Overview

1. Motivation and Introduction
2. Sensors
3. Sensor Output and Connections
4. Power and Sensors
5. Designing a Sensor System
6. Low-level Processing
7. Perceptual Components
8. Matching and Learning
9. Context and Situation
1. Motivation

How to describe a Situation?

It is difficult to describe and detect a situation
- A car is going to have a serious accident
- Two people are undecided what to buy
- Someone is sleeping in a room
- A family having dinner

…but often it is a prerequisite to recognized situations for building intelligent objects.

…and it is even harder to predict a future situation – but we (humans) do it all the time.
How is a situation characterized using sensor value?

Example: Someone is sleeping in a room in a care home

Sensors
- Motion sensor overseeing the room (ON/OFF)
- Weight sensor in each leg of the bed (0-100)
- Light sensor (0-100)
- Door sensor (OPEN/CLOSE)
- Pressure mat in a rag on the floor (ON/OFF)
- Microphone providing noise level (0-100)

Find a function that takes sensor values as input and that tells if someone in sleeping in the room or not

How is a situation characterized using sensor value?

Example: Someone is sleeping in a room in a care home

Issues
- Sensing over time required
- Calibration (at least initially)
- Function is dependent on the sensor setup and the user
- Function is not always correct (exceptions)
- Some sensors don’t contribute
- Learning as an option

Even in this simple case it is not trivial to set-up system
Perception in Nature I

- Having **perception** and cognitive functions are the **foundation of intelligent behavior** of creatures.

- **Acting and reacting with respect to the current situation is a basic property of most intelligent systems.**

- Looking at flora and fauna it is a major advantage in the struggle for survival to have the ability of **being adaptive**. The capability to adapt to new circumstance and situations is a vital quality for virtually all living organisms and a major advantage in the struggle for survival.

Perception in Nature II

- Senses in nature cannot be directly compared to sensors in a technical world.

- Senses comprise the whole process from the reception of the stimulus, translation from stimulus to signal, signal transport and the processing on several levels.

- **Vision**
- **Hearing**
- **Smell**
- **Taste**
- **Touch**
- **Temperature**
- **Gravity and acceleration**
- **Position and constellation of (body) parts**
- **magnetic fields**
- **Electric fields**
Perception in Nature III

• **To understand** or at least interpret information that is sensed from the environment *knowledge* or *experience* (or memory) is **required**.

• Creatures learn during their development how to **assign meaningful and abstract situations to complex stimuli** received by the sensory system. This is based on the presupposition that similar situations are characterised by similar stimuli.

• **Comprehension** of a situation or **understanding** of the implications given by a situation is a further step, which is to a great extent based on the **recall of experience**.

Situation and Context

• **Situation**
  A situation is the state of the real world at a certain moment or during an interval in time at a certain location.

• **Context**
  A Context is identified by a name and includes a description of a type of situation by its characteristic features.

• **Situation S belongs to a Context C**
  A situation S belongs to a context C when all conditions in the description of C evaluate to true in a given situation S.
Context Recognition

• It is assumed that for all situations that belong to the same context the sensory input of the characterizing features is similar.

• Creating a description of a context includes similar problems to creating a query for information retrieval. To assess the quality of a description measures such a precision and recall, well known from information retrieval.

• Based on these definitions context can be regarded as a pattern, which can be used to match situations of the same type.

Warning at the beginning
There are limitations…

“The physical world is a partially observable dynamic system ...”

“... sensors are physical devices have inherent accuracy and precision limitations”
2. Sensors

What is a Sensor?

• A sensor is a technological device or biological organ that detects, or senses, a signal or physical condition and chemical compounds.

• A electronic, electrical, micro-mechanic or electro-mechanical device that responds to a stimulus, such as heat, light, or pressure, and generates a signal that can be measured or interpreted.

• A function of time that returns a value (binary, number, vector, array) dependent on a measured parameter.
Some “classical” Sensors

- light sensors: photocells, phototransistors, CCDs,…
- sound sensors: microphones, seismic sensors…
- temperature sensors: thermometers, thermocouples, thermistors, …
- radiation sensors: Geiger counter, dosimeter
- electrical resistance sensors
- electrical current sensors
- electrical voltage sensors
- electrical power sensors
- magnetism sensors: magnetic compass, Hall effect device, …


Information “Sensors”

- Sensors that are related to the device or system
  - Examples
    - battery voltage,
    - RSSI,
    - real-time,
    - current packet loss,
    - current power consumption
    - location sensors
    - devices in vicinity

- Access to information over a network (e.g. WWW)
  - weather in New York
  - share price of GOOGLE
Bio-Sensors

- Sensors to measure physiological parameters in humans and animals
- Towards sensing emotions...

Example
- Galvanic skin response
- Heart rate
- Blood pressure
- Blood oxygen saturation
- EEG, ECG
- ...

Image from [http://affect.media.mit.edu/](http://affect.media.mit.edu/)

What can you measure?
Some Examples

- Temperature Sensor
  - weather / temperature
  - human proximity and touch
  - device in operation
  - indoor / outdoor?
  - speed?
  - ...

- Light Sensor
  - light level
  - light frequency (50Hz/60Hz)
  - indoor / outdoor?
  - movement?
  - usage of a environment
  - touch
  - ...

- Accelerometer
  - tilt
  - vibration
  - acceleration
  - gestures
  - shock
  - position?
  - Interaction?
  - ...

Dependent on the application a sensor can be used to measure different phenomena in the real world
Sensor in Detail
Analog Devices Accelerometer 1

- iMEMS
  integrate micro electrical mechanical systems
- Everything is integrated on a chip
  – mechanics and circuits!

From www.analog.com

Sensor in Detail:
Analog Devices Accelerometer 2

From Analog Devices, ADXL202
Read the datasheet …

\[ A(g) = \frac{1}{T_1/T_2 - 0.5y/12.5\%} \]
\[ 0g \leq 50\% \text{ DUTY CYCLE} \]
\[ T_2 = \frac{R_{\text{SET}}}{125\mu\Omega} \]
Sensor in Detail:
Load cell

Compressive load transducer

- Analog Output
- Stain Gauges measure change in length
- From RS Components Datasheet 232-5957

Sensor in Detail:
Gas Sensor 1

Model: TGS 21XX
Element type: S1
Package type: Plastic

Configuration: Single-sided, 1 element
Features: Simple structure
Benefits: Suitable for large volume and low-cost applications

Pin connection:
1. Sensor Electrode (-)
2. Common (+)
3. Heaters (+)

Circuit conditions:
V: 5V (DC)
V: 5V (DC)
R: Variable (50 kΩ)

From: Frigaro Gas Sensors Series 2000
Sensor in Detail: Gas Sensor 2

Problems with Sensors

- Need for calibration
- Sensors are Inaccurate (within a given specification)
- Sensors are unreliable (within a given specification)
- Noise and false readings are common
- Timing between processor and sensor is often critical

- Mechanical Issues, casing
  “Sensor may need a hole to see the world”
3. Sensor Output and Connections

Acquiring Sensor Data

- In many cases “Analog” phenomena are sensed
- Analog to digital conversion (ADC) is required
- Sampling rate as a central parameter that describes how often digital samples are taken from the analog signal
- ADC in the sensor, as extra component, or in MCU

- Issues for selecting the sampling rate
  - Speed of change of the parameter in the world
  - For reconstruction of the frequency \( f \) we need at least \( 2f \) (Nyquist)
  - Speed of change supported by the sensor
  - Capabilities of the ADC
  - Power consideration (energy saving)
Basic Electronics

(Do you remember from High school?)

- Voltage division
- Resistor may be a sensor
- Pull-up resistor
- Input is in a defined state

\[
Out = \frac{R_2}{R_1 + R_2} \cdot VCC
\]

\[
Out = \begin{cases} 
VCC & \text{if } S1 \text{ opened} \\
GND & \text{if } S1 \text{ closed}
\end{cases}
\]

Digital Output

- Sensors with binary state
e.g. on/off
- Typical sensors
  - Push button
  - Switch
  - Ball switch
- Output after threshold component (Schmitt-trigger) of an analog sensor
- To acquire read digital input pin

Repeat ... as long as S1 is pressed
\[
\text{while} \left( \neg \text{input}(\text{pin}_B) \right) \{ \ldots \}
\]
Analog Output

- Sensor with a analog output, e.g. IR distance sensor, PIR
- Sensors as variable resistor e.g. LDR, Pressure sensor, PTC

To acquire read analog port

```c
setup_adc(ALL_ANALOG);
set_adc_channel(0);
value=Read_ADC();
```

Pulse Width Modulation (PWM) Duty-Cycle

- Simple digital interface to communicate a number
- A value is code using the timing in a digital output
- Reading a digital line and measure the time
- Examples: Accelerometer, sensor modules
- Acquire using a digital in counting time for high and low

```
wait till B0 is high
counter=0
start counter
wait till B0 is low
T1=counter
wait till B0 is high
T2=counter
```

\[ A(g) = \frac{T1}{T2 - 0.5} \times 12.5\% \]

\[ O_g = 50\% \text{ DUTY CYCLE} \]

\[ T2 = R_{SET}/125\Omega \]
Frequency Output

• The output frequency is used to communicate the sensor value
• Similar to PWM
• Use a counter to read frequency

![Frequency Output Diagram](image)

I2C bidirectional Communication

• Bus topology
• Master-Slave protocol
• Usually MCU is master and sensor(s) slave
• Electrical Connection
  – SDA (Data)
  – SLC (Clock)
  – GND
• Devices
  – Different sensors
  – external memory
  – further components
• Read/write primitives

![I2C Connection Diagram](image)

http://www.esacademy.com/faq/i2c/
Serial Line (TTL, RS232)

- Protocol from the “terminal world” ;-
- Commonly used to interface to more complex sensors
- Minimal electrical connection
  - (TXD)
  - RXD
  - GND
- For 12V is a driver required (e.g. MAX233)
- Typical Examples
  - GPS
  - RFID Reader
  - Connecting sensor or receiver to a PC

Many More…
Communication with Smart Sensors

- 1-Wire Bus, data and power over one line
  E.g. I-Button, Temperature Sensor
- SPI digital output (and input)
  Serial digital interface
- RS485, RS422 serial bus for longer distances
- wired communication
- IEEE1451, protocol for smart sensors
- …

- Often depends on the sensors used and general requirements in the projects.
4. Power and Sensors

Minimizing Power Consumption
Generate Power from Sensors

- Use sensor to wake up a processor
- Use sensors that generate power
- Build circuits that generate an interrupt on change of sensor values

- Example
  - Solar cell as light sensor with no power
  - Piezoelectric Element

- Search for *Parasitic Power Harvesting*
Minimizing On-Time

• Use clever sampling strategies
  – switch the whole system (including sensors) off between sensor readings
  – sample at low speed in general
  – Increase speed when something interesting happens

• Example – detect gesture interaction
  – Switch of the system for a time that is acceptable as delay for recognition (e.g. 250 ms)
  – Switch system on, power sensor read a sample and compare to previous values (e.g. will take 5ms)
  – Only go into fast sampling mode if there is change
  – Results in much lower energy consumption (e.g. 2%)

Example
Check every 250ms for a change

```c
while(TRUE) {
    power_down_ms(250);
    power_sensor(SENSOR_1);
    x=read_sensor(SENSOR_1);
    if (diff(x,xold) > THRESHOLD) { recognizer(); } 
    else { xold=x; }
}
```

Power consumption

- 20 mA
- 500 uA
Sensor Hierarchies
Variable Processing Power

- Use a low-power (or no-power or energy harvesting) sensor to monitor
- Power the sensor with high power consumption only when a change is expected/predicted

- Example: Porcupine
  - Ball switches monitor change at low processing speed with minimal power consumption
  - In case of change accelerometers are powered and processing speed is increased
  - Kristof van Laerhoven, http://www.comp.lancs.ac.uk/~kristof/research/notes/porcupine/

Example Porcupine
Body Sensor Network

From: http://www.comp.lancs.ac.uk/~kristof/research/notes/porcupine/
5. Designing A Sensor System

Requirements on Sensing in a Ubiquitous Computing Environment

- Design and Usability
- Energy Consumption
- Calibration
- Start-up Time
- Robustness and Reliability
- Portability, Size and Weight
- Unobtrusiveness, Social Acceptance and User Concern
- Price and Introduced Cost
- Precision and Openness

A. Schmidt, and K. Van Laerhoven, *How to Build Smart Appliances?*, IEEE Personal Communications 8(4), August 2001
Arranging Sensors

- The position of sensors on an object or in the environment matters!
- Dependent on the position different phenomena will/can be measured
- The sensor in the "right" position can save processing and energy
- Embodiment – see robotics

Example: placement of acceleration sensors in an interactive cube

Multiple Sensors

- Multiple sensors (of the same type) can ease recognition of certain phenomena
- Correlation of sensor readings
- Networked sensors and time stamped readings

Example: detect the number of sound sources
  - very difficult with one microphone
  - much simpler with multiple distributed microphones
Sensing Options and Context Use

- Sensing
  - Observation from the outside (extrinsic)
  - Sensing from within (intrinsic)
  - combined

- Context used by
  - Entity
  - Observer
  - Anyone

<table>
<thead>
<tr>
<th>Context user</th>
<th>Entity</th>
<th>Observer</th>
<th>Anyone</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intrinsic</td>
<td>No communication</td>
<td>communication</td>
<td>communication</td>
</tr>
<tr>
<td>Extrinsic</td>
<td>Communication</td>
<td>No communication</td>
<td>communication</td>
</tr>
<tr>
<td>combined</td>
<td>communication</td>
<td>communication</td>
<td>communication</td>
</tr>
</tbody>
</table>

6. Low level Processing
Buffering Data, Histories

- Motivation
detect a conversation from audio?
  - Not possible with a snapshot
  - History is required

- In many cases sensor readings need to be considered over time to get meaningful information
- Sensor data is buffered (e.g. the last 50 values)
- Processing of sensor data from the buffer
- Issues
  - Size of buffer
  - Time stamp vs. samples at fixed intervals
  - Appropriate methods and algorithms for processing

Basic Statistics

- Motivation
data gathered is not perfect (e.g. outliers, faulty readings)
  - Features (e.g. change, average) are of interest rather than a single value

- Basic statistics are in many cases computationally cheap
- Can help to reduce effort for calibration
- Typical Features
  - average, median,
  - range, interquartile range,
  - variance, standard deviation
- Change of sensor values vs. absolute values
Filtering Sensor Readings

- Can be implemented in hardware or software
- Dealing with noisy data
- moving average
- low-pass filter
- Many more …

Time domain and Frequency domain

- Sensor values are sample in discrete time steps
- Often changes are of central interest (e.g. “it is getting colder”)

- Analyzing over time
  - derivatives, 2nd (higher) derivatives
  - summing up sensor values over time, integration
  - Summing up difference between sensor values

- Transformation into Frequency domain
  - Counting zero crossings to get base frequency
  - FFT
Feature Extraction

- Features over a given time interval are calculated
- Features are characteristic for a context
- Features are used in higher level processing and for recognition

Example
- Audio signal of 4 microphones over 30 seconds
- Possible Features (depend on contexts to recognize):
  - average audio level, variance of audio level
  - correlation between microphones
  - distribution of audio levels, number of distinct sound sources
  - frequency spectrum for 2 second periods

7. Perceptual Components
Sensing and Perception

- Bridging the gap between sensors and applications
- Sensors observe physical phenomena in real the world
- Applications use (implicit or explicit) world models
- Perception: associating sensor observations with meaning

- The world is represented as a set of collection of sensor reading
  - Numeric, symbolic or streams of data
  - Can be considered as observable variables
  - May contain meta data (e.g. time, location, confidence)

Perception Model
Basic Perception Component

- Transforming observed features/events/data to “higher level” features/events/data
- Transformation can be controlled by the system or by context
- Perception as multi-step process

Associating observations with entities

- grouping of observations
- entity corresponds to a physical object
- easy if sensors are directly associated with an object
- hard if sensors are external

Detecting relations

Grouping

- Determining relations between entities
- Easier with sensors external to the entities
- Harder with embedded sensors
Example: A Context-aware Table

• Idea: augment a table to be context-aware
  – Instrument with sensors to detect activity on the table surface
  – Use perception techniques to extract context
  – Use context information to support different applications

• What we might want to detect
  – Placement and removal of things on the table
  – Movement on the table surface
  – Identity of objects on the table

Sensor system for the table

• Load cell in each corner
• Measuring forces
• Trade-offs:
  – Accuracy
  – Speed, Sampling rate
  – Maximal load
Higher level Perception

- Basic event detection
  - Change in load at x,y
  - Increase in load by w: an object of weight w has been placed at x,y
  - Reduction in load: object removed from x,y

- Detecting movements
  - Track change in load distribution on surface
  - Continuous change is associated with movement

8. Matching and Learning
Rule based approach

• A (fixed) set of rules that specify a context
• Explicit definition of context parameters (features) to match a context
• In many application scenarios this is very simple to implement
• It is easy for small number of features and contexts that are well understood

• Example mobile phone
  - in_hand := (touch==TRUE) && (acceleration > EPSILON);
  - in_suitcase := (touch==FALSE) && (light == DARK)
  - ...

Off-line learning

Supervised training

• The context is learned by “experience”
• Data examples for a context is learned base on sensor data or feature
• The training data is collected in typical situations that belong to a context
• In a new situation the received stimulus (sensor data / features) are compared to the data learned
• Different algorithms, e.g.
  – Statistics
  – Nearest Neighbor matching
  – Backpropagation Neural Networks
• Useful for contexts that do not change but where the relation between sensor values and situation is not easily understood
On-line/Continuous learning
unsupervised learning

- sensor data or features are continuously used to learn a context
- clustering data and labeling clusters
- useful for changing environments
- various methods, e.g. Self Organizing Map

- Simple example – User’s favorite place
  - Base station ID as feature
  - Measure every minute the ID
  - “learn” the user favorite place
  - This relates to a time frame (e.g. favorite place over the last month)

Learning and adaptation

<table>
<thead>
<tr>
<th>Concept of Learning/adaptation</th>
<th>Usage</th>
<th>Algorithms</th>
<th>examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>No learning, fixed</td>
<td>Contexts are globally valid</td>
<td>Design time data analysis</td>
<td>Static Rule based systems, Preset Supervised NN</td>
</tr>
<tr>
<td>Learning phase</td>
<td>Contexts are stable but different depending on the use case</td>
<td>Training and/or data analysis capabilities built in</td>
<td>Dynamic Rule based systems Supervised Neural nets</td>
</tr>
<tr>
<td>Fully adaptive, always learning</td>
<td>Contexts are changing over time</td>
<td>Adaptive algorithms</td>
<td>SOM, ISL</td>
</tr>
</tbody>
</table>
9. Context and Situation

Modeling Context

• Modeling the domain
• Alternative approaches
  – Top-down
    Situation → Context → Features → Sensors
  – Bottom-up
    Sensors → Features → Context → Situation

• Do not try to model the world…
  …model your applications world!
Bottom-up Context Models

• Context is anchored in artifacts
  – Modeling and acquiring context on entity level
  – More general properties
  – Flexible, extensible, and simple model
  – Exploiting domain knowledge

• Augmenting artifacts with
  – Sensing
  – Processing
  – Communication

• Context related to interaction with the artifact
  – Combining context on a higher level
  – Time & space correlation

Bottom-up Context - Example

• sofa
  – free
  – occupied with one person
  – occupied with two people
  – occupied with three people

• door
  – open
  – closed
  – degree of openness
  – interaction

• briefcase
  – empty
  – loaded
  – open
  – closed
  – interaction

• sofa (over the top)
  – …
  – jumping on the sofa
  – motion of people on the sofa
  – temperature on the sofa
  – pouring orange juice on the sofa
  – pouring wine on the sofa
  – pouring milk on the sofa
  – cleaning the sofa
  – moving the sofa
  – sofa placed on the stairs
  – sofa upright
  – upside down
  – sofa flying in midair
  – …
Higher Level Context - Example
Time & Space correlation

- door → closed
- projector → on
- table → interaction
- chair 1 → occupied
- chair 2 → occupied
- light → on
- ... → ...

Layered Model for Sensor based Context Awareness

- Application primitives

  if enter(v, p, n) then action(i)
  if leave(v, p, n) then action(i)
  if in(v, p, m) then perform action(i)
Distributed Model for Sensor based Context Awareness

- Relevance of context information is related to physical distance

Summary

- Sensor provide means to see the real world
- Many different sensors and technologies are out there → read the datasheet
- Power saving and sensor system design are closely related
- Make abstraction/processing into components
- A leveled approach to derive concepts and contexts from sensor
- Learning is a key for many applications
References


• iMEMS, Accelerometers, http://www.analog.com/

• I2C FAQ, http://www.esacademy.com/faq/i2c/


• Kristof van Laerhoven, Porcupine Project http://www.comp.lancs.ac.uk/~kristof/research/notes/porcupine/

