Sensing Technologies and the Player-Middleware for Context-Awareness in Kitchen Environments

Matthias Kranz, Albrecht Schmidt
Research Group Embedded Interaction
University of Munich
Amalienstrasse 17, 80333 Munich, Germany
{matthias,albrecht}@hcilab.org

Radu Bogdan Rusu, Alexis Maldonado, Michael Beetz
Intellient Autonomous Systems Group
Technische Universität München (TUM)
Boltzmannstrasse 3, 85748 Garching b. München, Germany
{rusu, maldonad, beetz}@cs.tum.edu

Benedikt Hörnler, Gerhard Rigoll
Mensch-Maschine Kommunikation
Technische Universität München (TUM)
Arcisstr. 21, 80333 München, Germany
hbe@mmk.ei.tum.de, rigoll@ei.tum.de

Abstract— **In this paper we report on a use case of networked sensing technologies in the context of smart homes and specifically the kitchen as scenario for our research. We adapt, use and extend an existing middleware originating from robotics for pervasive computing. We report on initial results towards context recognition in this sensor enriched environment.**

I. Introduction

In this paper, we present the AwareKitchen research project. The setting for this scenario is a kitchen environment. Kitchens are an important part of everyday life. People spend a significant amount of time in them and do a wide spectrum of activities like preparing and eating meals, cleaning, and many social interactions in the families. The kitchen is in many households an important place for meeting and exchanging all kinds of information.

These activities also relate to many aspects of our life: personal health and health care, nutrition, the daily routine, and many more. Building systems and appliances that are capable of understanding behaviors and actions in the kitchen therefore can be generalized to be employed in other scenarios as well. This is especially true as in the course of e.g. preparing a full meal many tools are used - knifes, spoons, mixers, etc. This is also the case for many other work tasks - they involve physical tools operated by humans. Only recent advances in networked sensing technologies enabled their unobtrusive integration in everyday objects and environments.

Spoons and knifes are used without being perceived as tools, they are part of the everyday environment. Augmenting these everyday objects with sensing capabilities while keeping the look and feel and handling of the objects untouched is challenging. Thus this would allow us to gather data in-place and naturally - compared to laboratory settings which always influence behavior and the way actions are performed. As Intille et al. [1] argue, only the real-world setting allows to collect the most valuable data that leads to the development of the most robust and reliable algorithms. We computationally augmented several everyday artifacts commonly found in a kitchen and show their contribution towards user context acquisition.

With an aging population, the number of elderly people that need assistance, either in retirement castles or assisted living facilities, increases. With increasing costs for assisted living and decreasing costs for sensing systems, we believe that it is from many perspectives more desirable to invest in helpful technology, even if only for a few months, than to move people out of their loved homes. Research so far focussed on "smart" homes but without a specific focus on the kitchen as central place in a home. We therefore concentrate our research on this room, but the results can be generalized to other spaces.

We present two algorithms to derive the type of food currently handled by a person cooking. Our approach looks at two different means: a sensor-augmented knife and environmental microphones. We report on the algorithms employed and the results of the analysis of the force and torque sensor data of the knife and the audio data of the microphone, both analyzed in isolation and independently of each other.

The contribution of this work is as follows. First, we discuss related work in the field of pervasive healthcare and activity recognition based on the technologies we used. We consider the results we obtain for the kitchen to be transferable and applicable to other rooms and scenarios in a smart home context. Second, we present a novel middleware for unifying networked sensor I/O and actor support in pervasive computing. The Player/Stage/Gazebo middleware is a result of 10 years of continuous work in the field of autonomous intelligent robotic systems. It has not been reported that this middleware has been used in pervasive computing or in the context of recent sensor node systems. We report on our contributions towards this middleware for enabling a better support of pervasive computing technology, especially regarding added support for several standard sensor node technologies. Third, we report on the results of the application of modern sensing technology in a kitchen environment and discuss two examples in detail.

II. Related Work

Preparing and consuming food are, besides the social importance of a shared space like this, the main activities in a kitchen. The things we cook and eat have a direct impact on our personal well being, happiness and health.
Instrumenting the places where food is processed, like kitchens and dining rooms, can help to make people more aware of their nutrition and hence live a better, healthier life. The feasibility of this approach has been shown by Chang et al. in their work on a dietary monitoring system for a dining table [2]. We consider our kitchen and dining space to offer more options for research and ultimately for deployment due to the higher number and higher degree of unobtrusiveness of the augmented devices which also comprise everyday artifacts and specialized tools.

A. Related Work - Technological Discussion

Eating healthy already starts with buying and preparing meals before cooking them. Recent social investigations show that an increasing percentage of the population are overweight. Pervasive computing technology has shown that supportive system can raise nutritional awareness [3].
Acquiring context information from microphones in some areas seems more suitable than computer vision approaches. This is especially true for private settings as a home. This probably is due to the concern of the human inhabitants that so much additional and unconscious information can be derived from watching a video, e.g. the mood of the person. Microphones capture only things that happened, e.g. the usage of the dishwasher, but not things that e.g. were looked at. Additionally, the range of microphones can easily be restricted, e.g. by attaching them to a surface and only capturing the sound waves travelling through the material. Microphones can be embedded into all kinds of artifacts and do rely on less environmental conditions like lightning. Therefore, we chose the audio sensor approach for one experiment to determine the type of food handled.
Activity recognition using data acquired from environmental microphones has been demonstrated as promising approach. In a bathroom scenario, Chen et al. [4] were able to determine the activity of a person in the bathroom with minimal 85% accuracy. In this scenario, nearly all user would have rejected any vision technology. The authors though show that reliably activity recognition in arbitrary scenarios is possible. We could validate this in our kitchen context.
Amft et al. [5] propose a system for acquiring bone sound from a ear-worn sensor which allows detecting the kind of food consumed by analyzing chewing sounds. The goal is to support the human to change eating habits towards healthier food. Again, by analyzing sound related to food, they are capable of classifying more than 80% of the analyzed food correctly. They though only used a limited set of five 5 different classes of food. With our approach we do not try to classify greatly distinctive food like rice and chips correctly, but to classify all similar kinds of food, especially vegetables, correctly. In other contexts, e.g. for activity recognition using microphones [6], 85 % accuracy seems a reasonable value for future research, which we also achieve.
Augmenting the tools human use during their work ideally leads to task and process optimization, well-known in e.g. supply chain management. The goal of pervasive computing here is to transfer this idea to novel areas. Therefore selecting sensors and developing intelligent tools has attracted much research effort. This also has been the case for kitchen environments. Kitchen sinks that adapt in height to the size of the person, proposed by Bonanni [7] in 2005, are available as commercial products by now. In the same project, the MIT Counter Intelligence project[1], an intelligent spoon has been proposed by Cheng and Bonanni. It is a sensor equipped spoon that measures temperature, acidity and other environmental dimensions directly around the spoon. The idea is to support users while achieving their tasks in place by an integrated and embedded system. This integrated tool support for physical devices has also been chosen for our sensor-equipped knife [8]. The knife becomes an input devices for context acquisition.

B. Related Work - Everyday Objects

PlaceLab [1] is one of the most recently presented intelligent environments. The amount of sensors, e.g. Mites [9] and equipped every day objects, is huge and changeable from test to test. The importance of real-world collected sensor data as input for "robust and promising context detection algorithms" [1] has been identified as one of the key contributions of this project - along with the huge effort of annotating the raw data.
Unobtrusively and invisible augmented everyday objects are a key requirement for collecting data of processes and activities. The devices, e.g. wireless sensor nodes, have to be seamlessly integrated into the object to not change the usage behavior. Only then useful data can be gathered. Two further examples that achieve this are the TEA project [10] and the MediaCup. [11].
With the force and torque sensor in the knife, we try to get as close to the action as possible. Though not being invisible, the sensor does only add minimal additional weight and does not have any influence on the cutting process itself.

C. Related Work - Kitchen Activities

After discussing related work based on the technology used, we also want to place our work in context regarding the time line of events.
Before meals can be prepared, a certain dish has to be chosen. A shopping list has to be written (or kept in mind). Food has to be bought and brought home. The food now has to be prepared now, which is the point we specifically look at. After cooking, the food will be consumed. All these steps are necessary in this order. Much work in pervasive computing has specifically addressed these other steps of the process chain which we will present briefly.

The kitchen is a social place. Especially in the US, fridges are not covered and integrated within a shelf, but usually

[1] `http://www.media.mit.edu/ci/`

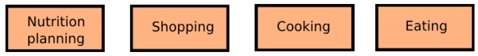

Fig. 1. Kitchen activities time line (from left to right). In the course of cooking and nurturing a family, many activities are involved. Research has so far analyzed the process of joint planning, shopping and eating. Cooking and food preparation activities have not recently been reported on. By our contribution, we try to fill this gap.

do have an exposed, metallic surface which is used for communication among house members. In Europe, usually a pin board servers this purpose. Taylor et al. investigated the use of fridge magnets [12] and identified planning of activities (e.g. shopping) as key issues supported by this surface. From a time perspective, Mankoff continues research on shopping lists for rising nutritional awareness [3]. By analyzing shopping lists and suggesting healthier and less calories alternatives, they give support when it comes to deciding which food to buy. After meals are cooked and prepared, the food is consumed. Chang et al. [2] show how, in this step, healthier eating and awareness regarding the amount and type of food during dinner can be supported. Amft et al. [5] take a different approach on identifying the type of food consumed, not regarding the quantity. We try to fill the gap in this sequence of steps (see Fig. 1) by enabling context-awareness and recognition in the course of preparing food for cooking.

III. Player/Stage/Gazebo for Pervasive Computing

In this section, we introduce the technical details of the Player/Stage/Gazebo (P/S/G) project, and present our contributions with respect to pervasive computing. We continue with a discussion of P/S/G's properties as middleware for pervasive computing. Finally, we present the simulation capabilities of P/S/G as they have been used in the context of the AwareKitchen project.

A. The Player device repository

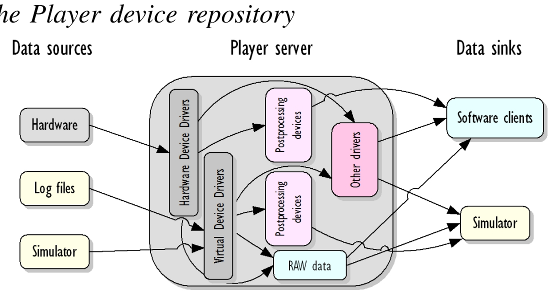

Fig. 2. Architectural overview. Data can originate either live from physical hardware, previously stored logfiles or a simulator. The data is acquired and provided by interfaces, abstracting from the physical device and its proprietary data format. It can be either made available directly or additionally be preprocessed on the Player server. The data is consumed by software clients or a simulator.

The Player component provides a simple and flexible interface for sensor and actuator control by realizing powerful classes of interface abstractions for interacting with real or simulated hardware. These abstractions enable the programmer to use devices with similar functionality identically from the code point of view, thus significantly increasing the robustness and transferability of the code.

Player's role in the sensor-actuator system is pretty similar to the abstraction layers present in modern operating systems. The same way an OS abstracts any hardware pointing device via a mouse interface, Player's abstraction layers decouple the user's program from the details of specific hardware. Any client that uses a specific Player interface, such as the **rfid** interface, will work in the same manner with any RFID readers that are supported in the Player repository. Furthermore, any number of clients can connect to the Player server and access data, send commands or request configuration changes to an existing device in the repository.

The most important working entity in Player is a device, which itself is composed of a driver and an interface. The set of *interfaces* are well defined and standardized, each of which describes the syntax and semantics for the allowable interactions with a particular class of drivers. In the context of pervasive computing, common interfaces include **wsn** and **rfid**, which respectively provide access to a Wireless Sensor Network and a RFID reader. Within the Player device concept, a *driver* does the work of directly controlling the hardware unit, by mapping its capabilities onto the corresponding interface. Within the Player concept, a driver can be:

- code that connects and communicates to a physical device;
- an algorithm that receives data from another device, processes it, then pushes it back through another channel (e.g. sensor fusion and processing or cognitive algorithms);
- *virtual*, meaning that it can create arbitrary data when needed.

In addition to providing access to hardware devices, Player drivers can implement sophisticated algorithms that use other drivers as sources and sinks for data, can generate data from the simulator or can play back previously recorded log files. Therefore, the Player driver system can be thought of a graph where nodes represent the drivers which interact with well-defined interfaces (edges). The generated data is processed through the appropriate interface provided by the device. For example, data generated by a Particle sensor node will be propagated through the *wsn* (Wireless Sensor Network) interface, or data generated by a video camera through the *camera* interface. A post-processing step can follow, in which the data will be made available to another driver, or simply just used as the input for another device (e.g. camera images can be the input for a marker detection algorithm that later provides position and orientation data through a *fiducial* interface). This is similar to the WaveScope project [13]. Finally, numerous data sinks are available, where the raw or processed data from any number of Player devices or servers can be used as input for other software programs, such as a higher-level middlewares like ECT or the Context Toolkit, or a 3D simulator like Gazebo.

B. Contributions for Pervasive Computing

Extending P/S/G towards pervasive sensing includes a development effort to provide drivers for sensors such as heterogeneous Wireless Sensor Networks, RFID technologies, Inertial Measurement Units, etc, as well as the appropriate logging tools, clients, visualizers and so on.

Initial work concentrated on defining and building the needed interfaces for the pervasive computing sensors, since the available pool of Player supported devices that were also used in pervasive computing environments were mainly only related to vision systems. Once the appropriate interfaces have been defined, a series of Player drivers for new pervasive hardware platforms were built and integrated in the freely available P/S/G sources, including:

- **Wireless Sensor Network** nodes - with a wide variety of different sensor nodes, ranging from the RCores and Particles from TecO/Particle Computers to the Porcupines, or the Mica2 and Mica2Dots from Crossbow;
- **RFID technologies** - several readers such as the Inside M/R300, the Skyetek M1 and the Skyetek M1-mini are now supported;
- **Inertial Measurement Units** - supporting the XSens MT9 as well as the XSens MTx, which provide drift-free 3D orientation and kinematic data.

Besides drivers and interfaces, the logging system was extended so that recording and playback of data from experiments is now possible. In addition, a number of virtual drivers which take care of sensor calibration, data fusion and automation feature extraction were also implemented.

The resulted devices can be accessed in the same way by the client, without the need to write additional code. This enables us to substitute drivers which provide the same interface, transparently for the client, and provides an important advantage: researchers can reuse the software without changes as long as they use the same interfaces (e.g. same context information). This greatly speeds up system and application development and is an issue so far neglected by nearly all existing systems. We integrated our drivers into the open source project to support other researchers realizing their own projects with P/S/G.

C. Simulation and Visualization with Player/Stage/Gazebo

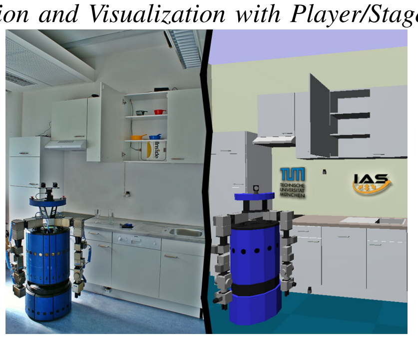

Fig. 3. Virtual and physical kitchen: the real environment is modeled and its physical properties are described so that sensors can acquire e.g. position, orientation, material density or gravity.

P/S/G comprises a 2D and a 3D simulator, called Stage and Gazebo. Both can be used for simulation and visualization. We will present the potential of this features after a short discussion of related work on simulators in pervasive computing. Simulation is important. Not all experiments can be done in the real world. Either they are too expensive or costly, too dangerous or are not easily repeatable. Therefore, researchers often have to work with simulated data before conducting the experiments to verify the results. An example are simulators for mobile phones, more and more used in pervasive computing for human-computer interaction. Most applications are thoroughly tested on a PC simulator before they are downloaded the first time to a real mobile phone. The debugging capabilities and the comfort associated with the simulator facilitate development a lot.

The question is why complex interactions, standard for pervasive computing, are normally *not* simulated and tested before going to the real site? Probably the answer simply is that there is no simulation system available that is comfortable and powerful enough to accomplish this task.

1) 2D Simulator Stage: Surfaces and surface interaction have attracted much research in the last years. Especially table top UIs and smart surfaces allow many interesting interactions between humans, devices and the environment.

Stage is the 2D simulator of the P/S/G project. It is especially useful for e.g. path planning and obstacle avoidance in robots. But it is easily imaginable that e.g. Pin & Play devices, modeled in Stage, could be used for simulating this technology and the interactions to develop application layer software. Also for visualization only purposes of an existing Pin & Play system, Stage can easily be used.

2) 3D Simulator Gazebo: The 3D simulator Gazebo of P/S/G basically is a OpenGL view on an environment modeled with the Open Dynamics Engine (ODE). ODE is useful for simulating vehicles, objects in virtual reality environments and virtual creatures. It is currently used in many computer games, 3D authoring tools and simulation tools.

We use the simulator (see Fig. III-C for a view on both the physical and virtual kitchen) for visualization purposes - displaying the sensor information in place and in a format easily understandable by humans. We are currently exploring the possibility of modeling not only common sensor platforms but also the possibilities of modeling human-computer interaction in the simulator. Already we are using the simulation environment as event generator, comparable but more powerful than the Location Event Simulator [14], to test our algorithms in early stages of the development process. A recently proposed simulator for pervasive computing is UbiReal [15], though it has only little middleware support.

IV. AwareKitchen: An Example for an Intelligent Environment running P/S/G

Understanding human activities and characterizing them into expressive and detailed activity models is one of the key issues of today's current pervasive computing systems.

In this context, we are building and developing specialized tools for acquiring user context in a kitchen scenario. We

also computationally augment everyday artefacts with sensing technologies which are unobtrusively integrated.
By providing an as natural as possible environment, we enable the humans in this environment to forget about the visible and invisible technologies and act as normal as possible. This has been a key principle during the project development.

A variety of sensors installed in the AwareKitchen provides the system with the necessary data to study the activities that take place in it. Several Hokuyo URG-04LX laser scanners allow us to locate people in the kitchen and follow their movements. Each cupboard door is equipped with a magnetic sensor to determine if it is opened or closed. There are also numerous RFID-Readers that detect the tags placed on the objects of interest, like glasses, pots, and other cooking equipment. One RFID reader is placed on a glove, which allows us to detect which objects are taken or manipulated.
Additionally, a great number of wireless sensor nodes are available, including Particles, RCores, Mica2s, Mica2Dots and Gumstix. The wireless nodes have accelerometers, magnetic sensors, light sensors, and microphones. Placed on objects or people, they offer a very easy and flexible way to collect data for new experiments.

Fig. 4. The AwareKitchen is an example for a sensor augmented intelligent environment. Everyday objects are tagged with RFID markers and partially equipped with wireless sensor nodes. Off-the-shelf and custom made components turn objects into context providers. Magnetic sensors in all doors detect opening and closing events. Mica2, Motes, Particles and Porcupine wireless sensor nodes enable unobtrusive data acquisition. RFID readers in the kitchen shelves detect the addition and removal of objects in various places, e.g. cupboards, tables, etc. An intelligent cutting board and a sensor knife enable context acquisition in the course of food processing and meal preparation. Laser scanners provide the position and number of people working in the kitchen. Robots learn from human behavior, e.g. from movement trajectories captured by inertial sensors and help completing tasks as part of the assistive health-care technology.

All systems presented deliver their data over the P/S/G middleware and allow to access and log the data in a well-defined, structured way. This supports and enables us to develop more quickly application layer software. Some examples of applications for this pervasive computing system we are currently working on are presented below. The P/S/G middleware as discussed and presented above is a suitable middleware for heterogeneous pervasive computing environments. We very shortly present examples of work-in-progress currently conducted in the AwareKitchen environment.

A. *Motion Analysis and Simulation*

Fig. 5. Food preparing experiments in the AwareKitchen together with the 3d reconstruction of the person's motion from XSens IMUs (left) and a visualization of the person's position from the 3 laser sensors, during another experiment in the AwareKitchen (right).

Using our XSens driver for P/S/G, we captured high-frequency position and orientation data of several kitchen activities like making a dinner table for a couple of persons or cutting vegetables. Fig. 5 shows a screenshot of the 3D visualization of cutting data (3D model of human) and the image of Matthias Kranz with the XSens while cutting the vegetables. Due to some limitations of the inertial sensors, work is currently being done to obtain position and orientation information from people using several cameras. Detecting how many people currently are in the AwareKitchen, what their position is relative to each other and to the kitchen, we are using the data from laser scanners. Fig. 5 (right) shows a visualization of the raw data.

B. *Context-Aware Tools for Kitchen Environments*

We built a sensor-augmented knife [8] that features (see Fig. 6) a three axis of force and three axes of torque sensor between handle and blade.

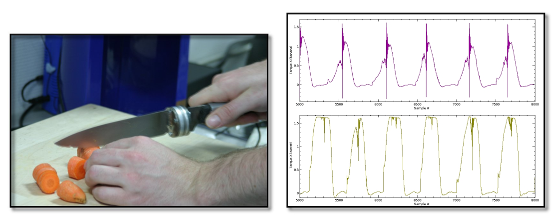

Fig. 6. Knife and Data visualization. The sensor between blade and handle measures each 3 perpendicular axes of force and torque. The sensor is soldered in the knife so it does not disturb normal use. The right image shows two data sets (just one axis of torque) of the knife, cutting a carrot and a banana. The data looks (as hoped) very different. Combing all 6 axes, we are able to distinguish which food is currently prepared for cooking.

We determine the type of ingredient that is cut to pieces with the knife in the course of preparing a meal. We are comparing the results of the acceleration sensor with an analysis of audio data from the cutting process. Both approaches can classify at least 85% of the food correctly. Though, both approaches differ significantly in the amount of features needed for this classification results.

V. Activity Recognition with Networked Sensing System via P/S/G

A. Cutting Board and Camera

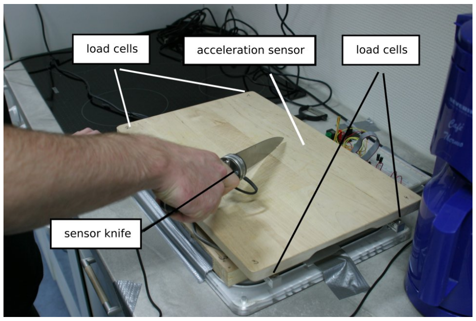

Fig. 7. The cutting board is suspended on four load cells (HBM 1-DF2SR-3/5K-C) in the edges of the wooded board. The load cells are directed towards the center of the board. In the middle of the cutting board, a ADXL203 two-axes acceleration sensor is placed. Acceleration is sensed perpendicular to the wooden board (z-axis). The cutting board can not only be used for implicit context acquisition, but also for explicit interaction as a mouse.

The cutting board, depicted in Fig. 7, has been suspended on four load cells which enable us to measure the total weight and weight distribution of the ingredients being processed on it.

With some domain knowledge applied we could e.g. also determine the calories the ingredients contain if we knew what type of food currently is processed. This would provide the same information as used in the diet aware dining table [2] without the introduction of a cumbersome scenario of a table no-one is allowed to lean on. We will later show that we can determine the type of food with a reasonable accuracy. A common kitchen scale, already present in most kitchens, can be replaced by this cutting board. It could even be used as amouse for explicit interaction, as demonstrated with the Load Table by Schmidt et al. [16].

We cut all ingredients for our experiments on this sensor augmented cutting board and collected the data. We can determine the type of processed food with the information about the initial weight, the changes in weight distribution (e.g. you move over the cutting board while cutting e.g. leek) and the acceleration information.

As depicted in Fig. 8, a small camera is additionally mounted above the cutting board. The camera only captures the cutting board and a few centimeters around it. We believe that a computer vision approach easily[2] will be able to recognize the food placed on the cutting board for meal preparation. We therefore also captured the video data but did not put any effort in object recognition. For limited niches as we do have it here (recognizing food in a defined area with stable lightning conditions), e.g. license plate recognition, has been demonstrated many times by computer vision researchers.

[2]At least easy for researchers concentrating on computer vision.

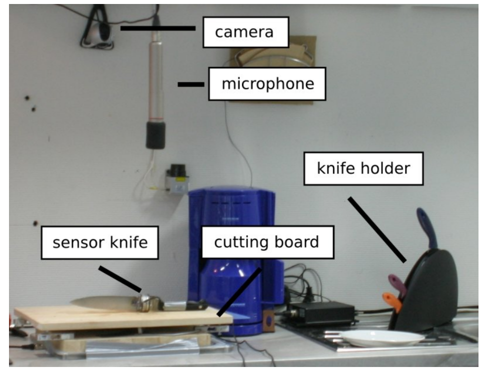

Fig. 8. Setup of the cutting board. A sensitive microphone is placed above the cutting board.

B. Microphone

The audio data is recorded with the microphone "AKG C1000S", which is placed approximately 30cm above the cutting board. Also the web cam "Logitech QuickCam Pro 4000" is mounted above. Both data streams are recorded to a single video file, so the audio- and video-signal are synchronized.

C. Knife

To optimally detect which type of food is being prepared in the kitchen, we need data that is produced very close to the action, and with the least possible noise from other activities. For this reason we chose to measure the forces and torques that result from the interaction between the knife and the food. We have equipped a knife (see Fig. 9) with a three axes force/torque sensor (SI-40-2 from ATI Industrial Automation). The sensor can measure torques up to 2 N-m and forces up to 120N, which is adequate for the normal tasks in the kitchen. The data is recorded using a National Instruments PCI-6221 data acquisition card. The chosen sampling rate was 1kHz.

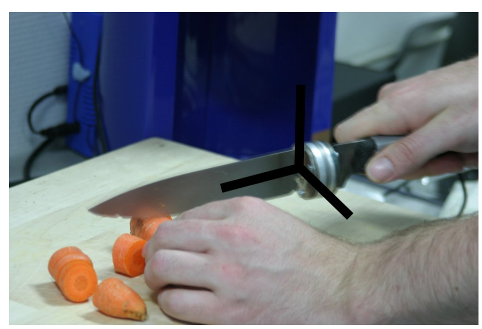

Fig. 9. Instrumented knife cutting carrots. A three axis force/torque sensor has been soldered between handle and blade. The weight of the sensor is minimal and nearly unnoticeable. The three axes are drawn as black lines in the image above. The sensor does not hinder normal "operation".

D. Data Collection

We did three data collection sessions. During these sessions four people cut different types of vegetables to small pieces. We cut carrots, banana, different types of leek, kohlrabi, ball pepper and apple. Carrots and kohlrabi have a hull that normally will be removed before cutting. We also collected audio data from peeling the hull of using a regular peeler. The peeling can be used as a part of the audio classification, but it is not present in the knife data.

E. Audio Data Analysis

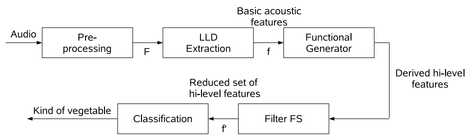

Fig. 10. Overview of the audio based classification of the cutting task.

For the audio database 15 video samples are separated into cutting-episodes, which represent one cutting or peeling action of a fruit or a vegetable. The total number of cutting-episodes is 269 and the distribution of the cutting-episodes is nearly uniform. The database contains 40 examples of apple cut, 43 of carrot peel, 42 of carrot cut, 31 of kohlrabi peel, 34 of kohlrabi cut, 48 of leek cut and 31 of bell pepper cut. To obtain good results for the classification it is important that the number of instances of each class is nearly equal.

To deal with the complex audio data, the signal is first pre-processed by applying a Hamming-window, and separating the signal into frames of 20ms each, windowed every 10ms. In this way, we obtain 100 frames per second.

It is very hard for any machine learning algorithm to process audio data directly. To deal with this issue, certain features are extracted from the waveforms of each frame. Around seven thousand different features in total are used, and they include for each frame: feature countours containing information about pitch, energy, amplitude, bandwitdh. Harmonics-to-Noise Ratio and many others.

The basic feature contours are provided by the package Speech-Filing-System (SFS) [17]. SFS is a free package combining several speech signal processing libraries within one framework.

The next stage performs a feature selection to reduce the dimensionality. This is needed because of the huge feature space and to save calculation and classification effort.

All the following results are performed on the audio database with 269 instances. The WEKA 3 data-mining environment [18] is used for classification. Only few of the available classifiers perform an analysis of this huge feature space in an acceptable time. The best result of the evaluation of the full feature space (around 7000 features) is obtained by a Support Vector Machine (SVM) with a polynomial kernel, complexity and exponent are set to one. This classifier achieves a performance of 85.5% of correct classified instances and the table I shows the confusion matrix.

After applying a feature selection algorithm, the training algorithm only has to deal with around 250 features, and has a very similar percentage of correctly classified instances. The learning phase is much faster with the reduced number of features.

classified as ->	apple cut	carrot cut	carrot peel	kohl. cut	kohl. peel	leek cut	pepper cut
apple cut	31	0	0	3	5	0	1
carrot cut	0	41	0	0	0	0	1
carrot peel	1	0	42	0	0	0	0
kohlrabi cut	4	1	0	24	0	2	3
kohlrabi peel	3	1	0	2	23	1	1
leek cut	1	0	0	0	0	46	1
pepper cut	4	3	0	1	0	0	23

TABLE I

CONFUSION MATRIX OF THE BEST CLASSIFICATION METHOD ON THE AUDIO DATA. THE MAJORITY OF AUDIO SECTIONS ARE CLASSIFIED CORRECTLY.

Some confusion between apple cut, kohlrabi cut and kohlrabi peel occurs, because these fruits and vegetables have a similar consistency [7]. The second reason is that at the end of each cutting episode, the sound of the knife touching the cutting board can be heard, and this sound is the same always.

F. Force/Torque Data Analysis

During the cutting action, the forces and torques produced have a signature that can be used to identify the type of food that was handled. To find this non-trivial signature, we will also use machine learning techniques.

We do the classification using only the torque data from one of the axes, with the intention of seeing the feasibility of cheap production of such instrumented knifes.

We first segment the cutting data automatically into episodes using thresholding with a minimum value for the cutting actions. The episodes also have to meet a minimum-length requirement of 100 ms. Each episode represents a cut into the fruit or vegetable.

Since the data is much cleaner than the audio recordings, a simpler set of features is enough for the classification. We used the following features to describe each episode: mean, variation, length, integral, maximum value, minimum value, Skewness, Kurtosis, and the first 50 coefficients of the Fourier Transform.

We collected a total of 854 episodes, including cutting of apples, carrots, kohlrabi, bananas, leek, and bell peppers.

To classify the data, we use WEKA [18], with the Multi-BoostAB classificator, using J48 decision trees internally. The resulting model is small, and can run in real time to give results as the people are using the knife. The results of the classification are shown in Tab. II.

From table II we can observe that the classificator has excellent performance for all the tested ingredients, with the exception of bell pepper.

The cutting episodes from the bell pepper are very hard to classify because the pepper presents a different resistance and structure during cutting: at the beginning it is very stiff and hollow, and the classifier confuses it with kohlrabi. Later when the pepper is open, and the cook cuts it into slices, the cutting action is very short and sharp, and its data looks like the cutting of carrots. The classificator is still able to correctly

classified as ->	apple	carrot	kohlrabi	banana	leek	pepper
apple	27	0	0	5	0	2
carrot	0	102	3	4	5	9
kohlrabi	3	0	183	0	1	9
banana	0	0	0	167	0	0
leek	0	2	0	2	179	4
pepper	2	17	29	1	9	89

TABLE II

CONFUSION MATRIX OF THE BEST CLASSIFICATION RESULT USING THE KNIFE DATA.

detect the bell pepper in the majority of the cases.
Using only the information from one axis of the force/torque sensor, we have obtained an accuracy of the classification higher than 85%. This information can be obtained by an inexpensive strain gage installed in regular knifes. The sampling frequency of 1kHz can also be easily reached by inexpensive micro controllers, which could collect the data and transfer wirelessly for recording. Since the cost would be very low, it is thinkable that such an instrumented knife will be present in kitchens in the future.

An accuracy of 85% might seem low, but this is only the accuracy of each cut. As a part of our future work, we plan to use the assumption that a person usually does several cuts in a row to the same ingredient, and that there are pauses between manipulation of different ingredients, to automatically segment the cutting episodes. Then we can use the classifications of all the cuts in one cutting segment to make a much better classification. We expect to achieve an accuracy higher than 95%.

VI. CONCLUSIONS

We present a common scenario, a kitchen, as fertile setting for investigating context-awareness in smart homes. We report on the rich sensor systems involved in this scenario and report on how a networked sensing system meaningful can be employed to gather information. We introduce a novel sensing middleware to the pervasive computing community and we report on our extensions to this middleware to leverage existing technologies and also share our drivers as part of the open source software Player/Stage/Gazebo to allow researchers to use them in their own projects. We discuss two examples of augmented tools in the kitchen environment in detail and report our findings. We also hope to attract researchers to explore the potentials of P/S/G in their work as step towards a 'standard' middleware for pervasive computing in e.g. intelligent environments.

REFERENCES

[1] S. S. Intille, K. Larson, E. M. Tapia, J. Beaudin, P. Kaushik, J. Nawyn, and R. Rockinson, "Using a live-in laboratory for ubiquitous computing research." in *Pervasive*, ser. Lecture Notes in Computer Science, K. P. Fishkin, B. Schiele, P. Nixon, and A. J. Quigley, Eds., vol. 3968. Springer, 2006, pp. 349–365.

[2] K. hao Chang, S. yen Liu, H.-H. Chu, J. Hsu, C. Chen, T. yun Lin, C. yu Chen, and P. Huang, "The diet-aware dining table: Observing dietary behaviors over a tabletop surface." in *Pervasive*, ser. Lecture Notes in Computer Science, K. P. Fishkin, B. Schiele, P. Nixon, and A. J. Quigley, Eds., vol. 3968. Springer, 2006, pp. 366–382.

[3] J. Mankoff, G. Hsieh, H. C. Hung, S. Lee, and E. Nitao, "Using low-cost sensing to support nutritional awareness," in *UbiComp '02: Proceedings of the 4th international conference on Ubiquitous Computing*. London, UK: Springer-Verlag, 2002, pp. 371–376.

[4] J. Chen, A. H. Kam, J. Zhang, N. Liu, and L. Shue, "Bathroom activity monitoring based on sound." in *Pervasive*, ser. Lecture Notes in Computer Science, H.-W. Gellersen, R. Want, and A. Schmidt, Eds., vol. 3468. Springer, 2005, pp. 47–61.

[5] O. Amft, M. Stäger, P. Lukowicz, and G. Tröster, "Analysis of chewing sounds for dietary monitoring," in *UbiComp 2005: Proceedings of the 7th International Conference on Ubiquitous Computing*, M. Beigl, S. Intille, and J. Rekimoto, Eds. Springer: Lecture Notes in Computer Science, Sept. 2005, pp. 56–72. [Online]. Available: http://www.springerlink.com/openurl.asp?genre=article&issn=0302-9743&volume=3660&spage=56

[6] P. Lukowicz, J. A. Ward, H. Junker, M. Stäger, G. Tröster, A. Atrash, and T. Starner, "Recognizing workshop activity using body worn microphones and accelerometers," in *Pervasive Computing: Proceedings of the 2nd International Conference*. Springer-Verlag Heidelberg: Lecture Notes in Computer Science, Apr. 2004, pp. 18–22. [Online]. Available: http://springerlink.metapress.com/openurl.asp?genre=article&issn=0302-9743&volume=3001&spage=18

[7] L. Bonanni, E. Arroyo, C.-H. Lee, and T. Selker, "Smart sinks: real-world opportunities for context-aware interaction," in *CHI '05: CHI '05 extended abstracts on Human factors in computing systems*. New York, NY, USA: ACM Press, 2005, pp. 1232–1235.

[8] M. Kranz, A. Schmidt, A. Maldonado, R. B. Rusu, M. Beetz, B. Hörnler, and G. Rigoll, "Context-aware kitchen utilities," in *TEI '07: Proceedings of the 1st international conference on Tangible and embedded interaction*. New York, NY, USA: ACM Press, 2007, pp. 213–214.

[9] E. M. Tapia, S. S. Intille, L. Lopez, and K. Larson, "The design of a portable kit of wireless sensors for naturalistic data collection." in *Pervasive*, ser. Lecture Notes in Computer Science, K. P. Fishkin, B. Schiele, P. Nixon, and A. J. Quigley, Eds., vol. 3968. Springer, 2006, pp. 117–134.

[10] H.-W. Gellersen and M. Beigl, "Multi-sensor context-awareness in mobile devices and smart artefacts," 2002. [Online]. Available: citeseer.ist.psu.edu/gellersen02multisensor.html

[11] H.-W. Gellersen, M. Beigl, and H. Krull, "The mediacup: Awareness technology embedded in a everyday object," in *HUC '99: Proceedings of the 1st international symposium on Handheld and Ubiquitous Computing*. London, UK: Springer-Verlag, 1999, pp. 308–310.

[12] A. S. Taylor, R. Eardley, S. Hodges, A. Sellen, K. R. Wood, and L. Swan, "Augmenting refrigerator magnets: why less is sometimes more," in *Fourth Nordic Conference on Human-Computer Interaction, Nordichi '06*, 2006.

[13] L. Girod, K. Jamieson, Y. Mei, R. Newton, S. Rost, A. Thiagarajan, H. Balakrishnan, and S. Madden, "The case for a signal-oriented data stream management system." in *CIDR*. www.crdrdb.org, 2007, pp. 397–406.

[14] K. Sanmugalingam and G. Coulouris, "A generic location event simulator," in *UbiComp '02: Proceedings of the 4th international conference on Ubiquitous Computing*. London, UK: Springer-Verlag, 2002, pp. 308–315.

[15] H. Nishikawa, S. Yamamoto, M. Tamai, K. Nishigaki, T. Kitani, N. Shibata, K. Yasumoto, and M. Ito, "UbiREAL: Realistic Smartspace Simulator for Systematic Testing," in *UbiComp 2006: Proceedings of the 8th International Conference on Ubiquitous Computing*, P. Dourish and A. Friday, Eds. Springer: Lecture Notes in Computer Science, Sept. 2006, pp. 459–467.

[16] A. Schmidt, M. Strohbach, K. van Laerhoven, A. Friday, and H.-W. Gellersen, "Context acquisition based on load sensing," in *UbiComp '02: Proceedings of the 4th international conference on Ubiquitous Computing*. London, UK: Springer-Verlag, 2002, pp. 333–350.

[17] Phonetics and Linguistics, University College London, "Speech filing system," On the Internet http://www.phon.ucl.ac.uk/resource/sfs/, 2006.

[18] I. H. Witten and E. Frank, *Data Mining: Practical machine learning tools and techniques*, 2nd ed. San Francisco: Morgan Kaufmann, 2005.

[19] K. P. Fishkin, B. Schiele, P. Nixon, and A. J. Quigley, Eds., *Pervasive Computing, 4th International Conference, PERVASIVE 2006, Dublin, Ireland, May 7-10, 2006, Proceedings*, ser. Lecture Notes in Computer Science, vol. 3968. Springer, 2006.