Wearable Activity Tracking in Car Manufacturing

A context-aware wearable computing system could support a production or maintenance worker by recognizing the worker's actions and delivering just-in-time information about activities to be performed.

Thomas Stiefmeier, Daniel Roggen, and Gerhard Tröster
ETH Zurich

Georg Ogris and Paul Lukowicz
University of Passau

Wearable computing systems should provide unobtrusive access to the IT infrastructure, so they should be physically unobtrusive, leave users' hands free, and allow freedom of motion. In an industrial scenario, they must also minimize workers' cognitive load and avoid distracting them from their primary tasks (see the sidebar "Related Work in Wearable Computing in Industrial Environments"). An appropriate user interface design can partially accomplish this. A better solution, however, is a proactive context-aware system that uses unobtrusive sensors to track each step of the performed task and presents the worker with the information needed at any given moment.

Little research has focused on complex activity recognition for industrial applications. In this field, system reliability and robustness are essential. We're working to develop and test such real-life industrial activity tracking systems as part of the European Union's 4.5-year-long wearIT@work project.[1] To this end, we conducted two case studies in cooperation with the European car manufacturer Skoda. Our first study involved context-aware worker training at a dedicated *learning island*, which consisted of an instrumented car body. The second considered wearable assistance for quality control in the production line, where workers use on-body sensors only.

After observing and videorecording workers in the Skoda factory, we made the first data recordings onsite. Because such recordings disrupt the production process, causing large costs, we recreated the environments in our lab using original tools, cars, and procedures. We brought an entire car body, identical to the Skoda learning island, into our premises. Also, Skoda provided us with a complete new car on which to reenact the quality-control steps from the videos and observations.

Based on experiments performed in these setups over the last two years, we present an overview of the challenges and approaches associated with using complex multimodal sensor systems for activity tracking in production environments.

Training assembly-line workers

Currently, Skoda assembly-line workers follow a two-step training process. First, they receive classroom instruction on the theoretical background of car assembly. Next, they practice on a learning island consisting of a specially prepared car body that allows repeated assembly and removal of relevant components. A supervisor assesses trainees' progress and decides when they're ready for the assembly line.

Our industrial partners' workplace studies revealed that workers can benefit from combining the theoretical and practical steps. Wearable assistance could give trainees online instructions about upcoming assembly steps while they're at the learning island, thereby streamlining the training. Devices could also provide warnings

 Published by the IEEE CS ■ 1536-1268/08/$25.00

when the worker doesn't properly follow the procedure and automatically generate a training score. Realizing such a system requires unobtrusively recognizing and tracking the detailed assembly steps from the worker's actions.

Experimental setup

We developed task-tracking methods on the learning island in our laboratory. The task we chose for our experiments—installation of the front lamp—is representative of a wide range of assembly tasks in terms of complexity. The task involves manipulative gestures, the use of hand-operated tools, and many interactions between the car body and assembly parts. It consists of four phases:

- inserting the lamp,
- mounting a supportive plastic bar using three screws and a cordless screwdriver,
- attaching the lamp body using two screws and a cordless screwdriver, and
- verifying the lamp's adjustment.

Tracking approach

We rely on sensors placed on the body and on the car to detect assembly steps (see figure 1). We augment tools with RFID tags to detect which ones workers are using (for example, a cordless screwdriver). An RFID reader placed between a worker's thumb and index finger picks up this information. We detect when a worker grasps tools or assembly parts from muscle contraction, captured by an array of force-sensitive resistors (FSRs) integrated into a strap worn around the lower arm. We infer when screws are completely tightened by detecting the onset of the screwdriver's torque limiter. This onset causes the tool, and the hand holding it, to shake and vibrate. We pick up this information with an Xsens inertial measurement unit (IMU), mounted on the back of the palm.

FSRs placed next to screw joints detect the attachment of assembly parts, such as the lamp body and the supportive plastic bar, by indicating the tightening of the corresponding screw. A magnetic switch placed inside the lamp cavity lets us know when the lamp is inserted. We also mount magnetic switches on the car body to detect when the worker verifies the lamp adjustment using a special alignment tool.

We use a combination of sensor modalities whenever possible to improve robustness. For instance, detecting a tool with the RFID reader, hand grasping with FSRs, hand vibration with IMUs, and screw pressure with car-mounted FSRs all convey bits of information that, combined, help us achieve robust activity recognition. Figure 2 illustrates sensor processing and fusion.

Modeling assembly tasks

We modeled the assembly process to map sensor information to assembly steps. We developed a task-modeling scheme akin to a finite state machine (FSM) with two elements:

- *states*, which correspond to defined assembly steps, and
- *transitions*, which indicate the possible sequences of states.

Related Work in Wearable Computing in Industrial Environments

In the October–December 2002 issue of *IEEE Pervasive Computing*, Vincent Stanford described early work on the deployment of wearable computing in industrial environments,[1] such as Boeing's pioneer work on wearable-computing-based guided assembly.[2] Several projects have used head-mounted displays and wireless technology to provide remote information access during maintenance work.[3] More recent work tried to combine wearable computing with virtual or augmented reality to support the worker.[4] Hendrik Witt and his colleagues tackled the question of how to interface a wearable system hands-free.[5] To give users proactive instructions during furniture assembly, Stavros Antifakos and his colleagues augmented parts and tools with sensors such as accelerometers, gyroscopes, and force-sensitive resistors.[6]

REFERENCES

1. V. Stanford, "Wearable Computing Goes Live in Industry," *IEEE Pervasive Computing*, vol. 1, no. 4, 2002, pp. 14–19.
2. W. Barfield and T. Caudell, "Boeing's Wire Bundle Experiment," *Fundamentals of Wearable Computers and Augmented Reality*, Lawrence Erlbaum Associates, 2001, pp. 447–467.
3. D. Siewiorek et al., "Adtranz: A Mobile Computing System for Maintenance and Collaboration," *Proc. IEEE Int'l Symp. Wearable Computers*, IEEE CS Press, 1998, pp. 25–32.
4. H. Regenbrecht, G. Baratoff, and W. Wilke, "Augmented Reality Projects in the Automotive and Aerospace Industries," *IEEE Computer Graphics and Applications*, vol. 25, no. 6, 2005, pp. 48–56.
5. H. Witt, T. Nicolai, and H. Kenn, "Designing a Wearable User Interface for Hands-Free Interaction in Maintenance Applications," *Proc. Pervasive Computing and Comm. Workshop*, sIEEE CS Press, 2006, pp. 4–7.
6. S. Antifakos, F. Michahelles, and B. Schiele, "Proactive Instructions for Furniture Assembly," *Proc. 4th Int'l Symp. Ubiquitous Computing* (UbiComp 02), LNCS 2498, Springer, 2002, pp. 351–360.

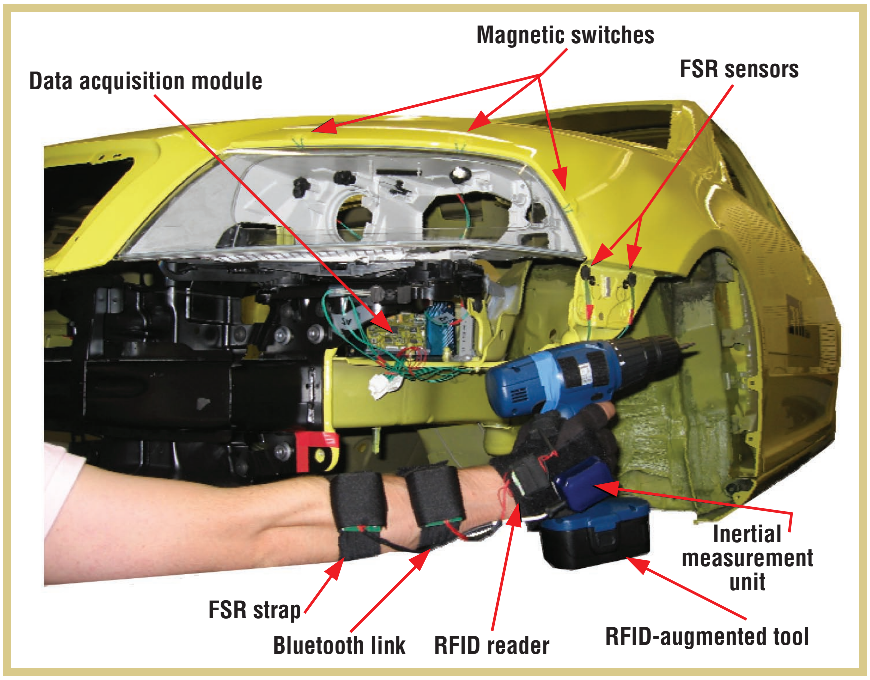


Figure 1. Sensors used in the training case study. The RFID reader, force-sensitive resistor (FSR) strap, and inertial measurement unit are on-body sensors; the magnetic switches and FSR sensors are environmental sensors.

Sensor signal conditions (that is, specific events or worker activities) trigger transitions between steps, effectively fusing multimodal sensor data.

Figure 3 details the tracking module and highlights the four lamp-assembly phases, which together take a worker about four minutes.

As an example, assume the FSM is in state *start* and the worker inserts the front lamp into the car body. Magnetic switch D1 detects the activity and the current state becomes *lamp inserted*. The worker then attaches a supportive plastic bar. After the worker tightens one of the three screws (A1, A2, or A3, detected by FSRs) with the appropriate screwdriver (RF1, detected by the RFID reader), the model transitions to state *1 bar screw*, provided that the FSR detected the grasping action (G) and the vibration (V). A detailed description of this platform appears elsewhere.[2]

Lessons learned

In this case study, we gained knowledge about sensor usage and task modeling in a realistic industrial setup.

Sensors. Several developers have suggested that redundant, multimodal sensor systems are crucial to reliable recognition of complex real-life activities.[3] Our work confirms this assertion on a particularly complex and diverse sensor system. The sensors on the car body are essential for the recognition task, whereas the wearable sensors mainly increase the system's robustness.

We made several findings about the individual sensor domains. First, the RFID reader's location is critical to obtaining a good coupling of the reader's antenna with the tags on the tools. Placing the reader between the thumb and index finger results in reliable user-independent operation.

Second, the FSR muscle activity strap's detection threshold is difficult to determine. Considerable variability exists among workers but also between grasping activities performed by the same worker, requiring user-specific calibration.

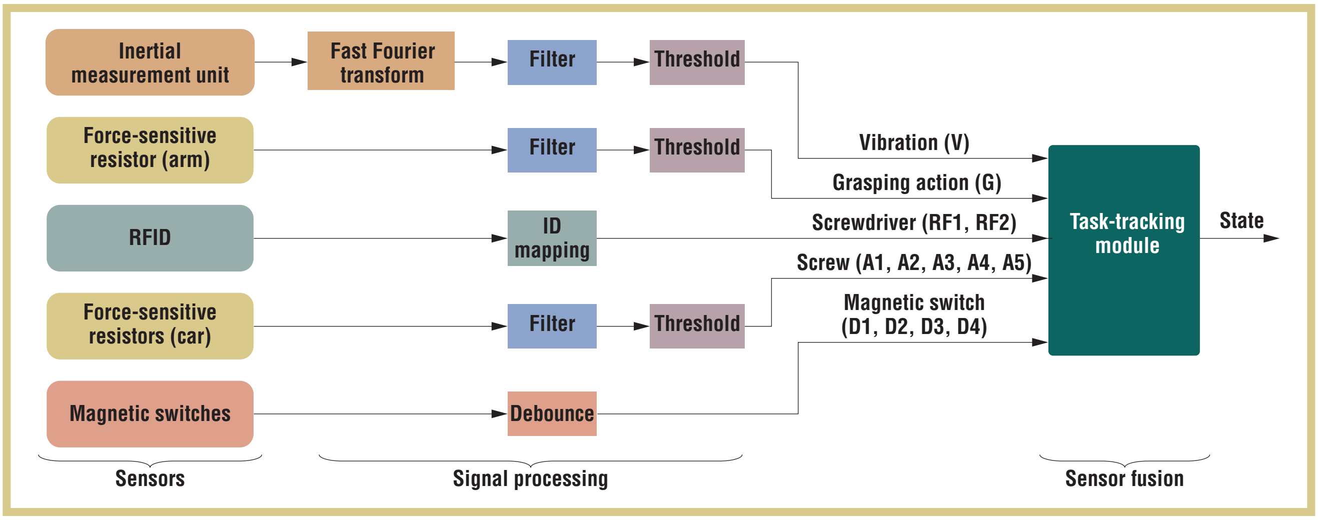


Figure 2. Sensor data processing and multimodal sensor fusion architecture.

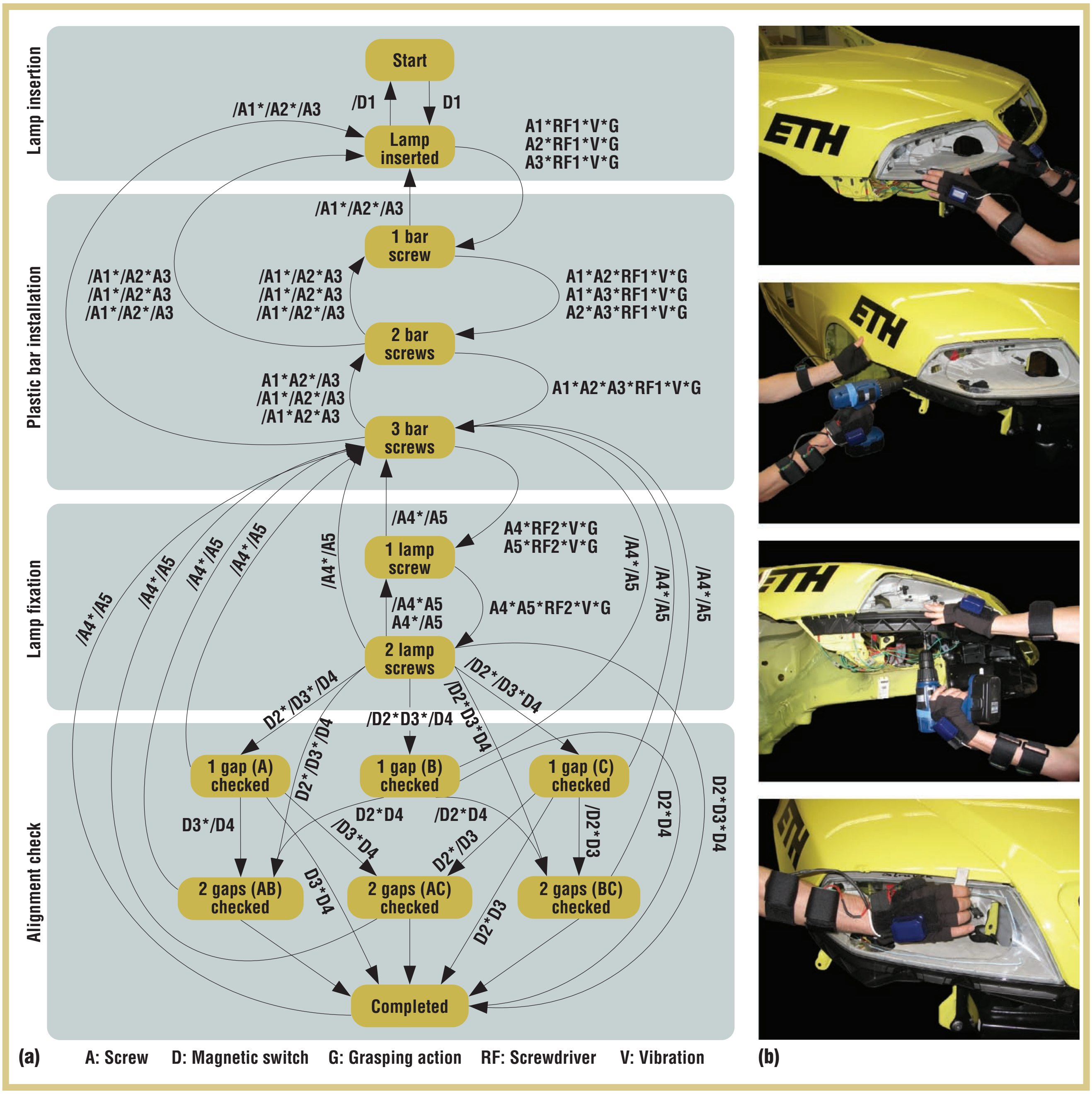


Figure 3. Front-lamp assembly. (a) We use a finite state machine to model the four phases of front-lamp assembly. The letter/number combinations (for example, D1) along the arrows define the logical condition that must exist to trigger the transition between the corresponding two states. A slash represents the logical operator NOT and an asterisk means AND. We combine individual logical conditions (such as A4 and A5 for analog FSR sensors) to more complex logical expressions such as "/A4 * /A5" ((NOT A4) AND (NOT A5)). (b) The photographs illustrate each phase's manipulative activity.

In addition, detecting the torque limiter's onset from the inertial sensor proved difficult, requiring sufficient mechanical coupling between hand and tool. This coupling depends on the worker's grasping strength and the angle between hand and tool (that is, it's user dependent). Placing the IMU directly on the tool might remove this constraint.

We also observed that the FSRs mounted next to screw joints on the car body performed reliably after initial calibration. The magnetic switches worked properly after initial adjustment without any further calibration.

Figure 4. Six checking activities. Workers (a) open the trunk, (b) close the engine hood, (c) open the back right door, (d) check the fuel filler cap mechanism, (e) open both right doors, and (f) check gaps at the front left door. The stick figures above each activity represent workers' upper body postures during task execution, captured by our motion jacket.

Finally, it takes one expert about half a day to mount and interconnect the sensors for the front lamp assembly task. So, this approach is restricted to a training environment in which the same instrumented car is used to train generations of workers. Currently, the learning island handles 18 training tasks of diverse complexity, requiring about one week of sensor installation and calibration. The number of necessary sensors scales linearly, and we estimate that we'd need about 120 sensors to detect all the tasks in the complete training program.

Our project partners performed a user study using Wizard-of-Oz techniques with a mock-up of our sensors to simulate the task tracking. The study revealed that the body-worn sensors didn't hamper workers in their primary task. Yet sensors were always noticeable, leaving room for future miniaturization and integration.

Task model. We considered approaches in which the system learns the behavior sequence from repeated task execution. To assign training material like pictures and videos to corresponding parts of the procedure, however, we'd need to identify specific assembly steps. Together with the well-defined activity structure, this requirement led us to consider the FSM approach. Our FSM approach requires perfect sensor operation because it doesn't contain probabilistic elements. However, the sensor reliability was high, leading to recognition rates close to 100 percent—that is, several users could successfully track the assembly task from start to finish. To allow for some sensor failure, we could introduce fuzzy operators or probabilistic methods.

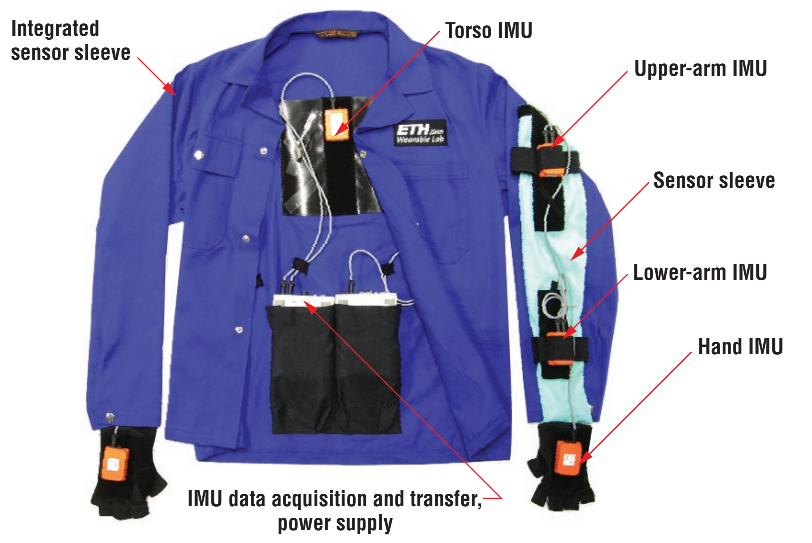
Contextual support in the assembly line

The last stage of car assembly is the quality-assurance checkpoint. The workers inspect the car and verify the functionality of user-accessible parts (doors, locks, seats, and so on). They carry paper sheets with fault matrices comprising a list of possible faults at all positions on and inside the car. They selectively read these sheets to find special characteristics of the car under inspection, such as nonstandard configuration (for example, steering wheel on the right side). If they detect a fault or any deviation from the car specification, they register it in the fault matrices.

A study from our industrial project partners revealed two important functionalities of wearable assistance in this scenario.

- The system should raise warnings whenever a worker misses a step in the checking procedure.
- Workers should be able to enter detected faults directly into the electronic database.

Figure 5. Our motion jacket is fully integrated into a standard worker's jacket. Seven inertial measurement units (IMUs) capture the worker's upper-body motion.



To minimize disturbances, the fault-entry system should have a context-aware interface that implicitly recognizes the location and, potentially, the type of detected fault. False positives aren't critical in this scenario because a simple confirmation mechanism could remove them based on speech recognition.

Task-tracking requirements

The key mechanism for enabling these functionalities is the automatic recognition of individual checking activities. Using quality-control lists and in situ observations of Skoda's process, we identified 46 trackable activities that form a complete, seven-minute checking procedure. Figure 4 illustrates some of these activities.

Sensor setup

Unlike in the training scenario, here tracking relies on body-worn sensors only because instrumenting cars in production is unfeasible. Figure 5 depicts the *motion jacket* we developed to integrate the required sensor modalities in an unobtrusive and robust working jacket. We capture the worker's upper body motion from seven IMUs within the jacket. The IMUs are on the lower and upper arms, the torso, and the back of the gloves. Two data-acquisition units collect the data from these IMUs. The IMUs let us capture arm and hand motion, which provides information about the worker's activity. We pick up the lower arms' muscle activity using custom-made thin sleeves equipped with arrays of FSRs. To estimate the worker's relative position to the car body, we use an ultrawideband system from Ubisense. Tags on the worker's shoulders let the system calculate the worker's position with respect to four reference transmitters placed around the car.

Experiments

As we mentioned earlier, we first collected data in situ at the Skoda production facility. A worker wearing our motion jacket performed the checking procedure on 10 cars while they were moving on the conveyor belt. We used these recordings to prove our sensor system's reliability and robustness under real-life conditions. In addition, we observed and filmed several other workers performing the procedure for later analysis.

Using our recreated setup (including the Skoda car), we recorded a data set with eight subjects (students, instructed from the video material). Each subject conducted 10 repetitions of the checking procedure. One experimenter annotated the start and end points of activities to provide an absolute reference (ground truth), while a second experimenter annotated the user's location ground truth simultaneously, both using the context recognition network toolbox[4] to synchronize the annotation streams with the data streams. We collected about 3,680 checking activities within 560 minutes of data. Figure 6 illustrates the recorded FSR and IMU signals.

Recognition of checking activities

Our data set contains a considerable amount of the null class (out of the 560 minutes of data collected, only 320 minutes cover actual activities). This includes all gestures and movements not directly related to a specific checking activity (for example, transitions between activities). Instances of identical activity classes can have different lengths and show variability (for example, the red FSR data of class "open trunk" in figure 6a), and data from different activity classes can show similarities (for example, the blue FSR data of classes "open trunk" and "close engine hood" in figure 6a and 6b). We segment the signal into interesting portions likely to contain an activity before classifying the data (see the "Activity Tracking" sidebar). These steps' algorithmic complexity must remain tractable for wearable computers with limited computational power.

To address these issues, we rely on cross-domain segmentation, which, for example, uses information about worker location and muscle activity to segment motion data. We also developed a new activity recognition algorithm, inspired by approximate string matching, that can perform signal segmentation and classification in a single step (see the "Activity Tracking" sidebar).

Lessons learned

We gained valuable experience from

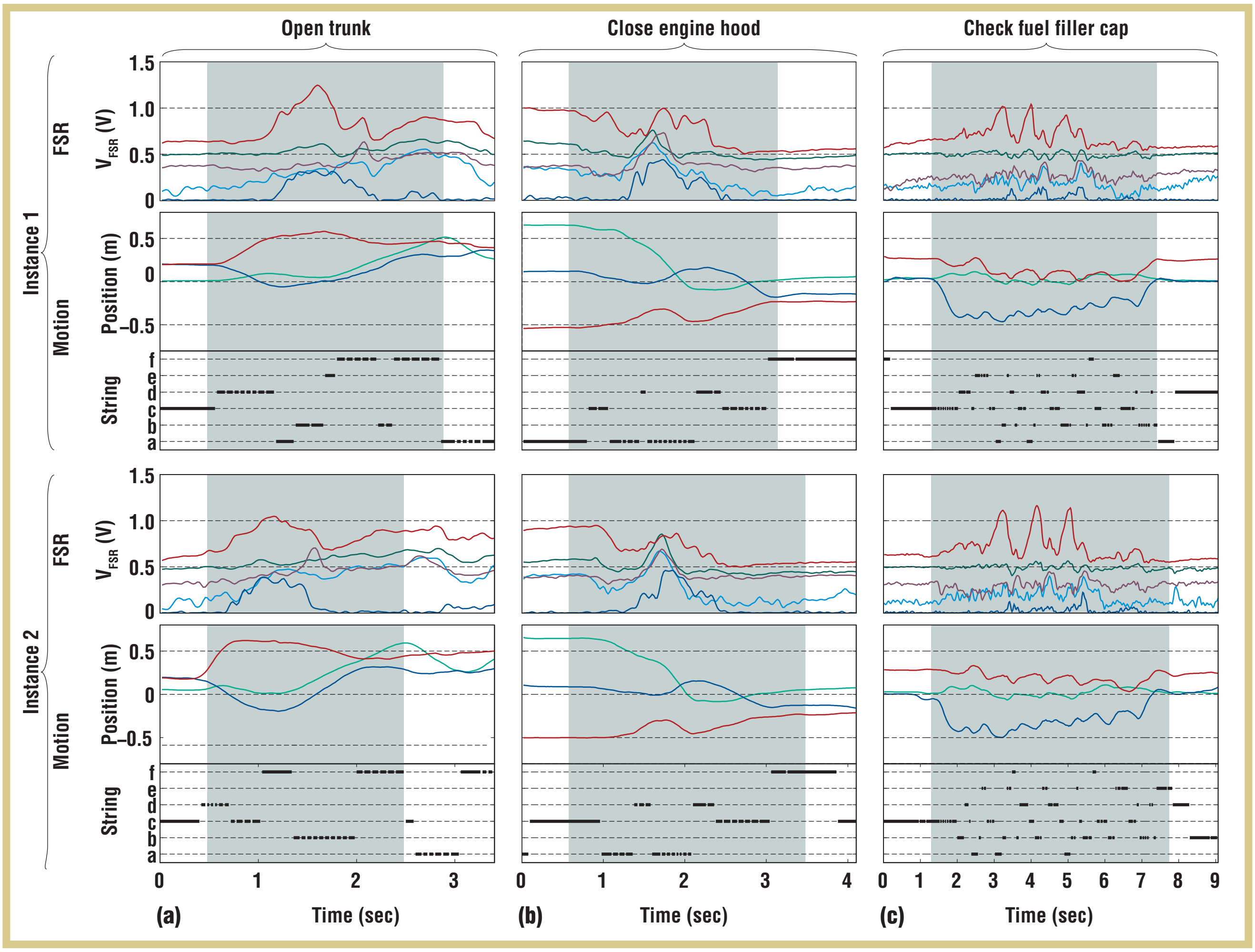


Figure 6. Signal segments from the muscle activity data (collected from FSRs) and right-hand motion data (collected from IMUs) for two instances of three activity classes: (a) open trunk, (b) close engine hood, and (c) check fuel filler cap. Motion data shows the hand position and corresponding encoded string representation. The shaded areas illustrate the manually annotated ground-truth information.

the checkpoint case study, which we think can be useful for other researchers working in the same field.

Data acquisition and annotation. We recorded data from 27 on-body sensors (seven IMUs, eight FSRs on each arm, and four Ubisense tags) running at different sampling frequencies while the worker freely performed a sequence of 46 diverse, at times subtle, activities. Generating a properly annotated, synchronized data stream was a major challenge. Our strategy included

- a hierarchical approach that synchronizes sensors belonging together as early as possible in the processing chain, preferably in hardware (this gave us five sensor groups: two IMU groups, two FSR arrays, and the Ubisense tags);
- dedicated software (the context recognition network toolbox) to synchronize the data from the five sensor groups together with the annotated labels during data collection;
- two experimenters working on separate, synchronized computers for the ground-truth annotation (activity plus location); and
- a continuous video record of all experiments to help us cope with potential labeling errors a posteriori.

Sensors. The motion jacket proved to be of great help with the recordings at the Skoda assembly plant. The workers didn't report any issue working with it. This confirmed our idea that embedding sensors into standard worker clothes would provide unobtrusive activity sensing. In our lab with the nonmoving car, the location estimation system's spatial resolution lies within the expected range of about 20 centimeters of absolute distance. In the production line, where we covered a working area

Activity Tracking

The first step of activity recognition is segmenting the continuous data stream by identifying sections that could contain meaningful activities. This is usually challenging because of the predominance of nonrelevant sections in the signal. Several methods for segmentation appear in the literature. These include hidden Markov models (HMMs),[1] dynamic time warping,[2] and methods based on feature similarity,[3] all of which are greedy in terms of computational complexity.

The second step is classifying the identified segments. You can use various classification algorithms: instance-based classifiers (such as nearest neighbor or nearest class center), rule-based classifiers (such as C4.5), stochastic approaches (such as HMM), or linear classifiers (such as support vector machines).

To recognize activities on wearable devices with limited computational power, we developed a string-matching-based segmentation and classification method that relies only on simple arithmetic operations.[4] Figure A illustrates the method's key processing steps. We capture and encode upper-body motion into a continuous string in which each character represents a distinct part of the motion (figure A1). We then match activity templates with this motion string (figure A2) using the operations equality (=), substitution (s), deletion (d), and insertion (i).

This gives a matching cost that indicates the similarity between the motion string and the activity template. As soon as the matching cost drops below a trained threshold, the method infers an activity occurrence of the corresponding template (figure A3).

REFERENCES

1. J. Deng and H. Tsui, "An HMM-Based Approach for Gesture Segmentation and Recognition," *Proc. 15th Int'l Conf. Pattern Recognition*, vol. 2, 2000, pp. 679–682.
2. M.H. Ko et al., "Online Context Recognition in Multisensor Systems Using Dynamic Time Warping," *Proc. Conf. Intelligent Sensors, Sensor Networks, and Information Processing*, IEEE CS Press, 2005, pp. 283–288.
3. E. Keogh et al., "An Online Algorithm for Segmenting Time Series," *Proc. Int'l Conf. Data Mining*, IEEE CS Press, 2001, pp. 289–296.
4. T. Stiefmeier, D. Roggen, and G. Tröster, "Fusion of String-Matched Templates for Continuous Activity Recognition," *Proc. 11th IEEE Int'l Symp. Wearable Computers*, IEEE CS Press, 2007, pp. 41–44.

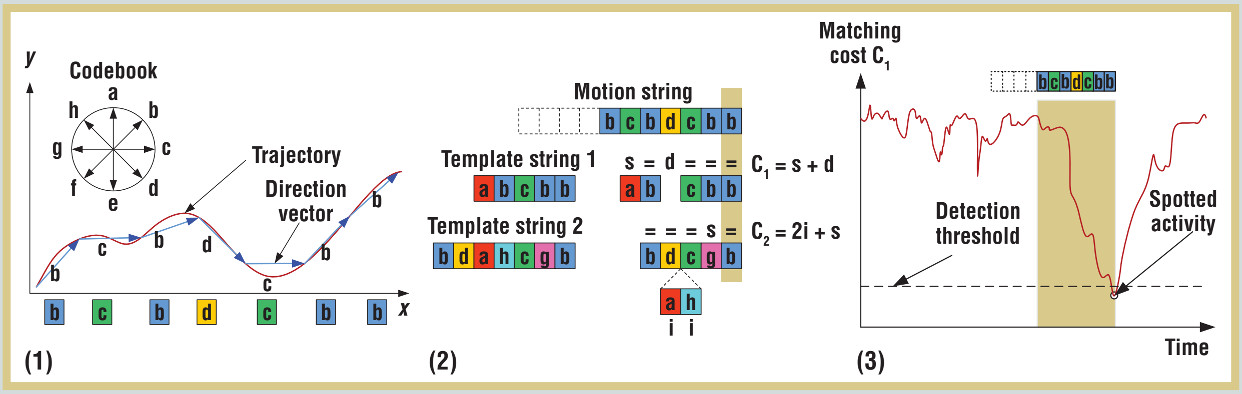


Figure A. Our string-matching-based segmentation and classification method: (1) upper-body-motion encoding using a codebook of quantized direction vectors; (2) activity-template matching with the continuous motion string, producing matching costs (C); and (3) activity spotting in the continuous matching costs (C_1).

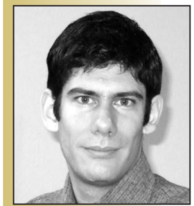
of 6 meters × 20 meters, the resolution is somewhat lower. The lower resolution is caused by stronger multipath propagation because of reflections, but it's still sufficient to discern the defined location classes around the cars on the conveyor belt.

Gesture segmentation and classification. We found that mere analysis of a single sensor domain isn't sufficient to spot workers' checking activities. So, we developed a multimodal segmentation method in which we use the information from one sensor domain to segment data in another sensor domain, effectively merging information from motion, muscle activity, and location. Our new string-matching-based method for spotting activities in continuous data streams has led to some promising results: On a subset of six activity classes, we reach a 74 percent accuracy rate in a 560-minute-long recording. This data set contains 480 relevant activity instances that all summed up are performed in 35 minutes.

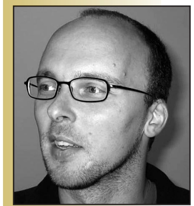
Industrial production is an attractive but demanding application field for activity recognition. Our work is an initial step toward

the AUTHORS

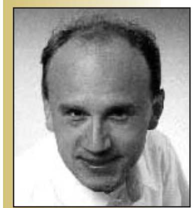
Thomas Stiefmeier is a doctoral candidate and member of the research staff at the Electronics Laboratory at ETH Zurich. His research interest is in context-aware computing and activity recognition. He received his master's degree in electrical engineering from the University of Technology Darmstadt. Contact him at Electronics Laboratory, ETH Zurich, Gloriastrasse 35, CH-8092 Zurich, Switzerland; stiefmeier@ife.ee.ethz.ch.



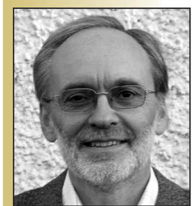
Daniel Roggen is a senior research fellow at the Electronics Laboratory at ETH Zurich. His research interests include context-aware and wearable computing and embedded and bio-inspired systems. He received his PhD in microengineering from the École Polytechnique Fédérale de Lausanne (EPFL). Contact him at Electronics Laboratory, ETH Zurich, Gloriastrasse 35, CH-8092 Zurich, Switzerland; droggen@ife.ee.ethz.ch.



Georg Ogris is a member of the research staff at the Embedded Systems Laboratory at the University of Passau. His research interest is in signal processing for context-aware computing. He received his master's degree in computer science from the University for Health Sciences, Austria, and his engineering degree in electronics from Technikum Wien, Austria. Contact him at ESL, Univ. Passau, Innstrasse 43, 94032 Passau, Germany; georg.ogris@uni-passau.de.



Paul Lukowicz is a full professor and chair of Embedded Systems and Pervasive Computing at the University of Passau. His research interests include wearable and mobile computer architecture, context and activity recognition, high-performance computing, and optoelectronic interconnection technology. He received his PhD in computer science from the University of Karlsruhe, Germany. Contact him at ESL, Univ. of Passau, Innstrasse 43, 94032 Passau, Germany; paul.lukowicz@uni-passau.de.



Gerhard Tröster is a professor and head of the Wearable Computing Lab at ETH Zurich, Switzerland. His research interests include wearable computing for healthcare and production, smart textiles, sensor networks, and electronic packaging. He received his PhD in electrical engineering from the Technical University Darmstadt. Contact him at Electronics Laboratory, ETH Zurich, Gloriastrasse 35, CH-8092 Zurich, Switzerland; troester@ife.ee.ethz.ch.

harnessing such applications. It has revealed several interesting insights:

- A good way to approach activity recognition in production environments is to start in a training setup, where it's easier to extensively augment the environment. In addition, collecting realistic data sets is less problematic during training than in production tasks.
- You can often place simple sensors in such a way that (nearly) unique signal combinations map to the individual activities. If you combine this with appropriate task modeling, you can achieve near-perfect recognition of even complex tasks, as our training scenario shows.
- In production, sensing is largely limited to body-worn sensors. Because many production activities are defined by a combination of body posture, arm and hand activity, and the worker's position with respect to the assembly line, our sensor combination is broadly applicable.
- Realistic recreation of production conditions in the lab involves considerable effort (in our case, it involved bringing entire cars into the lab). However, it's a worthwhile strategy if you need to collect large data sets.
- Using a large number of sensors is important for robust recognition, but it makes data collection difficult. So, you must carefully plan synchronization and labeling. Hierarchical synchronization, adequate streaming tools, and strict, redundant labeling protocols are crucial.

The next step in our work is a detailed evaluation of the quality-control data set. We're working on combining three sensor modalities (motion, muscle activity, and location) by fusing segmentation results from different sensor sources to complement the distributed pieces of information. The evaluation of our segmentation method will use such standard measures as correct, deletion, substitution, and insertion rates. The final goal is a reliable real-time recognition system.

ACKNOWLEDGMENTS

We performed all the work described in this article within the European Union project wearIT@work (contract number 004216). We acknowledge the contributions of our wearIT@work colleagues.

REFERENCES

1. P. Lukowicz et al., "The wearIT@work Project: Toward Real-World Industrial Wearable Computing," *IEEE Pervasive Computing*, vol. 6, no. 4, 2007, pp. 8–13.
2. T. Stiefmeier et al., "Event-Based Activity Tracking in Work Environments," *Proc. 3rd Int'l Forum Applied Wearable Computing*, Springer, 2006, pp. 91–100.
3. J.A. Ward et al., "Activity Recognition of Assembly Tasks Using Body-Worn Microphones and Accelerometers," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, 2006, pp. 1553–1567.
4. D. Bannach et al., "Distributed Modular Toolbox for Multimodal Context Recognition," *Proc. Architecture of Computing Systems Conf.*, LNCS 3894, Springer, 2006, pp. 99–113.