Distributed Human Activity Data Processing using HASC Tool

Nobuo Kawaguchi **Nobuhiro Ogawa** **Yohei Iwasaki** **Katsuhiko Kaji**
Graduate School of Engineering, Nagoya University
{kawaguti, hiro, iwasaki, kaji}@ucl.nuee.nagoya-u.ac.jp

ABSTRACT
To accelerate and simplify human activity recognition research, we have been developing a data processing tool named "HASC Tool." As the activity corpus becomes huge, it is not simple to handle the large number of files because it takes a lot of time to process. In this paper, we propose a distributed data processing mechanism which is implemented in the HASC Tool. By using the system, we can simply scale the local system into distributed processing. We also show the preliminary experimental result.

Author Keywords Activity Recognition, Activity Understandings, Distributed Data Processing, Large Scale Data Processing, HASC.

ACM Classification Keywords H.5.2 [User Interfaces]: Interaction styles (e.g., commands, menus, forms, direct manipulation) .

General Terms Measurement, Experimentation, Human Factors, Documentation, Standardization

INTRODUCTION

In the field of pattern recognition, a large number of try-and-errors are required to obtain better recognition results. If the training data becomes huge, the processing time becomes also huge. To enable the human activity recognition technology works in the real world, we have been constructing a large scale activity corpus through "HASC Challenge [1]". Evaluation efforts using such a large corpus require huge number of computations. For example, in the analysis of the effects of the number of subjects[2], we have performed more than 13,000 evaluations for more than 80 subjects with the multi-fold cross validation. This evaluation requires more than 24h even using high performance CPU such as AMD Opteron 2.8GHz.

Signal processing can be inherently performed parallel. However, the most of free signal processing toolkits such as SciLab, Octave and SciPy do not support distributed computing natively. If you add appropriate library or programs, it may support distributed computing. But it is not popular. MATLAB supports distributed computing through "Distributed Computing Toolbox." But it requires a license for each worker. So it is not handy to use.

UbiComp'11, September 17–21, 2011, Beijing, China.
ACM 978-1-4503-0630-0/11/09.

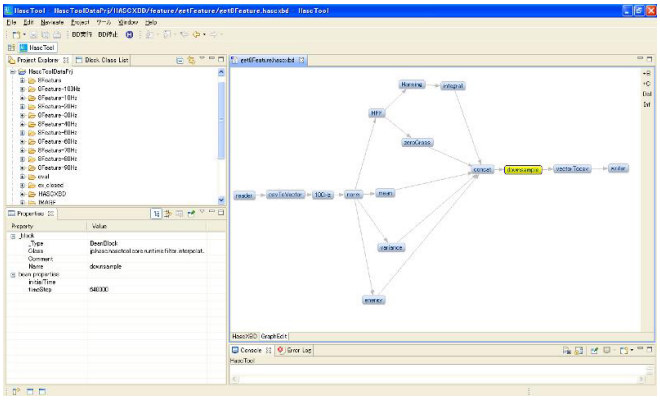

Figure 1. Screen shot of the HASC Tool

In this paper, we propose a distributed computing using HASC [1] Tool. HASC tool is a data processing tool developed for HASC Challenge.

HASC TOOL

To boost the data handling and trial-and-error process of the signal processing, we have developed a tool named "HASC Tool[3]." Figure 1 shows the screen shot of HASC Tool. HASC Tool is developed with Java and based on the famous IDE called Eclipse RCP. HASC Tool is open source software, so one can easily modify or improve the system. HASC Tool has following features.

-Showing accelerometer signals and label data

-Create a process block diagram graph called "XBD." By using "XBD," one can easily automate the various signal processing and file processing. Without this kind of automation, handling thousands of files is not easy.

-Real time / offline data acquisition with wireless sensors

-Integration with Weka Toolkit[5]

By using HASC Tool, we can exchange the process of activity recognition using XBD files.

Design choice on the distributed computing

Most of the distributed computing system requires some kind of special programming. For example, Hadoop[5] requires Map-Reduce style programming. However, we are trying to build a GUI-friendly signal processing tool, so we prefer not to require additional programming.

[1] HASC: Human Activity Sensing Consortium

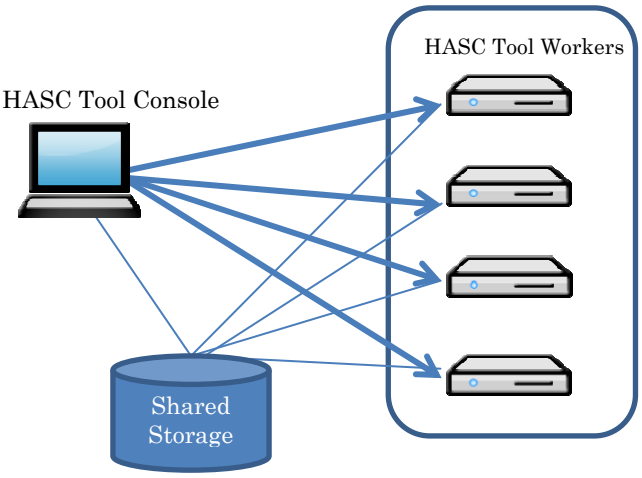

Figure 2. Distributed Data Processing using HASC Tool

Distribution of the message with shared storage

As shown in Figure 2, we finally fixed the style of distributed data processing on HASC Tool. In this distributed computing system, we require shared storage among the HASC console and workers. It can be distributed file system such as GFarm2[7], so there is no worry about bottlenecks on the file system. In HASC Tool, data processing can be described in XBD file. Each block in XBD can be hierarchically organized in the XBD file. Blocks are connected with "HASC Tool Message". In a single computer system, these messages are passed only within the system. For the distributed computing system, we add the functionality to pass the message through the network. Message for sub-XBD can contain the target file path. So the HASC Tool console can order to the each worker to process the file with the XBD.

Visualization of the distributed processing

It is very important to visualize the state of the system. In the HASC Tool console, we add the functionality to visualize the current status of the distributed process. In Figure 3, each small box shows the target files and the colored box show the processed files.

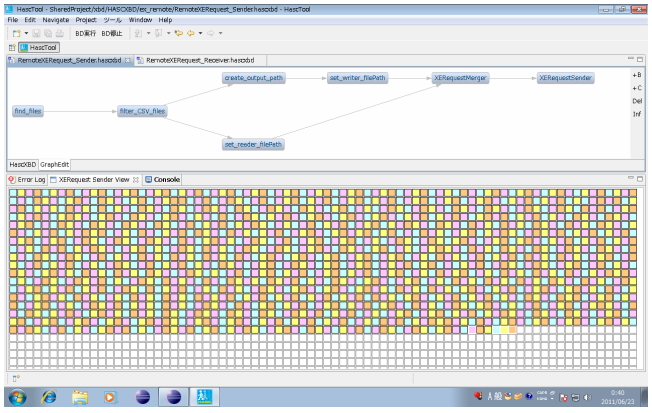

Figure 3. Visualization of the Distributed Data Processing

EVALUATION

To evaluate the effect of the distributed data processing, we have performed a simple evaluation with HASC2011corpus. We have calculated 7 features (mean, variance, zero cross rate, FFT with 4band) from 87 subjects as the part of training phase [2]. In Table 1, we show the result that clearly shows the effects of the number of workers.

Number of Workers	Processed time (sec)
1	5440
2	2616
3	1744
4	1296

Table 1. Result on the effect of number of workers.

CONCLUSION AND FUTURE WORKS

In this paper, we present our powerful data processing tool HASC Tool. By adding distributed data processing function into HASC Tool, one can easily utilize HASC tool from single computer to the large cluster.

ACKNOWLEDGMENTS

This work is partially supported by MEXT Grand-in-Aid for Scientific Research (A) 23240014 and JST CREST Kashin-Free project.

REFERENCES

1. Nobuo Kawaguchi, Nobuhiro Ogawa, Yohei Iwasaki, Katsuhiko Kaji, Tsutomu Terada, Kazuya Murao, Sozo Inoue, Yoshihiro Kawahara, Yasuyuki Sumi and Nobuhiko Nishio, HASC Challenge: Gathering Large Scale Human Activity Corpus for the Real-World Activity Understandings, *ACM Augmented Human (AH2011)*, pp. 27:1-27:5 (2011).
2. Nobuhiro Ogawa, Katsuhiko Kaji, Nobuo Kawaguchi, Effects of Number of Subjects on Activity Recognition - Findings from HASC2010corpus -, *International Workshop on Frontiers in Activity Recognition using Pervasive Sensing(IWFAR2011),* pp. 48-51(2011).
3. HASC Tool Project Website: http://sourceforge.jp/projects/hasc/
4. Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, Ian H. Witten, The WEKA Data Mining Software: An Update. *SIGKDD Explorations, Volume 11, Issue 1.*(2009).
5. Hadoop.: http://hadoop.apache.org/
6. Osamu Tatebe, Kohei Hiraga, Noriyuki Soda, "Gfarm Grid File System", New Generation Computing, Ohmsha, Ltd. and Springer, Vol. 28, No. 3, pp.257-275, (2010).