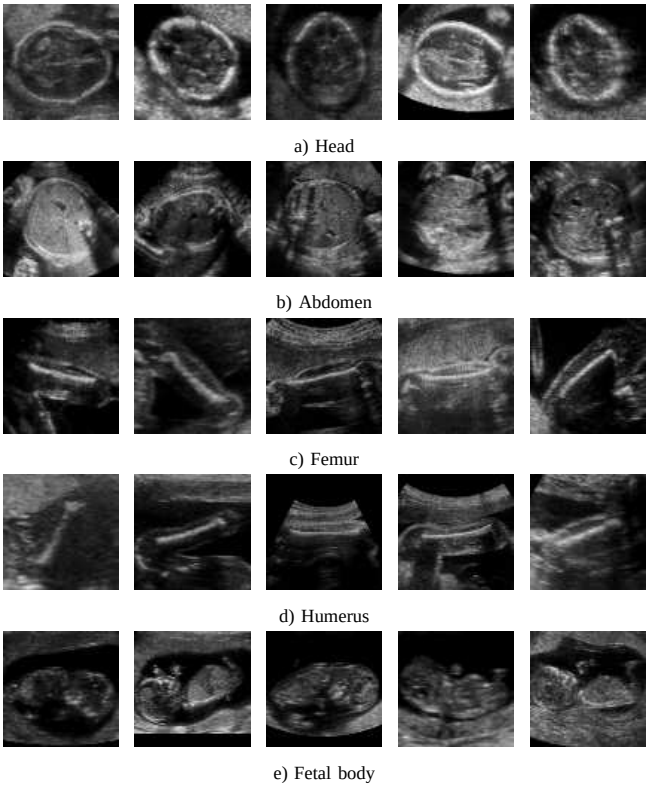


Fig. 8. Examples of the ROI training set for BPD and HC (a), AC (b), FL (c), HL (d), CRL (e).

stomach. Figures 5(d) and (e) display the femur and humerus annotations, respectively, where caliper $\mathbf{x}_1$ (red) and $\mathbf{x}_2$ (blue) are interchangeably located at the end points of the femur bone. Finally, Fig. 5(f) displays the fetal body annotation, respectively, where caliper $\mathbf{x}_1$ (red) is located at the bottom of the fetal body and and $\mathbf{x}_2$ (blue) is located at the head. This annotation protocol allows for building an aligned training set as the ones shown in Figure 7, with $\kappa = 1.5$ and $\eta = 0.38$ for femur and humerus and $\eta = 0.80$ for fetal body in (7) and (8). The values for η are defined based on the aspect ratio of the anatomical structure. Notice that the original image regions are transformed into a square size of 78×78 pixels (used linear interpolation) in the cases of head, abdomen, and fetal body, and into a rectangular size of 78×30 pixels (again, using bi-linear interpolation) for femur and humerus with aspect ratio $\frac{width}{height} = \frac{1}{\eta}$ for $\eta = 0.38$.

D. Training a Constrained Probabilistic Boosting Tree

As mentioned in Sec. IV-B, the training involves a sequence of classification problems of increasing complexity. Here, we rely on a training procedure (see Algorithm 1) involving three stages referred to as the region of interest (ROI) classification stage, the coarse classification stage and the fine classification stage (see Fig. 9).

For the ROI stage, the main goal is to use a subset of the initial parameter set in order to have a fast detection of hypothesis for sub-sequent classification stages. Recall from Section III-A that we rotate the image in intervals of δ_α and compute the integral image for each rotated version of the image. During detection, determining the parameter α in (1) requires loading the respective rotated integral image, which is in general a time consuming task because it is not possible to have all integral images loaded in cache (the usual image size is 600x800, where each pixel is represented by a float number; this means that each image has around 2MB). Therefore, leaving the parameter α out of the ROI classifier means a large gain in terms of detection efficiency. Another important observation for the ROI stage is that the aspect ratio σ_x/σ_y of the anatomy does not vary significantly in the training set. Specifically, for heads, abdomens, and fetal body, $\sigma_x/\sigma_y \in [0.8, 1.2]$ and for femurs and humerus, $\sigma_x/\sigma_y = 1/\eta$. Therefore, the parameter σ_y can also be left out from the ROI stage, and its estimation happens in the sub-sequent stages.

As a result, in the ROI stage, the positive samples are located in a region of the parameter space defined by:

$$\Delta_+^{\text{ROI}} = [\Delta_x^{\text{ROI}}, \Delta_y^{\text{ROI}}, X, \Delta_{\sigma_x}^{\text{ROI}}, X], \tag{9}$$

where $\Delta_x^{\text{ROI}} \in [x - \delta_x^{\text{ROI}}, x + \delta_x^{\text{ROI}}]$, $\Delta_y^{\text{ROI}} \in [y - \delta_y^{\text{ROI}}, y + \delta_y^{\text{ROI}}]$, $\Delta_{\sigma_x}^{\text{ROI}} \in [\sigma_x - \delta_{\sigma_x}^{\text{ROI}}, \sigma_x + \delta_{\sigma_x}^{\text{ROI}}]$, and X denotes a parameter that is not learned in this stage (in this case σ_y and α). In Fig. 4 we display this concept of training for a subset of the initial parameter set. Recall that the positive sample is located at $(x, y, \alpha, \sigma_x, \sigma_y)$ as defined in (1). On the other hand, the negative samples are located in the following region of the parameter space:

$$\Delta_-^{\text{ROI}} = \Theta - \Delta_+^{\text{ROI}}, \tag{10}$$

where Θ represents the whole parameter space. The ROI classifier is able to detect the position and scale of the object (within the limits of Δ_+^{ROI}), but not its rotation nor its aspect ratio (that is, $\alpha = 0$ and $\sigma_y = \sigma_x$ in (7) and (8) for this stage). This means that the training images are kept in

its original orientation and aspect ratio, resulting in training images aligned only in terms of position and scale, and these images are transformed to a square patch of size 78×78 pixels. In Figure 8, we show a few examples of training images for training the ROI classifier.

The coarse classifier is then trained with positive samples from the parameter subset:

$$\Delta_+^{\text{coarse}} = [\Delta_x^{\text{coarse}}, \Delta_y^{\text{coarse}}, \Delta_\alpha^{\text{coarse}}, \Delta_{\sigma_x}^{\text{coarse}}, \Delta_{\sigma_y}^{\text{coarse}}], \quad (11)$$

where $\Delta_x^{\text{coarse}} \in [x-\delta_x^{\text{coarse}}, x+\delta_x^{\text{coarse}}]$, $\Delta_y^{\text{coarse}} \in [y-\delta_y^{\text{coarse}}, y+\delta_y^{\text{coarse}}]$, $\Delta_\alpha^{\text{coarse}} \in [\alpha-\delta_\alpha^{\text{coarse}}, \alpha+\delta_\alpha^{\text{coarse}}]$, $\Delta_{\sigma_x}^{\text{coarse}} \in [\sigma_x-\delta_{\sigma_x}^{\text{coarse}}, \sigma_x+\delta_{\sigma_x}^{\text{coarse}}]$, and $\Delta_{\sigma_y}^{\text{coarse}} \in [\sigma_y-\delta_{\sigma_y}^{\text{coarse}}, \sigma_y+\delta_{\sigma_y}^{\text{coarse}}]$. In order to improve the precision of the detection from the ROI to the coarse classifier, we set $\delta^{\text{coarse}} < \delta^{\text{ROI}}$ in Eq. 9 for all parameters. The negative samples for the coarse classifier are located in the following region of the parameter space:

$$\Delta_-^{\text{coarse}} = \Delta_-^{\text{ROI}} - \Delta_+^{\text{coarse}}, \quad (12)$$

where Δ_-^{ROI} is defined in (10). Finally, the positive samples for the fine classifier are within the subset:

$$\Delta_+^{\text{fine}} = [\Delta_x^{\text{fine}}, \Delta_y^{\text{fine}}, \Delta_\alpha^{\text{fine}}, \Delta_{\sigma_x}^{\text{fine}}, \Delta_{\sigma_y}^{\text{fine}}], \quad (13)$$

where $\Delta_x^{\text{fine}} \in [x-\delta_x^{\text{fine}}, x+\delta_x^{\text{fine}}]$, $\Delta_y^{\text{fine}} \in [y-\delta_y^{\text{fine}}, y+\delta_y^{\text{fine}}]$, $\Delta_\alpha^{\text{fine}} \in [\alpha-\delta_\alpha^{\text{fine}}, \alpha+\delta_\alpha^{\text{fine}}]$, $\Delta_{\sigma_x}^{\text{fine}} \in [\sigma_x-\delta_{\sigma_x}^{\text{fine}}, \sigma_x+\delta_{\sigma_x}^{\text{fine}}]$, and $\Delta_{\sigma_y}^{\text{fine}} \in [\sigma_y-\delta_{\sigma_y}^{\text{fine}}, \sigma_y+\delta_{\sigma_y}^{\text{fine}}]$. The detection precision from the coarse to the fine classifier is improved by setting $\delta^{\text{fine}} < \delta^{\text{coarse}}$ in Eq. 11 for all parameters. The negative samples for the fine classifier are located in the following region of the parameter space:

$$\Delta_-^{\text{fine}} = \Delta_-^{\text{coarse}} - \Delta_+^{\text{fine}}, \quad (14)$$

where Δ_-^{coarse} is defined in (12).

Data : M training images with anatomy region $\{(I,\theta)_i\}_{i=1,..,M}$
Maximum height of each classifier tree: $H_{\text{ROI}}, H_{\text{coarse}}, H_{\text{fine}}$
Total number of nodes for each classifier: $N_{\text{ROI}}, N_{\text{coarse}}, N_{\text{fine}}$
$\mathcal{I}^+ = \emptyset$ and $\mathcal{I}^- = \emptyset$
for $i = 1, ..., M$ **do**
  Add P random samples from sub-space Δ_+^{ROI} (9) to $\mathcal{I}^+$
  Add N random samples from sub-space Δ_-^{ROI} (10) to $\mathcal{I}^-$
end
Train ROI classifier with H_{ROI} and N_{ROI} using $\mathcal{I}^+$ and $\mathcal{I}^-$.
$\mathcal{I}^+ = \emptyset$ and $\mathcal{I}^- = \emptyset$
for $i = 1, ..., M$ **do**
  Add P random samples from sub-space Δ_+^{coarse} (11) to $\mathcal{I}^+$
  Add N random samples from sub-space Δ_-^{coarse} (12) to $\mathcal{I}^-$
end
Train coarse classifier with H_{coarse} and N_{coarse} using $\mathcal{I}^+$ and $\mathcal{I}^-$.
$\mathcal{I}^+ = \emptyset$ and $\mathcal{I}^- = \emptyset$
for $i = 1, ..., M$ **do**
  Add P random samples from sub-space Δ_+^{fine} (13) to $\mathcal{I}^+$
  Add N random samples from sub-space Δ_-^{fine} (14) to $\mathcal{I}^-$
end
Train fine classifier with H_{fine} and N_{fine} using $\mathcal{I}^+$ and $\mathcal{I}^-$.
Result : ROI, coarse, and fine classifiers.

Algorithm 1: Training algorithm.

E. Detection

According to the training algorithm in Sec. IV-D, the detection algorithm must run in three stages, as described in Algorithm 2. The ROI detection samples the search space uniformly using the $\delta_{\{x,y,\sigma_x\}}^{\text{ROI}}$ as the sampling interval for position and scale. The coarse detection only classifies the positive samples for the ROI detector at smaller intervals of $\delta_{\{x,y,\alpha,\sigma_x,\sigma_y\}}^{\text{coarse}}$, while the fine detection searches the hypotheses selected from the coarse search at smaller intervals of $\delta_{\{x,y,\alpha,\sigma_x,\sigma_y\}}^{\text{fine}}$.

Fig. 9. Detection procedure.

Data : Test image and measurement to be performed (BPD, HC, AC, FL, HL, or CRL)
ROI, coarse, and fine classifiers
$\mathcal{H}_{\text{ROI}} = \emptyset$
for $\theta = [0,0,0,0,0] : \delta_{ROI} : [max(x), max(y), 0, max(\sigma_x), 0]$ **do**
  $\sigma_y = \sigma_x$
  Compute $P(y=+1|S)$ (6) using ROI classifier, where S is an image region determined by θ (1)
  $\mathcal{H}_{\text{ROI}} = \mathcal{H}_{\text{ROI}} \bigcup (\theta, P(y=+1|S))$
end
Assigned all hypotheses from $\mathcal{H}_{\text{ROI}}$ in terms of $P(y=+1|S)$ to $\mathcal{H}_{\text{coarse}}$
for $i = 1, ..., |\mathcal{H}_{coarse}|$ **do**
  Assume $(\theta_i, P_i) = i^{th}$element of $\mathcal{H}_{\text{coarse}}$
  for $\theta = [x_i - \delta_x^{ROI}, y_i - \delta_x^{ROI}, 0, \sigma_{x,i} - \delta_{\sigma_x}^{ROI}, 0] : \delta_{coarse} :$
  $[x_i + \delta_x^{ROI}, y_i + \delta_x^{ROI}, \max(\alpha), \sigma_{x,i} + \delta_{\sigma_x}^{ROI}, \max(\sigma_y)]$ **do**
    Compute $P(y=+1|S)$ (6) using coarse classifier, where S is an image region determined by θ (1)
    $\mathcal{H}_{\text{coarse}} = \mathcal{H}_{\text{coarse}} \bigcup (\theta, P(y=+1|S))$
  end
end
Assigned the top H hypotheses from $\mathcal{H}_{\text{coarse}}$ in terms of $P(y=+1|S)$ to $\mathcal{H}_{\text{fine}}$
for $i = 1, ..., |\mathcal{H}_{fine}|$ **do**
  Assume $(\theta_i, P_i) = i^{th}$element of $\mathcal{H}_{\text{fine}}$
  for $\theta = (\theta_i - \delta_{\{x,y,\alpha,\sigma_x,\sigma_y\}}^{coarse}) : \delta_{\{x,y,\alpha,\sigma_x,\sigma_y\}}^{fine} :$
  $(\theta_i + \delta_{\{x,y,\alpha,\sigma_x,\sigma_y\}}^{coarse})$ **do**
    Compute $P(y=+1|S)$ (6) using fine classifier, where S is an image region determined by θ (1)
    $\mathcal{H}_{\text{fine}} = \mathcal{H}_{\text{fine}} \bigcup (\theta, P(y=+1|S))$
  end
end
Select the top hypothesis from $\mathcal{H}_{\text{fine}}$ in terms of $P(y=+1|S)$, and display hypothesis if $P(y=+1|S) > \tau_{DET}$.
Result : Parameter θ of the top hypothesis.

Algorithm 2: Detection algorithm.

The value τ_{DET} was set in order to eliminate the bottom 5% of the cases in the *training set*. We found important to set such threshold in order to avoid large error cases. Therefore, after the detection process if $P(y=+1|S) < \tau_{DET}$, then the system outputs a message, which says "no anatomy detected".

F. Training Results

We have $1,426$ expert annotated training samples for head, $1,293$ for abdomen, $1,168$ for femur, 547 for humerus, 325 for fetal body. An ROI, a coarse, and a fine CPBT classifiers have been trained. We are interested in determining the tree structure of the classifier, where we want to constrain the tree to have the fewest possible number of nodes without affecting the classifier performance. Recall from Sections IV-D and IV-E that a smaller number of nodes produces more efficient

training and detection processes and a more generalizable classifier. Therefore, we compare the performance of the full binary tree against a tree constrained to have only one child per node. The number of weak classifiers is set to be at most 30 for the root node and its children (i.e., nodes at heights 0 and 1), and at most 30×(tree height) for the remaining nodes. Note that the actual number of weak classifiers is automatically determined by the AdaBoost algorithm [12]. The height of each tree is defined as $H_{\text{ROI}} \in [1, 7]$, $H_{\text{coarse}} \in [1, 10]$, and $H_{\text{fine}} \in [1, 15]$, with its specific value determined through the following stop condition: a node cannot be trained with less than 2,000 positives and negative samples (total of 4,000 samples). This stop condition basically avoids over-fitting of the training data. The sampling intervals values for each stage are $\delta_{\text{ROI}} = [15, 15, X, 15, X]$, $\delta_{\text{coarse}} = [8, 8, 20^o, 8, 8]$, and $\delta_{\text{fine}} = [4, 4, 10^o, 4, 4]$. Finally in Algorithm 1, the number of additional positives per image $P = 100$ and the number of negatives per image $N = 1000$.

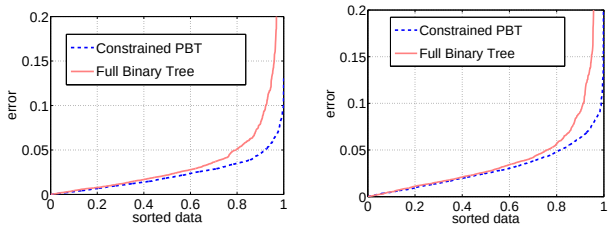
From the parameter $\theta = [x, y, \alpha, \sigma_x, \sigma_y]$ of the top hypothesis, each measurement is computed as follows:

- BPD = $\gamma\sigma_y$ using the response from the head detector, where $\gamma = 0.95$. This value for γ is estimated from the training set by computing $\gamma = \frac{1}{M}\sum_{i=1}^{M}\frac{BPD(i)}{2r_y(i)}$ with M being the number of training images for heads, $BPD(i)$ is the manual BPD measurement for image i, $r_y(i) = \frac{\sigma_y(i)}{2\kappa}$ with $\sigma_y(i)$ denoting the height of the rectangle which contains the head image i (see Eq. 7).
- HC = $\pi\left[3(r_x + r_y) - \sqrt{(3r_x + r_y)(r_x + 3r_y)}\right]$, where this value is the Ramanuja's approximation of the ellipse circumference with $r_x = \frac{\sigma_x}{2\kappa}$ and $r_y = \frac{\sigma_y}{2\kappa}$ (see Eq. 7).
- AC = $\pi\left[3(r_x + r_y) - \sqrt{(3r_x + r_y)(r_x + 3r_y)}\right]$, which is the same computation as for HC.
- FL,HL,CRL = $2r_x$, where $r_x = \frac{\sigma_x}{2\kappa}$(see Eq. 8).

Figure 10 shows the measurement errors for HC and BPD in the training set for the constrained tree and the full binary tree, where the training cases are sorted in terms of the error value. Assuming that the GT contains the expert annotation for BPD, HC, AC, FL, HL, or CRL and DT denotes the respective automatic measurement produced by the system, the error is computed as:

$$\text{error} = |GT - DT|/GT. \quad (15)$$

Notice that the performance of the constrained tree is better than that of the full binary tree. This is explained by the fact that the constrained tree is more regularized and should be able to generalize better than the full binary tree. Another key advantage of the constrained tree is the efficiency in training and testing. For the cases above, the training process for the full binary tree takes between seven to ten days, while for the constrained tree the whole training takes two to four days on a standard PC computer. The detection process for the constrained tree takes, on average, less than one second, while that of the full binary tree takes around three to four seconds. Hence, a constrained tree classifier is used in the experiments.



a) HC b) BPD

Fig. 10. Training comparison between the constrained PBT and full binary tree. The training cases are sorted in terms of the error measurement. The horizontal axes show the training set indices, which varies from 0 to 1, where 0 is the index to the training case with the smallest error, and 1 represents the case with the largest error.

V. Experimental Results

In this section we show qualitative and quantitative results of the database-guided image segmentation based on the CPBT classifier proposed in this paper. First, we describe the methodology to quantitatively assess the performance of our system, then, we describe the experimental protocol. Finally we show the quantitative results along with screen shots of the detection provided by the system.

A. Quantitative Assessment Methodology

For the quantitative assessment of our algorithm, we adopted the methodology proposed by Chalana et al. [7] and revised by Lopez et al. [3], which is briefly explained in this section.

Assume that the segmentation of the anatomy is produced by a curve $A = \{a_1, ..., a_m\}$, where $a_i \in \Re^2$ represent the image positions of the m control points that define this curve. Given another curve $B = \{b_1, ..., b_m\}$, the Hausdorff distance between these two curves is defined by

$$e(A, B) = \max(\max_i\{d(a_i, B)\}, \max_j\{d(b_j, A)\}), \quad (16)$$

where $d(a_i, B) = \min_j \|b_j - a_i\|$, with $\|.\|$ denoting Euclidean distance.

The gold standard measurement is obtained through the average of the user observations. Given that $GT_{(i,j)}$ represents the measurement of user $i \in \{1, ..., n\}$ on image $j \in \{1, ..., N\}$ (i.e., GT represents one of the six measurements considered in this work-BPD,HC,AC,FL,HL,CRL), then the gold standard measurement for image j is obtained as:

$$\bar{GT}_j = \frac{1}{n}\sum_{i=1}^{n} GT_{(i,j)}. \quad (17)$$

The following statistical evaluations compare the computer-generated segmentation to the multiple observers' segmentations. The main goal of these evaluations is to verify whether the computer-generated segmentations differ from the manual segmentations as much as the manual segmentations differ from one another. Assume that we have a database of curves, such as A and B in (16), represented by the variable $x_{i,j}$, with $i \in \{0, ..., n\}$ and $j \in 1, ..., N$, where i is a user index and j is an image index. User $i = 0$ shall always represent the computer-generated curve, while users $i \in \{1, ..., n\}$ are the curves defined from the manual segmentations. We use the

following two kinds of evaluations as proposed by Chalana [7]: 1)*modified Williams index*, and 2) *percentage statistic*. The modified Williams index is defined as:

$$I^{'} = \frac{\frac{1}{n}\sum_{j=1}^{n}\frac{1}{D_{0,j}}}{\frac{2}{n(n-1)}\sum_{j}\sum_{j':j'\neq j}\frac{1}{D_{j,j'}}}, \quad (18)$$

where $D_{j,j'} = \frac{1}{N}\sum_{i=1}^{N} e(x_{i,j}, x_{i,j'})$ with $e(.,.)$ defined in (16). A confidence interval (CI) is estimated using a jackknife non-parametric sampling technique [7], as follows:

$$I^{'}_{(.)} = \pm z_{0.95} se, \quad (19)$$

where $z_{0.95} = 1.96$ (representing the 95^{th} percentile of the standard normal distribution,

$$se = \left\{ \frac{1}{N-1}\sum_{i=1}^{N}[I^{'}_{(i)} - I^{'}_{(.)}]^2 \right\}^{1/2},$$

with $I^{'}_{(.)} = \frac{1}{N}\sum_{i=1}^{N} I^{'}_{(.)}$. Note that $I^{'}_{(i)}$ is the Williams index of (18) calculated by leaving image i out of the computation of $D_{j,j'}$. A successful measurement for the Williams index is to have $I^{'}_{(.)}$ close to 1.

The percentage statistic transform the computer-generated and manual curves into points in a $2m$-dimensional Euclidean space (recall from (16) that m is the number of control points of the segmentation curve), and the goal is to verify the percentage of times that computer-generated curve is within the convex hull formed by the manual curves. An approximation to this measure is computed by [7]

$$\max_{i}\{e(\mathcal{C}, \mathcal{O}_i)\} \leq \max_{i,j}\{e(\mathcal{O}_i, \mathcal{O}_j)\}, \quad (20)$$

where $\mathcal{C}$ is the computer-generated curve, $\mathcal{O}_i$ for $i \in \{1, ..., n\}$ are the observer-generated curves, and $e(.,.)$ defined in (16). The expected value for the percentage statistic depends on the number of observer-generated curves. According to Lopez et al. [3], who revised this value from [7], the successful expected value for the confidence interval of (20) should be greater than or equal to $\frac{n-1}{n+1}$, where n is the number of manual curves. The confidence interval for (20) is computed in the same way as in (19).

B. Experimental Protocol

This system was quantitatively evaluated in a clinical setting using typical ultrasound examination images. It is important to mention that all ultrasound images used in this evaluation were not included in the training set. The evaluation protocol was set up as follows:

1) User selects an ultrasound image of a fetal head, abdomen, femur, humerus, or fetal body.
2) User presses the relevant detection button (i.e., BPD or HC for head, AC for abdomen, FL for femur, HL for humerus, CRL for fetal body).
3) System displays automatic detection and measurement and saves the computer-generated curve.
4) User makes corrections to the automatic detection and saves the manual curve.

Three sets of data are available, as follows:

- *Set 1*: 10 distinct images of fetal heads, 10 distinct images of fetal abdomen, and 10 distinct images of fetal femur were evaluated by *five* expert users. Therefore, we have five different manual measurements per image (i.e., a total of $40 * 5 = 200$ measurements).
- *Set 2*: Five expert users annotated 59 head images, 53 abdomen images, and 50 femur images. In total, we have 295 head images, 265 abdomen images, and 250 femur images, which means that there is *no overlap* between images annotated by different users in this second set.
- *Set 3*: Three expert users annotated 30 humerus and 35 fetal body images. In total, we have 90 humerus images, and 105 fetal body images, which means that there is *no overlap* between images annotated by different users in this third set.

C. Results

In this section we show qualitative results in Fig. 11 and the quantitative assessment of our system using the Williams index and the percentage statistic described in Sec. V-A on the sets of data described in Sec. V-B.

Table I shows the error between control points of the curves generated by our system and by the manual measurements. The curves generated for the HC and AC measurements contain 16 control points, while the curve for BPD, FL, HL, and CRL have two control points (just the end points of the line). In addition to the Hausdorff distance, we also show results using the average distance, where $e(.,.)$ in (16) is substituted for

$$e(A, B) = \frac{1}{2}\left(\frac{1}{m}\sum_{i=1}^{M} d(a_i, B) + \frac{1}{m}\sum_{j=1}^{M} d(b_j, A)\right),$$

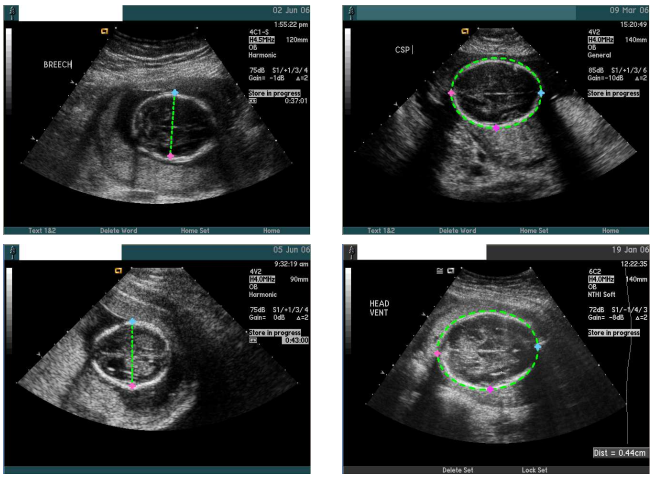
for curves A and B. The Williams index and its confidence interval are shown in Table I for Set 1. The computer-to-observer errors measured on Sets 2 and 3 are displayed in Table I (last two columns) [3]. Recall that the confidence interval for the Williams index has to be close to 1, so that it can be concluded that there is negligible statistical difference between the the computer-generated and user measurements.

The measurement errors computed from Set 1 are shown in Table II. Note that in this table we only consider the errors (15) computed from the measurements of BPD, HC, AC, and FL, and the gold-standard is obtained from the average of the five observers' measurements. We also present the correlation coefficient r, which denotes the Pearson correlation, defined as follows:

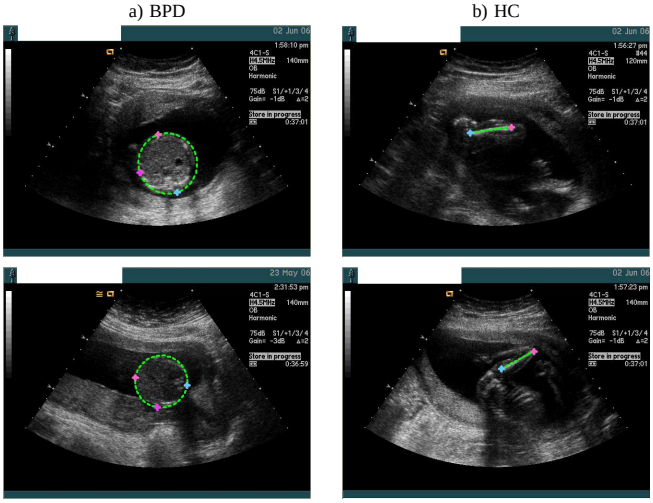
$$r = \frac{\sum_i \sum_j GT_i DT_j - \frac{\sum_i GT_i \sum_j DT_j}{\#images}}{\sqrt{\left(\sum_i GT_i^2 - \frac{(\sum_i GT_i)^2}{\#images}\right)\left(\sum_i DT_i^2 - \frac{(\sum_i DT_i)^2}{\#images}\right)}}, \quad (21)$$

where GT_i is the user measurement and DT_i is the system measurement for the i^{th} image (see Sec. IV-F). The measurement errors computed from Sets 2 and 3 are shown in Table III, where the gold-standard is simply the user measurement.

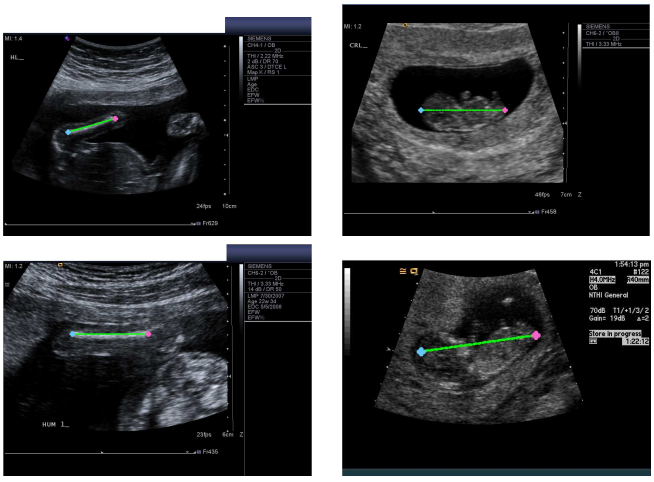
[3]We could not compute the Williams index for Sets 2 and 3 because we have only one user measurement per image



a) BPD b) HC



c) AC d) FL



e) HL f) CRL

Fig. 11. Detection and segmentation results.

Table IV shows the Williams index and percentage statistic with respect to the user measurements (as shown in [7]). Note that the confidence interval for the percentage statistic should be around $\frac{n-1}{n+1} = \frac{4}{6} = 0.66$, where $n = 5$ =number of manual measurements. Finally, Fig. 12 shows the average error in terms of days as a function of the gestational age (GA) of the fetus for Sets 1, 2, and 3. In this case the gestational age is computed as a function of each measurement using the Hadlock regression function [8]. The error is computed by taking the average error of the measurement (Tables II for Set 1, and III for Sets 2 and 3) and computing what that error represents in terms of number of days, but notice that this error varies as a function of the GA of the fetus.

For all cases above, notice that the confidence interval (CI) for the Williams index is around 1 for all measurements, and

TABLE I

COMPARISON OF THE COMPUTER GENERATED CURVES TO THE OBSERVERS' CURVES FOR FETAL HEAD, ABDOMEN, FEMUR, HUMERUS, AND BODY DETECTIONS ON SETS 1, 2, AND 3 (SEE SEC. V-B). CO = MEAN COMPUTER-TO-OBSERVER DISTANCE, IO = MEAN INTER-OBSERVER DISTANCE, WI = WILLIAMS INDEX, CI = CONFIDENCE INTERVAL.

	Set 1				Set 2	Set 3
Measure	CO (mm)	IO (mm)	WI	95% CI	CO (mm)	CO (mm)
Head					Head	Humerus
Hausdorff distance	2.13 (σ : 1.15)	2.25 (σ : 0.43)	0.88	(0.77, 0.98)	2.35 (σ : 2.26)	2.39 (σ : 1.62)
Average distance	1.44 (σ : 0.77)	1.49 (σ : 0.28)	0.86	(0.75, 0.97)	1.50 (σ : 1.46)	1.69 (σ : 1.65)
Abdomen					Abdomen	Body
Hausdorff distance	2.77 (σ : 1.64)	3.16 (σ : 1.15)	0.89	(0.77, 1.01)	3.49 (σ : 4.38)	2.86 (σ : 3.13)
Average distance	1.57 (σ : 0.89)	1.96 (σ : 0.48)	1.02	(0.92, 1.12)	2.03 (σ : 2.35)	2.11 (σ : 1.79)
Femur					Femur	
Hausdorff distance	0.76 (σ : 0.39)	0.52 (σ : 0.36)	1.15	(0.93, 1.37)	1.27 (σ : 2.94)	
Average distance	0.51 (σ : 0.26)	0.37 (σ : 0.25)	1.23	(1.04, 1.41)	0.79 (σ : 1.58)	

TABLE II

COMPARISON OF COMPUTER-GENERATED MEASUREMENTS TO THE GOLD-STANDARD (AVERAGE OF THE FIVE OBSERVERS' MEASUREMENTS) USING ABSOLUTE DIFFERENCES ON SET 1. r = CORRELATION COEFFICIENT.

	CO (mm)	CO (%)	IO (mm)	IO (%)	r
BPD	1.46 (σ : 1.48)	1.71 (σ : 1.76)	0.82 (σ : 0.61)	0.97 (σ : 0.59)	0.998
HC	4.80 (σ : 4.73)	1.02 (σ : 0.81)	4.11 (σ : 2.57)	0.89 (σ : 0.44)	0.999
AC	6.96 (σ : 9.14)	2.43 (σ : 3.51)	4.72 (σ : 6.49)	1.67 (σ : 2.45)	0.994
FL	0.45 (σ : 0.71)	1.36 (σ : 2.11)	0.16 (σ : 0.20)	0.53 (σ : 0.65)	0.996

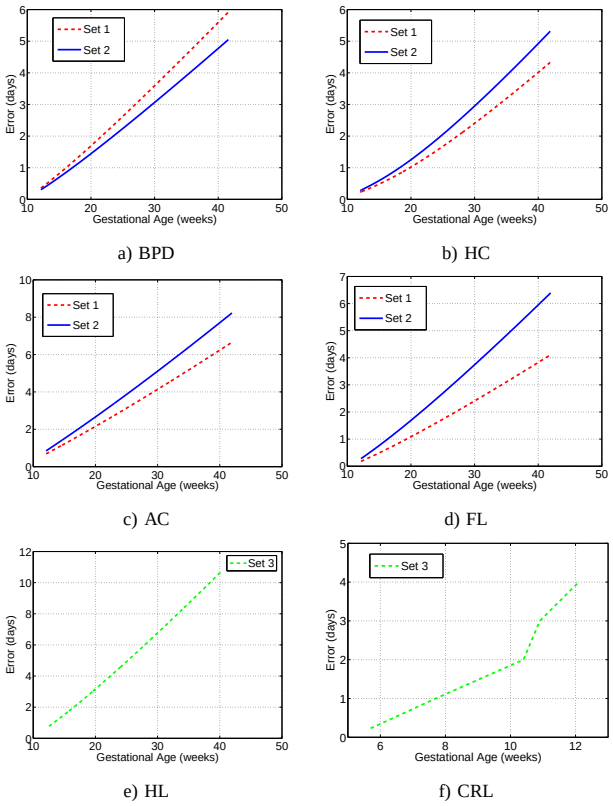
TABLE III

COMPARISON OF COMPUTER-GENERATED MEASUREMENTS TO THE GOLD-STANDARD (OBSERVERS' MEASUREMENTS) USING ABSOLUTE DIFFERENCES FOR SETS 2 AND 3. r = CORRELATION COEFFICIENT.

	CO (mm)	CO (%)	r
BPD	1.11 (σ : 1.44)	1.46 (σ : 1.74)	0.998
HC	5.07 (σ : 5.42)	1.25 (σ : 1.34)	0.999
AC	10.67 (σ : 18.80)	3.00 (σ : 6.16)	0.991
FL	0.89 (σ : 2.78)	2.11 (σ : 5.68)	0.986
HL	1.59 (σ : 1.53)	3.52 (σ : 3.72)	0.982
CRL	1.43 (σ : 1.49)	2.40 (σ : 2.30)	0.983

TABLE IV
WILLIAMS INDEX AND PERCENT STATISTIC FOR BPD, HC, AC, AND FL MEASUREMENTS ON SET 1. WI = WILLIAMS INDEX, P = PERCENT STATISTIC, CI = CONFIDENCE INTERVAL.

	WI	95% CI	P	95% CI
BPD	0.8246	(0.5791, 1.0702)	80.0	(75.38, 84.68)
HC	1.0567	(0.8924, 1.2211)	80.0	(75.38, 84.68)
AC	0.7086	(0.4520, 0.9652)	50.0	(44.14, 55.86)
FL	0.9201	(0.5774, 1.2628)	60.0	(54.26, 65.74)



a) BPD b) HC c) AC d) FL e) HL f) CRL

Fig. 12. Average error in days in terms of gestational age for Sets 1, 2, and 3.

the percentage statistic CI is close to the expected value of 0.66 for all measurements. The AC measurement shows a result slightly below this mark, but given that the Williams index result for AC and for the abdomen curve is always close to one, it is fair to say that AC is producing acceptable results. In general, the HL and CRL measurements present similar results compared to the other anatomies, even though their classifier models were built with much smaller training sets. Finally, it is interesting to see in Fig. 12 that the errors reported for each anatomy represent a deviation of only a couple of days when GA < 30 weeks and a few days (usually less than seven days) for GA > 30 weeks.

Chalana et al. [7] show the same experimental results for fetal heads and abdomens (see Tables V, VI, and VII), and in general, the results for head detection and measurements are comparable, but our results for abdomen detection and measurements are more accurate. Another interesting fact is that the inter-user variability is generally larger in Chalana's evaluation [7]. This fact increases the likelihood of more positive statistical evaluations (i.e., Williams index close to one, and higher percentage statistic). Finally, in Chalana's evaluation [7], there is no statistic assessment of the fetal femur, humerus, and fetal body measurements.

TABLE V
COMPARISON OF THE COMPUTER GENERATED CURVES TO THE FIVE OBSERVERS' CURVES FOR FETAL SKULL AND ABDOMEN DETECTIONS ON A SET OF 30 TEST IMAGES - TABLE FROM [7]. SEE TABLE I FOR DETAILS.

Measure	CO (mm)	IO (mm)	WI	95% CI
Head				
Hausdorff distance	4.64 (σ : 2.61)	3.83 (σ : 1.90)	0.83	(0.70, 0.96)
Average distance	2.09 (σ : 0.95)	1.92 (σ : 0.82)	0.92	(0.81, 1.03)
Abdomen				
Hausdorff distance	8.88 (σ : 6.25)	5.48 (σ : 5.22)	0.61	(0.49, 0.73)
Average distance	4.05 (σ : 3.13)	2.91 (σ : 3.49)	0.69	(0.57, 0.83)

TABLE VI
COMPARISON OF COMPUTER-GENERATED MEASUREMENTS TO THE GOLD-STANDARD (AVERAGE OF THE FIVE OBSERVERS' MEASUREMENTS) USING ABSOLUTE DIFFERENCES ON A SET OF 30 TEST IMAGES - TABLE FROM [7]. SEE TABLE II FOR DETAILS.

	CO (mm)	CO (%)	IO (mm)	IO (%)	r
BPD	0.71 (σ : 0.61)	1.19 (σ : 0.85)	0.83 (σ : 0.66)	1.33 (σ : 0.82)	0.999
HC	5.22 (σ : 5.27)	2.07 (σ : 1.67)	8.46 (σ : 3.28)	3.54 (σ : 0.99)	0.996
AC	12.6 (σ : 9.48)	6.35 (σ : 5.26)	11.62 (σ : 10.6)	5.65 (σ : 6.53)	0.974

The running time for our algorithm is on average 0.5 seconds for all measurements on a PC computer with the following configuration: Intel Core 2 CPU 6600 at 2.4 GHz, 2GB of RAM.

VI. CONCLUSIONS

We presented a system that automatically measures the BPD and HC from ultrasound images of fetal head, AC from images of fetal abdomen, FL in images of fetal femur, HL in images of fetal humerus, and CRL from images of fetal body. Our system exploits a large database of expert annotated images in order to model statistically the appearance of such anatomies. This is achieved through the training of a Constrained Probabilistic Boosting Tree classifier. The results show that our system produces accurate results, and the clinical evaluation shows results that are, on average, close to the accuracy of sonographers. A comparison with the method by Chalana [7] shows that our

TABLE VII
WILLIAMS INDEX AND PERCENT STATISTIC FOR BPD, HC, AC, AND FL MEASUREMENTS ON A SET OF 30 TEST IMAGES - TABLE FROM [7]. SEE TABLE IV FOR DETAILS.

	WI	95% CI	P	95% CI
BPD	1.07	(1.02, 1.11)	48.5	(33.9, 63.1)
HC	1.12	(1.09, 1.41)	66.7	(56.3, 83.1)
AC	0.82	(0.61, 1.03)	51.4	(37.3, 65.5)

method produces, in general, superior results. Moreover, the algorithm is extremely efficient and runs in under half second on a standard dual-core PC computer. Finally, the clinical evaluations showed a seamless integration of our system into the clinical workflow. We observed a reduction of up to 75% in the number of keystrokes when performing the automatic measurements (compared to the manual measurements).

ACKNOWLEDGEMENT

The authors would like to thank the reviewers and the area editor for providing comments and suggestions that substantially improved the paper. The authors would also like to thank Doctor Ivica Zalud and Kathleen Patchrapong, from the University of Hawaii at Manoa, for helping with the clinical evaluations.

REFERENCES

[1] The American Institute of Ultrasound in Medicine. AIUM Practice Guideline for the Performance of Obstetric Ultrasound Examinations. 2007.

[2] Y. Akgul and C. Kambhamettu. A coarse-to-fine deformable contour optimization framework. In *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 25 (2), pp. 174-186, 2003.

[3] C. Lopez, M. Fernandez, and J. Alzola. Comments on: A methodology for evaluation of boundary detection algorithms on medical images. In *IEEE Transaction on Medical Imaging*, 23 (5), pp. 658-660, 2004.

[4] E. Borenstein and S. Ullman. Class-specific, top-down segmentation. In *European Conference on Computer Vision*, Vol. 2, pp. 109-122, 2002.

[5] A. Chakraborty, H. Staib, and J. Duncan. Deformable Boundary Finding in Medical Images by Integrating Gradient and Region Information. In *IEEE Transactions Medical Imaging*, pp. 859-870, 1996.

[6] V. Chalana, T. Winter II, D. Cyr, D. Haynor, and Y. Kim. Automatic fetal head measurements from sonographic images. In *Acad. Radiology*, 3 (8), pp. 628-635, 1996.

[7] V. Chalana and Y. Kim. A methodology for evaluation of boundary detection algorithms on medical images. In *IEEE Transactions on Medical Imaging*, 16 (5), pp. 642-652, 1997.

[8] F. Chervenak and A. Kurjak. Current Perspectives on the Fetus as a Patient. ISBN-10: 1850707421, First Edition, 1996

[9] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23 (6), pp. 681-685, 2001.

[10] D. Cristinacce and T. Cootes. Facial feature detection using adaboost with shape constraints. In *British Machine Vision Conference*, Vol.1, pp. 231-240, 2003.

[11] Y. Freund. Boosting a weak learning algorithm by majority. In *Information and Computation*, 121(2), pp. 256-285, 1995.

[12] Y.'Freund and R. Schapire. A Decision-theoretic generalization of on-line learning and an application to boosting. In *Proceedings of the International Conference on Machine Learning*, 1996.

[13] B. Georgescu, X. Zhou, D. Comaniciu, and A. Gupta. Database-guided segmentation of anatomical structures with complex appearance. In *IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 2, pp. 429-436, 2005.

[14] C. Hanna and A. Youssef. Automated measurements in obstetric ultrasound images. In *International Conference on Image Processing*, Vol. 3, pp. 504-507, 1997.

[15] X. He, R. Zemel, V. Mnih. Topological map learning from outdoor image sequences. In *Journal of Field Robotics*. 23 (11-12), pp. 1091-1104, 2006.

[16] G. Jacob, J. Noble, C. Behrenbruch, A.Kelion and A. Banning. A shape-space-based approach to tracking myocardial borders and quantifying regional left-ventricular function applied in echocardiography. In *IEEE Transactions on Medical Imaging*, 21 (3), 2002.

[17] S. Jardim and M. Figueiredo. Segmentation of fetal ultrasound images *Ultrasound in Medicine and Biology*, 31 (2), pp. 243-250, 2005.

[18] M. Kumar, P. Torr, and A. Zisserman. Obj cut. In *IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 1, pp. 18-25, 2005.

[19] M. Leventon, W. Grimson, O. Faugeras. Statistical shape influence in geodesic active contours. In *IEEE Conference on Computer Vision and Pattern Recognition*, Vol. I, pp. 316-323, 2000.

[20] B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. In European Conference on Computer Vision - Workshop on Statistical Learning in Computer Vision, 2004.

[21] A. Levin, Y. Weiss, Learning to Combine Bottom-Up and Top-Down Segmentation. In *European Conference in Computer Vision*. Vol. 4, pp. 581-594, 2006.

[22] A. Madabhushi and D. Metaxas, Combining low-, high-level and empirical domain knowledge for automated segmentation of ultrasonic breast lesions In *IEEE Transactions on Medical Imaging*. 22 (2), pp. 155-169, 2003.

[23] G. Matsopoulos and S. Marshall. Use of morphological image processing techniques for the measurement of fetal head from ultrasound images. In *Pattern Recognition*, 27, pp. 1317-1324, 1994.

[24] P. Moral, A. Doucet, and G. Peters. Sequential monte carlo samplers. *J. R. Statist. Soc. B*, 68:411436, 2006.

[25] N. Paragios and R. Deriche. Geodesic active regions and level set methods for supervised texture segmentation. In *International Journal of Computer Vision*. 46 (3), pp. 223-247. 2002.

[26] N. Paragios. A level set approach for shape-driven segmentation and tracking of the left ventricle. In *IEEE Transactions on Patter Analysis and Machine Intelligence*, 22 (6), pp. 773-776, 2003.

[27] S. Pathak, V. Chalana, D. Haynor and Y. Kim. Edge-guided boundary delineation in prostate ultrasound images. *IEEE Transactions on Medical Imaging*, 19 (12), pp. 1211-1219, 2000.

[28] S. D. Pathak, V. Chalana and Y. Kim. Interactive automatic fetal head measurements from ultrasound images using multimedia computer technology. *Ultrasound in Medicine and Biology*, 23 (5), pp. 665-673, 1997.

[29] D. Pham,C. Xu, and J. Prince. Current methods in medical image segmentation. *Annual Review of Biomedical Engineering*, Vol. 2, pp. 315-337, 2000.

[30] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, and T. Poggio. Pedestrian detection using wavelet templates. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 193-199, 1997.

[31] M. Riesenhuber, and T. Poggio. Hierarchical models of object recognition in cortex. In *Nature Neuroscience*, 2, pp. 1019-1025, 1999.

[32] R. Schapire. The strength of weak learnability. In *Machine Learning*, 5(2), pp. 197227, 1990.

[33] P.J. Schluter, G. Pritchard, and M.A. Gill. Ultrasonic fetal size measurements in Brisbane, Australia. *Australasian Radiology*, 48 (4), pp. 480-486, 2004.

[34] K. Sidiqi, Y.-B. Lauziere, A. Tannenbaum, and S. Zucker. Area and length minimizing flows for shape segmentation. In *IEEE Transactions on Image Processing*, 7 (3), pp.433-443. 1998.

[35] J. Thomas, P. Jeanty, R. Peters II, E. Parrish Jr. Automatic measurements of fetal long bones. A feasibility study. *Journal of Ultrasound in Medicine*, 10 (7), pp. 381-5, 1991.

[36] Z. Tu. Probabilistic boosting-tree: learning discriminative models for classification, recognition, and clustering. *International Conference on Computer Vision*, Vol. 2, pp. 1589-1596, 2005.

[37] P. Viola and M. Jones. Robust real-time object detection. In *International Journal of Computer Vision*, 57 (2), pp. 137-154, 2004.

[38] G. Xiao, M. Brady,J. Noble and Y. Zhang. Segmentation of ultrasound B-mode images with intensityinhomogeneity correction In *IEEE Transactions on Medical Imaging*, 21 (1), pp. 48-57, 2002.

[39] Y. Zheng, A. Barbu, B. Georgescu, M. Scheuering, and D. Comaniciu. Fast automatic heart chamber segmentation from 3d ct data using marginal space learning and steerable features. *ICCV*, 2007.

[40] S. Zhu, and A. Yuille. Region competition: unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. In *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 18, pp.884-900. 1996.