# Offloading Image Recognition Processing Using Heterogeneous FPGAs for Care Robot Applications

EISUKE OKAZAKI[†1] HAYATO MORI[†2] GAI NAGAHASHI[†1]
MIKIKO SATO[†1] MIDORI SUGAYA[†2] TAKESHI OHKAWA[†3]

***Abstract***: Nursing care robots are expected to reduce caregiver labor by patrolling the care facility and detecting falls of older adults. However, the image recognition processing of robots has performance and power consumption issues. To solve these problems, our approach is to use heterogeneous FPGAs as the computational resource of the edge server. This paper shows the concept of offloading image recognition processing from the robot to the edge server. As an evaluation, the throughput of pose estimation and face detection was measured using heterogeneous FPGAs. The results show that throughput improvements of 8.71 times and 119 times in face detection and pose estimation, respectively, can be achieved compared to the Raspberry Pi 4B. In addition, results show that the use of heterogeneous FPGAs is advantageous because the FPGA with the DPU that achieves the highest throughput depends on the type of image recognition processing.

***Keywords***: FPGA, DPU, heterogeneous computing, offloading

## 1. Introduction

As the working population declines, there is a pressing need for automated solutions. Nursing care robots patrolling care facilities are expected to reduce the workload for human caregivers[1]. However, performance and power consumption issues exist in the image recognition processing required for robots. Therefore, high-throughput image recognition processing is expected using a heterogeneous computing system with accelerators such as Field-Programmable Gate Arrays (FPGAs) and General-Purpose Graphics Processing Units (GPGPUs) in the edge server.

The goal is to increase the efficiency and responsiveness of care robots, improve the quality of care, and reduce the burden on caregivers in the care environment by optimizing the throughput of image recognition processing using heterogeneous FPGAs. This paper presents the system concept for offloading from nursing care robots to the edge server and the throughput in heterogeneous FPGAs of image recognition processing.

## 2. Offloading Image Recognition Processing of Nursing Care Robot Applications

Nursing care robot applications to talk with residents or detect falls require face detection and pose estimation processing. It is not easy to process these applications only within the robot in terms of performance and power consumption.

Our approach is to use heterogeneous FPGAs with Deep-Learning Processing Units (DPUs), softcore computation engines optimized for convolutional neural networks, as the computational resource of the edge server to solve these issues. Figure 1 shows the concept for offloading the image recognition processing of nursing care robot applications to an edge server. The robot sends images to the edge server. Heterogeneous FPGAs on the edge server-side process the image recognition and respond results to the robot.

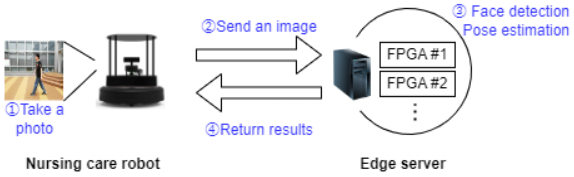


Figure 1 Concept of offloading system of nursing care robots

## 3. Throughput of Image Recognition Processing in Heterogeneous FPGAs

To obtain the throughput of face detection and pose estimation when offloaded to heterogeneous FPGAs, throughput was measured in both the Raspberry Pi 4B (8 GB) with ARM Cortex-A72, commonly used in robot control systems, and the two types of the FPGA environment shown in Table 1. M-KUBOS[2] is powered by a Zynq UltraScale+ MPSoC that is optimized for the edge. And Alveo U50 is an FPGA intended for use in data centers.

Table 2 shows the architectural names of the DPUs in the FPGA environment shown in Table 1 and their uses. DPU (a) and DPU (b) run at 300 MHz, and DPU (c) is at 333 MHz. DPU (a) was used for face detection and pose estimation. DPU (b) was used for face detection, and DPU (c) was used for pose estimation. DPU (a) and DPU (b) each contain two cores of DPUs in one bitstream. They were also measured for throughput during parallel execution on two DPUs. DPU (b) has three Processing Engines (PEs) as convolution engines, so the batch size limit is three. In this paper, to show the primary performance, the batch size was measured to be one for all DPUs.

The machine learning models used for face detection and pose estimation are the pre-trained DenseBox and OpenPose[3]. The input image sizes are 640x360 for face detection and 368x368 for pose estimation.

Table 3 shows the hardware usage of a single DPU. Since DPU (a) and DPU (b) have two cores, the entire bitstream uses twice as many hardware resources as in Table 3.

†1 Tokai University
†2 Shibaura Institute of Technology
†3 Kumamoto University

Table 1 Environment of FPGA systems

| FPGA | M-KUBOS[2] (xczu19eg-ffvc1760-2-i) | Alveo U50 machine (xcu50-fsvh2104-2-e) |
|---|---|---|
| CPU | ARM Cortex-A53 | Intel Core i5-12400 |
| RAM | 4 GB | 40 GB |
| OS | Petalinux 2022.2 | Ubuntu 18.04 |
| Vitis AI | 3.0 | 1.4.1 |

Table 2 DPU architectures on FPGAs and use in this study

| FPGA | DPU architecture | Face detection | Pose estimation |
|---|---|---|---|
| M-KUBOS | **(a)** DPUCZDX8G B4096 (x 2 cores) | ✓ | ✓ |
| Alveo U50 | **(b)** DPUCAHX8H 3PE (x 2 cores) | ✓ | - |
| | **(c)** DPUCAHX8L | - | ✓ |

Table 3 Hardware usage of a single DPU

| | LUT | FF | Block RAM | LUT RAM | Ultra RAM | DSP |
|---|---|---|---|---|---|---|
| DPU (a) | 51,185 | 97,926 | 255 | 7,061 | - | 710 |
| DPU (b) | 153,203 | 254,353 | 219.5 | - | 194 | 1,703 |
| DPU (c) | 212,860 | 299,342 | 459 | - | 312 | 2,452 |

Figure 2 shows the measured throughput of face detection on each device, which was 22.4 FPS on the Cortex-A72; in the FPGA environment, DPU (a) x2 implemented on M-KUBOS had the highest performance, which could be improved by 8.71 times. In addition, the measured throughput of pose estimation shown in Figure 3 was 0.1 FPS for the Cortex-A72 but 11.9 FPS for the DPU (c) implemented on Alveo U50, a throughput improvement of 119 times compared to the Cortex-A72.

Due to the difference in the number of cores, the throughput of face detection improved by a factor of 1.90 times in DPU (b), as shown in Figure 2. In addition, parallelization improved the throughput of pose estimation in DPU (a), shown in Figure 3, by a factor of 1.94 times.

## 4. Discussion

Regarding the FPGA hardware usage, DPU (b) uses more hardware than DPU (a). This is because the architecture of DPU (b) is equipped with three PEs to achieve highly parallel processing.

In addition, the measured throughputs of DPU(a) x1 and DPU(a) x2 are almost the same, as shown in Figure 2. This means that the pre-processing and post-processing in the CPU take more time than the process in the DPU. These processes include data transfer, probability calculation with softmax, and bounding box reduction with non-maximum suppression.

On the other hand, DPU(a) x2 shown in Figure 3, was 1.94 times larger than DPU(a) x1, indicating that parallelization was effective.

These results suggest that the appropriate choice of FPGA, supported DPUs, and the number of parallelisms are necessary to obtain high processing performance when offloading robot image processing tasks to FPGAs.

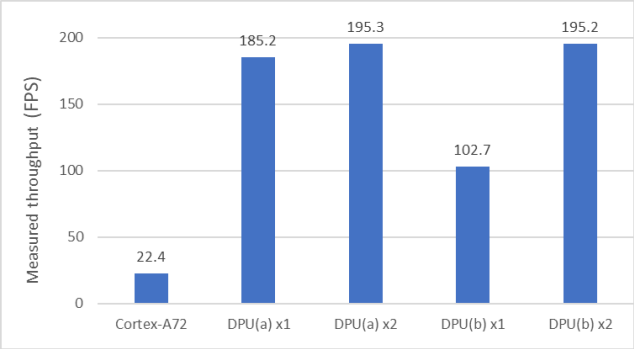


Figure 2 Measured throughput of face detection

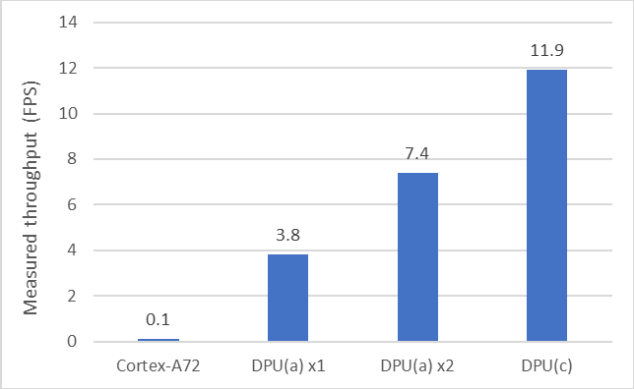


Figure 3 Measured throughput of pose estimation

## 5. Conclusion and Future Work

This paper presented a system concept for offloading pose estimation and face detection tasks of nursing care robot applications to edge servers. In measurement experiments of the processing throughput on heterogeneous FPGAs, face detection and pose estimation could be improved by 8.71 and 119 times compared to the Raspberry Pi 4B. In addition, it was revealed that measurements indicated that the FPGAs with the DPUs that provide the highest throughput differ depending on the type of image recognition processing. This fact indicates that offloading to heterogeneous FPGAs is effective.

Future work will measure latency and power efficiency when offloaded and how many robots can be handled using edge servers composed of heterogeneous FPGAs as computational resources.

## Reference

[1] Kawata, J., Morimoto, J., Kaji, Y., Higuchi, M., Matsumoto, K., Booka, M., and Fujisawa, S.: Development of a Care Robot Based on Needs Survey, *Journal of Robotics and Mechatronics*, Vol.33, No.4, pp.739-746 (2021).

[2] Inage, T., Hironaka, K., Iizuka, K., Ito, K., Fukushima, Y., Namiki, M., and Amano, H.: M-KUBOS/PYNQ Cluster for multi-access edge computing, *CANDAR*, pp.95-101 (2021).

[3] AMD Xilinx: Vitis AI Model Zoo (online), available from 〈https://docs.xilinx.com/r/1.4.1-English/ug1414-vitis-ai/Vitis-AI-Model-Zoo〉 (accessed 2023-09-17).

**Acknowledgments** This work was supported by JST CREST Grant Number JPMJCR19K1, Japan.