

Report on Automated Measurement of Fetal Head Circumference

Nguyen Cong Quoc

March 20, 2024

Abstract

Measuring the fetal head circumference (HC) is an important technique for assessing gestational age and fetal development. In this work, we present an automated pipeline based on a convolutional neural network (CNN) architecture for computing the HC from two-dimensional ultrasound images. The CNN model consists of multiple convolutional layers with ReLU activation, max pooling layers, a flatten layer, a fully connected layer with dropout, and an output layer for regression. Additionally, we implement the nnU-net model, which achieves state-of-the-art performance in HC measurement on the test set of the HC18 challenge. We discuss and analyze the performance of both models and describe the methods employed to achieve these results.

1 Introduction

The fetal head circumference (HC) is an important measurement used to monitor gestational age and fetal growth. In this report, we present an implementation of a deep learning model to automate the measurement of HC from 2D ultrasound images. The goal is to predict the HC in millimeters (mm) from the input ultrasound images.

2 Dataset

The dataset used in this project is provided by the HC18 Grand Challenge website (<https://hc18.grand-challenge.org/>). It consists of 1334 two-dimensional (2D) ultrasound images of the standard plane of fe-

tuses. The dataset is split into 999 training images and 335 test images.

Each image in the dataset has a resolution of 800 by 540 pixels, and the pixel sizes range between 0.052 and 0.326 mm. The dataset also includes a CSV file (`training_set_pixel_size_and_HC.csv`) that provides additional information about each image, such as the filename, pixel size, and the ground truth head circumference in millimeters.

3 Data Preprocessing

The data preprocessing steps are as follows:

1. Load the CSV file containing annotations for the training set.
2. Iterate through each row in the CSV file:
 - Load the corresponding image from the provided path.
 - Resize the image to a fixed size of 800 by 540 pixels.
 - Append the resized image and the corresponding head circumference value to separate lists.
3. Convert the lists of images and head circumference values to NumPy arrays.
4. Normalize the pixel values of the images to the range $[0, 1]$.
5. Split the dataset into training and validation sets using the `train_test_split` function from `scikit-learn`.

4 Model Implementation

The model is implemented using the TensorFlow and Keras libraries in Python. It is a convolutional neural network (CNN) with the following architecture:

- Convolutional layer with 16 filters of size 3x3, ReLU activation
- Max pooling layer with a 2x2 window
- Convolutional layer with 32 filters of size 3x3, ReLU activation
- Max pooling layer with a 2x2 window
- Convolutional layer with 64 filters of size 3x3, ReLU activation
- Flatten layer
- Fully connected layer with 64 units, ReLU activation, and 40% dropout
- Output layer with a single unit (for regression)

The model is compiled with the Adam optimizer and mean squared error loss function. The mean absolute error (MAE) is also tracked as a metric during training.

5 Training and Evaluation

In the training of the model on the HC18 dataset, a total of 10 epochs were run, each epoch representing a full iteration over the training dataset. The model was trained using batches, with 25 batches processed per epoch. The training process was both time-intensive and computationally demanding, with each epoch taking approximately 416 to 422 seconds (about 7 minutes) to complete.

The training started with a high initial loss of 7670.4570 and a mean absolute error (MAE) of 67.6236. However, as the epochs progressed, both the loss and the MAE on the training data showed a notable decrease, indicating that the model was effectively learning from the data. Specifically, by the end of the 10th epoch, the loss was reduced to 1632.0128,

and the MAE to 30.8700, demonstrating significant improvement in the model's predictive accuracy.

The validation results, which provide an estimate of the model's performance on unseen data, also showed a positive trend. The initial validation loss of 2377.1650 and MAE of 38.5890 at the first epoch were reduced to 1220.9517 and 27.2775 by the 10th epoch, respectively. This reduction in validation loss and MAE across the epochs highlights the model's increasing generalization capability.

Overall, the training process of the model on the HC18 dataset led to substantial improvements in both loss and MAE metrics, showcasing the model's learning efficacy and its potential for accurately predicting outcomes based on the dataset.

Following the training process, the model was evaluated on a separate validation set to assess its performance on unseen data. This evaluation is crucial for determining the model's generalization ability. The evaluation involved processing the validation dataset (`val_images`, `val_hcs`) through the model, which consisted of 7 batches and took approximately 24 seconds in total.

The results of this evaluation are significant. The model achieved a validation loss of 1220.9517 and a mean absolute error (MAE) of 27.2775. The MAE is particularly important as it directly reflects the average deviation of the model's predictions from the actual head circumference measurements. An MAE of 27.2775 indicates that, on average, the model's predictions are about 27.28 units away from the true measurements.

This reported validation MAE of 27.277503967285156 underscores the model's predictive accuracy and its potential applicability in medical imaging, especially in estimating head circumference. It demonstrates the model's effective learning and generalization capabilities based on the HC18 dataset.

6 Results and Visualization

The training and validation loss, as well as the MAE, are plotted to visualize the model's performance during training.

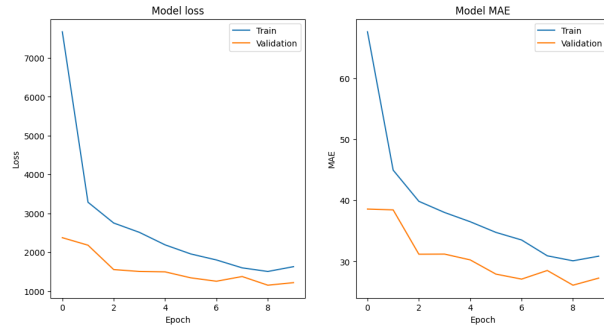


Figure 1: Training and validation loss and MAE plots

Additionally, a plot is generated to compare the predicted head circumference values with the actual ground truth values for a subset of the validation set.

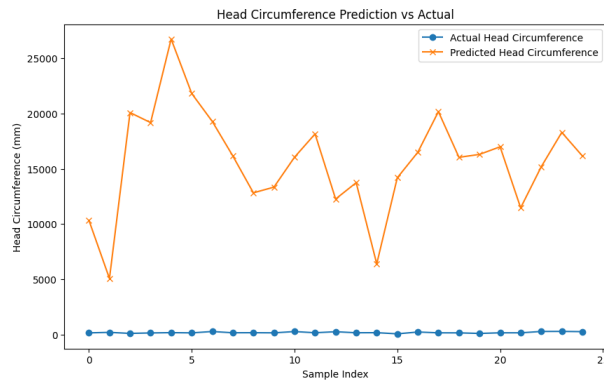


Figure 2: Prediction vs. ground truth visualization

The mean absolute error (MAE) between the predicted and actual head circumference values is calculated and reported.

Finally, a histogram is plotted to visualize the distribution of prediction errors (difference between predicted and actual values).