Crime Prevention System: Crashing Window Sound Detection Using AI Processor

Kwang Hyun Go, Student Member, IEEE, Chang Yeop Han, Student Member, IEEE,
Kwon Neung Cho, Student Member, IEEE, and Seung Eun Lee* Member, IEEE
Department of Electronics Engineering
Seoul National University of Science and Technology
Seoul, Republic of Korea
{gokwanghyun, hanchangyeop, chokwonneung, *seung.lee}@seoultech.ac.kr

2021 IEEE International Conference on Consumer Electronics (ICCE) | 978-1-7281-9766-1/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICCE50685.2021.9427630

***Abstract*— This paper introduces the AI system that is used as crime prevention system at house or store. This system checks intrusion by detecting the sound of crashing windows. The proposed system consists with a micro controller unit (MCU) and a AI processor. We employ various sounds such as crashing windows, crashing plates, and other sounds that occur in everyday life to verify the proposed system. In order to find the optimized learning model, we employ simulation model of the AI processor and implement the optimal learning model. We tested our proposed system and the result shows accuracy up to 91%.**

Keywords*— *AI processor, Internet of Things (IoT), Embedded system, Signal processing.

I. Introduction

The AI systems have been widely used because the AI systems offer many benefits. In line with this trend, many studies also have been conducted to apply the AI systems in embedded system [1], [2]. The most of the AI systems need high computing power and many resources. Generally, as the embedded system has low computing power and fewer resources, it is hard to perform high performance AI operation. For this reason, the home AI applications are implemented on the IoT devices that employ the resource of the cloud [3]. However, as the AI applications implemented on the IoT devices are affected by network environment, it is difficult to apply the IoT devices to crime prevention systems that require rapid response. Therefore, we propose the AI system using the micro controller unit (MCU) and the AI processor.

In the proposed system, the MCU transmits the sound data received from input devices such as microphone to the AI processor named Intellino. The Intellino is an AI processor that performs AI operation for the embedded system [4]. The Intellino stores the training dataset and classify the input data by calculating the distance with the trained data. When the Intellino transmits recognition result to the MCU, the MCU sends an alert to user when the recognition result is sounds of window crashing. The Intellino has flexible configuration that makes user to change the number of data and each data size. For the best AI performance, we utilize the software simulator to find the optimal configuration and learning model. After find the configuration, we implement the optimal configuration and learning model on the Intellino and test the recognition result. The results show that the proposed system classifies the crashing window sounds and other sounds successfully.

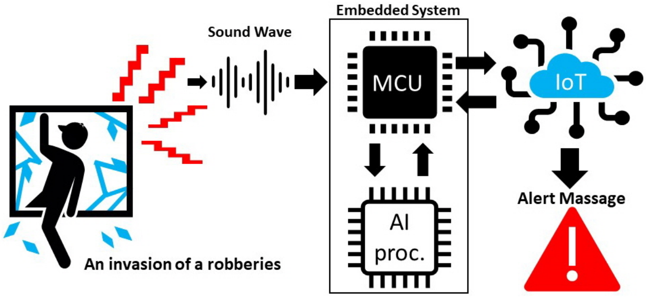

Fig. 1. Overview of the proposed system

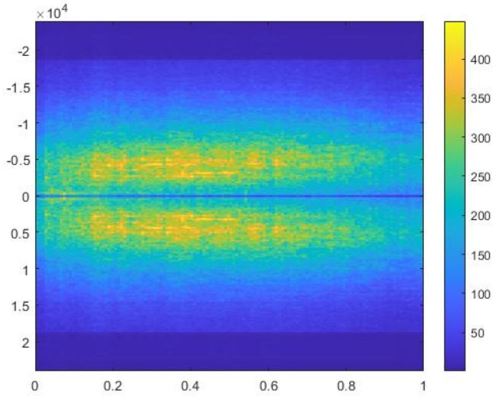

Fig. 2. Accumulated the STFT result of crashing window sounds

II. Backgrounds

The original sounds data have time and frequency data at once. In order to exploit the sounds data, the preprocessing is conducted to extract feature data. The short-time fourier transform (STFT) extracts the frequency data over time from sounds data. The extracted data have 3-dimension value and each dimension represent time, frequency, and magnitude.

In this paper, we need to classify the crashing window sounds. When the window is crashing, the sound has specific frequency data. Fig.2 shows the sum of the STFT results of crashing windows sounds. Y axis represents frequency, X axis represents time, the color represents magnitude. The crashing window sounds have a large magnitude at a certain frequency. Therefore, we exploit the frequency data to learn and recognize the sounds that we want.

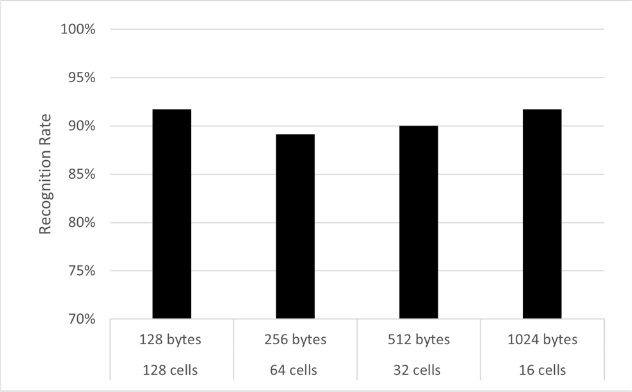

Fig 3. Recognition rate of software simulation

III. Simulation

A. Software Simulator

In order to measure the recognition rate of the AI processor in the proposed application, we employ the software simulator before hardware implementation. The software simulator performs the same functionality of the AI processor and supports flexible parameter of configurations such as the number of the training data or the size of the data. We measure the recognition rate under the configurations that 128 number of training data with 128 bytes data size, 64 number of data with 256 bytes, 32 number of data with 512 bytes and 16 number of data with 1024 bytes. By simulating with various parameters of configuration, the optimized hardware specification within the target application is determined.

B. Simulation procedure and result

For the recognition rate simulation, we use 358 number of data extracted by the STFT. The dataset consist of two category of data that represent the window crashing sounds and other sounds. The similar sounds data to the window crashing sounds are also included in the other sounds category. In the simulation procedure, the window crashing sounds and other sounds are sampled randomly with the same ratio. The sampled data are used to the training data and the test data are also sampled randomly in each category. By checking the classified result of the test data, recognition rates are calculated. In order to obtain high recognition rate, we perform the simulation procedure 10,000 times at each configuration. The maximum recognition rates of each configuration are represented in the Fig. 3. The model with the configuration of 128 number of cells and 128 bytes size of data has the highest recognition rate at 91.74%.

IV. implementation

In the software simulation, the simulation results show that the most well-learned configuration is having 128 bytes of data in each 128 cells, followed by 1024 bytes in each 16 cells. Therefore, we adopt the two configurations and datasets that show the highest recognition rates to our proposed system.

The proposed system learns and recognizes the data by the serial communication and classifies the result as the window crashing sounds and other sounds. Fig. 4 shows the result of recognizing the test dataset in two configurations. The classification is only verified window crashing sounds or not, however we also check the similar sounds are recognized as other sounds because detecting the similar sounds are important part of the system performance. The result shows the recognition rate of window crashing is lower than other recognition rates. However, as the window crashing sounds recognition results show 75.47% and 81.13%, the proposed system classifies the window crashing sound successfully.

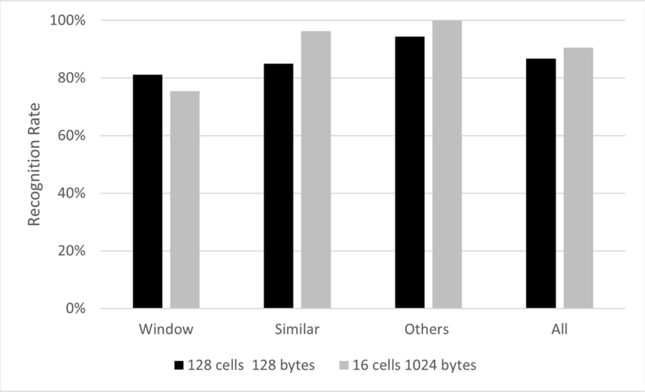

Fig 4. Recognition rates of sounds data

V. Conclusion

In this paper, we propose the AI IoT system for crime prevention by detecting the window crashing sound based on the AI processor. We perform the STFT to transform sounds data to feature data. The optimized learning models obtained through the software simulator are implemented on the AI processor. As a result, the recognition rate is about 81%. The proposed system successfully recognizes the sounds of window crashing. Therefore, the proposed system performs the AI applications without servers in the IoT systems.

Acknowledgment

This study was supported by the Ministry of Trade, Industry & Energy (MOTIE, Korea) under Industrial Technology Innovation Program. No. 20006566, ‘Development of embedded artificial intelligence module and system based on neuromorphic’

Reference

[1] D. Hou, M. D. Raymond Hou and J. Hou, "On-device Subspace Learning Chest X-Ray Screening," 2020 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, pp. 1-5, 2020

[2] S. Y. Jang, Y. H. Yoon and S. E. Lee, "Stochastic Computing based AI System for Mobile Devices," 2020 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, pp. 1-2, 2020

[3] S. B. Calo, M. Touna, D. C. Verma and A. Cullen, "Edge computing architecture for applying AI to IoT," 2017 IEEE International Conference on Big Data (Big Data), Boston, MA, pp. 3012-3016, 2017

[4] Y. H. Yoon, D. H. Hwang, J. H. Yang, S. E. Lee, "Intellino: Processor for Embedded Artificial Intelligence," Electronics 9, no. 7: 1169, 2020