Smart Nutrition Monitoring System Using Heterogeneous Internet of Things Platform

Bahman Javadi[1], Rodrigo N. Calheiros[1], Kenan M. Matawie[1], Athula Ginige[1], Amelia Cook[2]

[1] School of Computing, Engineering and Mathematics
[2] School of Science and Health
Western Sydney University, Australia
b.javadi@westernsydney.edu.au

Abstract. Poor nutrition impairs the health and wellbeing of the population and increases the risk of chronic diseases such as obesity and type 2 diabetes. Chronic diseases that require dietary management can be better managed if food and nutrition intake is monitored. Existing methods for measurement are inaccurate and not scalable as they are based on a person's ability to recall and self-report. In this paper, we propose a smart nutrition monitoring system based on Internet of Things (IoT) technologies to collect reliable nutrition intake data from heterogeneous sensors. The proposed method is non-invasive and consists of a combination of data sources from heterogeneous devices to increase accuracy. The system architecture is based on emerging Fog Computing concepts where data collection points are able to do the preprocessing and lightweight analytics before sending data to the Cloud. The system prototype is developed using various sensors including cameras to generate 3D images for food volume estimation.

1 Introduction

Healthy eating has a great impact on the lifestyle quality and can prevent diet-related diseases such as diabetes, obesity and cardiovascular disease[1]. Research data from the Institute for Health Metrics and Evaluation (IHME)[3] in 2015 revealed that dietary risks are the leading cause of disease burden in Australia and contributed to more health loss than smoking, alcohol, and drug use. There are also significant economic costs associated with diet-related chronic diseases. For instance, the estimated financial cost for type 2 diabetes alone is more than $14 billion per year in Australia[4]. This cost is about $200 billion per year in USA for obesity [2].

The ability to manage food and nutrition is vital to a healthy and productive society. Self-monitoring food intake, with the provision of automated and tailored feedback, is an effective strategy to build awareness of one's current diet

[3] http://www.healthdata.org/
[4] https://www.diabetesaustralia.com.au/

and eating patterns. This would require a smart nutrition monitoring system. Traditional dietary assessment methods (e.g., 24 hour recalls, food frequency questionnaires) are either unable to provide individual level feedback with the necessary precision, or they are impractical for self-monitoring, largely due to participant burden and low scalability [3]. More advanced approaches for food intake monitoring are based on smart phones [4], but still they rely heavily on user memory and recall, which is not practical for many, especially for people with memory disorders.

Recently, many researchers have attempted to address these issues using wearable sensor devices [5]. The main advantage of this approach is removing the burden of self-report. Although there are several studies to adopt these sensors to monitor food intake, most of them have been tested in laboratory conditions for limited number of participants [6]. Moreover, almost all of them are based on a single sensor, which limits the accuracy of data collection and consequently data reliability. The average accuracy for nutrition intake using these sensors is about 90% in laboratory environments [7]. So an open challenge is to find a non-invasive solution to monitor nutrition intake in real-life scenarios with a high degree of accuracy.

In this paper, we investigate the use of Internet of Things (IoT) technologies to collect nutrition intake data. The Internet of Things is an emerging technology consisting of several heterogeneous devices that can exchange data without human interaction [8]. This will enable non-invasive measurement of food intake by collecting data from different devices including wearable and environmental sensors. To limit the scope of this project, we only focus on take-away foods for young adults. We consider young adults (ages 18-35) as they are at a heightened risk of harmful effects of unhealthy "meals out", because they spend, proportionately, even more on the "Meals out and fast food" category than any other age group [9].

The rest of this paper is organized as follows. Related work is described in Section 2. In Section 3, we present the problem statement of smart nutrition monitoring system. Section 4 includes the detail of the proposed system model and its components. The system prototype and its implementation details are presented in Section 5. Conclusions and future work are presented in Section 6.

2 Related work

There are several research about measuring food and nutrition intake to manage and control the eating habit. There are two main approaches to address this challenge: manual and automatic techniques. Manual approaches are based on self-reports on paper [10], smart phones [9], or questionnaires [11]. In the paper-based methods, users need to provide the detail of the meals and beverages that they consumed during the day based on the personal recall. In a more advanced method, paper is replaced by smart phones so there will be a better user interface and faster response time from the dietitian or doctor. The last manual method is based on food frequency questionnaires where food consumption frequency is collected for a period of time (not for every single meal). All these manual methods are based on personal recall and might be imprecise. Moreover, these methods discourage people to continue recording due to lack of convenience and on average only 15% of the participants complete the program [12].

In order to address the issues in manual techniques, automatic methods based on environmental and wearable sensors are proposed [5]. The main advantage of the automatic approach is removing the burden of self-report. For instance, using environmental sensor such as cameras, automatic food intake detection from captured images of human faces during the eating process can be achieved [13]. However, this technique is not able to detect specific food types or food composition. Similar techniques based on motion sensors or pressure sensors have similar drawbacks while they have restriction to be used in free-living populations [14].

Automatic methods based on wearable sensors could be potential solutions as they do not rely on user's input and provide real time food intake monitoring. These sensors adapt various detection techniques such as chewing [15], swallowing [16] and wrist motion [17]. Although there are several studies to adopt these sensors to monitor food intake, most of them have been tested in laboratory conditions for limited number of participants [6]. The average accuracy for nutrition intake using these sensors is about 90% in laboratory environments [7]. Moreover, almost all of them are based on a single sensor, which limits the accuracy of data collection and consequently data reliability. So an open challenge is to find a non-invasive solution to monitor nutrition intake in real-life scenarios with a high degree of accuracy.

Because of the aforementioned limitations in existing methods for tracking nutrition intake, in this paper we investigate the use of Internet of Things (IoT) and Fog computing-based technologies to collect nutrition intake data from heterogeneous data sources. While using wearable sensors seem initially a better approach, they have many limitations in real-life scenarios, so our approach is based on environmental sensors. The approach is detailed in the next sections.

3 Smart Nutrition Monitoring System

We devise a Nutrition Smart Monitoring System that utilizes IoT, Fog computing, and hierarchical data analytics to provide an accurate understanding of the dietary habits of young adults, which can be used by users themselves as a motivator for change behavior and by dietitians to provide better guidance to their patients.

One of the main objectives when designing the system was *minimizing the amount of direct input and actions from users*, what may motivate continuous utilization of the system. Another objective of the system was achieving the goals of accurate data collection with inexpensive IoT devices. To circumvent limitations of such devices, we envision that *multiple and heterogeneous IoT devices will be used in a single data collection* and that statistical analysis will be carried out in the background to conciliate the information and increase accuracy. Finally, we aimed at leveraging emerging Fog Computing [18] capabilities in the architecture. By Fog Computing, we mean the capability of enabling some computation to occur in the edge of the network and the rest in the cloud. This decision is motivated by key capabilities of Fog Computing such as reduced communication latency (by enabling local computation) while taking advantage of the higher scale of cloud resources for heavier data analysis and computation.

The proposed smart nutrition monitoring system is composed of a kiosk where diverse sensors are installed. Since we are targeting take-away foods, this should be designed in a way that users can access that in the same place that they buy the food (e.g., restaurants). This kiosk will be equipped with various IoT sensors to collect weight, volume and structure (e.g., molecular pattern) of the food. The only action required from users is to authenticate with the kiosk (via a mobile app) and deposit the food in the kiosk for a couple of seconds while relevant information is obtained by the sensors. Once data is obtained, users can cease interaction with the kiosk and can proceed with their daily activities. Therefore, the data collection will be done with a non-invasive technique where the user does not need to enter any information about the food. The kiosk also has a built-in controller as part of Fog Computing system to process and communicate the collected data with other components of the system.

Data collected by the kiosks are sent to cloud servers where it is stored and processed. In a later time, reports and charts can be generated to the user (and optionally to their dietitian, if the user is the patient of a dietitian and wishes to share the information). Users and dietitians are the two actors in our envisioned system. Nevertheless, only users generate input data; dietitians can only generate reports and visualize their patients' information.

4 System Model and Architecture

Figure 1 presents the architecture of the Smart Nutrition Monitoring System. It is composed of two parts: the data collection points (kiosks) that analyze the presented food by user and the engine that carries out the relevant data storage, processing, reporting, and visualization. The kiosk has processing capabilities that enables it to play the role of edge device in a Fog Computing environment. Our system is complemented by two external components: cloud servers for data processing and an external database able to provide nutrition information about the food. Each component is discussed in the rest of this section. Implementation details of each component are discussed in Section 5.

4.1 Data Collection Points

The main interface between users and the system to enter information are the data collection points (i.e., kiosks). Users present the food in the kiosk and authenticate with the system via a mobile app. The same app triggers the process of activating the sensors so data is collected and send to the relevant systems.

Kiosks are designed to be modular: different kiosks can have different sensors in a given time. This gives flexibility to expand the units when new technologies

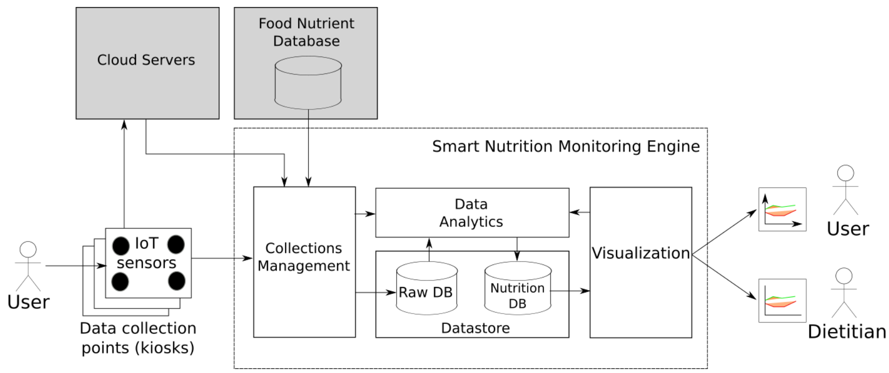

Fig. 1. Architecture of the Smart Nutrition Monitoring System.

are available and also enable data collection to progress even if some sensors are faulty. Regardless the particular sensors in operation, the concept behind kiosks is the following: kiosks are box-like structures that are large enough to accommodate a dish with food and sensors.

Data is collected by the heterogeneous sensors and aggregated before being transmitted. Cameras located in the kiosk also capture photos from the food from different angles and transmit them to the Cloud servers to generate a 3D model of the food, which will be used for food volume estimation.

4.2 Smart Nutrition Monitoring Engine

The Smart Nutrition Monitoring Engine has four components that are responsible for different aspects of the system, as follows.

Collections Management. This component provides an API that is accessed by the kiosks to upload food intake data and by the Cloud servers to return the results of computing-intensive operations (such as rendering of the 3D model of the food). In terms of functionality, it stores all the information in a raw database. The data that is stored is a combination of the information received by the other components and also nutritional value of the food that was presented. The latter is obtained from an external source which can be accessed, for example, via a nutrient database [5]. Another responsibility of the Collections Management is to request the Data Analytics module to perform specific operations when new data is collected from kiosks.

Data Analytics. This module is responsible for statistical analysis and machine learning activities in the architecture. This is used to generate reports and analyses that are relevant to users and dietitians and to identify the food that has been presented by users (and potentially its volume). Details of such analysis are discussed in Section 4.4.

[5] See, for example, `https://www.fatsecret.com`, which maintains such a database that can be accessed via RESTful APIs.

Datastore. This component stores both raw data (sourced from the user utilization of kiosks) and data that has been processed by the Data Analytics module.

Visualization. This module displays charts showing consumption of different nutrients over time and other forms of complex data analysis that are carried out by the Data Analytics module.

4.3 Cloud Servers

Cloud servers are adopted in the system for complex data processing such as 3D models rendering using multiple photos taken from different angles from the food deposited in the kiosk by users. The generation of the 3D model is a highly CPU-intensive application, and takes over 1 hour even for very simple samples.

Our architecture does not specify if the server capabilities are obtained via dedicated private clouds or by using public cloud services. In fact, the latter enables on-demand scaling to enable generation of multiple models in parallel when necessary. Regardless the choice, a strategy for adaptive provisioning and scheduling of cloud nodes for the architecture needs to be adopted, such as the approach proposed by Cai et al. [19]. Another alternative for this component is a hybrid approach, where a private cloud is used and public cloud nodes are added dynamically when the queue for model generation in the private cloud reaches a certain size. This is known in the literature as cloud burst, and there are solutions available for the problem [20, 21].

4.4 Data Analytics

Data collected for this project come from different sources and sensors applications, and can be complex and decomposite of different numerical and digital data. Integrating such multiple responses will greatly help to capture more accurate and valuable data that are useful to determine more significant statistical modeling and analysis that will serve different purposes in this area.

The automation of data collection will set the infrastructure and scale for machine learning tasks. This will include algorithms to integrate the data, detect objects, estimate volumes, and determine or estimate the associated nutritional composition values. Apart from visualization analysis and classification techniques used for object recognition, this analysis will involve various advanced statistical methodology such as Hidden Markov model, Support Vector machine, image and signal processing, clustering and prediction. Some of these techniques were used in wearable food intake monitoring, more details and review are given in [7].

The other part of this analytics is the accuracy and reliability of the collected data and the optimal form of the aggregated values. It is expected this will result in better and more significant predicted values that can be compared with standard nutritional calculations and tables. This will modify, update, and validate dietitians evaluation, interpretation, investigation, and accuracy of their values and recommendations. Daily and monthly food consumptions and other

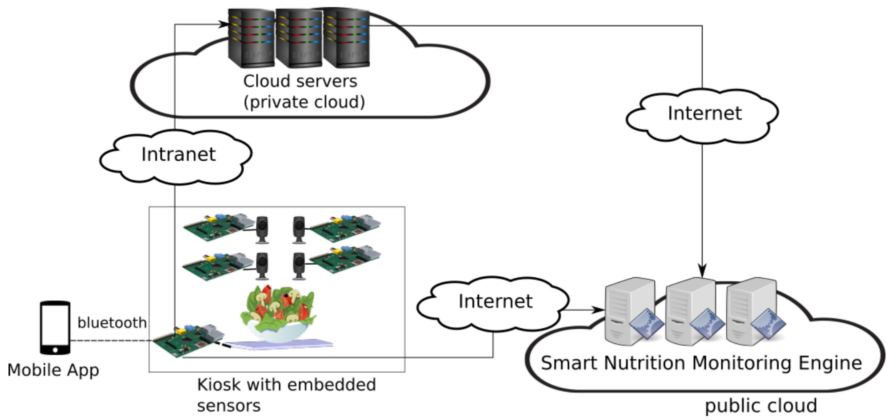

Fig. 2. Prototype of the Smart Nutrition Monitoring System.

predictions based on this approach can also be used for time-series analysis and prediction of food consumption.

5 System Prototype

To demonstrate the viability and feasibility of our approach, we have developed a prototype of the Smart Nutrition Monitoring System as shown in Figure 2. Details are discussed in the rest of this section.

5.1 Data Collection Points Prototype

The prototype version of the Data Collection Points, e.g., the Kiosk, utilizes Raspberry Pi 3 Model B boards (Quad Core 1.2GHz Broadcom BCM2837 CPU, 1GB of RAM) to interact with sensors and the rest of the architecture. There are 5 cameras with 8 megapixel resolution, each of which is attached to one Raspberry Pi. As the sensor device, we used the SITU Smart Scale[6], which is a smart food scale that communicates with other devices via Bluetooth. The vendor provides an app for iOS and an SDK that Android developers can use to develop their own apps. Besides the Raspberry Pis connected to cameras, there is an extra board that has a *Master* role. The Master is used to connect to the scale, receive the photos from the Raspberry Pis connected to cameras, and to interface with the architecture. To better integrate with the scale, the Master Raspberry Pi in our prototype had its Operating system replaced by emteria.OS[7] (Android-compatible Operating System that is optimized to run in Raspberry Pi 3).

[6] `http://situscale.com/`
[7] `https://emteria.com/`

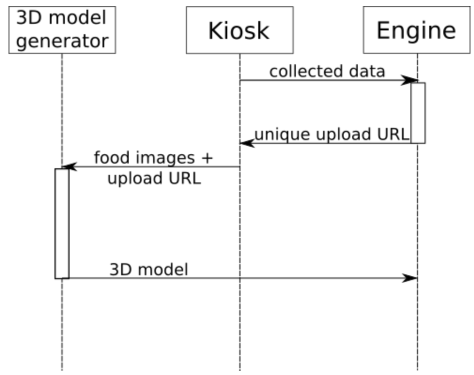

Fig. 3. Sequence diagram showing how the different components of the Smart Nutrition Monitoring System interact to enable capture and manage nutrition intake of users.

The process of information capture is triggered by users via a mobile app we developed. In this method, previously registered users of the system deposit the food dish in the kiosk, authenticate with the app, and tap a button on the app, which sends a message to a process running on the Master Raspberry Pi, indicating that the data collection process should start. The Master Raspberry Pi then collects the reading from the scale and from the other Raspberry Pis and sends all the relevant information to the Smart Nutrition Monitoring Engine via WiFi connection.

5.2 Smart Nutrition Monitoring Engine Prototype

The Smart Nutrition Monitoring Engine was designed with two key principles in mind (i) it should be built as a "cloud-first" application and (ii) it should leverage best practices in distributed software development.

To achieve these goals, the Engine's Collection Management module prototype has been developed in Java using the Jersey framework for RESTFul web services and is deployed on Tomcat 9.0. The Engine's Datastore module uses MySQL 5.7 and the Engine's Visualization module has been developed in ASP.Net. The external cloud servers used in the prototype consist of a computing cluster located at Western Sydney University and accessed via the local network by the kiosk.

5.3 Interaction between Components

Interaction with the Engine occurs via a RESTful API, and is depicted in Figure 3. Following best practices in the development of such APIs, this API is resource-centric: it is organized around the notion of a "collection" the represents a single use of the kiosk by a user to upload nutritional information.

When the kiosk is activated by the user (via the mobile app that also authenticates the user), it undertakes two actions: firstly, it uploads information

Table 1. HTTP status codes used in the Engine prototype.

HTTP Status Code	Use
`200 Ok`	Successful read or update of a collection
`201 Created`	A new food intake collection was received and is valid
`204 No Content`	Collection successfully deleted
`400 Bad Request`	Data uploaded cannot be successfully decoded in information that can be included in the database (due to missing information that are keys in the database)
`401 Unauthorized`	Attempt to connect without authentication
`404 Not Found`	The URL being accessed does not exist or an invalid operation for the URL was attempted

and requests the creation of a unique endpoint (URL) that can be used later to upload the 3D model of the food. This step occurs via a POST request to a specific endpoint. The unique URL generated as a response to the initial data upload is determined by the kiosk rather than by the engine. This is because the URL needs to be communicated to the cloud servers so it knows where to send the 3D food model once it is generated.

On successful upload of the information and creation of the unique endpoint, the Engine, via its Collections Management module, sends the photos for processing to the cluster alongside with the information about the URL where the model, once generated, needs to be uploaded. The successful creation of the unique URL address also enables that update, reading, and deletion of data related to this particular data collection can be carried out with PUT, GET, and DELETE HTTP operations, respectively.

Notice that the upload of the model is asynchronous with the rest of the data communication process. This is because of the huge difference in time scale in which the 3D model is created: while the rest of the data to be uploaded can be generated in seconds or milliseconds (depending on the sensors that collect the data and the latency to the cloud), the 3D model generator usually takes a few hours in servers with high CPU capacity.

Output of the operations are communicated via HTTP status codes. Table 1 lists the status codes and their meanings used in the prototype. The use of a generic `404 Not Found`, rather than a more specific return, when the URL being accessed does not exist or when an invalid operation for the URL is attempted, aims to reduce the amount of information given to malicious users accessing the platform. In particular, malicious users will not be able to know whether a specific endpoint exists or not if they try to access in the incorrect form (what is an evidence that the access is not occurring via the expected ways).

The API is designed to enable data collected from all the different sensors installed in a kiosk to be sent in a single HTTP message. In the JSON file sent by the kiosk in the body of the message, one of the tuples has as a value a list. Each element of the list represents the data collected by one sensor, in the form a JSON object that contains the sensor ID, sensor type, what is being measured, value, and unit of the data collected. If a single sensor generates more than one

```
{
"kioskId":"6947FA34B86",
"userId":"user@email.edu.au",
"scanId":"djklfj4980985fdsl",
"date":"13/09/2017",
"time":"15:30",
"items":[
{
"sensorId":"6874AB159",
"sensorType":"scale",
"metric":"weight"
"value":182.5,
"unit":"g"
},{
"sensorId":"C456D3450",
"sensorType":"composition",
"metric":"calories"
"value":397.5,
"unit":"kJ"}]
}
```


Fig. 4. Example of a JSON file representing the data being uploaded to the Engine.

data point, each data point becomes a different element (object) on the list, as shown in the code in Figure 4.

When designing our system, we aimed at following the principle of being permissive with the input received and rigorous on the output provided. This means that the output generated by the Engine and sent to clients conforms strictly with the expected formats; while great effort is put on receiving the input to extract as much relevant information as possible to enable successful data collection. This means that the Engine tolerates any missing field in the input that is not strictly required (i.e., which are not keys in the database tables), and only generates a `400 Bad Request` message if keys are missing or if the JSON file is malformed and thus relevant information cannot be extracted. One reason for tolerating missing fields is the fact that it is possible that in given moments some sensors in the kiosk might be faulty and may not generate output or generate meaningless output, and this should not compromise upload of measurements by non-faulty sensors in the same collection.

Besides data obtained from sensors in the kiosk, another source of data for the Engine is the external Food Nutrient Database. Our prototype utilizes the FatSecret database `https://www.fatsecret.com`, which is accessed via a RESTful API. The interaction with FatSecret is triggered when the Data Analytics module returns to the Collections Management Module a string with a food name (which can be the result of an analysis about the likely content of the food presented by the user). This name is used for a search in the FatSecret database (via the API) to determine nutrition facts about such food. This information is then stored in the Datastore.

All the nutrition data collected and stored in the database are used to generate daily, weekly, and monthly charts of intake of different nutrients and calories, for dietitians and users. Dietitians can only access data from users that are their patients (not patients from other dietitians).

6 Conclusions and Future Work

Understanding food habits and portion sizes consumed can form the basis to develop an information system to provide context-specific information to guide and improve these habits. This paper presented a novel smart nutrition monitoring system for take-away food. It contains a data collection module (kiosk) that contains heterogeneous sensors. Following a Fog Computing approach, data from such sensors are collected and pre-processed before being sent to the Smart Nutrition Monitoring Engine in the cloud for further processing, storage, and visualization. In contrast to previous approaches for food intake monitoring, this method is practical and non-invasive with minimum participants' burden.

As future work, we aim to develop new modules in the architecture to facilitate the addition of new sensors to the system with minimum burden for developers and administrators. We also aim to develop advanced analytics capabilities that can compensate for inaccuracy of individual sensors, increasing even further the accuracy of the data provided by the system.

References

1. Bazzano, L.A., He, J., Ogden, L.G., Loria, C.M., Vupputuri, S., Myers, L., Whelton, P.K.: Fruit and vegetable intake and risk of cardiovascular disease in us adults: the first national health and nutrition examination survey epidemiologic follow-up study. The American journal of clinical nutrition **76**(1) (2002) 93–99
2. Volkow, N.D., Wang, G.J., Baler, R.D.: Reward, dopamine and the control of food intake: implications for obesity. Trends in cognitive sciences **15**(1) (2011) 37–46
3. Basiotis, P.P., Welsh, S.O., Cronin, F.J., Kelsay, J.L., Mertz, W., et al.: Number of days of food intake records required to estimate individual and group nutrient intakes with defined confidence. J nutr **117**(9) (1987) 1638–1641
4. Darby, A., Strum, M.W., Holmes, E., Gatwood, J.: A review of nutritional tracking mobile applications for diabetes patient use. Diabetes technology & therapeutics **18**(3) (2016) 200–212
5. Fontana, J.M., Sazonov, E.: Detection and characterization of food intake by wearable sensors. Wearable Sensors (2014) 591–616
6. Passler, S., Fischer, W.J.: Food intake monitoring: Automated chew event detection in chewing sounds. IEEE journal of biomedical and health informatics **18**(1) (2014) 278–289
7. Vu, T., Lin, F., Alshurafa, N., Xu, W.: Wearable food intake monitoring technologies: A comprehensive review. Computers **6**(1) (2017) 4

8. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of Things (IoT): A vision, architectural elements, and future directions. Future generation computer systems **29**(7) (2013) 1645–1660
9. Hebden, L., Cook, A., van der Ploeg, H.P., Allman-Farinelli, M.: Development of smartphone applications for nutrition and physical activity behavior change. JMIR research protocols **1**(2) (2012)
10. Block, G.: A review of validations of dietary assessment methods. American journal of epidemiology **115**(4) (1982) 492–505
11. Fallaize, R., Forster, H., Macready, A.L., Walsh, M.C., Mathers, J.C., Brennan, L., Gibney, E.R., Gibney, M.J., Lovegrove, J.A.: Online dietary intake estimation: reproducibility and validity of the food4me food frequency questionnaire against a 4-day weighed food record. Journal of medical Internet research **16**(8) (2014)
12. Bingham, S.A., Gill, C., Welch, A., Day, K., Cassidy, A., Khaw, K., Sneyd, M., Key, T., Roe, L., Day, N.: Comparison of dietary assessment methods in nutritional epidemiology: weighed records v. 24 h recalls, food-frequency questionnaires and estimated-diet records. British Journal of Nutrition **72**(4) (1994) 619–643
13. Cadavid, S., Abdel-Mottaleb, M., Helal, A.: Exploiting visual quasi-periodicity for real-time chewing event detection using active appearance models and support vector machines. Personal and Ubiquitous Computing **16**(6) (2012) 729–739
14. Zhou, B., Cheng, J., Sundholm, M., Reiss, A., Huang, W., Amft, O., Lukowicz, P.: Smart table surface: A novel approach to pervasive dining monitoring. In: Pervasive Computing and Communications (PerCom), 2015 IEEE International Conference on, IEEE (2015) 155–162
15. Amft, O., Kusserow, M., Tröster, G.: Bite weight prediction from acoustic recognition of chewing. IEEE transactions on biomedical engineering **56**(6) (2009) 1663–1672
16. Sazonov, E.S., Makeyev, O., Schuckers, S., Lopez-Meyer, P., Melanson, E.L., Neuman, M.R.: Automatic detection of swallowing events by acoustical means for applications of monitoring of ingestive behavior. IEEE Transactions on Biomedical Engineering **57**(3) (2010) 626–633
17. Dong, Y., Hoover, A., Scisco, J., Muth, E.: A new method for measuring meal intake in humans via automated wrist motion tracking. Applied psychophysiology and biofeedback **37**(3) (2012) 205–215
18. Mehdipour, F., Javadi, B., Mahanti, A.: FOG-engine: towards big data analytics in the Fog. In: Dependable, Autonomic and Secure Computing, 14th Intl Conf on Pervasive Intelligence and Computing, 2nd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), 2016 IEEE 14th Intl C, IEEE (2016) 640–646
19. Cai, Z., Li, X., Ruiz, R., Li, Q.: A delay-based dynamic scheduling algorithm for bag-of-task workflows with stochastic task execution times in clouds. Future Generation Computer Systems **71** (June 2017) 57–72
20. Calheiros, R.N., Buyya, R.: Cost-effective provisioning and scheduling of deadline-constrained applications in hybrid clouds. In Wang, X.S., Cruz, I.F., Delis, A., Huang, G., eds.: Proceedings of the 13th International Conference on Web Information Systems Engineering (WISE'12). Volume 7651 of Lecture Notes in Computer Science., Berlin, Germany, Springer (Nov. 2012) 171–184
21. Duan, R., Prodan, R., Li, X.: Multi-objective game theoretic scheduling of bag-of-tasks workflows on hybrid clouds. IEEE Transactions on Cloud Computing **2**(1) (March 2014) 29–42