INTEGRATED DETECTION NETWORK (IDN) FOR POSE AND BOUNDARY ESTIMATION IN MEDICAL IMAGES

Michal Sofka[⋆] *Kristóf Ralovich*[†] *Neil Birkbeck*[‡] *Jingdan Zhang*[⋆] *S.Kevin Zhou*[⋆]

[⋆] Siemens Corporate Research, 775 College Road East, Princeton, NJ USA
[†] Technical University of Munich, Boltzmannstr. 3, 85748 Garching bei München, Germany
[‡] Dept. of Computing Science, 2-32 Athabasca Hall, University of Alberta, Edmonton, Alberta, Canada

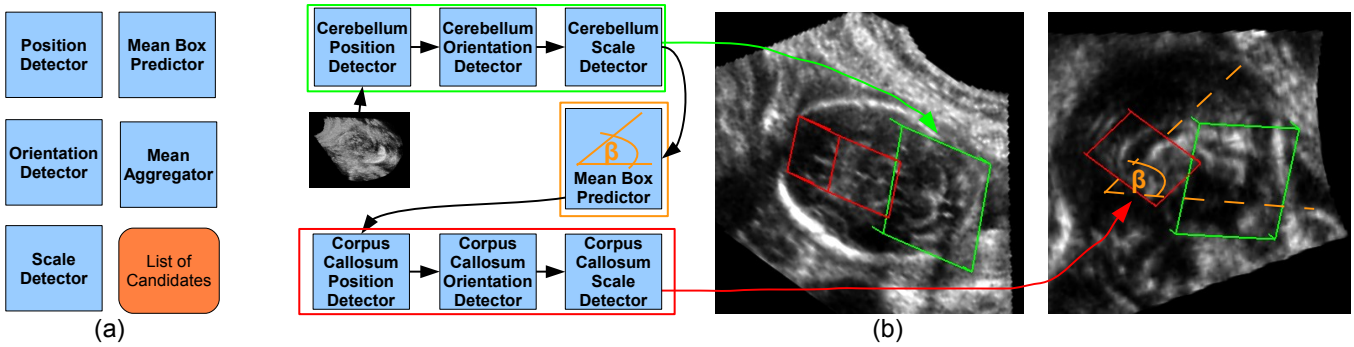

Fig. 1: (a) Most important parts of the Integrated Detection Network. (b) With the Integrated Detection Network, expressing existing morphological relationship among anatomies is straightforward. In case of the fetal head, the pose of the Cerebellum constrains the plane the Corpus Callosum is situated along.

ABSTRACT

The expanding role of complex object detection algorithms introduces a need for flexible architectures that simplify interfacing with machine learning techniques and offer easy-to-use training and detection procedures. To address this need, the Integrated Detection Network (IDN) proposes a conceptual design for rapid prototyping of object and boundary detection systems. The IDN uses a strong spatial prior present in the medical imaging domain and a large annotated database of images to train robust detectors. The best detection hypotheses are propagated throughout the detection network using sequential sampling techniques. The effectiveness of the IDN is demonstrated on two learning-based algorithms: (1) automatic detection of fetal brain structures in ultrasound volumes, and (2) liver boundary detection in MRI volumes. Modifying the detection pipeline is simple and allows for immediate adaptation to the variations of the desired algorithms. Both systems achieved low detection error (3.09 and 4.20 mm for two brain structures and 2.53 mm for boundary).

Index Terms— detection systems, discriminative learning, corpus callosum detection, cerebellum detection, liver segmentation

Second and third author performed the work while at Siemens Corporate Research.

1. INTRODUCTION

In the recent years many accurate and domain-specific object detection algorithms have been built around the well established machine learning and pattern recognition techniques [1, 2]. Often, these algorithms result in highly sophisticated description of the detection systems that go well beyond detecting single objects [3, 4, 5, 6]. In these cases, it becomes challenging to manage large numbers of detectors, maintain their training and detection pipelines, and navigate through the parameter settings. In this paper, we propose a conceptual framework, called Integrated Detection Network (IDN), that enables efficient prototyping of large scale and robust systems for object pose and boundary detection. In the IDN framework the detection systems are decomposed into a network of *modules* and the *data* associations between modules (Figure 2). This decomposition simplifies design, modification, tuning, implementation, and encourages experimentation.

In large scale systems, it is often difficult to modify the detection pipeline when a new theory is developed, additional modules are included, or existing modules need to be rearranged (Figure 4). In the medical imaging domain, these changes are necessary to handle additional anatomical structures, different acquisition protocols, various types of pathological cases, and imaging artifacts. Unless there is a clear design concept, such modifications become cumbersome and

time consuming.

In this paper, we focus on two representative systems of algorithms that account for complex spatial interdependencies between objects and span applications of detection and segmentation. In the first system, anatomical structure detection in 3D fetal ultrasound volumes, we propose modules for detecting position, orientation, and scale at different resolutions (Figure 4 and 5). The relationships are realized in terms of hypothesized candidates for each detector. In the second application, liver boundary detection in 3D MRI scans, we propose modules for estimating an organ shape model. The relationships are represented by PCA coefficients and a free form organ boundary.

Both systems are built using a hierarchical learning-based algorithm with one detector trained for each structure and a resolution level. At the coarsest level, the search region is the entire image. At each subsequent resolution level, the detector search region is defined by the image neighborhoods surrounding the highest probability candidates from the previous level. This way, the candidates are propagated and refined throughout the detection network. IDN proposes a flexible interface for re-arranging modules such that this refinement is correctly handled for both training and detection.

In summary, the paper makes the following contributions: (1) Conceptual framework for designing large scale detection systems, (2) Formalism for propagating detection hypotheses through the detection pipeline, (3) Algorithm and two different pipelines for detecting cerebellum and corpus callosum in fetal ultrasound volumes, and (4) Technique for detecting liver boundary in 3D MRI scans.

2. BACKGROUND

The Integrated Detection Network (IDN) uses discriminative learning techniques that rely on large database of annotated images. It has been previously shown, that the localized detectors can be improved by modeling interdependence of objects using contextual [3, 5] and semantic information [4]. The detectors are improved even further by exploiting the strong prior information embedded in our domain of medical images. In our approach, we detect multiple objects one-by-one using sequential sampling techniques [6]. In the IDN design, these techniques are encapsulated into a common framework for object and boundary detection.

The detection of fetal anatomical structures in ultrasound images is complicated by the low quality of images that contain speckle noise, shadows, blurry edges, and appearance differences due to varying gestational age. These challenges have been previously addressed by learning-based approaches [7, 6] and by a multi-resolution hierarchy of detectors [6]. In this paper, we will show how to refine the selection of the pose estimation hierarchy by removing orientation and scale detectors, when their models might be noisy (e.g. at coarser levels). In addition, we apply the IDN to detect corpus callosum by refining its predicted pose parameters from cerebellum. To the best of our knowledge, this is the first time an automatic method for detection and visualization of corpus callosum in fetal brain ultrasound images has been proposed in literature.

Previously, there have been several techniques proposed for the boundary detection of the liver (or liver segmentation) in CT images [8]. The algorithms for MRI boundary detection have been based on graph cuts [9] and level sets [10]. The design of these algorithms is complicated by high variation of image intensities inside the liver parenchyma and neighboring structures [11]. In our approach, we use a learning-based boundary detector that adapts to the differences of the images in the training set and focuses on what is consistent.

3. MULTI-OBJECT DETECTION

Our detection algorithms are built using discriminative models trained from a large annotated database of medical images (Section 3.1). In Section 3.2, we will describe how to use the basic IDN blocks (modules and data) to build a detection network. We will then focus on two specific networks: (1) IDN for detecting anatomical structures in 3D ultrasound images of fetal brain (Section 3.4) and (2) IDN for detecting boundary of liver in 3D MRI scans (Section 3.5).

3.1. Hierarchical Detection Network (HDN)

In our multi-object detection systems, we adopt Hierarchical Detection Network (HDN) [6] that samples a sequential order of probability distributions to obtain the best pose estimate of each object one by one. Let's denote the pose parameters (position, orientation, and size) of the object t as $\boldsymbol{\theta}_t$ and the sequence of multiple object detections as $\boldsymbol{\theta}_{0:t} = \{\boldsymbol{\theta}_0, \boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_t\}$. The set of observations (features) for object t is denoted V_t, and the sequence as $V_{0:t} = \{V_0, V_1, \ldots, V_t\}$. The multi-object detection problem is solved by recursively applying *prediction* and *update* steps to obtain the posterior distribution $f(\boldsymbol{\theta}_{0:t}|V_{0:t})$. The prediction step computes the probability density of the state of the object t using the state of the previous object, $t-1$, and previous observations of all objects up to $t-1$:

$$f(\boldsymbol{\theta}_{0:t}|V_{0:t-1}) = f(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{0:t-1}) f(\boldsymbol{\theta}_{0:t-1}|V_{0:t-1}). \quad (1)$$

When detecting object t, the observation V_t is used to compute the estimate during the update step as:

$$f(\boldsymbol{\theta}_{0:t}|V_{0:t}) = \frac{f(V_t|\boldsymbol{\theta}_t) f(\boldsymbol{\theta}_{0:t}|V_{0:t-1})}{f(V_t|V_{0:t-1})}, \quad (2)$$

where $f(V_t|V_{0:t-1})$ is the normalizing constant. The observation model is defined by a discriminative classifier (e.g. PBT [2]):

$$f(V_t|\boldsymbol{\theta}_t) = f(y_t = +1|\boldsymbol{\theta}_t, V_t), \quad (3)$$

where the random variable $y \in \{-1, +1\}$, indicates the presence/absence of the object and $f(y_t = +1|\boldsymbol{\theta}_t, V_t)$ is posterior probability of object presence at $\boldsymbol{\theta}_t$ in V_t. The transition kernel defines a pairwise dependency

$$f(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{0:t-1}) = f(\boldsymbol{\theta}_t|\boldsymbol{\theta}_j), \quad j \in \{0, 1, \ldots, t-1\}. \tag{4}$$

where $f(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{0:t-1})$ is a Gaussian distribution estimated from the training data, and j indicates object precursor specified using a prior knowledge or determined automatically [6].

3.2. IDN Abstraction Model

IDN is defined around a conceptually simple and minimal abstraction model: *Modules* perform an operation on the input *Data* and produce zero or more output *Data*. The input (consumption) and output (production) is realized through connecting *Data* objects to the input/output slots of a *Module* object interface.

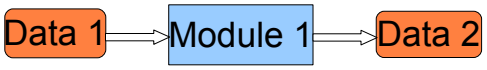

Fig. 2: The two fundamental building blocks of IDN are *Data* (orange in all figures) and *Modules (blue)*.

The network is a heterogenous combination of *Modules* and *Data* objects organized into directed acyclic graph (DAG). The acyclic property is important to ensure that the network is able to do self discovery of the connections and to propagate certain signals (e.g. detection, training, traversal of graph structure) without falling into infinite recursion.

The DAG is implemented as a heterogenous tree data structure. Each level of the DAG consists of either only types of *Data* or *Module*, in other words, operations relate to each other only through data. This heterogenity allows for a flexible, pluggable approach. Both new operations and new data types can be added to the framework and these can become part of any network. As long as the output data type of a *Module* matches the input data type of another *Module*, the two can be directly connected through that *Data* type. *Modules* and *Data* are designed such that it is possible to train and detect with the same network and therefore handle a single image as well as a collection of images.

3.3. Rigid Detector in IDN

Figure 3 depicts a simple network that corresponds to the Marginal Space Learning algorithm (MSL) for estimating pose (9-dimensional similarity transform) [12]. The object localization happens in 3 operations. Given an input image (specified in *Experiment Parameters*), the initial position estimate is obtained from the *Position Detector*. The output of this operation is the list of most probable locations (3D *Position Candidates*). These locations are used during orientation

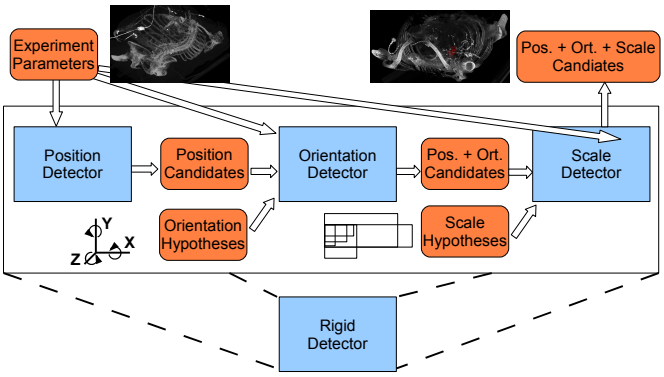

Fig. 3: Rigid Detector in IDN. The detector network estimates the parameters of a 9-dimensional similarity with three modules. See text for description.

estimation (in *Orientation Detector*) to produce the position and orientation candidates (6D *Pos. + Ort. Candidates*). Finally, the 6D candidates are used in the (*Scale Detector*) to estimate the object pose candidates, output as a list 9D pose parameters (*Pos. + Ort. + Scale Candidates*). Furthermore, the *Orientation* and *Scale Detector*s take a set of possible rotation (*Orientation Hypotheses*) and size (*Scale Hypotheses*) parameters as additional inputs. In this paper we refer to such a network as a *Rigid Dectector*.

3.4. Detecting Brain Structures in 3D Fetal US

We used the modules described in the previous sections to build a system for detecting brain anatomical structures in fetal head ultrasound volumes. The structures we are concerned with in this paper are cerebellum and corpus callosum. The output of the system is a visualization of the plane with correct orientation and centering of each structure. The structures are used in OB/GYN practice to asses the fetus health and growth.

Cerebellum pose is found using a hierarchy of rigid detectors (Section 3.3). The detection hypotheses from a lower resolution image are propagated to the higher resolution image in both training and detection. The next detector is constrained to only search within the region of interest defined by the union of neighborhood regions surrounding candidates with the highest probability. The structure at different resolutions is therefore treated as another object and the sampling of the probability distributions for computing the prediction and update steps follows Eqs. 1 and 2. This way, the search space at each resolution level is decreased which results in higher efficiency and robustness.

Figure 4 details two networks for cerebellum detection. Both networks consist of a hierarchy of detectors using volumes at resolution 4 mm, 2 mm, and 1 mm. Pipeline A uses position detectors at 4 mm and 2 mm resolutions whereas the Pipeline B uses rigid detectors. For both networks A and B,

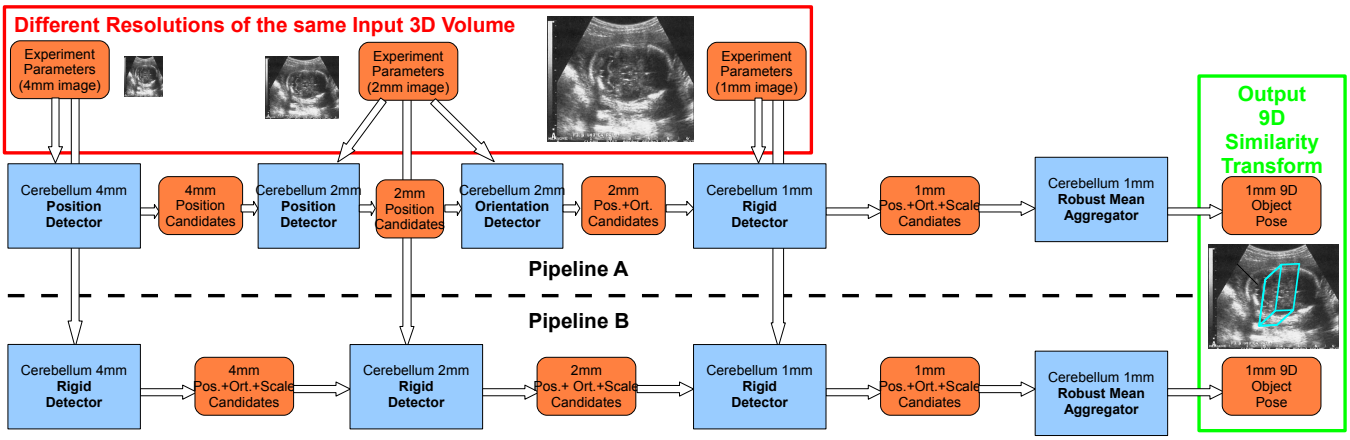

Fig. 4: Two IDN configurations for localizing cerebellum in ultrasound volumes of the fetal head. The pipelines A and B have different subnetworks at 4 mm and 2 mm resolutions and same modules at 1 mm resolution.

the cerebellum 3D pose candidates are obtained from a *Rigid Detector* (Figure 3). The final 9D similarity transformation is output by a robust *mean aggregator* that combines the 9D pose candidates weighted by their probability (Eq.3).

Once cerebellum is detected, the candidates with the highest probability are used to predict the pose parameters of the corpus callosum. This sampling and prediction is performed following Eq. 1 and using the prediction kernel from Eq. 4. The prediction kernel is Gaussian and is implemented in the *mean box predictor* module. Using the candidates with the highest probability, the corpus callosum detection continues using a rigid detector module (Section 3.3).

In this paper, we propose two corpus callosum detection pipelines (Figure 5). The Pipeline C uses cerebellum candidates from 2 mm resolution and performs detection using 2 mm resolution volume. The Pipeline D uses 1 mm candidates and volume.

3.5. Detecting Liver Boundary in 3D Liver MRI

Boundary detection occurs in a similar way as HDN, and first proceeds by detecting a shape in a learned sub-space[13]. Given a mean shape, $\hat{P} = \{\hat{p}_i \in \Re^3\}_{i=1}^n$, and a few modes (e.g., 3) of shape variation, $U_j = \{u_i^j\}_{i=1}^n$ (obtained by procrustus analysis of training data and PCA analysis), a new shape in the subspace can be synthesized as a linear combination of the modes:

$$p_i(\lambda_j, \mathbf{r}, \mathbf{s}, \mathbf{t}) = T(\mathbf{r}, \mathbf{s}, \mathbf{t})(\hat{p}_i + \sum_j \lambda_j u_i^j)$$

where $T(\mathbf{r}, \mathbf{s}, \mathbf{t})$ is a similarity matrix defined by rotation, $\mathbf{r}$, scale $\mathbf{s}$, and translation $\mathbf{t}$. The parameters in the shape space, $\theta_{pca} = \{\lambda_1, \lambda_2, \lambda_3\}$, are estimated using a discriminative classifier (Eq. 3), and the transformation $(\mathbf{r}, \mathbf{s}, \mathbf{t})$ comes directly from estimating the pose.

The second step is a free-form refinement of the mesh [13]. In this phase, the parameters θ are the locations of mesh vertices. The update, $p_i \leftarrow p_i + \alpha_i n_i$, is computed in the direction of the normal, n_i. Again, the α_i is obtained through the use of a trained discriminative model:

$$\alpha_i = \texttt{argmax}_{-\tau \leq \alpha \leq \tau} f(y_i = +1 | V_{0:t}, p_i + \alpha n_i,),$$

where τ is the search range along the normal.

This update is interleaved with surface smoothing and updating of the normal, n_i. In practice, the mesh refinement is done on a three level mesh hierarchy, where $\hat{P}_2 \subset \hat{P}_1 \subset \hat{P}_0$, with the coarser levels being detected first.

The boundary detection algorithm naturally maps to the concepts in IDN (Figure 5). The *PCA Detector*, takes as input a shape subspace and a single similarity transform (e.g., computed by aggregating a set of candidates), which it augments with the detected PCA coefficients, λ_j. The *Mesh Synthesizer*, uses these candidates along with the shape subspace to generate an output *Mesh* data. Finally, a *Boundary Detector* accepts an input *Mesh*, and outputs the refined *Mesh*. Internally, the boundary detector optionally upsamples the input *Mesh* before performing detection.

4. EXPERIMENTS

Our experiments are on detecting brain structures in fetal head ultrasound volumes using IDN networks presented in Section 3.4 and on detecting boundary of the liver in MRI volumes using IDN network from Section 3.5.

4.1. Detecting Brain Structures in 3D Fetal US

The cerebellum detection networks (Figure 4) were trained with 990 expert-annotated volumes and the corpus callosum networks (Figure 5) with 636 volumes. The volumes have average size $250 \times 200 \times 150$ mm. The cerebellum annotation line was drawn in the cerebellum measurement plane and the corpus callosum line was drawn from the bottom of the genu inside the body of corpus callosum (see Figure 7). The annotation planes and lines define the pose of each structure. A total of 107 volumes were used for testing. The separation

of the volumes into disjoint training and testing data sets was random.

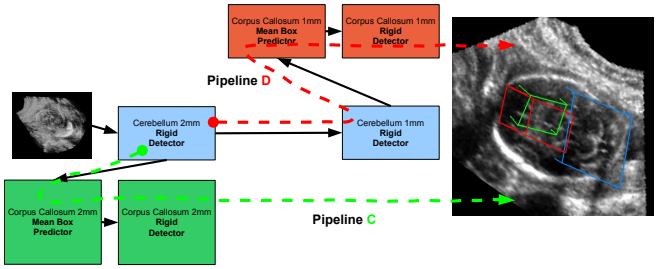

Fig. 5: Detection of corpus callosum using two different pipelines. Pipeline C and D uses candidate detections from cerebellum at 2 mm and 1 mm resolution, respectively.

Quantitative evaluation of the automatic cerebellum detection and measurement is in Table 1. The median measurement error[1] of Pipeline A and B is 3.09 mm and 3.38 mm, respectively. Pipeline A provides more accurate measurements despite the fact that the network is simpler (it uses only *position detector* at 4 mm resolution and *position* and *orientation* detector at 2 mm resolution as opposed to *rigid detector* as in Pipeline B). This is caused by an insufficient amount of detail at the 4 mm resolution to disambiguate the orientation of the fetus skull (see Figure 6). Several examples of automatic measurements are in the top of Figure 7.

Fig. 6: The coarse 4 mm resolution volumes have insufficient details causing ambiguity of the fetus skull orientation. The annotation line highlights the cerebellum which is difficult to distinguish at 4 mm but is much more clear at 2 mm and 1 mm resolutions.

Quantitative evaluation of the automatic corpus callosum detection is in Table 1. The median measurement error of Pipeline C and D is 4.83 mm and 4.20 mm, respectively. The results at 1 mm resolution (Pipeline C) are more accurate thanks to the more reliable cerebellum candidates at this resolution.

	Median	Std.D.		Median	Std.D.
P.A	3.09	1.71	P.C (2 mm)	4.83	2.38
P.B	3.38	1.78	P.D (1 mm)	4.20	2.13

Table 1: Detection error [mm] for Cerebellum (left) and corpus callosum (right). Pipelines A and D have lower error. See text for discussion.

[1]The measurment error for 3D Fetal Structures is computed as the maximum of the two distances between corresponding end points of the annotation line and the detection line. Average annotation line length is 19.78 mm and 10.26 mm for cerebellum and corpus callosum, respectively.

4.2. Detecting Liver Boundary in 3D Liver MRI

To detect and segment the liver boundary in 3D MRI data, we have configured our boundary detection modules as illustrated in Figure 8. The mean liver mesh and shape subspace are built by performing Procrustes analysis on manually annotated training examples. The mesh hierarchy consists of a low, medium, and high resolution meshes with 602, 1202, and 2402 vertices, respectively. The low and medium resolution boundary detectors use 3 mm resolution volumes and the high resolution boundary uses 1 mm resolution volumes.

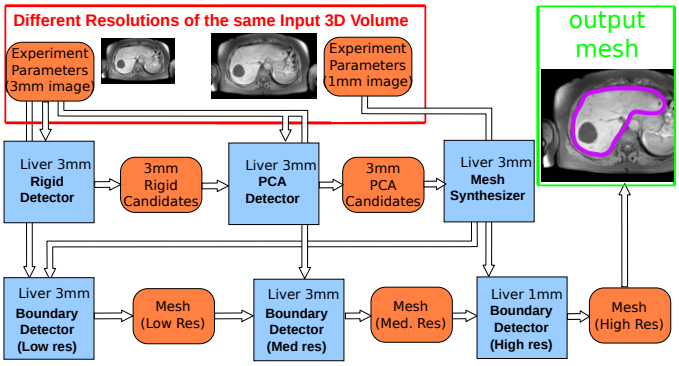

Fig. 8: The MRI liver segmentation network uses the *rigid detector* to locate the liver. Then several layers boundary detection is performed on different image and mesh resolutions.

The pipeline was trained on 59 annotated input volumes with size as large as $420 \times 300 \times 432$ mm. Using 3-fold cross-validation, we computed the mesh-to-mesh distance of the detected results. Through IDN, we easily reconfigured our detection pipeline and removed intermediate modules such that different stages of the algorithm can be evaluated (Table 2). The table shows that all detection phases are necessary to achieve the highest accuracy. Figure 9 illustrates some boundary detection results for the entire pipeline.

Boundary pipeline	Mean	Mean Std	Median
Entire	2.53	1.82	1.81
No med.	3.72	2.17	3.28
No Pca,no med.	5.26	2.38	4.79

Table 2: Mesh-to-mesh statistics on Liver MRI boundary detection using the full pipeline, on pipelines without 3mm medium resolution boundary detector (No med.), and without PCA and medium resolution boundary (No Pca, no med).

5. CONCLUSION

We have proposed the Integrated Detection Network (IDN) framework as a flexible design to manage complex large-scale learning-based detection algorithms. The framework relies on a simple but powerful abstraction of representing algorithm components as either *Modules* or *Data*. IDN allows rapid prototyping, tuning, and reconfiguration of detection algorithms,

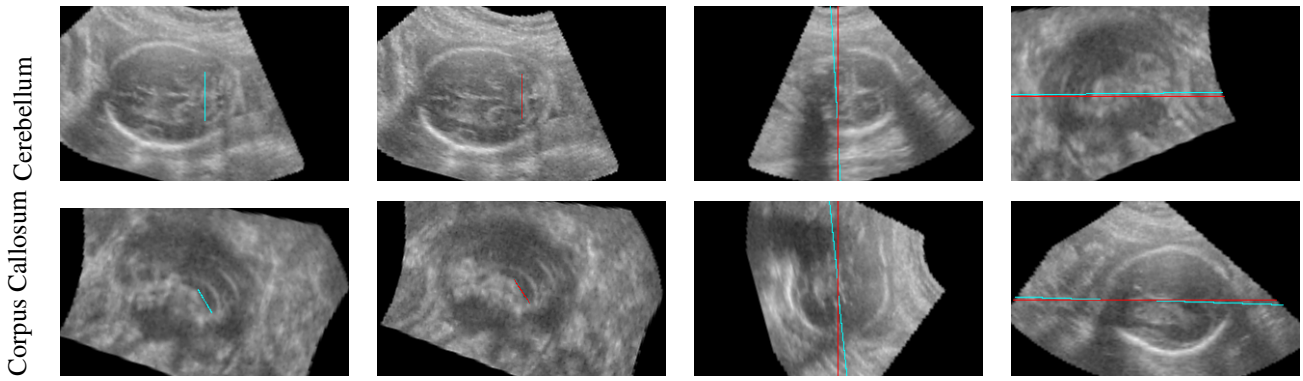

Fig. 7: Final hierarchical detection result of Pipelines A and D (cyan) compared to ground truth (red). The last two columns show the agreement of the detection plane in the sagittal and coronal cross section.

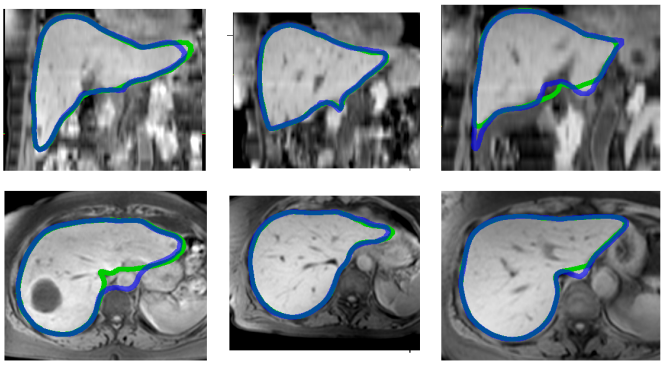

Fig. 9: Sample liver boundary detection results (blue) in MRI with ground truth (green).

and ensures that training uses the same inter-object dependencies as detection.

In this work, we proposed two different networks and their variants for hierarchical learning-based detection. The first network, anatomical structure detection in 3D fetal US, achieved accuracy of 3.09 mm (cerebellum) and 4.20 mm (corpus callosum). The second, accurate 3D liver boundary detection in MRI images, achieved accuracy of 2.53 mm. The modularity of IDN promotes reuse, since these modules can readily be reconfigured for other applications and newly developed algorithms can be easily integrated into existing detection pipelines.

6. REFERENCES

[1] P. Viola and M. J. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. CVPR*, 2001, vol. 1, pp. 511–518.

[2] Z. Tu, "Probabilistic boosting-tree: Learning discriminative models for classification, recognition, and clustering," in *Proc. ICCV*, 2005, vol. 2, pp. 1589–1596.

[3] C. Desai, D. Ramanan, and C. Fowlkes, "Discriminative models for multi-class object layout," in *Proc. ICCV*, 2009.

[4] C. Galleguillos, A. Rabinovich, and S. Belongie, "Object categorization using co-occurrence, location and appearance," in *Proc. CVPR*, Anchorage, AK, 2008.

[5] S. Kumar and M. Hebert, "Discriminative random fields: a discriminative framework for contextual interaction in classification," in *Proc. ICCV*, 2003, vol. 2, pp. 1150–1157.

[6] M. Sofka, J. Zhang, S. K. Zhou, and D. Comaniciu, "Multiple object detection by sequential Monte Carlo and hierarchical detection network," in *Proc. CVPR*, San Francisco, CA, 13–18 June 2010.

[7] G. Carneiro, F. Amat, B. Georgescu, S. Good, and D. Comaniciu, "Semantic-based indexing of fetal anatomies from 3-D ultrasound data using global/semi-local context and sequential sampling," in *Proc. CVPR*, Anchorage, AK, 2008.

[8] T. Heimann, B. van Ginneken, and M.A. Styner et al., "Comparison and evaluation of methods for liver segmentation from ct datasets," *IEEE T. Med. Imaging*, vol. 28, no. 8, pp. 1251–1265, 2009.

[9] L. Massoptier and S. Casciaro, "Fully automatic liver segmentation through graph-cut technique," in *Proc. EMBS*, 2007, pp. 5243–5246.

[10] Kan Cheng, Lixu Gu, Jianghua Wu, and Jianrong Xu Wei Li, "A novel level set based shape prior method for liver segmentation from MRI images," in *Proc. Medical Imaging and Augmented Reality*, 2008, pp. 150–159.

[11] U. Vovk, F. Pernus, and B. Likar, "A review of methods for correction of intensity inhomogeneity in MRI," *IEEE T. Med. Imaging*, vol. 26, no. 3, pp. 405–421, 2007.

[12] Y. Zheng, A. Barbu, B. Georgescu, M. Scheuering, and D. Comaniciu, "Four-chamber heart modeling and automatic segmentation for 3-D cardiac CT volumes using marginal space learning and steerable features," *IEEE T. Med. Imaging*, vol. 27, no. 11, pp. 1668 –1681, nov. 2008.

[13] H. Ling, S. K. Zhou, Y. Zheng, B. Georgescu, M. Suehling, and D. Comaniciu, "Hierarchical, learning-based automatic liver segmentation," in *Proc. CVPR*, Los Alamitos, CA, USA, 2008.