big data and cognitive computing

MDPI

Article

Deep Learning-Based Computer-Aided Classification of Amniotic Fluid Using Ultrasound Images from Saudi Arabia

Irfan Ullah Khan [1], Nida Aslam [1,*], Fatima M. Anis [1], Samiha Mirza [1], Alanoud AlOwayed [1], Reef M. Aljuaid [1], Razan M. Bakr [1] and Nourah Hasan Al Qahtani [2]

[1] Department of Computer Science, College of Computer Science and Information Technology, Imam Abdulrahman Bin Faisal University, P.O. Box 1982, Dammam 31441, Saudi Arabia
[2] Department of Obstetrics and Gynecology, College of Medicine, Imam Abdulrahman Bin Faisal University, P.O. Box 1982, Dammam 31441, Saudi Arabia
* Correspondence: Correspondence: naslam@iau.edu.sa

Abstract: Amniotic Fluid (AF) refers to a protective liquid surrounding the fetus inside the amniotic sac, serving multiple purposes, and hence is a key indicator of fetal health. Determining the AF levels at an early stage helps to ascertain the maturation of lungs and gastrointestinal development, etc. Low AF entails the risk of premature birth, perinatal mortality, and thereby admission to intensive care unit (ICU). Moreover, AF level is also a critical factor in determining early deliveries. Hence, AF detection is a vital measurement required during early ultrasound (US), and its automation is essential. The detection of AF is usually a time-consuming process as it is patient specific. Furthermore, its measurement and accuracy are prone to errors as it heavily depends on the sonographer's experience. However, automating this process by developing robust, precise, and effective methods for detection will be beneficial to the healthcare community. Therefore, in this paper, we utilized transfer learning models in order to classify the AF levels as normal or abnormal using the US images. The dataset used consisted of 166 US images of pregnant women, and initially the dataset was preprocessed before training the model. Five transfer learning models, namely, Xception, Densenet, InceptionResNet, MobileNet, and ResNet, were applied. The results showed that MobileNet achieved an overall accuracy of 0.94. Overall, the proposed study produces an effective result in successfully classifying the AF levels, thereby building automated, effective models reliant on transfer learning in order to aid sonographers in evaluating fetal health.

Keywords: amniotic fluid (AF); artificial intelligence; deep learning; transfer learning; ultrasound; classification

Citation: Khan, I.U.; Aslam, N.; Anis, F.M.; Mirza, S.; AlOwayed, A.; Aljuaid, R.M.; Bakr, R.M.; Qahtani, N.H.A. Deep Learning-Based Computer-Aided Classification of Amniotic Fluid Using Ultrasound Images from Saudi Arabia. *Big Data Cogn. Comput.* **2022**, *6*, 107. https://doi.org/10.3390/bdcc6040107

Academic Editors: Nadav Rappoport, Yuval Shahar and Hyojung Paik

Received: 8 August 2022
Accepted: 26 September 2022
Published: 3 October 2022

1. Introduction

Technological advances in the medical field have had a major impact on the treatment of many conditions that were once thought to be incurable or irreversible. Certainly, a fundamental step of any treatment is an adequate and correct diagnosis. Medical diagnostic and therapeutic procedures were primitive and could not take into account the numerous medical symptoms before the introduction of contemporary technologies. Fortunately, modern diagnostics have made great advances [1]. However, it has some limitations. Within the medical field, obstetrics and gynecology is a highly diversified discipline of medicine that includes surgery, prenatal care, gynecological care, oncology, and female preventive medicine [1,2]. More specifically, this paper deals with prenatal diagnostic procedures, more specifically with the determination of the amniotic fluid level (AF).

AF is a liquid seen in the amniotic sac that envelops the developing fetus in the uterus, serving a variety of roles, and is essential for embryonic development. Although, complications might arise if the volume of AF within the uterus is either too little or too high. Abnormal AF volumes can lead to significant complications including oligohydramnios (inadequate AF level) and polyhydramnios (excess AF level). Oligohydramnios is

associated with a higher incidence of stillbirth or miscarriage [3]. Because the AF serves an important role in the respiratory system growth during the second trimester, it can sometimes result in anomalies including severe lung defects [3]. It can also cause umbilical cord constriction and other complications. Polyhydramnios is also responsible for other problems throughout pregnancy and childbirth. This condition is known to induce preterm labor spasms, premature birth, trouble breathing, limited air flow to the fetus due to the umbilical cord becoming caught underneath the fetus, and other complications [3].

The detection of AF is usually a time-consuming process, and it is patient specific. Moreover, its measurement and accuracy are subject to human errors, as it heavily depends on the sonographer's experience [4]. The four-quadrant AF index (AFI) or the single deep vertical pocket (SDP) approach are commonly used to determine the AF volume. Sonographers locate an appropriate AF region and afterwards analyze the depth of the AF region by finding a specific point to measure the AF levels. Traditional AFI assessment is primarily dependent on the sonographer's abilities and knowledge, regardless of the fact that AFI and SDP techniques are proven to be reproducible and semi-quantitative [5]. Even after extended training, sonographers struggle to precisely quantify the AF level in the fetus [4]. As a result, the entire examination is a time-consuming process that can contribute to erroneous results. However, automating this process by developing robust, precise, and effective methods for detection will be beneficial to the healthcare community.

Nevertheless, though some studies have already been conducted in relation to the AF classification, limited research has been dedicated to classifying AF in US images using transfer learning [1]. Hence, this study will significantly contribute to automating the detection process, which will benefit physicians as the automated detection helps them to assess fetal health and development as well as perinatal prognosis. Furthermore, accurate detection of AF levels in a quick and efficient manner is very critical. Therefore, utilizing transfer learning to detect AF levels with accurate results and by processing US data will equip gynecologists with a great tool that will help in improving the accuracy and efficiency of their diagnosis and thus will help to monitor fetuses' health. This paper utilized several transfer learning models such as Xception, DenseNet, InceptionResNet, MobileNet, and ResNet to classify the AF levels as normal or abnormal. The dataset used consisted of 166 US images obtained from King Fahd Hospital of the University (KFHU) and Elite Clinic in Dammam, Saudi Arabia. The paper makes the following contributions:

1. Applying transfer learning models to classify the AF levels as abnormal or normal, using ultrasound images and achieving high predictive performance;
2. Develops a predictive model using the real dataset from the hospital in the Kingdom of Saudi Arabia (KSA). As per the authors knowledge, the proposed study is the first study related to the AF classification using the KSA hospital dataset;
3. Initially, the preprocessing was applied using various techniques such as cropping, enhancing, and augmenting, etc., to build more robust predictive models;
4. Performs cross-validation and provides a comprehensive discussion on the obtained results on classifying the AF levels using the US images.

The paper is structured as follows: Section 2 investigates the related studies in the field of utilizing AI in AF detection. Section 3 presents the material and methods used in classification of AF, which include processing of the dataset, classification models applied, and the metrics used to evaluate the models. Section 4 presents the experimental setup, while Section 5 contains the results and discussion. Finally, Section 6 gives a conclusion to summarize the overall study as well as the intended future work.

2. Related Studies

A plethora of studies have been carried out that have focused on using AI methods to detect AF. This section examines the theories and applicable concepts on the subject in the present literature, as well as their findings in order to identify the gap in literature. However, limited studies have focused on the classification on AF using US images/videos. Likewise, Ayu et al. [6] used machine learning (ML) algorithms, namely, rule-based SDP and the

Random Forest (RF) algorithm, in order to classify the AF into 6 groups: oligohydramnios clear and echogenic, polyhydramnios clear and echogenic, and normal clear and echogenic. The dataset comprised 95 US images acquired from a local hospital. Before SDP feature extraction, the images were cropped and transformed from red–green–blue (RGB) to greyscale during preprocessing phase. After training the classifiers, the accuracy of the model was 0.9052, and was higher than that of previous research. Furthermore, Ayu and Hartati [7] conducted another study utilizing a pixel-based classification method to distinguish AF areas on US images with noise, distortions, low contrast, and fuzzy margins. The images were classified into four classes including the AF, fetal body, placenta, and uterus. The accuracy obtained using RF was 0.995, using 50 test US images. Finally, Amuthadevi and Subarnan [8] deployed fuzzy techniques to measure the AF index and the geometric properties of AF at different phases of gestation. The anomalies in head circumference and infant weight, etc., were forecasted using the fuzzy techniques. The AFI was classified as oligohydramnios, borderline, normal, or polyhydramnios. The classification accuracy obtained was 0.925.

On the other hand, various research has employed AI algorithms to segment AF from US images. Following this ideology, a DL model called AF-net was used to segment the AF pockets, developed by Cho et al. [4]. The AF-net is a version of U-net, which combines several ideas: dilated convolution, multiscale side-input and side-output layer using 435 US images dataset, and 5-folds cross-validation was employed. For AF segmentation, the suggested model achieved a precision of 0.898 ± 0.111 and a dice similarity coefficient (DSC) of 0.877 ± 0.086. Similarly, Sun et al. [5] attempted to estimate the AF volume from US images by segmenting the AF using a dual path DL network, which was composed of AF-net and an auxiliary network. The dataset contains 2380 US images, which were preprocessed using the following methods: resizing, trimming, augmenting, normalizing, and applying 5-fold cross-validation. The model achieved a DSC of 0.8599. Furthermore, Li et al. [9] also deployed DL in order to segment the AF in the US images. The dataset constituted US videos of 4 patients, where each video length was 20 s. Key frame extraction images were selected; 900 training images and 400 testing images were collected. The model achieved an accuracy of 0.93, by applying 3 inner layers to the kernel in the applied encoder–decoder network.

In another study, Ayu et al. [10] employed 50 fetal B-Mode US images to carry out AF segmentation. To perform the segmentation, a pixel classification centered on the RF was utilized. For comparison, the images were first brought into two window-size proportions (3×3 and 5×5). After that, multiple points pertaining to 3 classes—AF, fetal body, and uterus—were labeled by a radiologist expert. The results demonstrated that images with a window size of 5×5 reached an accuracy of 0.8586, and images with a window size of 3×3 scored an accuracy of 0.8145. Furthermore, Ayu et. al. [11] also conducted another study where they performed segmentation using pixel classification by applying several classifiers such as decision tree (DT), RF, naive bayes (NB), support vector machine (SVM), and K-nearest neighbor (KNN). The dataset used comprised 55 US images, and the RF classifier gave the best results, attaining a DSC of 0.876 and pixel accuracy of 0.857.

Additionally, Looney et al. [11] attempted to segment the AF, placenta, and fetus by building a multiclass CNN model. In order to segment the placenta, 2093 images were used, and fully CNN (FCNN) was deployed. The highest DCS of 0.85 was attained after 17,000 training steps for placenta segmentation. For multiclass segmentation, 300 images were employed to combine a two-pathway hybrid model, and a DSC of 0.84 was obtained. Finally, Anquez et al. [12] investigated the utero-fetal unit (UFU) segmentation by employing 19 3D US images using the fuzzy technique. All these images belonged to the first trimester of the fetal stage. Automating the fetal tissue and AF extraction was their primary goal. An average accuracy of 0.89 was obtained in the study.

Most of the aforementioned studies focused on segmenting the AF from the US images. Some of these studies achieved high DSC. For instance, Ayu et al. [13] obtained a DSC of 0.876. However, segmenting the AF alone does not help in estimating if the AF level in

the fetus is in the normal range. We also need to model techniques that can successfully classify by using the US images, as a normal AF level or not. There was, however, one study that focused on calculating the AF index from the segmented AF. This study was conducted by Cho et al. [4] by using US videos, and they achieved an overall precision of 0.898. Furthermore, among the studies that pursued classification, the highest accuracy of 0.995 was obtained by Ayu and Hartati [7]. However, in this study, they did not focus on classifying the AF levels in the US images. Rather, they focused on classifying the US images into four classes: AF, fetal body, placenta, and uterus. Therefore, in the current study we focused on classifying the US images as having normal AF or abnormal AF. To accomplish this, the US images dataset was collected from a local hospital, and several transfer learning models were deployed to develop a model that could make accurate predictions. Hence, the proposed methods have successfully classified the AF images and thereby aid the sonographers/physicians in determining fetal health.

3. Materials and Methods

This section provides a breakdown of the proposed methodology along with a comprehensive analysis of data preprocessing methods, data-partitioning techniques, classification models applied, and evaluation metrics used. Figure 1 summarizes the methodology deployed in this study. The dataset was first passed through the preprocessing steps, and then for training, the model and the preprocessed images were split into a training set and test set, with the training set further split into training and validation sets. The transfer learning model was trained and validated by the training and validation sets. After training, the model was evaluated with a test set and the results were collected. The section below contains the details of proposed methodology steps.

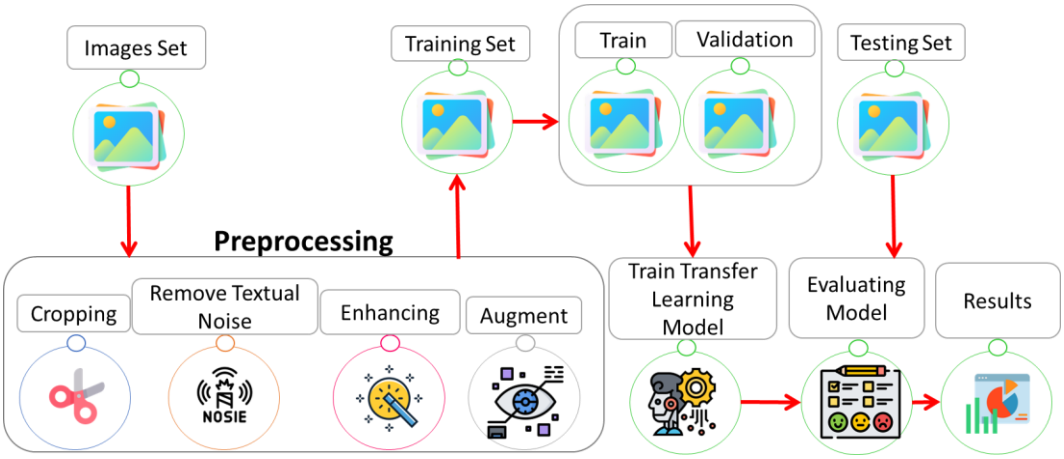


Figure 1. Proposed methodology.

3.1. Dataset Description

The US images are classified into two classes based on the AF levels: abnormal AF and normal AF.

Abnormal AF: The abnormal AF corresponds to a patient suffering from oligohydramnios or polyhydramnios conditions. Our focal point of the research is recognizing the US images having these abnormal AF levels.

Normal AF: Normal AF pertains to the patients not suffering from the aforementioned conditions. The normal AF US images of patients are required to enable the classification models to distinguish the abnormal from the normal ones.

Hence, as an initial step to develop the models to classify these AF levels in images, we acquired the US images from King Fahd Hospital of the University (KFHU) and Elite Clinic in Dammam, KSA. The US device that was used to collect the images was the GE

Voluson P6, and we collected a total of 166 US images, among which 100 cases belonged to normal AF levels and 66 cases belonged to abnormal AF levels.

3.2. Preprocessing

The preprocessing involved several steps to prepare the data to be ready for further processing. The following steps were performed during preprocessing

3.2.1. Cropping and Enhancement

The first step after acquiring the images was cropping them in order to remove the textual information in the US images, particularly the top right corner of each image, which contains the patient's name and gestational age, etc., as well as the scale present on the left bar of each image. This information is removed to preserve the patient's privacy. The images were cropped using Python. This step removed the textual noise present on the images.

The next step after removing the textual noise was enhancing the US images. Image enhancement is typically carried out in order to improve the quality of the images. The PIL library in Python contains an image enhancement algorithm. Hence, the images were enhanced using the ImageEnhance method from the PIL library.

3.2.2. Augmentation

Augmentation in image processing is carried out in order to expand the size of the dataset by generating new images from the original image dataset. The preprocessing library in TensorFlow contains the ImageDataGenerator class, which can perform the process of augmentation in real time while the model training is performed. This function will perform random rotation, height shift, zooming, and rescaling, etc., thereby making the model more robust by ensuring that it receives new variations of the images at each epoch. An essential thing to be noted is that the ImageDataGenerator only returns the new (transformed) images instead of adding to the original set of images. The reason behind utilizing this in ours is that if the models were to see the original images multiple times, it would end up suffering from the problem of overfitting. Furthermore, the ImageDataGenerator class also helps to save memory by loading all the images at once instead of loading them in batches. Therefore, the process of augmentation was carried out using the ImageDataGenerator class.

3.3. Classification Models

Transfer learning is the process of reusing an already trained model for another task. In this paper we applied several transfer learning models. Below are the five main transfer learning models that we applied, which yielded good results, as will be shown in the results section.

3.3.1. Xception

A unique deep convolution method reliant on depthwise separable convolutions proved to excel the regular Inception V3 regarding the ImageNet dataset, alongside superior performance on a complex classifier dataset involving 350 million images with 17,000 classes [14]. The term Xception, which means "extreme inception", thus stems from the fact that the CNNs' feature maps' cross-channels and spatial correlations mapping can be completely detached. Due to the depthwise separable convolution layers accompanying the residual connections, the architecture is easily adaptable using high-level libraries such as Keras and TensorFlow-Slim.

3.3.2. Resnet50V2

The winner of the ILSVRC 2015 classification task, residual network (ResNet), was introduced by He K. et al. [15] to enhance the efficiency of CNN and accelerate the computational time. ResNet can contain thousands of layers without negatively affecting the

performance, making it ideal for image recognition, object localization and detection, and even establishing acceptable accuracy for non-vision tasks. Furthermore, the model was proposed to surpass the problem of the vanishing/exploding gradient in DNN by applying shortcut connections—also known as residual blocks—for identity mapping, which does not increase the parameter number nor the computational time. The goal was to stabilize the error rate for the higher neural network layers and diminish the impacts on the lower layers. For example, authors were reformulating the original mapping into M(a): =G(a) + a, where M(a) is the resulted mapping, and a is the input for these layers. This reformulation was considered to add a prerequisite for the degradation problem, which states that an increase in the depth of the network increases the error rate on both training and testing data. ResNet50V2 is one of the ResNet latest versions, with 50 layers deep and batch normalization for each weight layer. The 50th layered model has the same architecture as ResNet34 but adds an extra bottleneck block instead of 2 layers, resulting in achieving higher accuracy than the ResNet34.

3.3.3. DenseNet121

Dense convolutional network (DenseNet) is an architecture that makes deep learning networks much more efficient to train in comparison to the standard convolutional neural network (CNN). In particular, in standard CNN each convolutional layer receives the input from the previous layer. Contrastingly, in DenseNet each layer is connected to all other layers in the network to maximum information flow between the layers. Each layer obtains inputs from all the previous layers and passes on its own feature maps to all the layers after that layer, in order to preserve the feed-forward nature. DenseNet combines the features by concatenating them where the “ith” layer has “i" inputs and consists of feature maps of all its previous convolutional layers. For “L” layers, there are L(L + 1)/2 direct connections rather than just “I” connections as in standard CNN architectures. Thus, it requires fewer parameters as there is no need to learn unimportant feature maps, and results in more compact models and achieves high performances and better results across competitive datasets [16].

3.3.4. MobileNet

MobileNet is a type of CNN architecture that was mainly designed to be utilized for computer vision in mobile applications. It has been open sourced by Google and can be used for training the classifiers faster. The MobileNet uses the mechanism of depthwise separable convolutions, which entails splitting the computation into two main steps: depthwise convolution and pointwise convolution. The depthwise convolution first applies the same filter to each input channel [17]. Then, 1×1 convolution is applied at the pointwise convolution step in order to combine the outputs from the depthwise convolution step. Therefore, the depthwise separable convolution splits the architecture into two layers, i.e., one for filtering and the other for combining. This separation of layers dramatically reduces the model size and computation. Traditionally, the CNN architecture consists of single 3×3 convolution layers, which is followed by batch normalization and ReLu activation. However, MobileNet splits the convolutions into 3×3 depthwise convolution, which is followed by 1×1 pointwise convolution.

3.3.5. InceptionResnetV2

This is a CNN model pretrained on around a million images from the ImageNet database. Images can be classified into numerous categories, including keyboard, mouse, and pencil, using its 164 layers deep network. With a 299-by-299 image input size, it has a highly exclusive features’ representation capability. The two underlying constituents of the network are the inception structure and residual connection. Degradation and time elongation issues are prevented by the implementation of residual connections [18].

3.4. Evaluation Metrics

To ensure the reliability and to demonstrate the proposed model's performance, several evaluation parameters were utilized. Accuracy, balanced accuracy, precision, recall, F1 score, and AUC–ROC are some of the commonly used performance metrics to measure performance for similar models. Accuracy is the ratio of correctly predicted observation to the total observations. To calculate an accuracy, first we need to calculate the true positive, true negative, false positive, and false negative, and then utilize the following equation:

$$Accuracy = \frac{(TP + TN)}{(TP + FP + TN + FN)}$$

where TP is true positive, TN is true negative, FP is false positive, and FN is false negative.

Precision can be calculated by finding the issue of correctly predicted positive observation to the total predicted positive observation, as shown in the following equation:

$$Precision = \frac{TP}{(TP + FP)}$$

Moreover, to show how many actual positive cases we would be able to predict in our model, recall will be calculated, which is the ratio of correctly predicted positive observation to all observations and actual class using the following equation:

$$Recall = \frac{TP}{(TP + FN)}$$

By calculating precision and recall, F-score, another useful tool, is calculated to evaluate the model performance represented by the weighted average of precision and recall using the following equation:

$$F\text{-}score = \frac{2 \times \text{Recall } \times \text{Precision}}{(Recall + Precision)}$$

Additionally, balanced accuracy is calculated, which is balance accuracy, and is used in binary and multiclass classification problems to account for the imbalance dataset. It is calculated by the average of recall of each class.

The area under the curve (AUC), which measures the quality of the model predictions and has a scale from zero to one, where the best value is 1 and the worst value is 0, is also calculated. Additionally, AUC–ROC is typically used to measure performance for classification problems. ROC (receiver operating characteristics) is a probability curve, and AUC represents the degree of separability, which explains the model capability of distinguishing between classes.

4. Experimental Setup

The study was implemented using Python programming language (ver. 3.7.12). Several libraries were used for developing the model such as PIL (ver. 8.4.0), TensorFlow (ver. 2), Numpy (ver. 1.22.0), and Matplotlib (ver. 2.2). The experiments were performed on the Google Colab with the GPU setting. The dataset was divided using five-fold cross-validation for training and testing. The training dataset was further divided into training and validation. The model optimization was performed using the Adam optimizer. Five transfer learning models were trained such as MobileNet, InceptionResnetV2, DenseNet121, ResNet50V2, and Xception. The results of each model will be discussed below.

5. Results and Discussion

To assess the performance of the applied models, the results of the test data were compared with the results provided by the gynecologist from a local hospital. Since five-fold cross-validation was applied as it yielded better results when compared with holdout (train–test split), we recorded the performance metric values obtained at each

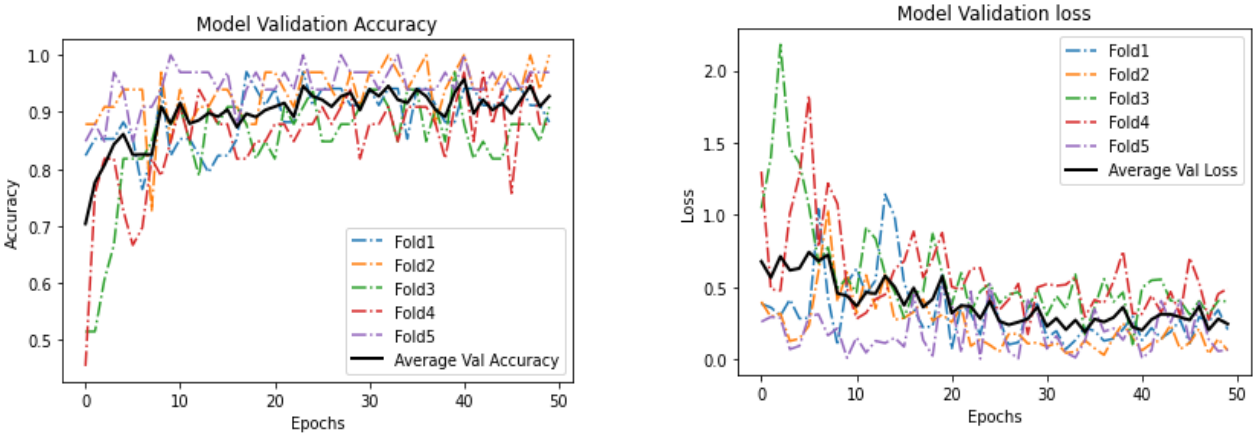
fold. The average of the performance values was taken for each metric for each fold and included the results for each model. These results are included in the respective subsections. Furthermore, the loss and accuracy curve at each epoch was used to monitor and assess the model's performance.

5.1. MobileNet

The results obtained by the MobileNet model have been summarized in Table 1 below across each fold. Among all the folds, the highest accuracy was achieved by the first fold. However, most of the performance metrics gave similar values. The mean performance using all the metrics have been included in the last column. It is observed that the model achieved an overall average accuracy of 0.945, which means the model can correctly predict 94.5% of the test cases. Other metrics are also indicated in the last column. Figure 2 (a and b) shows two plots, i.e., validation data accuracy at each epoch 2(a) and validation loss at each epoch 2(b). From these curves we found that the performance varied at each epoch but remained in the same ranges as the model was learning.

Table 1. K-folds performance results using MobileNet.

Measures	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Average
Accuracy	0.9706	0.9697	0.9091	0.9091	0.9697	0.9456
Bal-Accuracy	0.9722	0.9615	0.9115	0.9111	0.9444	0.9402
Precision	1	0.9524	0.9474	0.9412	0.96	0.9602
Recall	0.9444	1	0.9	0.8889	1	0.9467
F1 score	0.9714	0.9756	0.9231	0.9143	0.9796	0.9528
Auc roc	0.9722	0.9615	0.9115	0.9111	0.9444	0.9402



(a) (b)

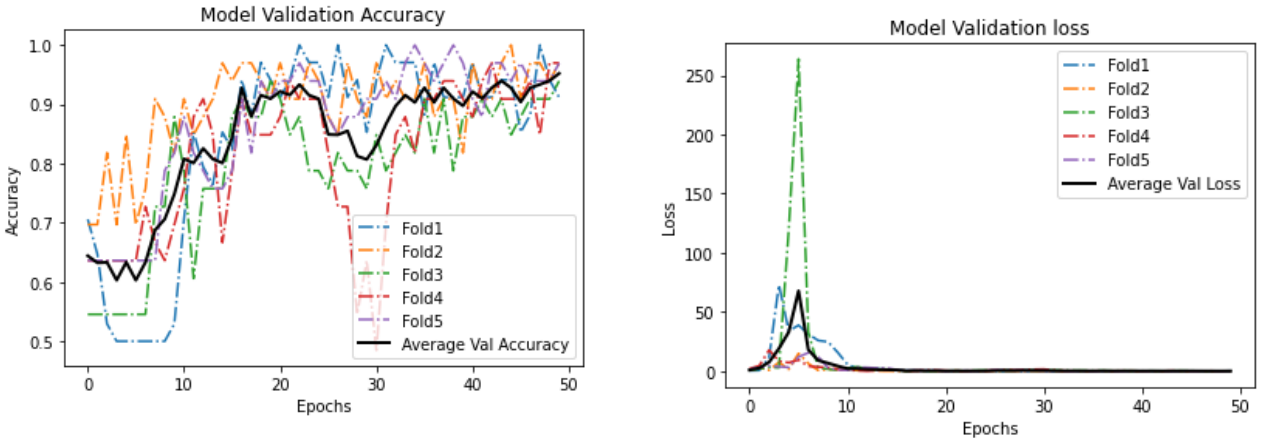
Figure 2. Validation accuracy and loss curve for MobileNet. (**a**) Model Validation Accuracy; (**b**) Model Validation Loss.

5.2. DenseNet121

Table 2 below gives the performance results obtained by DenseNet121. In this model, the final fold (fold 5) showed the highest performance. The mean values are added in the last column, and as we can see, the accuracy achieved was 0.927, which is less than that achieved by MobileNet, but the metrics values for both models are comparatively close. Figure 3 (a & b) shows the validation accuracy (3a) and loss curves (3b) for DensNet121, plotted at each epoch. It was observed that the accuracy gradually kept increasing at every epoch, which indicates that the model was learning well during training. Moreover, the loss increased till the 5th epoch, after which it decreased and stayed in the same range at the end of the epochs.

Table 2. K-folds performance results using DenseNet121.

Measures	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Average
Accuracy	0.9118	0.9091	0.9091	0.9394	0.9697	0.9278
Bal-Accuracy	0.9118	0.9065	0.9111	0.9524	0.9762	0.9316
Precision	0.85	0.9545	0.9412	1	1	0.9491
Recall	1	0.913	0.8889	0.9048	0.9524	0.9318
F1 score	0.9189	0.9333	0.9143	0.95	0.9756	0.9384
Auc roc	0.9118	0.9065	0.9111	0.9524	0.9762	0.9316



(a) (b)

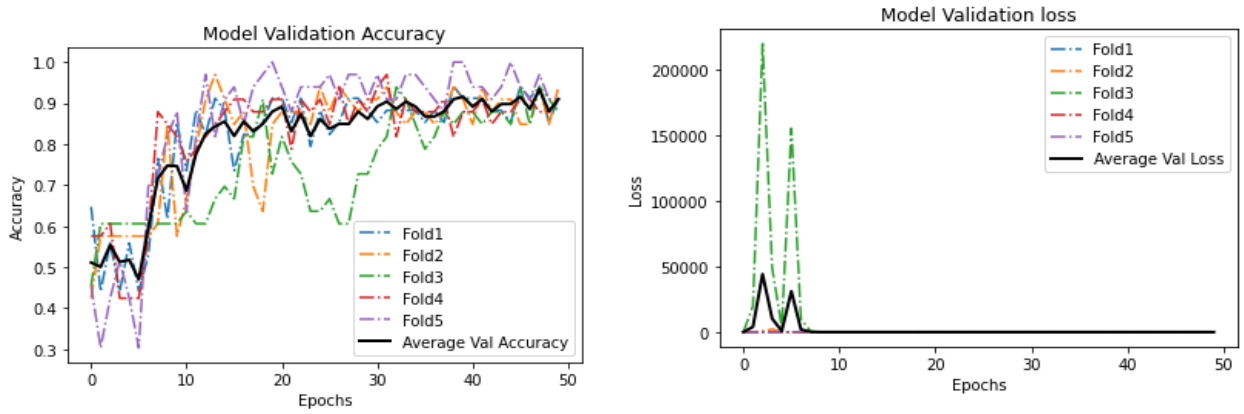
Figure 3. Validation accuracy and loss curve for DenseNet121. (**a**) Model Validation Accuracy; (**b**) Model Validation Loss.

5.3. InceptionResnetV2

Table 3 illustrates the performance results obtained by InceptionResnetV2. In this model, the final fold (Fold 5) showed the highest performance. The mean values are added in the last column, and we see that the accuracy achieved was 0.892, which was less than that achieved by both MobileNet and DenseNet121. The validation accuracy and loss curves for InceptionResNetV2 are illustrated in Figure 4 (a & b), plotted at each epoch. From the figures, it has been found that the accuracy gradually kept increasing at a fluctuating rate at every epoch, indicating that this model too was learning well during training. Moreover, the average loss fluctuated up and downwards between the 5th and 10th epoch, after which it decreased and stayed the same until the end of the run.

Table 3. K-folds performance results using InceptionResnetV2.

Measures	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Average
Accuracy	0.8824	0.9091	0.8485	0.8788	0.9394	0.8916
Bal-Accuracy	0.8807	0.9117	0.8346	0.8665	0.9283	0.8844
Precision	0.8947	0.9444	0.8571	0.8571	0.9565	0.902
Recall	0.8947	0.8947	0.9	0.9474	0.9565	0.9187
F1 score	0.8947	0.9189	0.878	0.9	0.9565	0.9096
Auc roc	0.8807	0.9117	0.8346	0.8665	0.9283	0.8844



(a) (b)

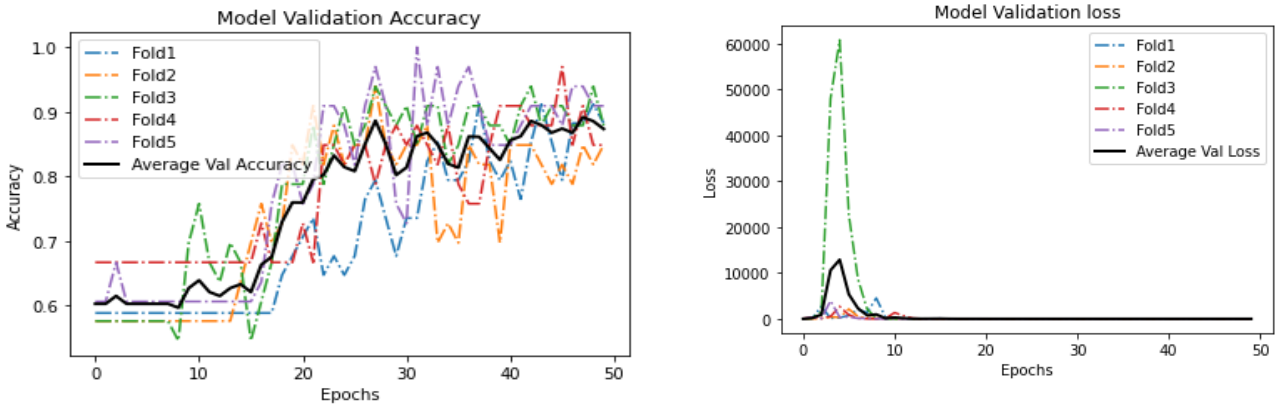
Figure 4. Validation accuracy and loss curve for InceptionResnetV2. (**a**) Model Validation Accuracy; (**b**) Model Validation Loss.

5.4. ResNet50V2

The performance results obtained by ResNet50V2 are depicted in Table 4. In this model, the final fold (Fold 5) showed the highest performance, and it reached 100% for all the metrics. However, the difference between the metrics values was high at each fold. The model was not consistent in learning. The mean values are added in the last column, and the accuracy achieved was 0.862, which was less than the accuracy achieved by the previous three models discussed. The validation accuracy and loss curves for ResNet50V2 are illustrated in Figure 5 (a & b). In these curves we see that the accuracy gradually kept increasing from 15th epoch onwards. Moreover, the average loss decreased immediately at the 5th epoch and stayed almost the same until the end of the run.

Table 4. K-folds performance results using ResNet50V2.

Measures	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Average
Accuracy	0.8529	0.7879	0.8182	0.8485	1	0.8615
Bal-Accuracy	0.8536	0.7688	0.8233	0.8182	1	0.8528
Precision	0.8947	0.7727	0.8824	0.8696	1	0.8839
Recall	0.85	0.8947	0.7895	0.9091	1	0.8887
F1 score	0.8718	0.8293	0.8333	0.8889	1	0.8847
Auc roc	0.8536	0.7688	0.8233	0.8182	1	0.8528



(a) (b)

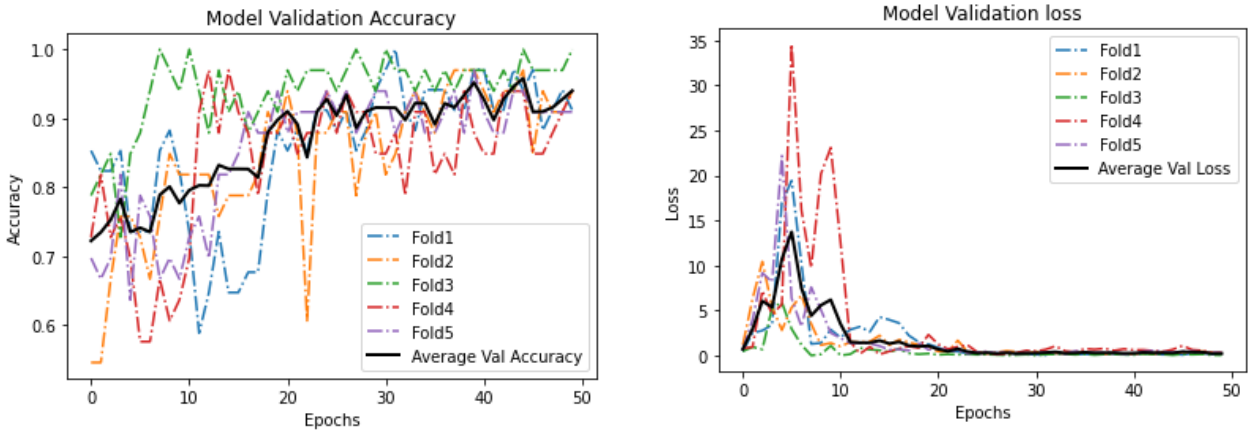
Figure 5. Validation accuracy and loss curve for ResNet50V2. (**a**) Model Validation Accuracy; (**b**) Model Validation Loss.

5.5. Xception

Finally, Table 5 summarizes the performance results obtained by Xception. In this model, the first fold (Fold 1) showed the highest performance, and the value difference was inconsistent between each fold, meaning sometimes it decreased and sometimes there was an increase. The mean values are added in the last column, and we see that the accuracy achieved was 0.92, which was similar to that achieved by both MobileNet and DenseNet121. The validation accuracy and loss curves for Xception are illustrated in Figure 6 (a & b). Here, we see that the accuracy gradually kept increasing at a fluctuating rate at every epoch, which indicates that the model was learning well. Furthermore, the average loss moved up and down until the 10th epoch, after which it decreased and stayed in the same range until the end of the run.

Table 5. K-folds performance results using Xception.

Measures	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Average
Accuracy	0.9706	0.9091	0.9697	0.8485	0.9091	0.9214
Bal-Accuracy	0.98	0.9056	0.9773	0.859	0.875	0.9194
Precision	1	0.8947	1	0.7647	0.875	0.9069
Recall	0.96	0.9444	0.9545	0.9286	1	0.9575
F1 score	0.9796	0.9189	0.9767	0.8387	0.9333	0.9295
Auc roc	0.98	0.9056	0.9773	0.859	0.875	0.9194



(a) (b)

Figure 6. Validation accuracy and loss curve for Xception. (**a**) Model Validation Accuracy; (**b**) Model Validation Loss.

Implementing AI techniques in fetal health has proven to exhibit significant results for the diagnosis of major fetus health issues such as premature birth and perinatal mortality [19]. The first process of preprocessing involved the cropping of images for the removal of patient information in the form of text, followed by other noise removal from various locations within the image for clarification. Finally, augmentation was implemented for the expansion of the dataset size using the ImageDataGenerator class available on the TensorFlow library.

With five-fold cross-validation providing better results in comparison to holdout, its performance metric values were recorded at each fold and its average calculated. The MobileNet model's first fold produced the highest accuracy and best overall performance, with accuracy, precision, recall, and F1-scores of 0.94, 0.96, 0.94, and 0.95, respectively.

6. Conclusions

The classification of the AF level is a crucial method to diagnose fetus health and development. Early diagnosis of normal AF levels can help in identifying major fetus

health issues such as premature birth and perinatal mortality. Moreover, detection of AF is a lengthy process that requires accurate measurements of patients' US data. This process requires experience to accurately measure the levels and avoid any errors, in order for gynecologists to successfully assess the cases and ensure fetus health. Furthermore, accurate detection of AF levels in a quick and efficient manner is very critical. Therefore, utilizing AI to detect AF levels with accurate results and by processing US data will equip gynecologists with a great tool, which will help in improving the accuracy and efficiency of their diagnosis, and thus to monitor fetuses' health.

In this paper, we utilized transfer learning models that analyzed US images to detect the AF level. The model was trained and tested using a set of US images obtained from KFHU and Elite Clinic. The models were evaluated using performance measures such as accuracy, precision, recall, F1-score, balanced accuracy, and AUC-ROC, etc. The proposed models development consisted of two phases: the first phase is preprocessing, and the second phase is classification. In the first phase, starting with the cropping of the images in order to remove the text labels visible on most of them, this process was then followed by the removal of textual noise located in different locations within the images to further clarify the image and remove any noise that might negatively affect the accuracy, then enhancing the image to improve its quality and thus ensuring proper feature detection in the next phase. Lastly came augmentation, which was used to expand the size of the dataset using the ImageDataGenerator class available on the TensorFlow library. The second phase is AF classification, which was conducted using TensorFlow's five transfer learning models that include Xception, MobileNet, InceptionResnetV2, DenseNet121, and ResNet50V2, which are used to train the model on 50 epochs and used five-fold cross-validation on the data and batch size of 16, in order to accurately classify the AF images to help diagnose normal or abnormal AF levels. Upon analyzing the results, MobileNet gave us the best performance, in which it achieved an accuracy, precision, recall, and f1-scores of 0.94, 0.96, 0.94, and 0.95 respectively.

By developing this model, we aim to help gynecologists and physicians to perform accurate assessments of their cases and thus save the lives of fetuses and avoid premature birth or any other medical complications. For future work, we plan to expand the score to multiclass classification with three classes—normal, polyhydramnios, and oligohydramnios—instead of a binary class classification that includes normal and abnormal, to provide physicians and gynecologists with a more accurate prediction about the AF level diagnosis. Multiclass classification was not implemented in the current study due to the very small number of patients with oligohydramnios and polyhydramnios.

Author Contributions: Conceptualization, I.U.K., N.A., F.M.A., S.M., A.A., R.M.A., R.M.B. and N.H.A.Q.; methodology, I.U.K., N.A., F.M.A., S.M., A.A., R.M.A. and R.M.B.; formal analysis, F.M.A., S.M., A.A., R.M.A. and R.M.B.; investigation, F.M.A., S.M., A.A., R.M.A. and R.M.B.; resources, I.U.K., N.A., F.M.A., S.M., A.A., R.M.A. and R.M.B.; data curation, I.U.K., N.A., F.M.A., S.M., A.A., R.M.A., N.H.A.Q. and R.M.B.; writing—original draft preparation, F.M.A., S.M., A.A., R.M.A. and R.M.B.; writing—review and editing, I.U.K. and N.A.; visualization, F.M.A., S.M., A.A., R.M.A. and R.M.B.; supervision, I.U.K. and N.A.; project administration, I.U.K. and N.A.; funding acquisition, I.U.K., N.A., F.M.A., S.M., A.A., R.M.A., R.M.B. and N.H.A.Q. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki and approved by the Institutional Review Board (or Ethics Committee) of Imam Abdulrahman Bin Faisal University, Dammam, KSA (IRB-2022-09-065, 10-0-2022).

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset used in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Khan, I.U.; Aslam, N.; Anis, F.M.; Mirza, S.; AlOwayed, A.; Aljuaid, R.M.; Bakr, R.M. Amniotic Fluid Classification and Artificial Intelligence: Challenges and Opportunities. *Sensors* **2022**, *22*, 4570. [CrossRef] [PubMed]
2. Chen, Z.; Liu, Z.; Du, M.; Wang, Z. Artificial Intelligence in Obstetric Ultrasound: An Update and Future Applications. *Front. Med.* **2021**, *8*, 733468. [CrossRef] [PubMed]
3. Magann, E.F.; Chauhan, S.P.; Doherty, D.A.; Lutgendorf, M.A.; Magann, M.I.; Morrison, J.C. A review of idiopathic hydramnios and pregnancy outcomes. *Obstet. Gynecol. Surv.* **2007**, *62*, 795–802. [CrossRef] [PubMed]
4. Cho, H.C.; Sun, S.; Hyun, C.M.; Kwon, J.; Kim, B.; Park, Y.; Seo, J.K. Automated ultrasound assessment of amniotic fluid index using deep learning. *Med. Image Anal.* **2021**, *69*, 101951. [CrossRef] [PubMed]
5. Sun, S.; Kwon, J.Y.; Park, Y.; Cho, H.C.; Hyun, C.M.; Seo, J.K. Complementary Network for Accurate Amniotic Fluid Segmentation from Ultrasound Images. *IEEE Access* **2021**, *9*, 108223–108235. [CrossRef]
6. Ayu, P.D.W.; Hartati, S.; Musdholifah, A.; Nurdiati, D.S. *Amniotic Fluids Classification Using Combination of Rules-Based and Random Forest Algorithm BT—Soft Computing in Data Science*; Springer: Singapore, 2021; pp. 267–285.
7. Ayu, P.; Hartati, S. Pixel Classification Based on Local Gray Level Rectangle Window Sampling for Amniotic Fluid Segmentation. *Int. J. Intell. Eng. Syst.* **2021**, *14*, 420–432. [CrossRef]
8. Amuthadevi, C.; Subarnan, G.M. Development of fuzzy approach to predict the fetus safety and growth using AFI. *J. Supercomput.* **2020**, *76*, 5981–5995. [CrossRef]
9. Li, Y.; Xu, R.; Ohya, J.; Iwata, H. Automatic fetal body and amniotic fluid segmentation from fetal ultrasound images by encoder-decoder network with inner layers. In Proceedings of the 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Jeju, Kore, 11–15 July 2017. [CrossRef]
10. Ayu, D.W.; Hartati, S.; Musdholifah, A. Amniotic Fluid Segmentation by Pixel Classification in B-Mode Ultrasound Image for Computer Assisted Diagnosis. In *Communications in Computer and Information Science*; Springer: Singapore, 2019; Volume 1100. [CrossRef]
11. Looney, P.; Yin, Y.; Collins, S.L.; Nicolaides, K.H.; Plasencia, W.; Molloholli, M.; Natsis, S.; Stevenso, G.N. Fully Automated 3-D Ultrasound Segmentation of the Placenta, Amniotic Fluid, and Fetus for Early Pregnancy Assessment. *IEEE Trans. Ultrason. Ferroelectr. Freq. Control* **2021**, *68*, 2038–2047. [CrossRef] [PubMed]
12. Anquez, J.; Angelini, E.D.; Grange, G.; Bloch, I. Automatic segmentation of antenatal 3-D ultrasound images. *IEEE Trans. Biomed. Eng.* **2013**, *60*, 1388–1400. [CrossRef] [PubMed]
13. Ayu, P.D.W.; Hartati, S.; Musdholifah, A.; Nurdiati, D.S. Amniotic fluid segmentation based on pixel classification using local window information and distance angle pixel. *Appl. Soft Comput.* **2021**, *107*, 107196. [CrossRef]
14. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258.
15. He, J.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
16. Huang, G.; Liu, Z.; van der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
17. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. 2017. Available online: http://arxiv.org/abs/1704.04861 (accessed on 24 July 2022).
18. Keras InceptionResNetV2. Available online: https://jkjung-avt.github.io/keras-inceptionresnetv2/ (accessed on 12 May 2022).
19. Aslam, N.; Khan, I.U.; Aljishi, R.F.; Alnamer, Z.M.; Alzawad, Z.M.; Almomen, F.A.; Alramadan, F.A. Explainable Computational Intelligence Model for Antepartum Fetal Monitoring to Predict the Risk of IUGR. *Electronics* **2022**, *11*, 593. [CrossRef]