A Deep Convolutional Neural Network-Based Framework for Automatic Fetal Facial Standard Plane Recognition

Zhen Yu, Ee-Leng Tan, Dong Ni, *Member, IEEE*, Jing Qin, *Member, IEEE*, Siping Chen, Shengli Li, Baiying Lei, *Member, IEEE*, and Tianfu Wang

***Abstract*—Ultrasound imaging has become a prevalent examination method in prenatal diagnosis. Accurate acquisition of fetal facial standard plane (FFSP) is the most important precondition for subsequent diagnosis and measurement. In the past few years, considerable effort has been devoted to FFSP recognition using various hand-crafted features, but the recognition performance is still unsatisfactory due to the high intraclass variation of FFSPs and the high degree of visual similarity between FFSPs and other non-FFSPs. To improve the recognition performance, we propose a method to automatically recognize FFSP via a deep convolutional neural network (DCNN) architecture. The proposed DCNN consists of 16 convolutional layers with small 3 × 3 size kernels and three fully connected layers. A global average pooling is adopted in the last pooling layer to significantly reduce network parameters, which alleviates the overfitting problems and improves the performance under limited training data. Both the transfer learning strategy and a data augmentation technique tailored for FFSP are implemented to further boost the recognition performance. Extensive experiments demonstrate the advantage of our proposed method over traditional approaches and the effectiveness of DCNN to recognize FFSP for clinical diagnosis.**

***Index Terms*—Deep convolutional neural network, standard plane recognition, transfer learning, ultrasound image.**

Manuscript received September 21, 2016; revised March 24, 2017 and May 8, 2017; accepted May 9, 2017. Date of publication May 16, 2017; date of current version May 3, 2018. This work was supported in part by National Natural Science Foundation of China under Grants 81571758, 61571304, 61402296, 61571304, and 61427806, in part by National Key Research and Develop Program 2016YFC0104703, in part by Guangdong Medical under Grant B2016094, in part by Shenzhen Peacock Plan KQTD2016053112051497, in part by Shenzhen Key Basic Research Project JCYJ20150525092940986 and JCYJ20150525092940988, and in part by the National Natural Science Foundation of Shenzhen University 827000197. *(Corresponding author: Baiying Lei and Tianfu Wang.)*

Z. Yu, D. Ni, S. Chen, B. Lei, and T. Wang are with the National Regional Key Technology Engineering Laboratory for Medical Ultrasound, Guangdong Key Laboratory for Biomedical Measurements and Ultrasound Imaging, School of Biomedical Engineering, Health Science Center, Shenzhen University, Shenzhen 518060, China (e-mail: yishon555@outlook.com; nidong@szu.edu.cn; chensiping@szu.edu.cn; leiby@szu.edu.cn; tfwang@szu.edu.cn).

E.-L. Tan is with the Beijing Sesame World Co., Ltd, Beijing 100190, China (e-mail: eltan@ntu.edu.sg).

J. Qin is with the Centre for Smart Health, School of Nursing, The Hong Kong Polytechnic University, Hong Kong (e-mail: harry.qin@polyu.edu.hk).

S. Li is with the Department of Ultrasound, Affiliated Shenzhen Maternal and Child Healthcare Hospital of Nanfang Medical University, Shenzhen 518060, China (e-mail: lishengli63@126.com).

Digital Object Identifier 10.1109/JBHI.2017.2705031

I. Introduction

ULTRASOUND (US) screening is commonly used for pregnancy diagnosis in routine clinical examination as US screening is low-cost and radiation-free [1]–[8]. Clinically, antenatal US screening is usually performed between 18 and 24 weeks of gestation. Following a standardized protocol, serial standard images of fetal structures are acquired for biometric measurement (e.g. measurement of biparietal diameter) and detection of malformation [9]. Thus, acquisition of the standard planes (e.g. fetal facial standard plane, FFSP) from US is essential for accurate fetal diagnosis and subsequent measurement [1], [3], [5], [6]. Clinicians must go through substantial training and have extensive knowledge to effectively identify and evaluate the FFSP. However, this labor-intensive and subjective assessment is too time-consuming and can be unreliable due to huge variations of assessments among different clinicians. Furthermore, there is a lack of experienced clinicians in the unprivileged regions. Therefore, an automatic FFSP recognition method is highly desired.

FFSP has high intra-class and low inter-class variations caused by various fetal postures [1], different scanning orientations, and numerous artifacts such as speckle noise and shadow [10], [11]. As shown in Fig. 1, there is no distinguishing difference between standard planes and other planes (non-FFSP). Hence, it is a challenging task to select the standard plane, particularly for automatic recognition algorithms. To address this challenge, numerous methods have been proposed in the past few years [1], [3], [5], [6]. The typical pipeline of these methods is composed of two distinct steps: feature extraction and classification [12]–[15]. Traditional methods mainly used low-level hand-crafted features [3], [7], [8], [16]–[18] (i.e., scale-invariant feature transform (SIFT) [19], Dense -SIFT (DSIFT) [20], Haar, and histogram -of-gradient (HOG) [21]) as image descriptors to represent the images. Low-level features are then encoded by popular algorithms such as bag of visual words (BoVW) [22], [23], vector of locally aggregated descriptor (VLAD) [24], Fisher vector (FV) [25], [26], and multi-layer Fisher vector (MFV) [1] to enhance the effectiveness of recognition [1], [2]. Finally, support vector machine (SVM)

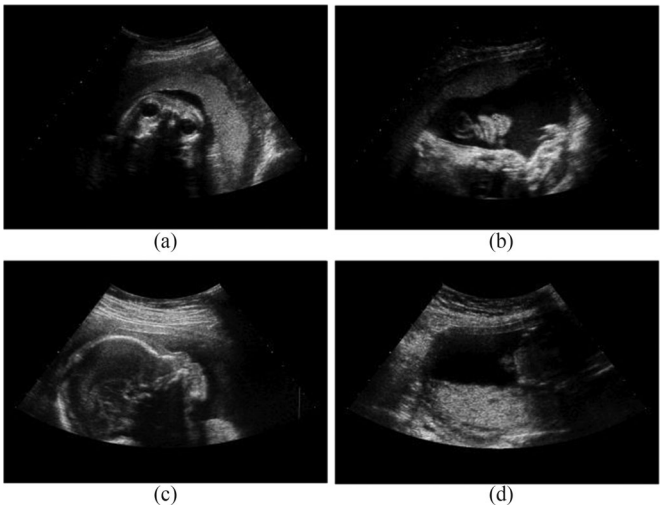

Fig. 1. Illustration of the high intra-class variation and low inter-class variation of FFSP images: (a) axial plane; (b) coronal plane; (c) sagittal plane; (d) other planes (non-FFSP).

is adopted to classify these features. However, existing hand-crafted features extracted from the consecutive US images are still inadequate for accurate FFSP recognition.

In the past few years, great success has been witnessed in image recognition tasks primarily because of the available large-scale annotated datasets (e.g., ImageNet) and powerful representation ability of deep convolutional neural network (DCNN) [27], [28]. In general, DCNN consists of alternating convolutional layers and pooling layers. DCNN automatically learns feature representations from raw data for recognition or detection without any manually designed features. Furthermore, DCNN can approximate certain function classes far more efficiently than shallow networks [29]. Hence, deep representations discovered by DCNN are more robust and sophisticated than the standard hand-crafted features. In literature [29]–[33], there are various studies using deep representations such as 16-layer VGG Net [32], 30-layer batch normalization (BN) Net [31], and even 152-layer ResNet [30], which demonstrate outstanding performance in numerous challenging benchmark datasets (i.e., ImageNet). Due to the limited availability of large medical datasets and convergence problem, only a few studies have applied these deep models on medical US image datasets. Although many medical image recognition and detection tasks [34]–[39] have significantly benefited from DCNN model, the application of DCNN model in US image is less studied. Studies on systematically investigating and analyzing the performance of DCNN models for US image processing are scarce too. Inspired by the impressive results achieved by DCNN in a myriad of fields, a DCNN model is applied for our FFSP recognition task.

We propose to recognize FFSP via a DCNN model (19 layers) with small size kernel (3×3) to capture the complexity of the planes. Specifically, a global average pooling (GAP) in the last pooling layer is configured with our DCNN model to further improve performance and efficiency. To address the issue of the optimization and underlying overfitting problem caused by small training dataset size, we utilize the transfer learning strategy based on ImageNet dataset and data augmentation technique tailored for FFSP detection. To the best of our knowledge, this is the first automatic FFSP recognition method using DCNN architecture. Our experiments reveal that the proposed method exhibits great potential in the practical application of routine US examination and prenatal care.

We mainly focus on the following issues in this paper: i) how does the depth (complexity) of the DCNN affect the FFSP recognition results? ii) how do the key DCNN elements affect the FFSP recognition performance? iii) how is the FFSP recognition performance via DCNN compared with the traditional hand-crafted feature and classification models?

The rest of this paper is organized as follows. Section II reviews various features and models for FFSP recognition. Section III introduces the methodology of the proposed method. Experimental setup and results are presented in Section IV. Section V discusses the properties of our proposed method and the research directions. Finally, the conclusions are presented in Section VI.

II. Related Work

A. Hand-Crafted Feature Based Classification

Traditional hand-crafted feature based classification models generally consist of three steps: (i) feature extraction [6], [40], (ii) feature encoding [12], [24], [41], [42], and (iii) feature classification [43]. Spatial pyramid matching (SPM) [23], [44] can be adopted to incorporate spatial information of image features for feature enhancement.

Carefully designed data-specific features are particularly important in conventional models. In the last few years, significant effort has been made to devising and designing appropriate features for image representation [3], [7], [8], [16]. The typical and prevalent feature representations include SIFT, DSIFT, Haar, HoG, and combination of these feature representation with intensity, shape, motion, and edges [3], [7], [8], [16]. To further improve classification performance, feature encoding approaches (e.g., BoVW, VLAD, FV and MFV) [2], [23], [24], [26] have been introduced to produce more powerful representative and stable information. For example, Lei *et al*. [1] proposed a hand-crafted DSIFT feature representation method based on a MFV feature encoding method for FFSP recognition.

B. Deep Convolutional Neural Network

DCNN is endowed with an impressive representation capability for recognition or detection task according to the given training dataset [45]. Generally, DCNN model composes of multiple processing layers to learn different level features. Combining these hierarchy features preserves extremely discriminative and effective deep representations [29], [45]. Hence, state-of-art performance is achieved in numerous applications [28], [36], [37], [45], [46].

The deep hierarchy architecture of DCNN model is of vital importance due to its powerful capability of representation learning. Well-designed initialization strategies and activation functions [47], [48], efficient intermediate

regularization strategies [31], [49], [50] are proposed in recent years which significantly improves the optimization of deep models [34], [51], [52]. With remarkable performance in natural image processing and natural language processing, DCNN has achieved its dominance in machine learning domain and has been widely applied in the medical image analysis field [4], [35], [53]–[56]. Many studies have reported promising results in various applications such as object recognition [53]–[55], [57], detection [4], [58], and segmentation [56], [59]–[61]. For instance, Chen *et al*. [5] used a DCNN model to recognize fetal abdominal standard plane (FASP) in fetal ultrasound application. This method was based on natural image dataset (i.e., ImageNet) pre-trained AlexNet (an eight layers DCNN model). Chen *et al.* [62] proposed a framework based on convolutional and recurrent neural networks to detect three different fetal ultrasound standard planes from US videos.

III. Methodology

A. Network Architecture

1) Convolutional Neural Network (CNN): Inspired by the biological neural system, CNN has achieved promising performance in object recognition [28], [30], [33], [34], detection [36], and segmentation [37], [38]. The main component of CNN is the convolutional (Conv) layer, which includes a set of neurons. Each neuron has a group of learnable weights and one bias. Neurons in the Conv layer take local receptive fields of feature maps in the previous layers as input. Assuming χ_j^l is the *j*-th feature map in *l*-th layer, and $\chi_m^{l-1}(m = 1, \ldots, M)$ are the outputs of *l*-1th layer, χ_j^l is calculated by

$$\chi_j^l = \delta\left(\sum_{m=1}^{M} w_{jm}^l * \chi_m^{l-1} + b_j^l\right), \tag{1}$$

where w_{jm}^l is the weight connected to the *m*-th feature map in the previous layer, b_j^l is the *j*-th bias of the *l*-th layer, and $\delta(\blacksquare)$ is the rectified linear unit. In general, several pooling layers are periodically inserted in-between successive Conv layers to progressively decrease output scale of the intermediate activation maps. In fully-connected (FC) layer, neurons have connections to all activations of the previous layer. The final classification task is performed by softmax layer. Let $I_i(i = 1, \ldots, N)$ be the input images, $T_i \in \{0, 1, \ldots, K\}$ be the corresponding ground-truth labels for the input FFSP image I_i, the loss function is defined as

$$L_i = -\sum_{i=1}^{N}\sum_{k=1}^{K}\left(1\{T_i = C_k\}\left(\log p\left(I_i \in C_k | w, b\right)\right)\right) \tag{2}$$

$$\log p\ (I_i \in C_k | w, b) = \frac{e^{f_{y\,i}}}{\sum_{j=1}^{K} e^{f_j}}\ k = 1, \ldots, K \tag{3}$$

where $p(I_i \in C_k | w, b)$ is the probability output of sample I_i, C_k is all categories, $\mathbf{1}\{\mathrm{T}_i = C_k\}$ is an indicator function, when $I_i \in T_i$, the output is 1, otherwise 0, f_j is the output of network before softmax layer for training image I_i.

TABLE I
Detailed Configuration of Proposed DCNN Model

Input (224 × 224 RGB image)	
Layer	Parameter
Conv Layer 1-2	3 × 3, 64, stride = 1
Max pool	2 × 2, stride = 2
Conv Layer 3-4	3 × 3, 128, stride = 1
Max pool	2 × 2, stride = 2
Conv Layer 5-8	3 × 3, 256, stride = 1
Max pool	2 × 2, stride = 2
Conv Layer 9-12	3 × 3, 512, stride = 1
Max pool	2 × 2, stride = 2
Conv Layer 13-16	3 × 3, 512, stride = 1
Global average pool	14 × 14, stride = 0
FC 1-2, 1 × 1, 1024	
FC 3, 1 × 1, 4	
Softmax	

2) Proposed DCNN Model: Appropriate design of network architecture can improve the performance significantly. Our DCNN model shares the basic architecture of the typical VGG-Net [32]. Specifically, it contains sixteen Conv layers, and three FC layers. Moreover, the kernel size of all the Conv layers is 3 × 3, and convolution strides are fixed to one pixel. The use of a small receptive field is usually regarded as a positive option for model design, which not only significantly reduces the number of model parameters, but also allows for increasing the depth in limited computational budget and captures subtle notions [32], [53].

Due to the large kernel size and dense connection of the first FC layer, the FC layers occupy most of the free parameters, and thus the model is prone to overfitting. In [29], GAP is proposed as structural regularizer to bring feature maps into correspondence with categories. Supposing in *l*-th convolutional layer, we obtain $w_i^l \times h_i^l \times d^l$ spatial feature map $\mathcal{M}_i^l$ for *i*-th input image, where w_i^l and h_i^l denote the width and height, d^l is the depth or channels of the feature map. Then, GAP accumulates the spatial information and converts the spatial activation maps into a single feature vector:

$$f_i^l = \frac{1}{w_i^l \times h_i^l}\sum \mathcal{M}_i^l(:, :, j),\ j = 1, 2, \ldots d^l, \tag{4}$$

which provide more geometric invariance of the learned representation. In this paper, we utilize the GAP technique in the last pooling layer to further reduce network parameters and enhance FFSP recognition performance. We denote our model as CNN-19-GAP in this paper hereafter. Table I illustrates the detailed configurations of the proposed DCNN model.

In our DCNN model, the channels of Conv layers are increased from 64 to 512 to ensure the complexity and abstraction of learned features. For the subsequent FC layers, we set the depth dimension as 1024-1024-4. Since our FFSP recognition task contains four classes, the last FC layer has four channels. Compared with the original VGG-Net, we significantly reduce the number of kernels of FC layers to shorten the time required for training. The study in [63] indicates that the dimensionality of CNN FC layers can be reduced substantially without an

adverse effect on performance. For this reason, dimensionality reduction is also performed on our DCNN model for FFSP recognition, and the dimensionality is reduced from 1024 to 512, and further to 256.

B. Image Preprocessing and Data Augmentation

Since DCNN models usually have a large number of parameters to learn, they cannot be effectively trained unless a sufficient number of training images are provided. However, collecting and annotating a large number of medical images are extremely tedious. Enlarging or augmenting the original dataset by the label-preserving transformations is one of the most common and effective approaches [28] to address this issue. Based on this idea, we generate new images by randomly extracting sub-images and their horizontal reflections from the original US images after excluding non-US regions.

In our dataset, there is a data imbalance issue, namely, there are more non-FFSP images than FFSP images. Hence, we adopt different sampling rates for FFSP and non-FFSP images to resolve this unequal distribution of our dataset [64]. Specifically, for each 768 $\times$ 576 original training image, we first crop the corner region (i.e., non-US regions) and rescale the short side of remaining part to 256 while keep aspect ratio. Subsequently, if the image belong to FFSP, five 224 $\times$ 224 sub-images and their horizontal reflections as FFSP images and one sub-image as non-FFSP images. To evaluate the efficacy and reliable result of this strategy, training our DCNN with and without data augmentation is introduced in the experiment section.

C. Knowledge Transferred DCNN for FFSP Recognition

Even though DCNN produces effective feature representation that can be applied in many medical fields, the only limited training data is available. Overfitting issue tends to occur in fully supervised training of deep architectures and downgrades the learning performance. To address this issue, transferring parameters from DCNN pre-trained on large dataset is a useful approach [51]. Although there are some differences between natural images and medical images, the recent studies illustrate the potential of knowledge transferring from natural image domain to medical image domain [4], [5], [65]. The common transfer learning strategy involves pre-training a base network and then transferring its parameters of a certain number of layers of the pre-trained network to the target network. The remaining layers of the target network are initialized with random weights and trained on the target data.

There are currently two major transfer learning approaches: 1) keep the transferred layers frozen, make the transferred layers as a fixed feature extractor for the new dataset, and only training random initialized layers from scratch; 2) fine-tuning these base layers during training using the new dataset. The efficacy of transfer learning is mainly derived from the hierarchical attribute of features extracted by different learning layers. Features are more generalized in the early layer and more dataset-specific features are found in the subsequent layers. This transition of general features to specific features occurs in a CNN network, which is detailed in [51].

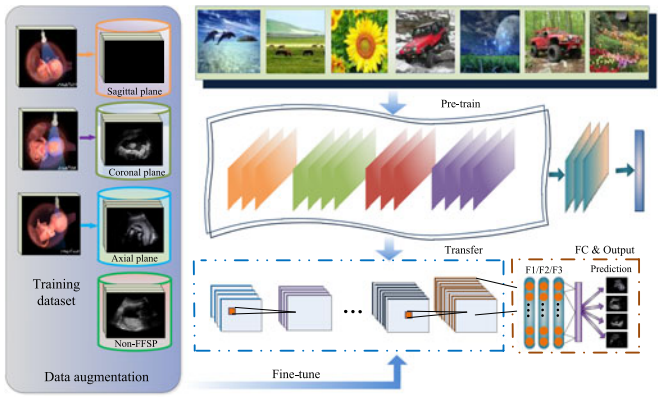

Fig. 2. Flowchart of proposed fine-tuning strategy. All Conv layers of pre-trained network are transferred to target network.

Considering the distinct differences between our FFSP dataset and ImageNet dataset, we choose to fine-tune the pre-trained network on our dataset. As suggested in [65], the depth of transfer learning is essential for medical image applications. In our fine-tuning strategy setting, we transfer all the Conv layers except the last three FC layers of Alex-Net pre-trained on ImageNet, and we denote it as CNN-8-TR hereafter. Also, we transfer all the Conv layers except the last three FC layers of VGG-Net pre-trained on ImageNet, and we denote them as CNN-16-TR, CNN-19-TR, and their variant using GAP strategy as CNN-16-GAP-TR, CNN-19-GAP-TR hereafter. Fig. 2 illustrates the flowchart of our fine-tuning approach.

D. Network Training and Classification

Several theoretical studies revealed that deep network is difficult to train [47], [48]. Deep models tend to suffer from the convergence problem as the depth of the model increases. To tackle this, enhanced techniques have been proposed [31], [47]–[50]. For instance, batch normalization (BN) [31] is an effective approach to resolve the convergence problem and accelerates the training process. The BN algorithm addresses the internal covariate shift issue during the training of network by normalizing layer inputs. In general, the normalization is employed before the nonlinearity. Interested readers can refer to [31] for more details.

For our experiment, BN is utilized to assist in training CNN-16 (GAP) and CNN-19 (GAP) using random initialization. The models cannot converge without BN algorithm. The network training procedure mainly follows the study in [28]. We adopt a stochastic gradient decent algorithm with a mini-batch size of 64 (32 for CNN-16 (GAP) and CNN-19 (GAP)). The momentum variable and weight decay for the weights updating in all CNN models are 0.9 and 0.0005, respectively. During the training process, we use different learning rate (LR) for the transferred layers and randomly initialize these layers. For the models trained via random initialization, the learning rate is initialized as 0.01. For the models trained via fine-tuning strategy, the initial LR is set as 0.01 and 0.001 for the transferred Conv layers and last three FC layers, respectively. LR is decreased at every epoch throughout the training.

To classify a test image, the same prepossessing in Section III-B is applied. For each sub-image, we subtract the mean image pixel calculated from training dataset and input it into CNN model. Finally, the output prediction of the model is based on the class scores, and the output prediction with the highest-class score gets the classification result.

IV. Experimental Setup and Results

A. Experimental Setup

1) FFSP Dataset: The study protocol was reviewed and approved by the ethics committee of our institution, and informed consent was obtained from all subjects.

Training dataset: The experiments are conducted using our in-house collected FFSP dataset extracted from US videos. These videos were acquired by a US scanner from Siemens Acuson Sequoia 512 at the Shenzhen Maternal and Child Health Hospital. The fetal gestational ages from the images range from 20 to 36 weeks. Experienced obstetricians are engaged to manually annotate the US images. Our training dataset includes 375 images of axial plane, 257 images of coronal plane, 405 images of sagittal plane, and 3812 images randomly extracted from the remaining images without any standard planes (Non-FFSP). Data augmentation described in Section III is used to enlarge the dataset and balance the numbers of US images for each class. The data augmentation increases our dataset to 3750 axial planes, 2570 coronal planes, 4050 sagittal planes, and 3812 non-FFSP planes. In our experiment, the whole training dataset is randomly divided into two parts, where 4/5 of them are adopted for training and 1/5 of them are used for validation.

Test dataset: Our test data is based on a total of 2418 images (491 axial planes, 127 coronal planes, 174 sagittal planes, and 1626 non-FFSP planes). As described before, the classification result of a test image is based on its sub-images.

2) Evaluation Methods: For the evaluation of different FFSP recognition methods, training-validation-test strategy is used. Training of all the models is performed based on the training data, and the validation set is adopted for tuning the hyper-parameters. The overall classification performance of each system is evaluated from the test dataset.

Popular evaluation metrics such as overall accuracy, precision, recall, and F1-score are computed to quantitatively evaluate all models. For DCNN models, we qualitatively assess the classification results by visualizing the high-level feature representations and intermediate feature maps.

B. Quantitative Evaluation of the Proposed Method

We evaluate and analyze various factors in our proposed DCNN based approach for FFSP image recognition.

1) Effectiveness of Model Depth: To demonstrate the effectiveness of DCNN architecture depth, our evaluation consists of two parts. First, we train the DCNN models with different depths, and then obtain the evaluation results from the test images. Second, the training and test datasets are fed into trained models to get the deep representations. We subsequently train and test SVM classifiers with these representations. Similar to hand-crafted feature based classification models, different kernels are exploited and the stochastic dual coordinate ascent optimization algorithm is adopted [66].

TABLE II
FFSP Recognition Results With Various Models (%)

Model		Kernel	Accuracy	Precision	Recall	F1-score
(a)	DSIFT-BoVW	Linear	51.08	56.51	78.83	65.83
		Hell	55.00	57.66	80.00	67.02
		Chi2	81.21	70.49	82.41	75.99
(b)	DSIFT-VLAD	Linear	84.63	77.82	74.24	75.99
		Hell	83.95	74.45	77.27	75.84
		Chi2	84.66	76.49	77.39	76.94
(c)	DSIFT-FV	Linear	84.29	73.65	79.01	76.23
		Hell	73.12	62.21	83.03	71.13
		Chi2	79.21	67.83	83.23	74.74
(d)	DSIFT-MFV	Linear	81.29	71.51	83.97	77.24
		Hell	81.29	70.91	85.50	77.53
		Chi2	86.58	77.86	80.50	79.16
(e)	CNN-8-SVM	Linear	75.57	67.80	89.20	77.04
		Hell	76.83	68.56	89.63	77.69
		Chi2	78.79	70.25	90.24	79.00
(f)	CNN-16-SVM	Linear	86.22	76.02	91.77	83.15
		Hell	86.14	75.73	92.33	83.21
		Chi2	86.72	76.36	92.40	83.62
(g)	CNN-16-GAP-SVM	Linear	94.61	87.46	95.67	91.38
		Hell	94.99	88.34	96.09	92.05
		Chi2	95.32	88.94	95.94	92.31
(h)	CNN-19-SVM	Linear	92.98	85.63	95.95	90.50
		Hell	93.11	85.73	96.00	90.57
		Chi2	93.53	86.25	96.15	90.93
(i)	CNN-19-GAP-SVM	Linear	96.32	91.91	96.92	94.35
		Hell	96.36	91.88	96.94	94.34
		Chi2	96.32	91.86	96.92	94.32
(j)	CNN-8-RI	—	81.21	87.77	88.47	88.10
(k)	CNN-8-TR	—	78.74	90.33	91.25	90.79
(l)	CNN-16-RI	—	81.92	89.78	90.21	89.99
(m)	CNN-16-TR	—	87.59	92.50	92.76	92.63
(n)	CNN-16-GAP-RI	—	88.52	86.69	88.26	87.47
(o)	**CNN-16-GAP-TR**	—	**95.07**	**95.18**	**95.32**	**95.25**
(p)	CNN-19-RI	—	81.34	89.14	89.37	89.25
(q)	CNN-19-TR	—	93.03	95.58	95.85	95.71
(r)	CNN-19-GAP-RI	—	84.47	77.88	82.83	80.82
(s)	**CNN-19-GAP-TR**	—	**96.53**	**96.98**	**97.00**	**96.99**

Note that linear means linear kernel, Hell denotes the Hellinger's kernel, and Chi2 represents x^2 kernel.

Apart from the proposed DCNN, several DCNN architectures with varied network depths are trained for comparison. Each network is trained by random initialization and fine-tuning via the corresponding ImageNet pre-trained model. In addition, we compare various algorithms with the recent developed CNN architectures (e.g. CNN-8 based on AlexNet [28], CNN-16 and CNN-19 based on VGG-Net [32]) to investigate the effect of DCNN depth (complexity).

Detailed comparison of the FFSP recognition results are summarized in Table II (note that RI means random initialization, TR denotes the transfer learning via fine-tuning). When the depth increases from 8 to 16 layers, we observe an improvement of ∼0.7%, ∼2%, ∼1.8%, and ∼1.9% in accuracy, precision, recall, and F1-score, respectively, in case of training via random initialization ((j) vs. (l)). We also observe that the recognition performance of CNN-19-RI is slightly lower than CNN-16-RI ((l) vs. (p)). Similar observation is found in our proposed CNN-19-GAP model ((n) vs. (r)). The degraded performance demonstrate that

TABLE III
PARAMETER MEMORY OF MODELS

Model	Parameter Memory
CNN-16	158 M
CNN-16-GAP	62 M
CNN-19	178 M
CNN-19-GAP	82 M

TABLE IV
EFFECT OF FC LAYER CHANNELS (%)

Model	Accuracy	Precision	Recall	F1-score
CNN-19-TR-256	91.06	93.75	93.88	93.81
CNN-19-TR-512	90.77	94.45	94.71	94.58
CNN-19-TR-1024	93.03	95.58	95.85	95.71

it is more difficult to train DCNN via random initialization with the increase of model depth. By contrast, we observe that the model with 19 layers trained with fine-tuning strategy outperforms the model with 16 layers. In the fine-tuning case, deeper models yield considerable and remarkable improvements over the rest (e.g., a performance improvement of 2%∼10% in accuracy ((k) vs. (m) vs. (o), and (o) vs. (s)).

To further investigate the effect of deep representations from different DCNN models, we extract the output feature vector before classification layers of different networks for each FFSP image and train this feature with SVM classifier. In our experiment, features are extracted from the fine-tuned version of each network. As shown in Table II, features extracted from deeper model show superiority over the rest with considerable improvement ((e) vs. (f), (f) (g) vs. (h) (i)). Overall, deep features with Chi2 kernel outperform linear and hell kernels by 1%∼3%, and the CNN-19-GAP features achieve the best result with an accuracy of 96.32%, a precision of 91.91%, a recall of 96.92%, and a F1-score of 94.35%.

2) Effectiveness of Global Average Pooling: To investigate the impact of GAP adopted in our DCNN model, we replace the conventional max pooling in last pooling layer by GAP in our configuration and evaluate the performance. Similarly, the deep representations of FFSP images are extracted for evaluation. The same evaluation is also conducted on the model of CNN-16 to further investigate the generalization of the GAP. As illustrated in Table II, the DCNNs equipped with GAP remarkably outperform the corresponding model configured with conventional max pooling. A distinct boosting of approximately 7% and 3% is observed on each metric for CNN-16 versus CNN-16-GAP ((n) vs. (l), (o) vs. (m)), and CNN-19 versus CNN-19-GAP ((r) vs. (p), (s) vs. (q)), respectively. Similarly, the improvements also occur when we extract the deep representations to train SVM classifier ((f) vs. (g), (h) vs. (i)). It is worth noting that there is no significant performance improvement, and the performance was degraded when the model is trained from scratch among network with different depths ((r) vs. (n)). This demonstrates that the deeper network leads to the optimization difficulty due to random initialization, which downgrades the recognition performance.

As illustrated in previous studies [29], [30], [33], the special pooling also significantly reduces the parameter memory of a network. Table III summarizes the specific memory value for each model. We can see that there is a drop of memory by approximately 50% when the GAP configuration is adopted. From the experimental results of the GAP in our model, it is evident that GAP configuration not only makes the training more efficient by reducing the complexity of network, but also significantly enhances the representation ability of the deep models.

3) Effectiveness of FC Layer Channels: We exploit the DCNN with different channels of FC layers. As shown in Table IV, there is only a slight drop of approximately 2% in accuracy and 1% in other metrics when the channels of FC layer is reduced from 1024 to 512. Meanwhile, there is a drop of approximately 2% in terms of all metrics when the dimensionality is further reduced to 256.

4) Effectiveness of Transfer Learning: To evaluate the impact of fine-tuning strategy adopted in our DCNN model, we compare the DCNN model trained via random initialization and DCNN model trained with fine-tuning strategy. The parameter setting of the training is introduced in Section III-C.

As summarized in Table II, our proposed DCNN model achieves an accuracy of 96.53%, a precision of 96.98%, a recall of 97.00%, and a F1-score of 96.99% with the boosting of fine-tuning strategy. It is clear that our model achieves the best result among all the models. Compared with the counterpart trained via random initialization, there is a significant improvement of around 12% in accuracy ((r) vs. (s)).

The evaluation results based on confusion matrices and ROC curves are shown in Figs. 3 and 4, respectively. Other fine-tuned DCNN models also significantly outperform their corresponding fully trained DCNN models. Detailed results are also shown in Table II. Upon careful comparison, we observe that the increase in depth of deep models hardly exhibit any increase in accuracy when training via random initialization ((l) and (p), (n) and (r)). In addition, the accuracy is improved significantly when training via fine-tuning strategy.

5) Effectiveness of Data Augmentation: With the adoption of data augmentation, our FFSP dataset is substantially enlarged. To evaluate the efficacy of data augmentation, we compare the performance with and without this technique.

As previously mentioned, the amount of our FFSP dataset increases from 4849 to 14182 with data augmentation. We fine-tune our DCNN with 4849 images and 14182 images respectively, and then evaluate the test set. In case of training without augmentation, we crop and exclude the non-US region of each image and directly resize the rest to 224 × 224. The same process is adopted for the test image.

The evaluation results in terms of various metrics are shown in Fig. 5, where data augmentation and fine-tuning are denoted as Data Aug and FT, respectively. We can see that the model trained with the data augmentation consistently outperform the counterpart trained without data augmentation. There is a considerable improvement of ∼9% accuracy margin for the fine-tuned case (No Data Aug + FT vs. Data Aug + FT), and

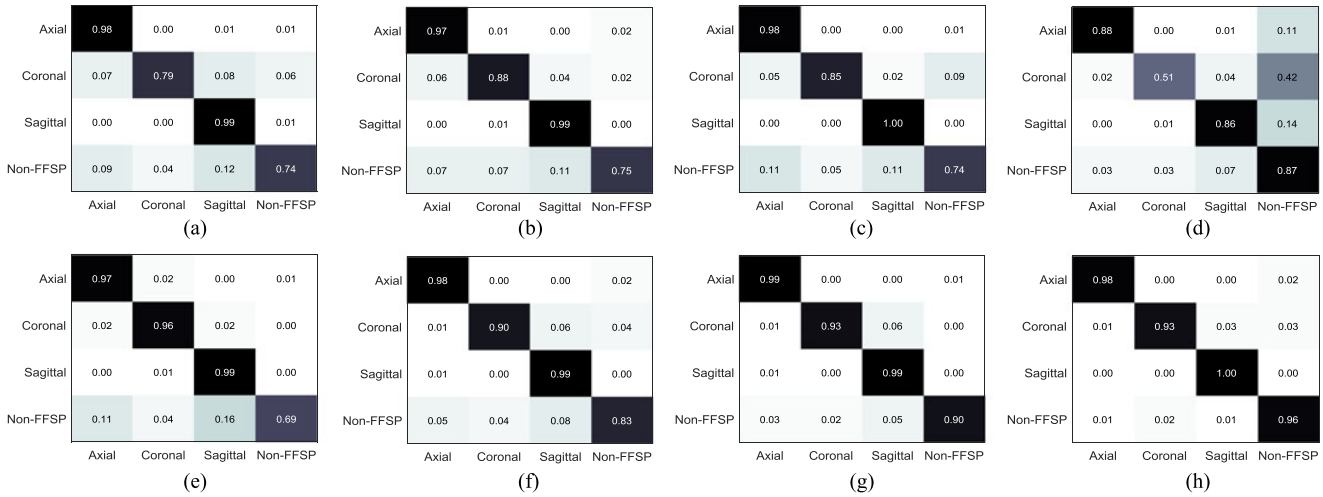

Fig. 3. Confusion matrices of DCNN models. (a) CNN-8-RI; (b) CNN-16-RI; (c) CNN-19-RI; (d) CNN-19-GAP-RI; (e) CNN-8-TR; (f) CNN-16-TR; (g) CNN-19-TR; (h) CNN-19-GAP-TR.

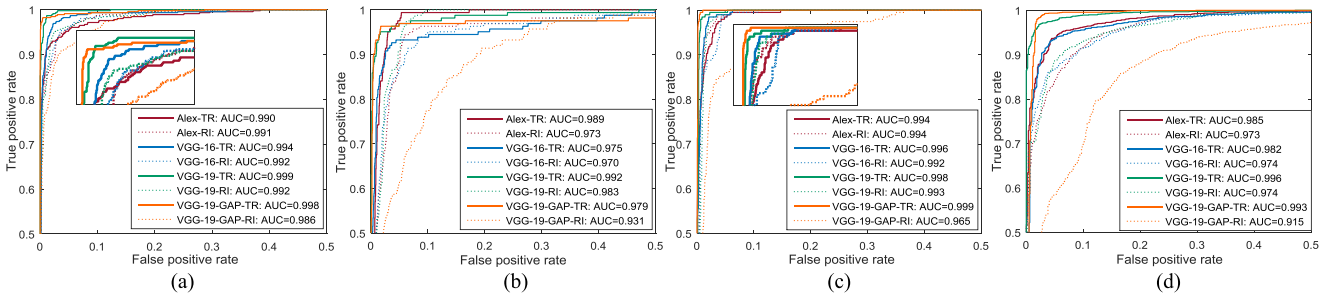

Fig. 4. ROC curves of DCNN models on different fetal facial image planes, (a) axial plane, (b) coronal plane, (c) sagittal plane, (d) non-FFSP.

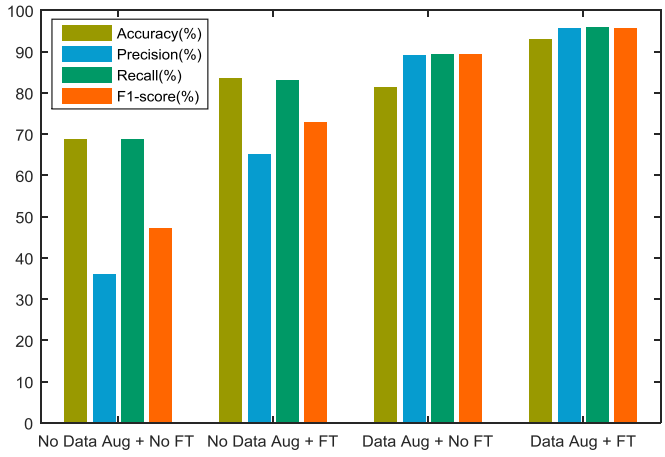

Fig. 5. Evaluation metrics: Accuracy, Precision, Recall, and F1-score of DCNN trained with and without data augmentation (Note that Aug is short for augmentation, and FT is short for fine-tuning).

improvement of ∼14% in accuracy in no fine-tuned case (No Data Aug + No FT vs. Data Aug + No FT). The primary explanation is that data augmentation can assist in learning the underlying characteristics and distribution of the FFSP images. Also, more samples within an image are available after augmentation which make the model more robust to spatial variance of the input. The significant performance boosting by this strategy indicates the effectiveness of data augmentation for successful FFSP recognition.

C. Visualization of DCNN Features

1) Visualization of Convolutional Feature Maps: Visualization: is an effective tool to get insights of the intermediate response of a DCNN model. For the purpose of understanding the model trained with FFSP images and gaining insights for the FFSP recognition with the DCNN models, we visualize the intermediate feature maps and filters learned by our DCNN.

The kernels learned by our proposed DCNN are presented in Fig. 6. For better illustration, the activation maps of a FFSP image obtained with these kernels at different layers are also illustrated in Fig. 7. For each Conv layer, we randomly extract a few feature maps from each layer for efficient and better presentation. We can see that these feature maps preserve the spatial information of FFSP image. The size of feature maps decreases and the features become more abstract when the input is forward passed into the higher layers. Among the activation maps of last convolutional layers, various semantic regions of the image are captured (fetal facial part and background), which assist in triggering the CNN for final prediction.

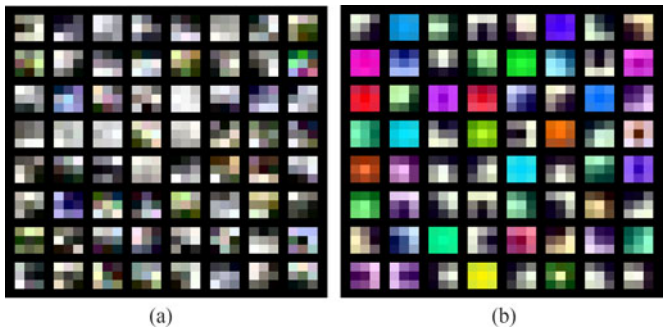

Fig. 6. Filters of first Conv layer learned by proposed DCNN. (a) filters trained via random initialization; (b) filters trained via fine-tuning.

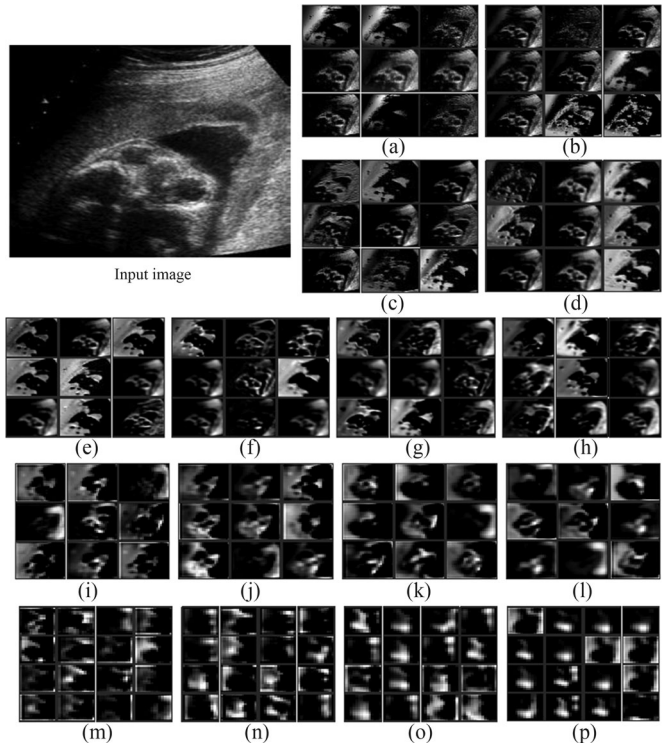

Fig. 7. Feature maps of Conv layers learned by proposed DCNN models (CNN-19-GAP-TR). (a)-(p) correspond to C1-C16 layers feature maps.

2) 2) Visualization of High-level Feature Representations: Since data visualization has a great impact on feature representation, we adopt the t-SNE method [64] to visualize our datasets and deep representations extracted from different networks. Specifically, for the original images, we transform the pixels of each image into a row vector, and concatenate all the samples' vector value along column dimension. We feed the pixel matrix with their labels into the t-SNE function. Similarly, output feature vectors of first FC layers of the DCNNs are extracted by inputting the training and test data, the resulting representations are adopted for t-SNE visualization.

The visualization results are illustrated in Fig. 8, where colors are used to represent different planes (modena for axial plane, blue for coronal plane, celadon for sagittal plane, and yellow for non-FFSP plane). The mixed distribution of training and test data in the original domain illustrates the high intra-class variations of FFSP and the low inter-class variations between FFSP and non-FFSP, which makes the FFSP recognition challenging. The learned representations of different DCNN models for FFSP images are clearly illustrated in the t-SNE visualization, and we have the same observations in the previous sections but in a more intuitive way. By and large, for these five models, the features learned using fine-tuning strategy are more separable compared that directly training from scratch. In the 2D feature space, we find that deeper model CNN-16 (CNN-16-GAP) and CNN-19 (CNN-19-GAP) are prone to suffer from overfitting problem. For instance, in Fig. 8(n–q), there is significant overlap in underlying distribution of the representations of testing data with different categories even with the adoption of fine-tuning (Fig. 8(s) and (u)). However, by using GAP, complexity (i.e. number of parameters) of models is reduced and deep representations are learned for separation (Fig. 8(t) vs. (s), Fig. 8(v) vs. (u)).

D. Comparison With the Traditional Classifier Model

1) Experimental Setting: To ensure consistency and fair comparison, the same image preprocessing described in Section III-B is adopted for hand-crafted feature based classification models. For each FFSP image, DSIFT descriptors are extracted with a stride of three pixels. For the BoVW model, the visual words are learned from the local descriptors generated from the training images via k-means clustering. The dictionary size of visual words is set to 1024. SPM is also adopted to divide each image into 2×2, and 3×1 regions, and sum pooling is used to compute the features of each spatial region, which leads to a 7168-dimensional representation of each FFSP image. The same setting is utilized for the VLAD model. Also, the 128-dimension DSIFT descriptor is reduced and de-correlated to a dimension of 100 by principle component analysis (PCA) whitening. The final VLAD dimension of the representation is 44800.

In FV model, we project and de-correlate the descriptors to a dimension of 80 using PCA. In addition, the number of Gaussian components to learn the generation of descriptors is fixed to 64 without spatial information (i.e., no SPM), hence, the dimensionality of the resulting representation is 10240. For the MFV model, we maintain the same encoding parameters and processing of descriptors, but we use the SPM technique to split the input into 1×1, 2×2, and 3×1 spatial subdivisions, which leads to an 81920-dimensional representation for each image. The dimension of the DSIFT descriptors are first reduced by PCA, and then encoded by FV and MFV via the corresponding models.

For the SVM classifier, different kernels are exploited, including linear kernel, Hellinger's kernel (short for Hell) and x^2 kernel (short for Chi2) [43], [67]. During the optimization of SVM, the stochastic dual coordinate ascent algorithm replaces the conventional stochastic gradient descent due to its efficiency and fast convergence rate. All the implementations of the models are based on the publicly available VLFeat toolbox [68].

2) Comparison Results: The BoVW, VLAD, FV, MFV, and DCNN models are compared using the same training and test

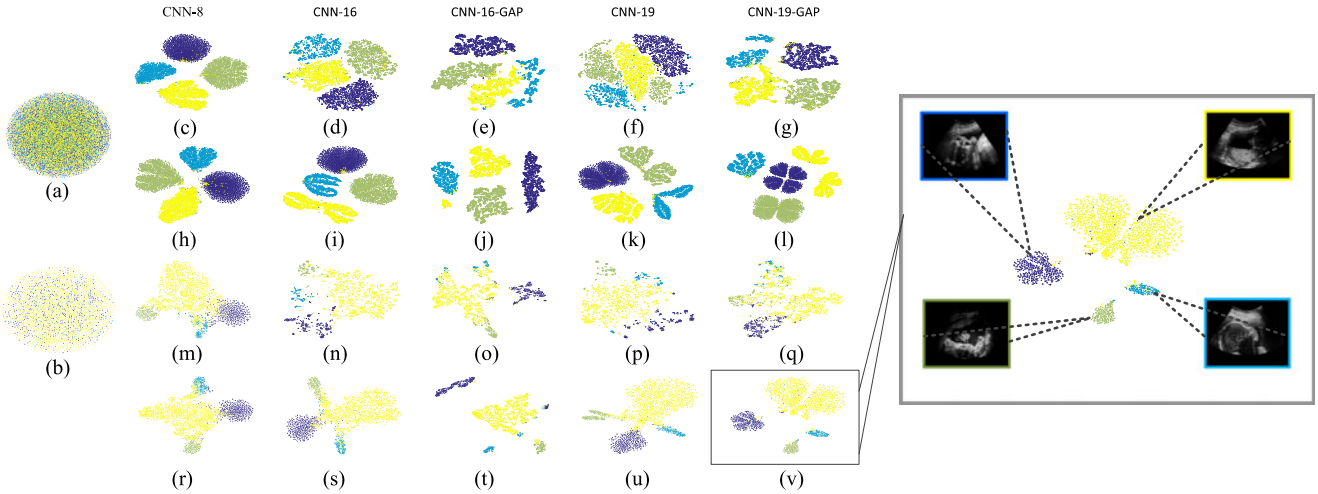

Fig. 8. t-SNE visualizations results to illustrate deep representations of our training data and test data learned by the DCNN models. Different FFSP planes are represented by four colors (e.g., Modena points represent axial plane, blue points represent coronal plane, celadon points represent sagittal plane, and yellow points represent others plane (non-FFSP)). (a) raw training data; (b) raw test data; (c)–(g) features of training data learned via random initialization; (h)–(l) features of training data learned via fine-tuning DCNNs; (m)–(q) features of test data learned by DCNNs trained via random initialization; (r)–(v) features of test data learned via fine-tuning DCNNs.

dataset. To be consistent with the experimental setting of DCNN models, each test FFSP image is processed and predicted in the same way. Table II provides a detailed comparison of the proposed DCNN with the state-of-art methods using hand-crafted features and kernel based SVM classifiers.

As illustrated in Table II, Chi2 kernel obtains the best results for the BoVW, VLAD, FV and MFV models than other kernels, which is consistent with our previous work [1], [6]. The MFV model with Chi2 kernel achieves the best performance (86.58% accuracy) among all the hand-crafted feature based classification models. Compared with the corresponding FV model, MFV has a noticeable improvement of 7% in terms of accuracy. The DCNN based methods consistently outperform the encoding based methods in terms of the performance of accuracy, precision, recall, and F1-score. The accuracy of CNN-8, CNN-16-RI, and CNN-19-RI are slightly higher than the MFV model with the best performance. The results of kernel learning and data augmentation in DCNN architectures are far more effective than the hand-crafted feature based classification models even with the improvement of SPM.

In this work, all the experiments are conducted on a PC with 2.70 GHz, 8 processors central processor unit (CPU) and 128 GB memory. For CNN model based classification methods, the training of networks consumes the most time in the whole pipeline. The training loss of networks via fine-tuning from the pre-trained model is stabilized within 20 epochs. For the models trained via random initialization, 40 epochs are sufficient to converge with the BN acceleration. The trade-off between number of training epoch and accuracy of the proposed DCNN model is shown in Fig. 9. Table V illustrates the data memory and time (training and test) requirements for different models up to 20 epochs. It is noteworthy that we only accelerate the training of CNN-8 with graphics processing unit (GPU) because our computer resource is still insufficient for training deep models.

For hand-crafted features based classification models, building visual vocabulary and encoding the descriptors are the most

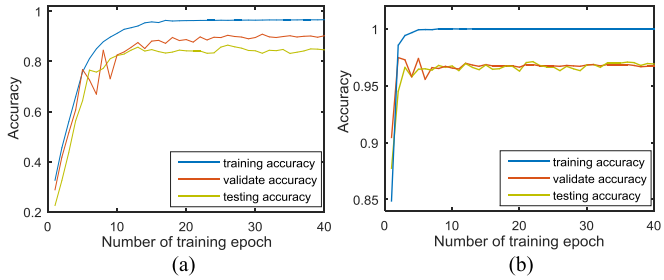

Fig. 9. Trade-off between number of training epoch and accuracy of the proposed DCNN model. (a) training from scratch; (b) training with fine-tuning strategy.

TABLE V
DATA MEMORY AND COMPUTATION TIME REQUIRED FOR THE MFV MODEL AND DCNN MODELS

Model	MFV	CNN-8	CNN-16	CNN-19
Traing time	6 h 15 m	1 h 49 m(2-GPUs)	67 h 20 m	87 h
Test time	0.01 s	0.76 s	4.29 s	4.39 s
Memory/batch	29M/64	502M/64	3G/32	4G/32

time-consuming parts due to the utilization of data augmentation technique. For example, building a vocabulary of 1024 using the Gaussian mixture model of 64 components takes around six hours in our experiment. The implementation time of the MFV model is listed in Table V.

V. DISCUSSION

We elaborated the proposed method for FFSP recognition, and extensive experiments are performed to investigate its effectiveness. In fact, one of the main challenges of leveraging DCNN in medical applications is the insufficiency of training data, which is easily put it in the jeopardy of overfitting or

Fig. 10. Examples of false positives recognized by our method. (a) Recognized as axial plane. (b) Recognized as coronal plane. (c) Recognized as sagittal plane.

performance degradation. In this case, we usually need to employ effective training policies to tackle this challenge and tap the potential of the limited training data. In this regard, training a DCNN with powerful discrimination capability for a medical image processing and analysis task is not always an easy task. In this work, we seamlessly integrate a set of modern techniques to overcome the bottleneck of DCNN model for FFSP recognition with limited US image data. While the data augmentation scheme is effective to address the issue of the paucity of training data, the GAP is adopted to reduce the complexity of the deep model, which circumvents the overfitting problem to some degree. Furthermore, the fine-tuning strategy based on natural image pre-trained network is adopted to boost the model optimization. Compared with conventional hand-crafted feature based algorithms (e.g. MFV model), the DCNN model possesses better generality in which case the learned knowledge can be transferred to other US standard planes recognition task. Also, the trained network can be further used to detect ROI of fetal anatomical structure, which is our planned future work as well.

Beyond the impressive results, there are still several limitations. First, the current study mainly focuses on normal clinical images, which are obtained from the healthy babies and mothers. In our future study, we shall validate the proposed framework on more pathological cases. In addition, 2418 images may be still insufficient to evaluate the method. More test images are needed. In our future work, larger and more representative dataset will be collected. Second, as shown in Fig. 10, although an impressive precision result (96.98%) has been achieved, the presented method still mis-classifies some FFSP. Examples of false positives recognized by our model are shown in Fig. 10. Most of these samples are quite similar to the FFSP, and the discriminative information only exists in the local small regions, which pose great challenge for recognition. In this regard, we are considering to integrate some low-level local cues in our framework to improve its discrimination capability for these hard mimics in the future. In our DCNN model, the non-FFSP is more sensitive to the preprocessing step compared with the FFSPs. The main explanation is that the sample rate of non-FFSP is low, where fewer sub-images within an image are extracted and accessed to the network. Therefore, we will explore the influence of the sample rate of preprocessing on the FFSP recognition in our future study.

VI. Conclusion

In this paper, we proposed an automatic FFSP recognition method based on a DCNN model with powerful feature representation and classification capability. We optimized the architecture of our model to enhance the recognition performance. Data augmentation and fine-tuning strategy were also adopted for performance boosting. We extensively analyzed the key elements of proposed DCNN and the effectiveness of deep representations. Both model architecture and fine-tuning strategy were investigated as well. The extensive experiments on our collected FFSP dataset demonstrated the superiority of our method over the traditional classification models for FFSP recognition. In addition, our experiments showed the effectiveness of data augmentation, especially when training data is insufficient. The impressive performance of the proposed method indicated the great potential of the DCNN for the clinic diagnosis.

References

[1] B. Lei *et al.*, "Automatic recognition of fetal facial standard plane in ultrasound image via fisher vector," *PLoS One*, vol. 10, no. 5, 2015, Art. no. e0121838.

[2] B. Lei *et al.*, "Discriminative learning for automatic staging of placental maturity via multi-layer fisher vector," *Sci. Rep.*, vol. 5, no. 2015, Art. no. 12818.

[3] B. Rahmatullah, A. Papageorghiou, and J. A. Noble, "Automated selection of standardized planes from ultrasound volume," in *Proc. Mach. Learn. Med. Imag.*, 2011, pp. 35–42.

[4] H. Chen *et al.*, "Automatic fetal ultrasound standard pane detection using knowledge transferred recurrent neural networks," in *Proc. Med. Imag. Comput. Comput. Assist. Interv.*, 2015, pp. 507–514.

[5] H. Chen *et al.*, "Standard plane localization in fetal ultrasound via domain transferred deep neural networks," *IEEE J. Biomed. Health. Inf.*, vol. 19, no. 5, pp. 1627–1636, 2015.

[6] B. Lei, L. Zhuo, S. Chen, S. Li, D. Ni, and T. Wang, "Automatic recognition of fetal standard plane in ultrasound image," in *Proc. IEEE 11th Int. Symp. Biomed. Imag.*, 2014, pp. 85–88.

[7] L. Zhang, S. Chen, C. T. Chin, T. Wang, and S. Li, "Intelligent scanning: Automated standard plane selection and biometric measurement of early gestational sac in routine ultrasound examination," *Med. Phys.*, vol. 39, no. 8, pp. 5015–5027, 2012.

[8] B. Rahmatullah and J. A. Noble, "Anatomical object detection in fetal ultrasound: Computer-expert agreements," in *Proc. Biomed. Inf. Technol.*, 2014, pp. 207–218.

[9] N. Dudley and E. Chapman, "The importance of quality management in fetal measurement," *Ultrasound Obstet. Gynecol.*, vol. 19, no. 2, pp. 190–196, 2002.

[10] M. Yaqub, B. Kelly, A. Papageorghiou, and J. A. Noble, "Guided random forests for identification of key fetal anatomy and image categorization in

ultrasound scans," in *Proc. Med. Imag. Comput. Comput. Assist. Interv.*, 2015, pp. 687–694.
[11] C. F. Baumgartner, K. Kamnitsas, J. Matthew, S. Smith, B. Kainz, and D. Rueckert, "Real-time standard scan plane detection and localisation in fetal ultrasound using fully convolutional neural networks," in *Proc. Med. Imag. Comput. Comput. Assist. Interv.*, 2016, pp. 203–211.
[12] K. Chatfield, V. S. Lempitsky, A. Vedaldi, and A. Zisserman, "The devil is in the details: An evaluation of recent feature encoding methods," in *Proc. Brit. Mach. Vis. Conf.*, 2011, pp. 76.1–76.12.
[13] J. Shi, Q. Jiang, R. Mao, M. Lu, and T. Wang, "FR-KECA: Fuzzy robust kernel entropy component analysis," *Neurocomputing*, vol. 149, Part C, pp. 1415–1423, 2015.
[14] J. Shi, J. Wu, Y. Li, Q. Zhang, and S. Ying, "Histopathological image classification with color pattern random binary hashing based PCANet and matrix-form classifier," *IEEE J. Biomed. Health Inf.*, doi: 10.1109/JBHI.2016.2602823.
[15] J. Shi, S. Zhou, X. Liu, Q. Zhang, M. Lu, and T. Wang, "Stacked deep polynomial network based representation learning for tumor classification with small ultrasound image dataset," *Neurocomputing*, vol. 194, pp. 87–94, 2016.
[16] D. Ni *et al.*, "Standard plane localization in ultrasound by radial component model and selective search," *Ultrasound Med. Biol.*, vol. 40, no. 11, pp. 2728–2742, 2014.
[17] B. Lei, E.-L. Tan, S. Chen, D. Ni, and T. Wang, "Saliency-driven image classification method based on histogram mining and image score," *Pattern Recognit*., vol. 48, no. 8, pp. 2567–2580, 2015.
[18] X. Zhu, X. Li, and S. Zhang, "Block-row sparse multiview multilabel learning for image classification," *IEEE Trans. Cybern.*, vol. 46, no. 2, pp. 450–461, Feb. 2016.
[19] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[20] C. Liu, J. Yuen, and A. Torralba, "SIFT Flow: Dense correspondence across scenes and its applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 978–994, May 2011.
[21] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 886–893.
[22] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proc. Eur. Conf. Comput. Vis. Workshop Statist. Learn. Comput. Vis.*, pp. 950–953, 2011.
[23] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2006, pp. 2169–2178.
[24] H. Jegou, F. Perronnin, M. Douze, J. Sanchez, P. Perez, and C. Schmid, "Aggregating local image descriptors into compact codes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 1704–1716.
[25] F. Perronnin, J. Sanchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *Proc. 11th Eur. Conf. Comput. Vis.*, 2010, pp. 143–156.
[26] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image classification with the fisher vector: Theory and practice," *Int. J. Comput. Vis.*, vol. 105, no. 3, pp. 222–245, 2013.
[27] J. Deng, W. Dong, R. Socher, K. L. L.-J. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.
[28] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
[29] M. Lin, Qi. Chen, and S. Yan, "Network in network," arXiv: 1312.4400, 2013.
[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[31] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
[32] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–14.
[33] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.
[34] J. Donahue *et al.*, "Decaf: A deep convolutional activation feature for generic visual recognition," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 647–655.
[35] Q. Dou *et al.*, "Automatic detection of cerebral microbleeds from MR images via 3D convolutional neural networks," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1182–1195, Feb. 2016.
[36] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 580–587.
[37] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.
[38] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Med. Imag. Comput. Comput. Assist. Interv.*, 2015, pp. 234–241.
[39] Q. Li, W. Cai, X. Wang, Y. Zhou, D. Feng, and M. Chen, "Medical image classification with convolutional neural network," in *Proc. 13th Int. Conf. Control Autom. Robot. Vis.*, 2014, pp. 844–848.
[40] X. Zhu, H. Suk, L. Wang, S.-W. Lee, and D. Shen, "A novel relational regularization feature selection method for joint regression and classification in AD diagnosis," *Med. Imag. Anal.*, vol. 38, pp. 205–214, 2017.
[41] X. Zhu, X. Li, S. Zhang, C. Ju, and X. Wu, "Robust joint graph sparse coding for unsupervised spectral feature selection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 6, pp. 1263–1275, Jun. 2017.
[42] X. Zhu, L. Zhang, and Z. Huang, "A sparse embedding and least variance encoding approach to hashing," *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 3737–3750, Sep. 2014.
[43] S. Maji, A. C. Berg, and J. Malik, "Classification using intersection kernel support vector machines is efficient," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.
[44] K. Grauman and T. Darrell, "The pyramid match kernel: Discriminative classification with sets of image features," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2005, pp. 1458–1465.
[45] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
[46] H. Greenspan, B. van Ginneken, and R. M. Summers, "Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1153–1159, May 2016.
[47] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2010, pp. 249–256.
[48] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1026–1034.
[49] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[50] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. C. Courville, and Y. Bengio, "Maxout networks," in *Proc. 30th Int. Conf. Mach. Learn.*, 2013, pp. 1319–1327.
[51] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," in *Proc. Adv. Neural Inform. Process. Syst.*, 2014, pp. 3320–3328.
[52] Y. Bengio, "Deep learning of representations for unsupervised and transfer learning," in *Proc. ICML Workshop Unsupervised Transfer Learn.*, 2012, pp. 1–20.
[53] M. Anthimopoulos, S. Christodoulidis, A. C. L. Ebner, and S. Mougiakakou, "Lung pattern classification for interstitial lung diseases using a deep convolutional neural network," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1207–1216, May 2016.
[54] Z. Gao, L. Wang, L. Zhou, and J. Zhang, "Hep-2 cell image classification with deep convolutional neural networks," *IEEE J. Biomed. Health. Inf.*, vol. 21, no. 2, pp. 416–428, Mar. 2017.
[55] Z. Yan *et al.*, "Multi-instance deep learning: Discover discriminative local anatomies for bodypart recognition," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1332–1343, May 2016.
[56] Y. Song, L. Zhang, S. Chen, D. Ni, B. Lei, and T. Wang, "Accurate segmentation of cervical cytoplasm and nuclei based on multiscale convolutional network and graph partitioning," *IEEE Trans. Biomed. Eng.*, vol. 62, no. 10, pp. 2421–2433, Oct. 2015.
[57] Z. Yu, D. Ni, S. Chen, S. Li, T. Wang, and B. Lei, "Fetal facial standard plane recognition via very deep convolutional networks," in *Proc IEEE 38th Annu. Int. Conf. Eng. Med. Biol. Soc*, 2016, pp. 627–630.
[58] H. Chen, Q. Dou, X. Wang, J. Qin, and P. A. Heng, "Mitosis detection in breast cancer histology images via deep cascaded networks," in *Proc. AAAI Conf. Artif. Intell.*, 2016, pp. 1160–1166.
[59] Y. Song, L. He, F. Zhou, S. Chen, D. Ni, B. Lei, and T. Wang, "Segmentation, splitting, and classification of overlapping bacteria in microscope images for automatic bacterial vaginosis diagnosis," doi: 10.1109/JBHI.2016.2594239.

[60] H. Chen, X. Qi, L. Yu, Q. Dou, J. Qin, and P. A. Heng, "DCAN: Deep contour-aware networks for object instance segmentation from histology images," *Med. Image Anal.,* vol. 36, no. pp. 135–146, 2017.

[61] H. Chen, Q. Dou, L. Yu, J. Qin, and P. A. Heng, "VoxResNet: Deep voxelwise residual networks for brain segmentation from 3D MR images," *Neuroimage*, doi: https://doi.org/10.1016/j.neuroimage.2017.04.041.

[62] H. Chen, L. Wu, Q. Dou, J. Qin, S. Li, J. Z. Cheng, D. Ni, and P. A. Heng, "Ultrasound standard plane detection using a composite neural network framework," *IEEE Trans. Cybern.*, vol. 47, no. 6, pp. 1576–1586, Jun. 2017.

[63] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *Proc. Brit. Mach. Vis. Conf.*, 2014, pp. 1–12.

[64] P. Hensman and D. Masko, "The impact of imbalanced training data for convolutional neural networks," Degree Project in Computer Science DD143X, 2015.

[65] N. Tajbakhsh *et al.*, "Convolutional neural networks for medical image analysis: Full training or fine tuning?," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1299–1312, May 2016.

[66] S. Shalev-Shwartz and Z. Tong, "Stochastic dual coordinate ascent methods for regularized loss minimization," *J. Mach. Learn. Res.*, vol. 14, no. 2, pp. 567–599, 2013.

[67] B. Scholkopf and A. J. Smola, *Learning With Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*.Cambridge, MA, USA: MIT Press, 2001.

[68] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," in *Proc. 18th Int. Conf. Multimedia*, 2010, pp. 1469–1472.