NEWS
OF THENATIONAL ACADEMY OF SCIENCES OF THE REPUBLIC OF KAZAKHSTAN
PHYSICO-MATHEMATICAL SERIES
ISSN 1991-346X **https://doi.org/10.32014/2020.2518-1726.66**
Volume 4, Number 332 (2020), 61 – 67

УДК 004.89-004.31
МРНТИ 28.23.33

Y.T. Kozhagulov, D.M. Zhexebay, S.A. Sarmanbetov, A.A. Sagatbayeva, D. Zholdas

Al-Farabi Kazakh National University, Almaty, Kazakhstan.
Email: kaznu.kz@gmail.com, zhexebay92@gmail.com, sarmanbetov.sanzhar@gmail.com, sagalua95@gmail.com, dauletzholdas@mail.ru

COMPARATIVE ANALYSIS OF OBJECT DETECTION PROCESSING SPEED ON THE BASIS OF NEUROPROCESSORS AND NEUROACCELERATORS

Abstract. This paper is devoted to a comparative analysis of neural network models based on neuroprocessors. The following neural network models were selected: MobileNetSSD v1, SSD MobileNet v2. As a research task, the authors determined an attempt to compare several platforms that differ in size, computational capabilities and cost: Coral Dev Board, NVIDIA Jetson Nano, Coral USB Accelerator, Neural Compute Stick 2, Raspberry Pi 4. Local data processing offers a number of advantages compared to downloading calculations to a remote server or data center. Firstly, downloading data to remote servers takes a lot of time, as well as additional costs for infrastructure with energy, financial and computer equipment. It also requires high bandwidth and reliability, as data transfer may not be completed in case of a bad signal. Secondly, data transfer can lead to security and privacy issues. Finally, local processing can reduce the amount of data transferred to the cloud, which allows to performing tasks of a higher level.

The aim of this paper is a comparative analysis of platforms for object recognition tasks (object detection) with MobileNetSSD v1 / v2 models. For training, a cloud service based on the Jupyter Notebook, which gives access to incredibly fast GPUs and TPUs was used. The paper addresses the topic of determining small objects using the example of car detection.

Based on the study of platforms and models, it was found that the MobileNetSSD v1 model is effective for NVIDIA Jetson Nano by NVIDIA (61FPS), but in turn, the MobileNet v2 SSD model is less efficient (11FPS). Google's Coral Dev Board is more productive than other devices (47.8FPS and 63FPS). Raspberry Pi 4 (0.8FPS and 1.4FPS) turned out to be less effective. Among neuroprocessors, Neural Compute Stick 2 (9FPS and 7.1FPS) showed poor performance.

Key words: convolutional neural network, neuroaccelerators, neuroprocessor, object detection, deep learning.

1. Introduction

In modern conditions, the tasks of intelligent image processing are relevant, so the question arises of choosing a hardware platform with minimal power consumption, small size and high computing power [1-3]. There are a number of neural network models for object detection: Single Shot Detector (SSD), Faster Region-based Convolutional Neural Networks (R-CNN) and Region based Fully Convolutional Networks (R-FCN) [4]. These models, combined with various models such as VGG, Resnet101 and Mobilenet, determine the optimal combination of performance and speed. The combination of Faster R-CNN with Resnet-101 provides the best accuracy for object definition with 33.7% average accuracy and 396 ms of time in the GPU. The SSD-MobileNet model provides 19% and 40 ms in the GPU.

Other studies are aimed at improving the available models. For example, in [5], the performance of the traditional Faster R-CNN method is optimized by connecting blocks to the Fast R-CNN model, this new network is called RF-RCNN. The traditional Faster R-CNN method has an accuracy of about 61%, and with RF-RCNN this value improves to 75%.

The paper [6] describes the analysis in terms of the power and computational capabilities of the CNN-based model for an object detection system. The implementation uses a platform for embedded devices with support for a graphics processor (GPU), and the results are compared with a traditional platform based on a personal computer (PC).

In [7], the Movidius Neural Computer Stick is used to implement real-time object detection systems on the Raspberry Pi 3B. The results show that Movidius reaches 3.5 FPS. In most cases, detection models are used in modern computers with a GPU, the reason is that models such as Faster R-CNN, R-FCN or GoogleLenet require high processing, which cannot be satisfied with the built-in device. The most suitable deployment model with embedded devices and additional graphics processing elements is SSD-Mobilnet, which is strictly designed for devices with low performance. And also, the MobileNet SSD is an advanced convolution network architecture (model) that allows for fast recognition [8-9]. MobileNet SSDs are 20 times faster than their peers. The aim of this work is a comparative analysis of the computing capabilities of neuroprocessors and neuroaccelerators such as Coral Dev Board, NVIDIA Jetson Nano, Coral USB Accelerator, Movidus Neural Compute.

2. Hardware testing platforms

Platform NVIDIA Jetson Nano single-board computer for computing in the field of Artificial Intelligence (AI). A small computer with CUDA-X AI library support delivers 472 gigaflops to run modern AI workloads. The solution expands the capabilities of developers in terms of creating IoT applications, including for entry-level DVRs, home robots and smart gateways with analytical capabilities.

Figure 1 - NVIDIA Jetson Nano

Table 1 - NVIDIA Jetson Nano Features

Parameter	Value
CPU	Quad-core ARM A57 @ 1.43 GHz
GPU	128-core Maxwell
RAM	4 GB 64-bit LPDDR4 25.6 GB/s
Cost	100$

Intel® Neural Compute Stick 2 is a plug-and-play development kit for AI. The module can work without connecting to cloud technologies and allows you to create prototypes using inexpensive end devices such as Raspberry Pi4, etc.

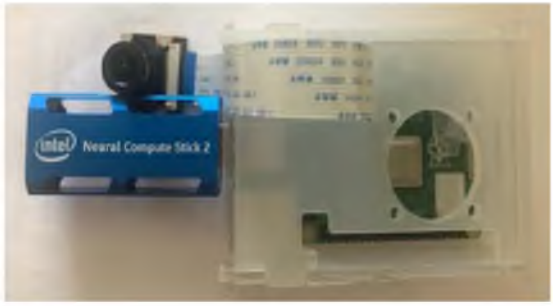


Figure 2 - Intel® Neural Compute Stick 2+Rb Pi4

Table 2 - Intel® Neural Compute Stick 2 Specifications

Parameter	Value
VPU	Intel Movidius Myriad X
Power supply	USB 3.0 Type-A
Sizes	72,5 x 27 x 14 mm
Cost	100$

Coral USB Accelerator is a specialized ASIC developed by Google, which is considered a lightweight version of the TPU (Tensor Processing Unit) provided as part of cloud services for training neural networks.

Coral USB Accelerator supports two different operating modes: at standard frequency and maximum frequency. At maximum frequency, output performance grows in two.

Figure 3 - Coral USB Accelerator+Rb Pi4

Table 3 - Coral USB Accelerator Specifications

Parameter	Value
ML accelerator	Google Edge TPU coprocessor
Connector	USB 3.0 Type-C* (data/power)
Sizes	65 mm x 30 mm
Cost	75 $

Coral Dev Board is a single-board computer that is ideal for quickly performing machine learning (ML) operations in a small form factor (technical product standard). You can use Dev Board to prototype your embedded system and then scale it to production using the integrated Coral System-on-Module (SoM) system in combination with custom PCB hardware.

Figure 4 - Coral Dev Board

Table 4 - Coral Dev Board Specifications

Parameter	Value
CPU	NXP i.MX 8M SoC (quad Cortex-A53, Cortex-M4F)
GPU	Integrated GC7000 Lite Graphics
ML accelerator	Google Edge TPU coprocessor
RAM	1 GB LPDDR4
Flash memory	8 GB eMMC
Sizes	48mm x 40mm x 5mm
Cost	150 $

The Raspberry Pi 4 is a single-board computer the size of a bank card, originally developed as a budget system for teaching computer science, but later gaining wider application and fame.

Figure 5 - Raspberry Pi 4

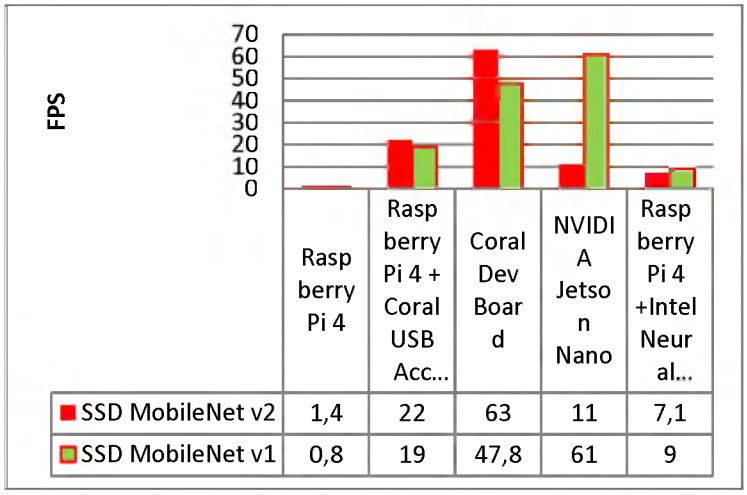
Table 5 - Raspberry Pi 4 Specifications

Parameter	Value
SoC	Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
GPU	VideoCore VI c OpenGL ES
Sizes	88 x 58 mm.
Cost	35$

3. The results of a comparative analysis of neuroprocessors

The following neural network models were selected as objects for testing: MobileNetSSD v1, SSD MobileNet v2. Models are trained on common objects in the Common Objects in Context (COCO) database of a data set. For training, we used a cloud service based on the Jupyter Notebook, which gives access to incredibly fast GPUs and TPUs.

Graph 1 below shows the results of a study on the processing speed of data on single-board platforms.



Graph 1 - The result of testing platforms in FPS (the number of frames per second)

The diagram shows that the MobileNetSSD v1 model is an order of magnitude faster with the NVIDIA Jetson Nano (61FPS). The results obtained were tested for the task of recognizing the flow of cars in real time (Figure 6).

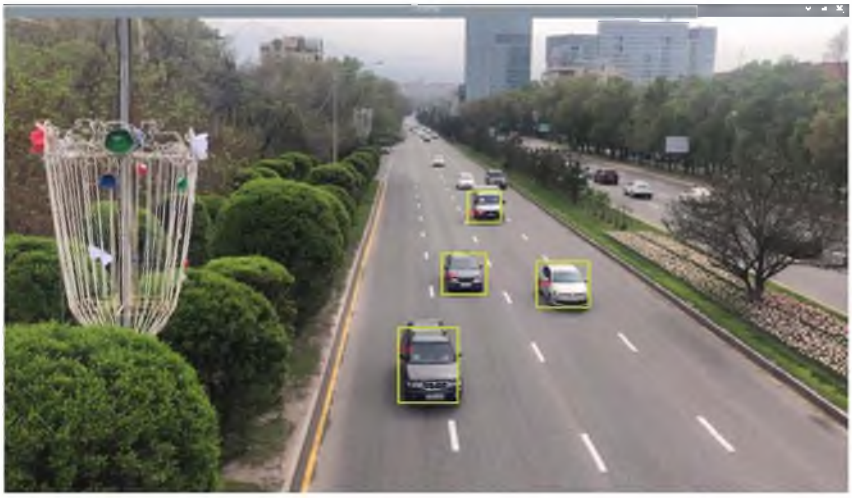


Figure 6 - Test result of MobileNet v1 / v2 SSD models in FPS (frames per second) trained using the COCO dataset

Based on the result (Figure 6), it can be seen that the detector missed small cars. The solution to this problem is to increase the resolution of input images. Vehicles that are far away are not a problem for the model. As the approach getting closer, models will be able to recognize their presence. The second problem that may encounter is that the models do not distinguish between the "front" and "frontal" type of vehicle. These incorrect classifications are partly due to the fact that the front and rear parts of vehicles have many visually similar characteristics. Despite this, MobileNet SSDs achieve high recognition accuracy.

4. Conclusion

The results of this paper show that the MobileNetSSD v1 model is effective for NVIDIA Jetson Nano from NVIDIA (61FPS), however, this device for the MobileNet v2 SSD model showed a low performance (11FPS) compared to other devices. Google's Coral Dev Board is more productive than NVIDIA Jetson Nano, whose results for neural network models are 47.8FPS and 63FPS, respectively. The Raspberry Pi 4 (0.8FPS and 1.4FPS) turned out to be not effective, since the platform does not have a built-in neuroprocessor and a neuro accelerator. Among neuroprocessors, Neural Compute Stick 2 (9FPS and 7.1FPS) showed poor performance.

Е.Т. Кожагулов, Д.М. Жексебай, С.А. Сарманбетов, А.А. Сағатбаева, Д. Жолдас

әл-Фараби атындағы Қазақ Ұлттық Университеті, Алматы, Казахстан

НЕЙРОПРОЦЕССОРЛАР МЕН НЕЙРОЖЫЛДАМДАТҚЫШТАРДЫҢ БАЗАСЫНДА ОБЪЕКТІЛЕРДІ АНЫҚТАУ ЖЫЛДАМДЫҒЫН САЛЫСТЫРМАЛЫ ТАЛДАУ

Аннотация. Ақпараттық технологияға деген сұраныстың артуы нейрондық желілер мен алгоритмдердің дамуына әкелді. Соңғы жылдары зерттеушілердің қызығушылығын арттырған машиналық оқыту алгоритмдері, соның ішінде, нейрондық желілер басты назарға ілігін отыр. Нейрондық желі алгоритмдеріне кескінді өңдеу, мәтіндік тілді өңдеу, мәліметтерді талдау және т.б. жатады, нейрондық желі көптеген салаларда жоғары нәтижелерге қол жеткізеді. Бұл жетістіктер зерттеушілерді адам өмірінің барлық салаларында нейрондық алгоритмдерді қолдануға шабыттандырады. Бүгінгі таңда зияткерлік бейнені өңдеу міндеттері өзекті болып отыр.

Жұмыс нейропроңессорлар неғізінде нейрондық желі моделдерін салыстырмалы түрде талдауға арналған. Зерттеуге нейрондық желінің келесі моделдері таңдалды: MobileNetSSD v1, SSD MobileNet v2.

Жұмыстық мақсаты MobileNetSSD v1/v2 модельдері мен объектілерді анықтауға арналған алаңдарды салыстырмалы түрде талдау. Моделдер жалпы нысандардағы Common Objects in Context (COCO) базасында оқытылды. Оқыту үшін біз Jupyter Notebook-ке неғізделген қызметті қолдандық, ол GPU мен TPU-ға жылдам қол жеткізуе мүмкіндік береді. Жұмыс автомобильдерді анықтау мысалын қолдана отырып, кішкентай заттарды анықтауда қарастырылған.

Көзделген мақсатқа қол жеткізу үшін келесі міндеттер қойылды:

Coral Dev Board, NVIDIA Jetson Nano, Coral USB Accelerator, Neural Compute Stick 2, Raspberry Pi 4 платформаларын салыстыру. Бір платалы процессорлар - бұл қол жетімді бағамен машиналық оқытудың озық үлгілерін орналастыруға мүмкіндік беретін керемет құралдар болып табылады. Jetson Nano - бұл толық функционалдығы бар кішкентай Linux компьютері, оны бағдарламалық жасақтаманы қолдану тұргысынан икемді етеді. Ол TensorFlow, Caffe, PyTorch, Keras және MXNet сияқты машиналарды оқытудың барлық орталарымен жұмыс істей алады. Салыстыру үшін, Coral Dev Board, Coral USB Accelerator, Neural Compute Stick 2, Raspberry Pi - Jetson Nano сияқты фреймворктарды қолдану тұргысынан икемді емес, өйткені аппараттық архитектурасына сәйкес келетін арнайы форматтагы модельдерді гана қолдана алады. Сонымен жогарыда атылып кеткен платформалар өздерінің шектеулі мүмкіндіктеріне қарамастан оларды машиналық оқытуда пайдалануда тиімді ететін бірнеше факторлар бар. Шағын қуатты тұтыну арқасында кірістірілген жүйелер деректерді жинау орнында өңдеуге мүмкіндік береді, мысалы ретінде IoT құрылгысын, роботтарды, автономды автомобиль немесе дрондарды жатқызсақ болады. Жергілікті деректерді өңдеу қашықтагы серверге немесе деректерді өңдеу орталыгына есептеулерді жүктеумен салыстырганда бірқатар артықшылықтарды береді. Біріншіден деректерді қашықтагы серверлерге жүктеу үлкен кідірістер алады, сонымен қатар энергетикалық, қаржылық және есептеу техникаларымен инфраструктураға қосымша шығындар алып келеді. Сондай-ақ, бұл жогары өткізу қабілеттілігі мен сенімділікті қажет етеді, өйткені нашар сигнал болган жағдайда деректерді жіберу аяқталмай қалуы мүмкін. Екіншіден, деректерді жіберу қауінсіздік пен құпиялылық мәселелеріне алып келуі мүмкін. Соңында, жергілікті өңдеу бұлтқа берілетін деректер көлемін азайтуы мүмкін, бұл оган жогары децгейдегі тапсырмаларды орындауга мүмкіндік береді.

Олардың есептеу қабілеттері мен құны бойынша ерекшеліктерін бірнеше платформаларды салыстыра отырып анықтау;

Платформалар мен модельдерді зерттеу негізінде NVIDIA Jetson Nano NVIDIA-ден (61FPS) үшін MobileNetSSD v1 моделі тиімді екені анықталды, бірақ SSD MobileNet v2 моделі MobileNetSSD v1 моделімен салыстырганда тімділігі төмен. Google Coral Dev Board басқа құрылғыларга қараганда тиімді (47,8FPS және 63FPS). Ал Raspberry Pi 4 (0,8FPS и 1,4FPS) тінмділігі төмен. Сонымен қатар Neural Compute Stick 2 (9FPS и 7,1FPS) моделдері нейропроцессорлардың арасында төменгі көрсеткіштер көрсетті.

Түйін сөздер: жинақталған нейрондық желі, нейроүдеткіштер, нейропроцессор, объектіні анықтау, тереңдетін оқыту.

Е.Т. Кожагулов, Д.М. Жексебай, С.А. Сарманбетов, А.А. Сағатбаева, Д. Жолдас

Казахский национальный университет имени аль-Фараби, Алматы, Казахстан.

СРАВНИТЕЛЬНЫЙ АНАЛИЗ СКОРОСТИ ОБРАБОТКИ ОБНАРУЖЕНИЯ ОБЪЕКТОВ НА БАЗЕ НЕЙРОПРОЦЕССОРОВ И НЕЙРОУСКОРИТЕЛЕЙ

Аннотация. Быстрый рост данных, внедрение нейронных сетей и появление различных технологий, ускоряющих процесс обучения, привели к разработке нейропроцессоров и нейроускорителей. В последние годы алгоритмы машинного обучения привлекают большое внимание исследователей, это связано с необходимостью обработки большого массива данных. Алгоритмы нейронной сети включают обработку изображений, обработку естественного языка, анализ данных и т. д. достигает высоких результатов во многих областях. Эти достижения вдохновили исследователей на применение алгоритмов нейронных сетей в сложных областях человеческой жизни, требующей использование нетрадиционных алгоритмов аналитических вычислений. На сегоднящний день задачи интеллектуальной обработки изображений актуальны.

Работа посвящена сравнительному анализу нейросетевых моделей на базе нейропроцессоров. Были выбраны следующие нейросетевые модели: MobileNetSSD v1, SSD MobileNet v2. В качестве исследовательской задачи авторами была определена попытка сравненить несколько платформ, которые отличаются по размеру, вычислительным возможностям и стоимости: Coral Dev Board, NVIDIA Jetson Nano, Coral USB Accelerator, Neural Compute Stick 2, Raspberry Pi 4. Из-за низкого энергопотребления встроенные системы позволяют обрабатывать данные в точке сбора, например, устройства IoT, роботы, автономные транспортные средства или дроны. Локальная обработка данных предлагает ряд преимуществ по сравнению с загрузкой вычислений на удаленный сервер или в центр обработки данных. Во-первых, загрузка данных на удаленные серверы занимает много времени, а также дополнительные расходы на инфраструктуру с энергетическим, финансовым и компьютерным оборудованием. Это также требует высокой пропускной способности и надежности, поскольку передача данных может быть не завершена в случае плохого сигнала. Во-вторых, передача данных может привести к проблемам безопасности и конфиденциальности. Наконец,

локальная обработка может уменьшить объем данных, передаваемых в облако, что позволяет выполнять задачи более высокого уровня. Одноплатные процессоры являются отличными инструментами, которые позволяют развертывать передовые модели машинного обучения по доступной цене. Jetson Nano - это небольшой полнофункциональный компьютер Linux, который делает его гибким в плане использования программного обеспечения. Он может работать со всеми средами машинного обучения, такими как TensorFlow, Caffe, PyTorch, Keras и MXNet. Для сравнения, он является негибким с точки зрения использования таких платформ, как Coral Dev Board, Coral USB Accelerator, Neural Compute Stick 2, Raspberry Pi -Jetson Nano, поскольку может использовать модели только в специальных форматах, соответствующих аппаратной архитектуре.

Целью работы является сравнительный анализ платформ для задач распознавания объектов (object detection) с моделями MobileNetSSD v1/v2. В сатье модели обучены на общие объекты в базе Common Objects in Context (COCO) набора данных. Для обучения использовалась облачный сервис на основе Jupyter Notebook, которая дает доступ к невероятно быстрым GPU и TPU. В работе затрагивается тема определения маленьких объектов на примере детектирования автомобилей.

На основе изучения платформ и моделей установлено ,что модель MobileNetSSD v1 эффективна для NVIDIA Jetson Nano от NVIDIA (61FPS), но в свою очередь модель SSD MobileNet v2 менее эффективна (11FPS). Coral Dev Board от Google является более производительнее чем другие устройства (47,8FPS и 63FPS). Менее эффективным оказался Raspberry Pi 4 (0,8FPS и 1,4FPS). Среди нейропроцессоров низкую производительность показал Neural Compute Stick 2 (9FPS и 7,1FPS).

Ключевые слова: сверточная нейронная сеть, нейроускорители, нейропроцессор, обнаружение объектов, глубокое обучение.

Information about authors:

Kozhagulov YE., Lead researcher, PhD, Lecturer of Department of Physics and Technology, Al-Farabi Kazakh National University, Almaty, Kazakhstan, kazgu.kz@gmail.com ,https://orcid.org/0000-0001-5714-832X;

Zhexebay D., PhD candidate, zhexebay92@gmail.com, https://orcid.org/0000-0002-3974-0896;

Sarmanbetov S., Master of Natural Sciences,sarmanbetov.sanzhar@gmail.com, https://orcid.org/0000-0003-1749-2163;

Sagatbayeva A., Master of Engineering and Technology,sagalua95@gmail.com, https://orcid.org/0000-0002-8601-8900;

Zholdas D., Student,dauletzholdas@mail.ru, https://orcid.org/0000-0001-5183-8354;

REFERENCES

[1] Barthélemy J., Verstaevel N., Forehead H., Perez P. Edge-computing video analytics for real-time traffic monitoring in a smart city //Sensors. 2019. Vol. 19, №. 9. P. 2048.

[2] Zhong Q., Li C., Zhang Y., Xie D., Yang S., Pu S. Cascade region proposal and global context for deep object detection //Neurocomputing. 2019.

[3] Antonini M., Vu T.H., Min C., Montanari A., Mathur A., Kawsar F. Resource Characterisation of Personal-Scale Sensing Models on Edge Accelerators //Proceedings of the First International Workshop on Challenges in Artificial Intelligence and Machine Learning for Internet of Things. 2019. P. 49-55.

[4] Nikhil Yadav U. B., "Comparative study of object detection algorithms," in International Research Journal of Engineering and Technology (IRJET), May 2017, pp. 586–591.

[5] M. Roh and J. Lee, "Refining faster-rcnn for accurate object detection," in 2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA), May 2017, pp. 514–517.

[6] Z. Chen, T. Ellis, and S. A. Velastin, "Vehicle detection, tracking and classification in urban traffic," in 2012 15th International IEEE Conference on Intelligent Transportation Systems, Sept 2012, pp. 951– 956.

[7] R. H. Pea-Gonzlez and M. A. Nuo-Maganda, "Computer vision based real-time vehicle tracking and classification system," in 2014 IEEE 57th International Midwest Symposium on Circuits and Systems (MWSCAS), Aug 2014, pp. 679–682.

[8] Nicholas D Lane. 2019. EmBench: Quantifying Performance Variations of Deep.

Neural Networks across Modern Commodity Devices. In The 3rd InternationalWorkshop on Deep Learning for Mobile Systems and Applications. ACM, 1–6.

[9] Saisakul Chernbumroong, Shuang Cang, Anthony Atkins, and Hongnian Yu. 2013. Elderly activities recognition and classification for applications in assisted living. Expert Systems with Applications 40, 5 (2013).