An Infant Cry Recognition based on Convolutional Neural Network Method

K. Teeravajanadet[1], N. Siwilai[2], K. Thanaselanggul[3], N. Ponsiricharoenphan[4],
S. Tungjitkusolmun[5], P. Phasukkit[6]

Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang
Bangkok, Thailand
[1]tvd.rooshdee@gmail.com, [2]natapatsiwilai@gmail.com, [3]kamonnut.21@gmail.com, [4]nutthapol.p.24@gmail.com,
[5]supan.tu@kmitl.ac.th, [6]pattarapong.ph@kmitl.ac.th

***Abstract*—In this paper, an investigation of crying signal spectra is used to classify categories of infant cries. Three different types of crying considered in this work are hungry, sleepy and burping need. These cries are preprocessed and converted for calculation of Mel-Frequency Cepstral Coefficients (MFCC) before being classified by Convolutional Neural Network (CNN). Experimental results show that CNN based deep learning achieves high performance of 84%.**

***Index Terms*—CNN, deep learning, MFCC, infant crying classification**

I. INTRODUCTION

For infants, crying is one of few ways to communicate or express how they feel or need to their parents. Because newborns may cry for several reasons, this may make new parents feel tough to handle or respond to their baby's need. However, a study in [1] has discovered the repeatable pattern pronunciation related to infant behaviour. For example, 'Neh' means hungry, 'Owh' presents their tired or sleepy and 'Eh' tells burping of the baby. These specific patterns have been reflected into such non-stationary crying signal and required complicated preprocessing techniques for feature extraction before feeding to machine learning algorithm for classifying [2]. However, by using deep learning model, these harsh procedures can be partly ignored because the model will learn from the information itself. While a study in [3] used deep learning for detecting crying onset, another study [4] used spectrogram derived from short-time-Fourier transform (STFT) to be an input of the model. The overall accuracy was 76%. However, STFT can introduce unrelated information due to its linear-frequency scaling. Thus, this paper will apply deep learning to classify crying into three categories; hungry, sleepy and burping (multi-class classification) by using MFCC features.

II. BACKGROUND

A crying signal is composed of not only a single crying unit such as cry utterances or expiratory sound but also non-voice, silence or other artefacts. To deal with the problem, short-time energy and zero-crossing rate [3] together with root-mean-square-energy calculation are used to find for threshold setting.

978-1-7281-3752-0/19/$31.00 ©2019 IEEE

A. Short-time Energy

Short-time energy is the mean of the square of samples in a suitable window:

$$E(n)= \frac{1}{N}\sum_{m=0}^{N-1}[w(m)x(n-m)]^2 \quad (1)$$

Where w(m) are coefficients of the window function (Hamming window) of length N, m stands for window index, and n is an index of samples.

B. Short-time Zero-Crossing Rate

This metric is defined as the rate of change of signal sign:

$$Z(n)= \frac{1}{N}\sum_{m=0}^{N-1}\left|\text{sign}\big(x(n-m)\big)-\text{sign}\big(x(n-m-1)\big)\right| \quad (2)$$

where $\text{sign}(x(m))=\begin{cases}1, & x(m)\geq 0\\ -1, & x(m)<0\end{cases}$ and x(m) is sample.

C. Root-Mean-Square Energy

Equation of RMSE is as following:

$$R(n)= \sqrt{\frac{1}{N}\sum_{m=0}^{N}|x(n-m)|^2} \quad (3)$$

D. Mel-Frequency-Cepstral Coefficient (MFCC)

Characteristics of the cry signal is a representation of events in the vocal tract system. By exploiting MFCC features of each cry [5], this can enhance discriminant spectral information.

1) Pre-emphasis: a kind of filter which is required to reduce noise and get a smoother spectrum. The input is a time-domain signal expressed in the equation:

$$y(n)=x(n)-ax(n-1) \quad (4)$$

where a is emphasis filter constant, $0.9 < a < 1.0$.

2) Frame blocking and windowing: segment signal into multiple overlapped frames. This segmented frame will be multiplied with the FIR filter to remove alias signal:

$$y_1(n)=x_1(n)w(n) \ , \ 0 \le n \le 1 \tag{5}$$

where w(n) represents the Hamming window:

$$w(n)=0.54-0.46\cos\left(\frac{2\pi n}{N-1}\right) , 0 \le n \le N-1 \tag{6}$$

3) Fast Fourier Transform: conversion of time-series signal into frequency spectrum:

$$X_n=\sum_{k=0}^{N-1} x_k e^{-\frac{2\pi jkn}{N}} \tag{7}$$

4) Mel-Frequency Wrapping: the Mel-scaled filter banks (triangular filters) mimicked from human perception:

$$F_{mel} = 2595*\log_{10}(1+\frac{F_{HZ}}{700}) \tag{8}$$

5) Compute log spectrum from filter banks:

$$X_i= \log_{10}\left(\sum_{k=0}^{N-1}|x(k)|H_i(k)\right) \tag{9}$$

where i = 1, 2, 3, …, M (M is the number of triangle filters) and $H_i(k)$ is the value of the i-triangle filter for acoustic frequency of k.

6) Cepstrum: convert Mel-spectrum into the time domain by using a Discrete Cosine Transform (DCT):

$$C_j=\sum_{j=1}^{K} X_j \cos\left(\frac{j(i-1)}{2\frac{\pi}{k}}\right) \tag{10}$$

where C_j is the MFCC coefficient, X_j is the power spectrum of Mel frequency and j= 1,2,3…, k (k is numbers of desired coefficients).

E. Convolutional Neural Network

There have been widespread applications of convolutional neural network in the areas of natural language processing and computer vision, especially where the number of data has to be learned and classified. Like ordinary neural network, CNN contains lots of hidden layers linked by neurons which have adjustable and learnable weights. CNN is also composed of several filters which are applied to outputs provided by the previous layer using the convolution operation. CNNs learn the filters during the training process, which can be thought of a way to generate crucial information out of the data. Thus, in contrast to conventional classification models, the need for dependence on prior knowledge is a significant advantage of CNNs. Figure1 shows a brief process of how CNN learn the data.

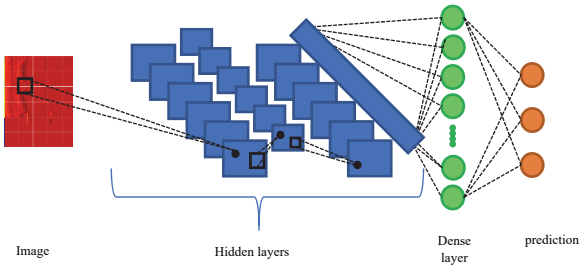


Fig.1. CNN algorithm workflow

III. METHODS

From figure 2, we use Python programming language and Jupyter notebook as our Integrated Development Editor (IDE) for code and function development plus essential modules such as pandas, Numpy, Matplotlib (for matrix calculation and visualization) Librosa (for audio processing and feature calculation) and Scikit Learn and Keras-Tensor Flow (for classification model). The proposed method is composed of three main parts. Firstly, a raw crying signal embedded with noise or silence is fed to the unit of artefact removal. Then, the enhanced crying signal is taken to compute MFCC and its derivative for input feature vectors of the CNN model. After constructing CNN, 80% of input features are randomly stratified sampled to be and input for model learning the data. During this step, some hyperparameters are fine-tuned to improve accuracy and prevent overfitting. Finally, the remaining 20% of the features will be fed to test the accuracy of the classification model.

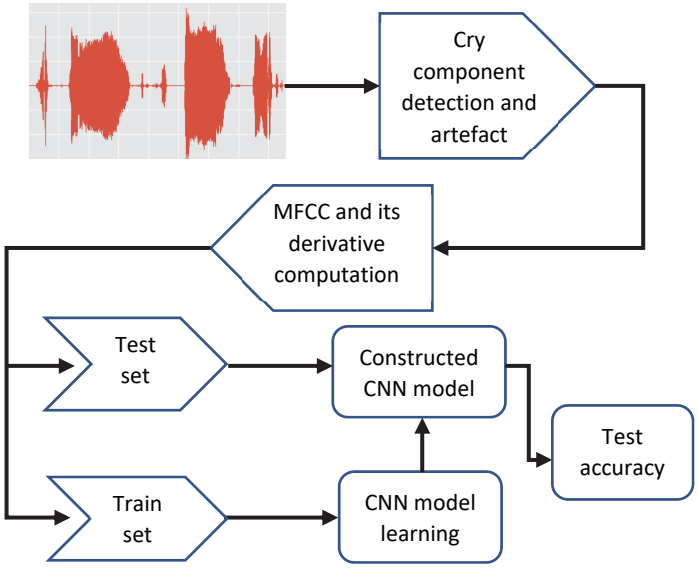


Fig.2. The proposed method

IV. EXPERIMENTS AND RESULTS

A. Dataset

In this paper, we use infant-crying data from the Dunstan Baby Language, and the signal sampling rate is 16 kHz. These data are divided into 80% train and 20% test set, both have three

categories, namely hungry, sleepy and burping. The number of three classes are 46, 60 and 75, which is shown in table 1.

TABLE 1
NUMBER OF DATA IN TRAIN AND TEST SET

DATASET	HUNGRY	SLEEPY	BURPING
TRAIN	36	48	60
TEST	10	12	15

B. Preprocessing

After collecting newborn-crying signal, we analyze the recorded signal by first detecting silence and non-voiced artefacts such as breathing or whimpering. This process utilizes a combination of three short-time methods including Energy, Root Mean Square Energy (RMSE) and Zero Crossing Rate, with the setting of 20 ms frame length and 10 ms overlapping. Figure 3. shows the result of the preprocessed signal. Since the normal infant cry duration is about 1.6 seconds, three quantifiable threshold values are set followings to detect the artefacts:

1) energy > 0.5
2) RMSE > 0.05
3) Zero-crossing rate > 10

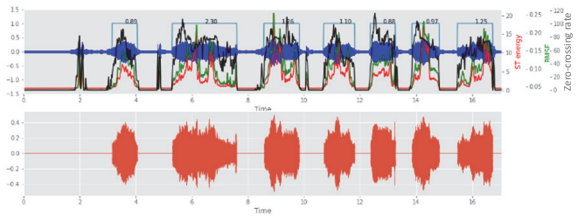


Fig.3. Crying signal before and after preprocessing. (Top) A normalized raw audio signal (blue) is used to perform three short-time calculations; energy (red), RMSE (green) and zero-crossing rate (black). (Bottom) Preprocessed signal (Red)

C. Mel-Frequency Cepstral Coefficients (MFCC)

After eliminating artefacts, the desired signals are then stacked into 5 seconds for a proper calculation. If the signal duration is more than the threshold, remove the remaining, otherwise, pad signal with zero. MFCC method provided by Librosa module is then taken to calculate parameters by setting n_mfcc, n_fft and hop_length to be 16, 256 and 128 respectively. Apart from this, delta feature methods are applied to MFCC to enhance higher essential information. MFCC result is shown in figure 4, while its first and second delta features are also given in figure 5 and figure 6. This complex information will then be fed to CNN for constructing classification models.

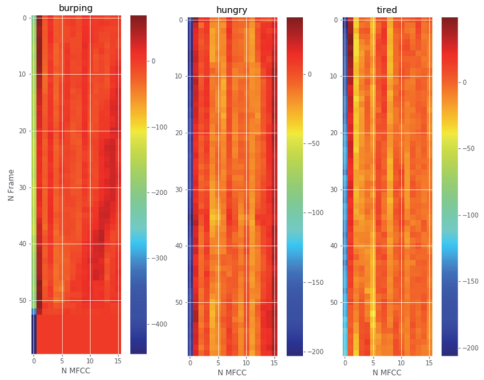
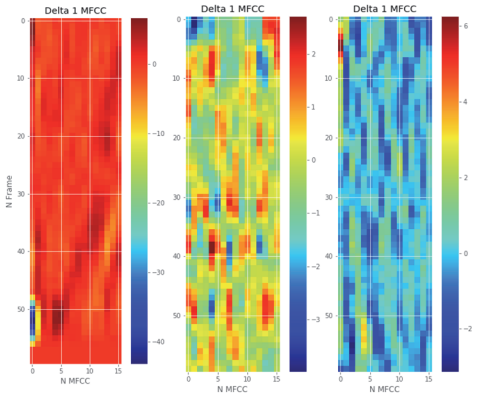


Fig.4. Examples of MFCC for burping, hungry and sleepy(tired)

Fig.5. Delta feature order 1 of burping (left), hungry (middle), sleepy (right)



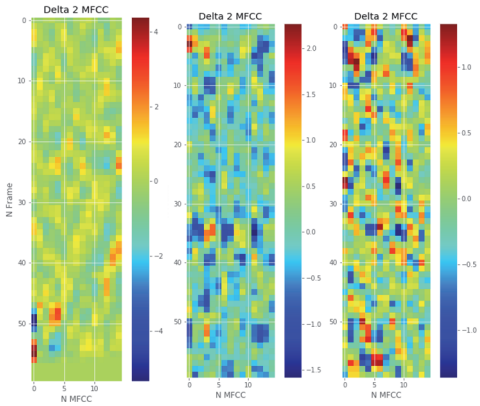


Fig.6. Delta feature order 2 of burping (left), hungry (centre), tired (right)

D. Convolutional Neural Network Model (CNN) Classification

The deep learning framework Keras using Tensor Flow backend is used in this paper. The model is composed of six deep two-dimensional convolutional layers (CONV2D). Parameters set in each layer are 32-128 filters and 3x3 kernel size. Max pooling process is then applied at the first and last layer that helps to reduce computational consumption by reducing dimensionality; however, still, maintain important information. Softmax method is used for activation function, and dropout technique which helps reducing overfitting is set to be 0.5. Another two parameters for constructing model are optimizer and loss function which are set to be Adam and categorical-cross entropy respectively. After the model is constructed, training dataset of 626x48 input shape is then fed to fit the model by setting learning rate of $4x10^{-4}$, 128 batch size and 200 epochs. Table 3 presents the proposed CNN architecture in details.

TABLE 3
DESIGNED CNN ARCHITECTURE (f is kernel size, s is strides)

N^{th} layer	Layer	Number of filters	Padding	Activation shape
0	Input	-	-	(626,48,1)
1	Batch normalization ReLU	-	-	(626,48,1)
	ConV2D (f =3,s=1)	32	Valid	(626,48,32)
	Max Pooling	-	-	(313,24,32)
2	Batch normalization ReLU	-	-	(313,24,32)
	ConV2D (f =3,s=1)	32	Valid	(313,24,32)
3	Batch normalization ReLU	-	-	(313,24,32)
	ConV2D (f =3,s=2)	64	Valid	(157,12,64)
4	Batch normalization ReLU	-	-	(157,12,64)

	ConV2D (f =3,s=1)	64	Valid	(157,12,64)
5	Batch normalization ReLU	-	-	(157,12,64)
	ConV2D (f =3,s=2)	128	Valid	(79,6,128)
6	Batch normalization ReLU	-	-	(79,6,128)
	ConV2D (f =3,s=2)	128	Valid	(79,6,128)
	Global Max pooling	-	-	(128,1)
7	Dense(fully-connected)	-	-	(1024,1)
8	Output	-	-	(3,1)

After running 200 iterations, the network achieves 84% test accuracy, and a confusion matrix is provided in Table 4. From figure 7, history of accuracy and loss between train and test set of each running epoch are consistent after epoch 150, and this has assured that the model is not influenced by overfitting.

TABLE 4
CNN MODEL PREDICTION RESULT OF TEST DATA
(NORMALIZED SCORE)

ACTUAL				
	BURPING	0.80	0.19	0.01
	HUNGRY	0.10	0.80	0.10
	SLEEPY	0	0.08	0.92
		BURPING	HUNGRY	SLEEPY

PREDICTED

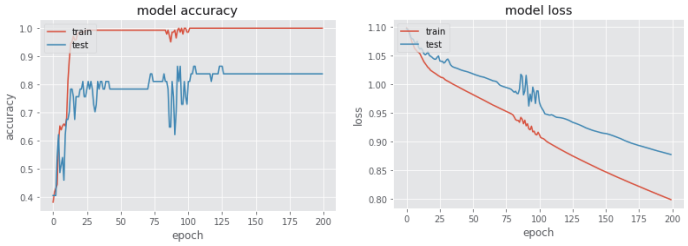


Fig.7. Model accuracy (left) and model loss (right) history

V. CONCLUSION

This paper has utilized MFCC and its secondary characteristics derived from different newborn-crying signals to classify into three types; hungry, sleepy and burping. From table 3, CNN can learn information from the complicated MFCCs spectra with a total accuracy of 84%. Each class accuracy is 80%, 80% and 92% for burping, hungry and sleepy respectively. For future work, more data need to be collected to enhance classification accuracy and consistency, which will be integrated into the infant care system.

REFERENCES

[1] E. Rincon, J. Beltran, M. Tentori, J. Favela and E. Chavez, "A Context-Aware Baby Monitor for the Automatic Selective Archiving of the Language of Infants," in *2013 Mexican International Conference on Computer Science (ENC)*, Morelia, MichoacÃ¡n, Mexico, 2013 pp. 60-67.

[2] A. Osmani, M. Hamidi and A. Chibani, "Machine Learning Approach for Infant Cry Interpretation," in *2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI)*, Boston, MA, USA, 2017 pp. 182-186.

[3] L. Liu, W. Li, X. Wu and B. X. Zhou, "Infant cry language analysis and recognition: an experimental approach," in *IEEE/CAA Journal of Automatica Sinica*, vol. 6, no. 3, pp. 778-788, May 2019.

[4] C. Chang and J. Li, "Application of deep learning for recognizing infant cries," *2016 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW)*, Nantou, 2016, pp. 1-2.

[5] D. Anggraeni, W.S.M. Sanjaya, M.Y.S. Nurasyidiek, M. Munawwaroh, "The Implementation of Speech Recognition using Mel-Frequency Cepstrum Coefficients (MFCC) and Support Vector Machine (SVM) method based on Python to Control Robot Arm," in *2018 IOP Conference Series: Materials Science and Engineering,* vol.288.