Reality Media Workshop — Introduction to Data Analysis

Biased, Best Practices

Kai Kunze

Disclaimer

My background is Pervasive/Wearable Computing.

So I can show you the tools we are using in the community.

Data analysis/processing is a very broad term

- A lot of different definitions
- A lot of different tools

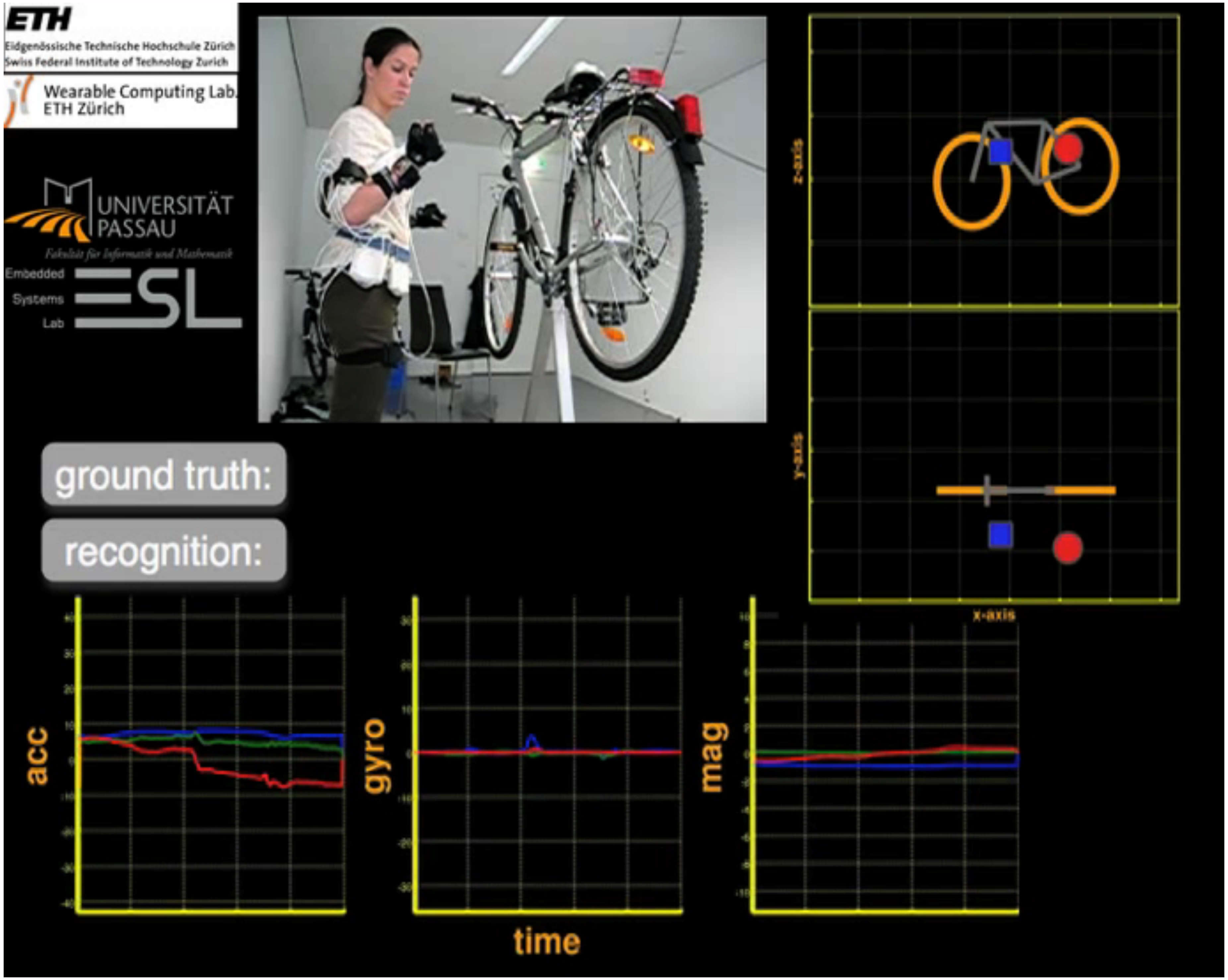
In this tutorial I show you what works for me:

- Methodology
- Useful tools

Most important: Have fun!

Overview

Activity Recognition



Research Methodology

As researcher your product is not data or code

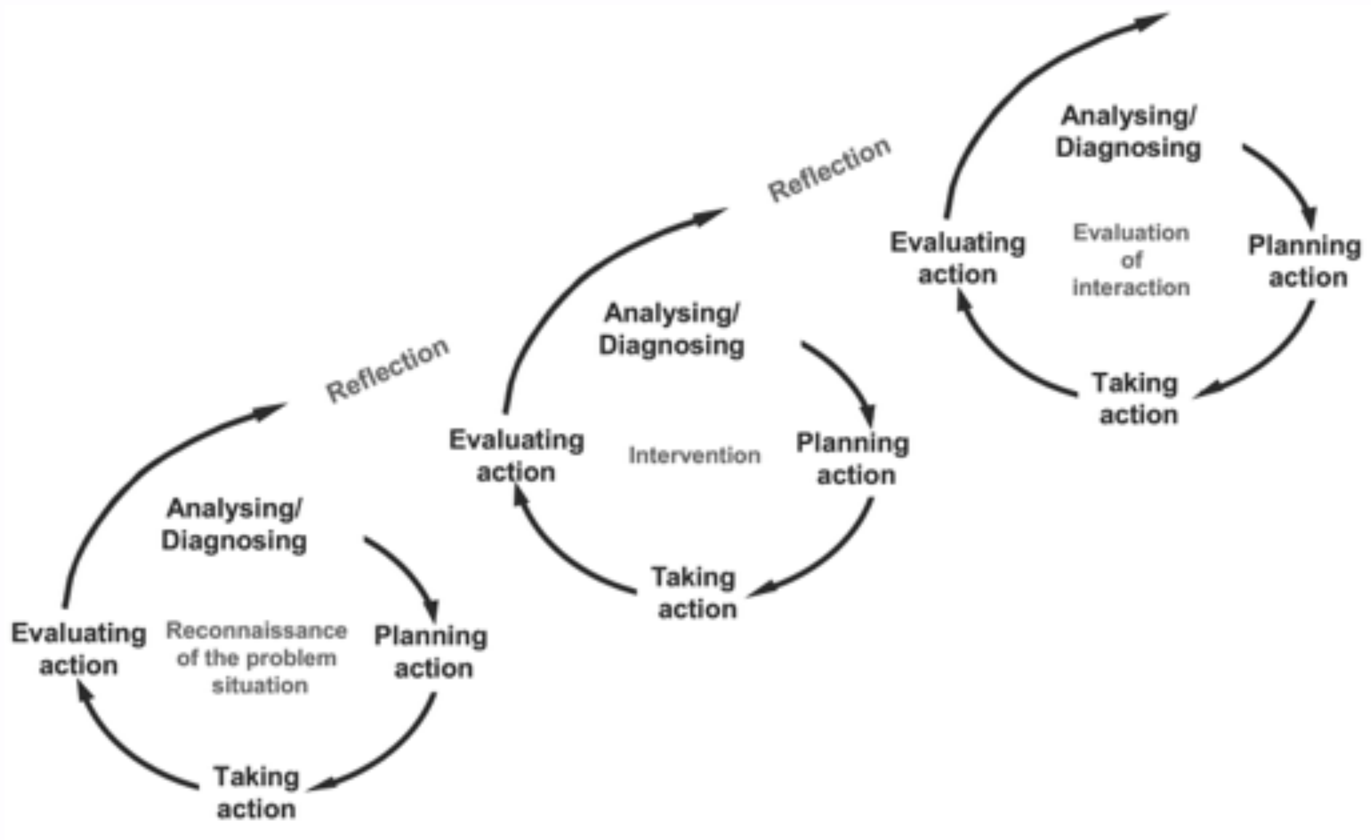
Your product is knowledge

Empirical science uses data and code to obtain knowledge

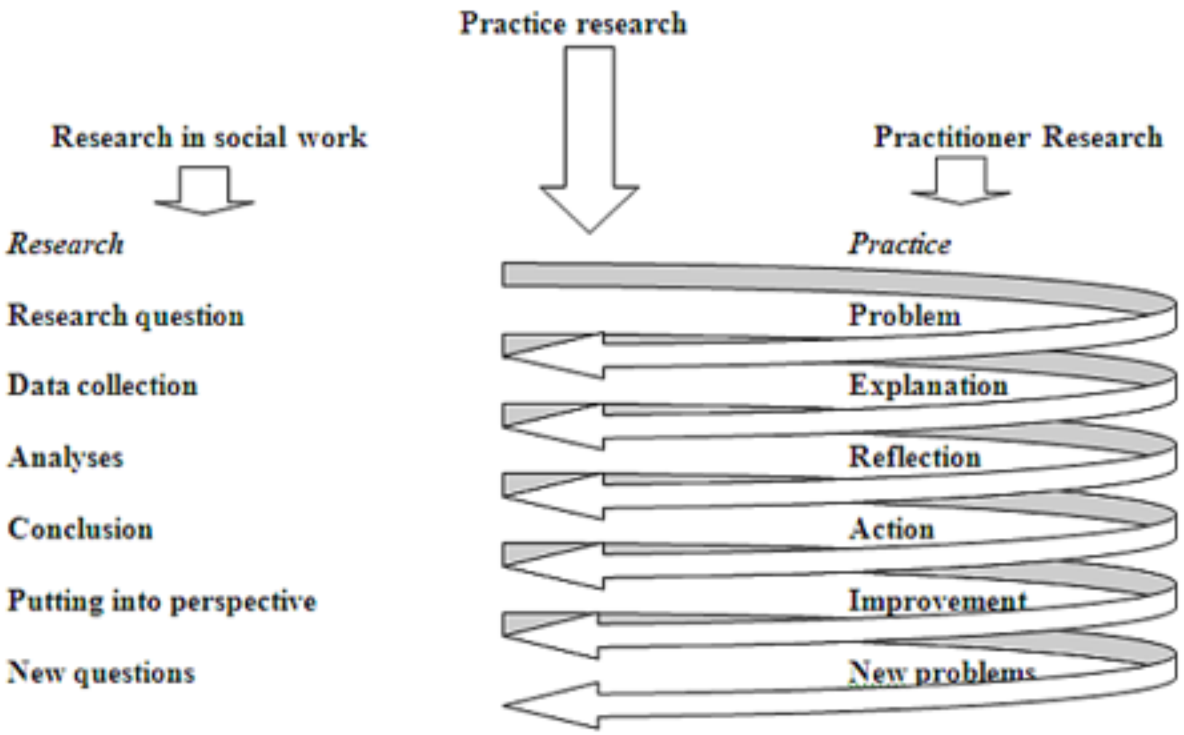
Good research is about reproducibility

Good design is about experience

Research "Life" Cycle



Source: Adapted from Coghlan and Brannick (2001), p. 19; Cardno and Piggot-Irvine (1996), p. 19



Formulate your Problem

What do you want to do?

Try to be as concrete as possible

“I want to use wearable technology to help users to read more”

Versus

“I want to use physiological sensors on smart eyewear to track user’s concentration while reading and design interventions to help them focus more”

Where to get data?

Standard Datasets

Own Experimental Design

Difficult !! Don't underestimate the design

Some good starting points (for UI design experiments, yet also valid for other designs)

http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-831-user-interface-design-and-implementation-spring-2011/lecture-notes/MIT6_831S11_lec14.pdf

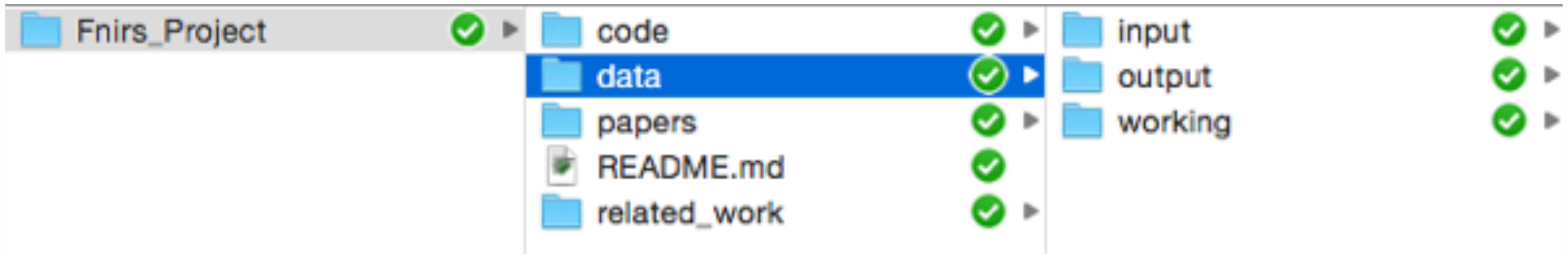
http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-831-user-interface-design-and-implementation-spring-2011/lecture-notes/MIT6_831S11_lec15.pdf

How I handle data

- Separate code from data
  - One data set, multiple types of analysis
- Have separate directories for data
  - Suggestion: input, working, output
  - input: never changes!
  - working: calculated features, processing steps
  - output: classification results etc.
- NEVER change the raw data directly
- NEVER do changes by hand to the data …

Sample Project Folder Layout

- My_Project
  - Data
    - Working
    - Input
    - Output
  - Code
    - Matlab
    - Python
    - C
  - README.md



Fnirs_Project

code
data
papers
README.md
related_work

input
output
working

Version Control

- Use a distributed version control system
  - My “favorite”: git
- I use it for everything … (except DATA)
  - Papers
  - My website
  - Every text file
  - For most writing I use Markdown
- Text files are your friends ☺
- Don’t use git for data (use .gitignore file to exclude it)

Version Control with Git

Demo

How I use Git — Notes

```
git co -b <new feature I want to work on>

git tag <conference shortage of paper submitted e.g. ah2015>

git co -b <old analysis branch> <tag name e.g ah2015>
```

For more background checkout tutorials on the web or the git book:

http://git-scm.com/book/en/v2

Save and Backup

- Save everything to disk frequently
  - Features you calculated
  - Data preprocessing steps
  - Your models, your results
- Disk-space is cheap, use it
- Use naming conventions:
  - For example one of my working directories:
  - Working/2012-12-02-features-accel-sw100.mat
- Dropbox is nice
  - if the data is not too large and sensitive (privacy!!)
  - works for code (I use github working copies in Dropbox)

Test Driven Development

Test the data + code as early and often as possible

- Work with input /output files

General tests used for:

- Prototyping language (perl, python ...)
- Demo implementation in faster language (C, C++ ...)

Estimate the timing of your methods

My work is usually paper deadline driven

This means I need good **time estimates!**

Take Away Messages

Experimental Design is difficult

- start early, try to get as much input from other people as possible

Use Version Control

Use folder structures (with documentation and tests)

Use Text files

Save everything frequently (intermediate steps)

Make it easy to execute part of the analysis (modular setup)

Pick the right tool for the right purpose

These things work for me … maybe not for you.

So you know one programming language and want to use it for everything? Forget it …

Remember we want to **produce knowledge**

There are a lot of data processing/analysis software out there:

Matlab, Mathematica, Maple, S, Strata

Octave, Sage, R

Libraries in c, java, pyhton, ruby, javascript, perl

Every software comes with advantages and disadvantages!

These things work for me … maybe not for you :)

Editor of choice: vim

Commandline tools

- screen, cat, grep, sed, head, tail, awk, find, xargs, sort, wc …
- Zsh, vim
- I use YADR (for mac): http://skwp.github.com/dotfiles/

For the initial data processing

- Matlab, python (ipython, scipy …)

For plots sometimes: R (R-Studio is nice and cross platform)

For demos, production code:

Depends, whatever does the job

A good knowledge of C is very helpful ☺

First Case Study: Workshop Scenario

Let’s assume you need to build a prototype for a project

Support of a worker during an assembly scenario

Requirements:

- You should recognize the following tasks:
- hammering, screw driving, sand papering, sawing
- The classifier should run on an embedded platform
  - Linux –arm, C implementation

How do you start?

Plotting

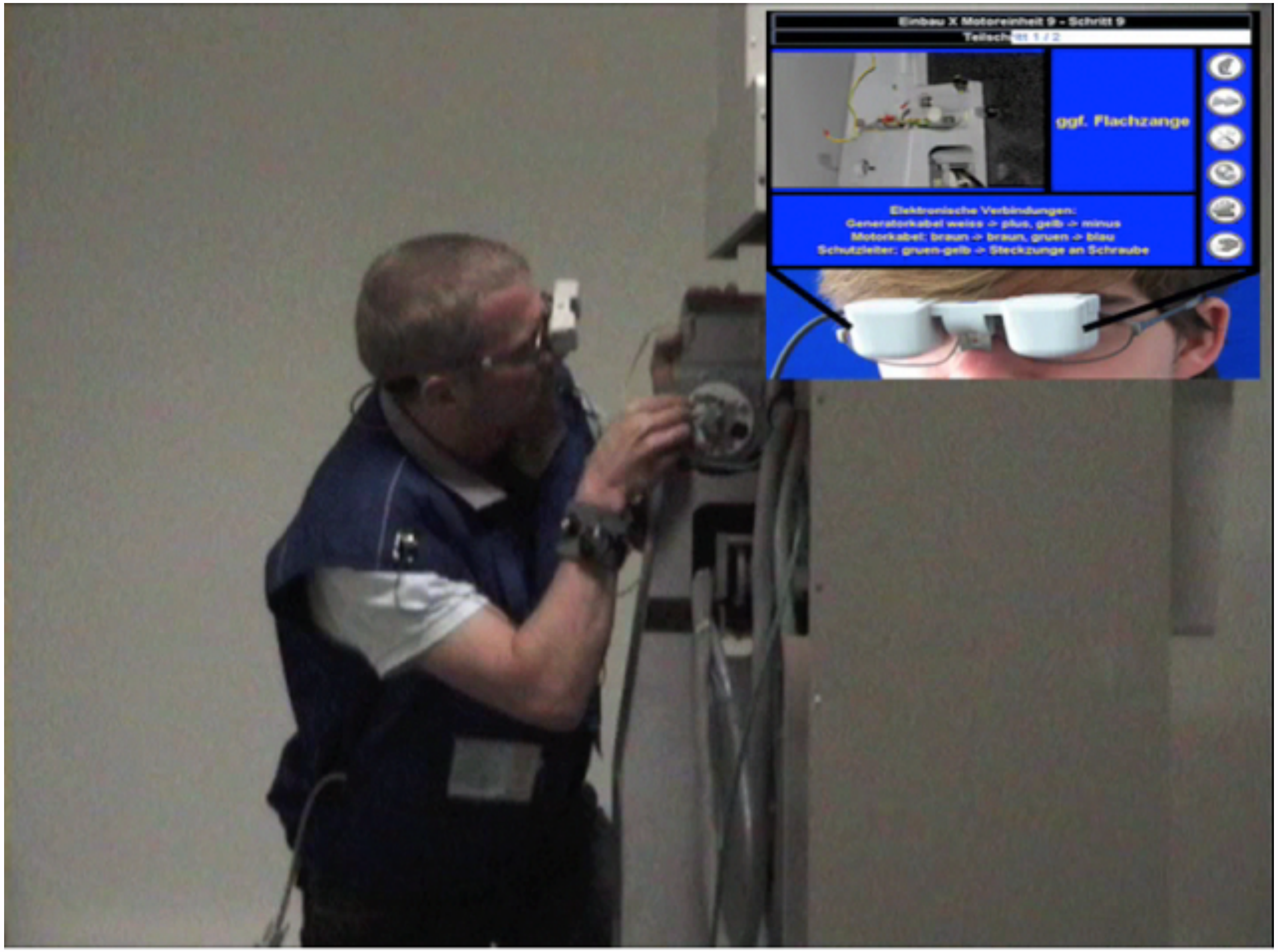
Don’t use Excel!

Matplotlib got better.

I often use R (although I cannot program R)

Let’s take a look at Matlab/ ipython etc.

Maintenance Scenario —Zeiss—



Einbau X Motoreinheit 9 - Schritt 9

Teilschritt 1 / 2

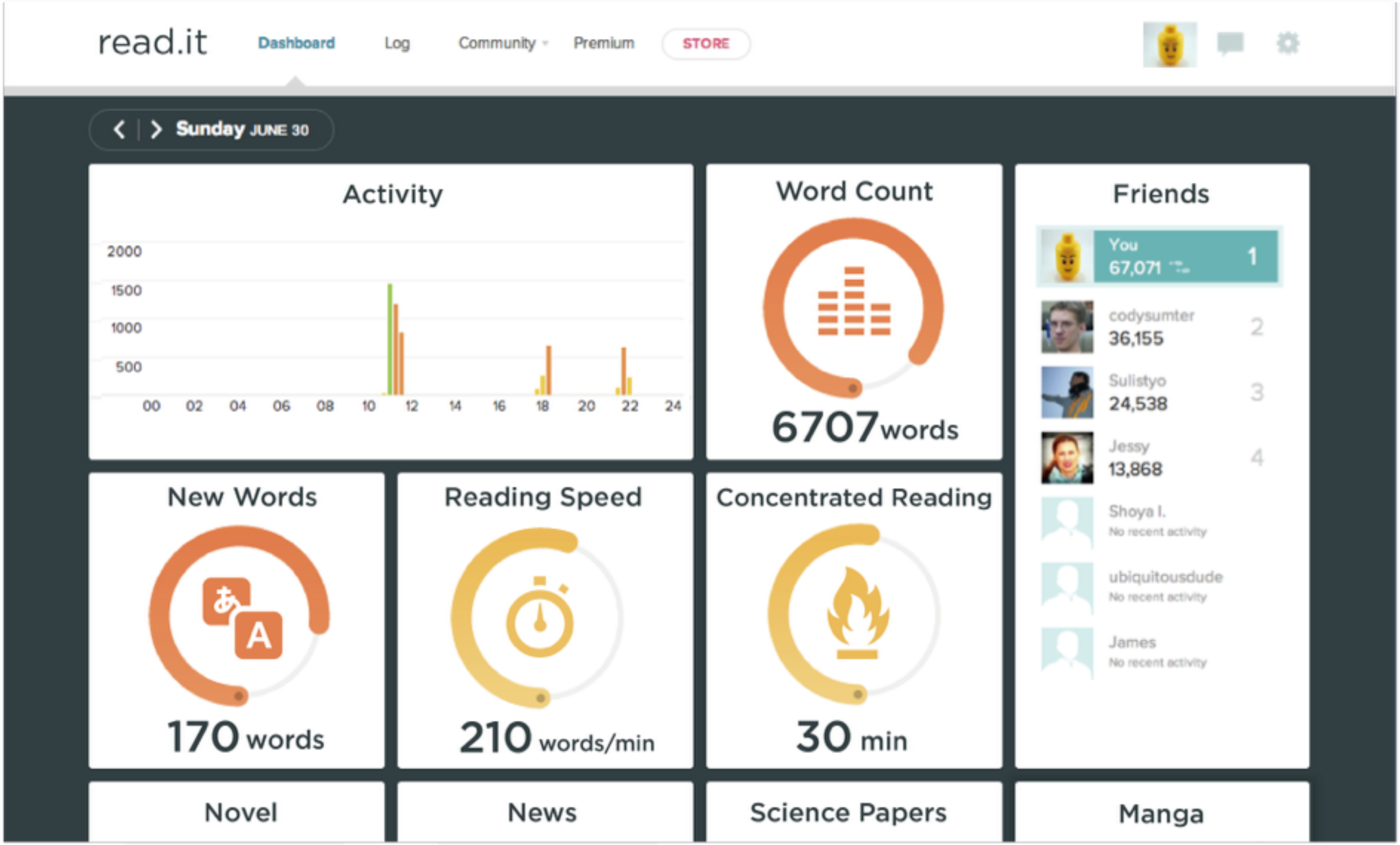
ggf. Flachzange

Elektronische Verbindungen:
Generatorkabel weiss -> plus, gelb -> minus
Motorkabel: braun -> braun, gruen -> blau
Schutzleiter: gruen-gelb -> Steckzunge an Schraube

Kunze, K.,Wagner, F., Kartal, E., Morales Kluge, E., and Lukowicz, P. Does Context Matter ? - A Quantitative Evaluation in a Real World Maintenance Scenario. *In Proceedings of the 7th international Conference on Pervasive Computing* Nara, Japan, May 11 - 14, 2009.

Early Prototype (around 2003-2004)

Second Case Study: Document Type Recognition



Sunday JUNE 30

Activity

Word Count
6707 words

Friends

You	67,071	1
codysumter	36,155	2
Sulistyo	24,538	3
Jessy	13,868	4
Shoya I.	No recent activity	
ubiquitousdude	No recent activity	
James	No recent activity	

New Words
170 words

Reading Speed
210 words/min

Concentrated Reading
30 min

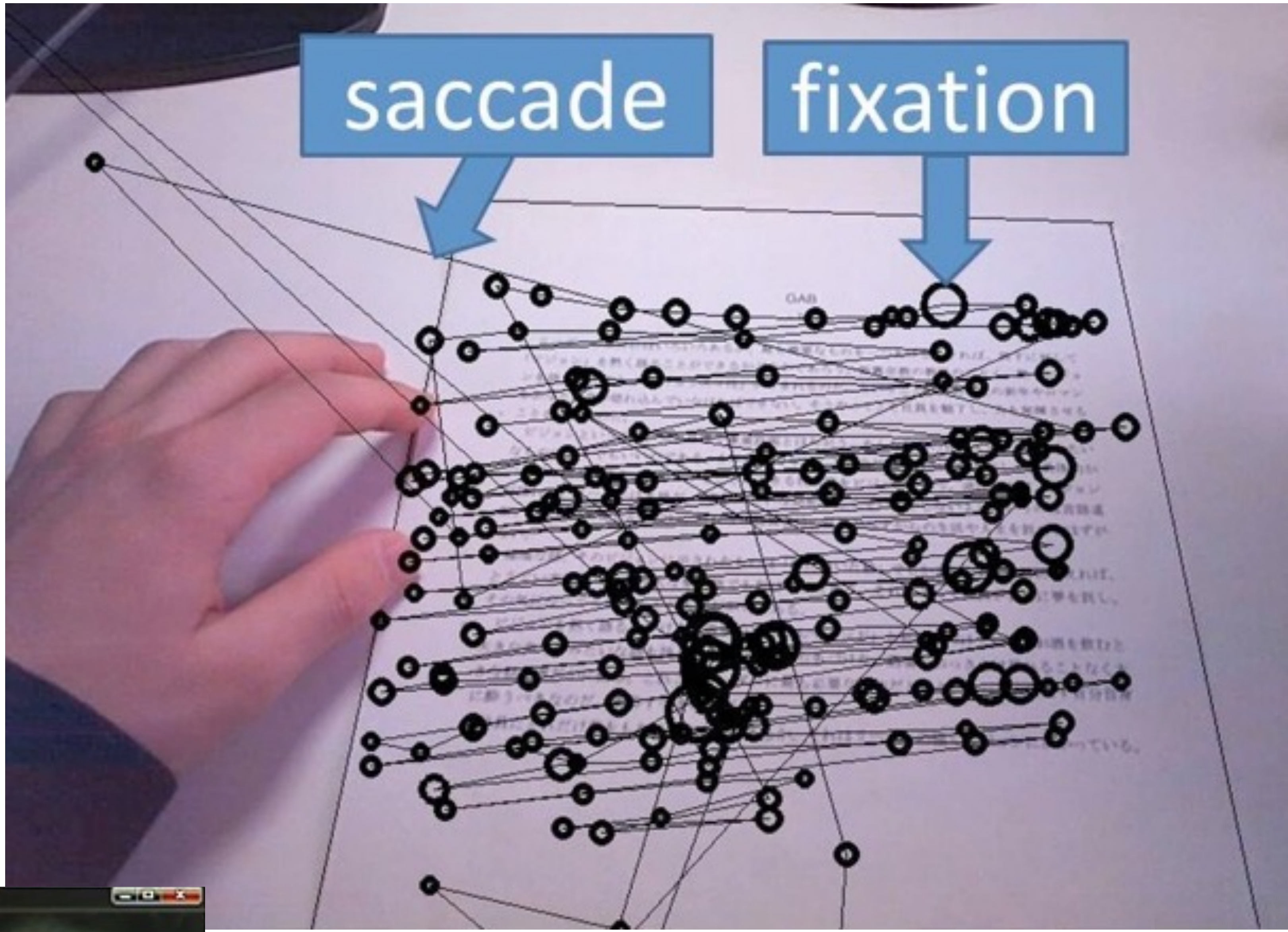
Novel

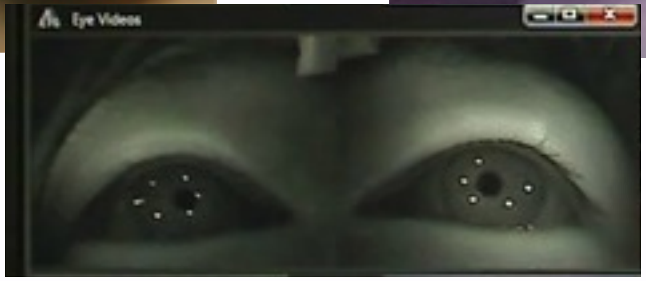
News

Science Papers

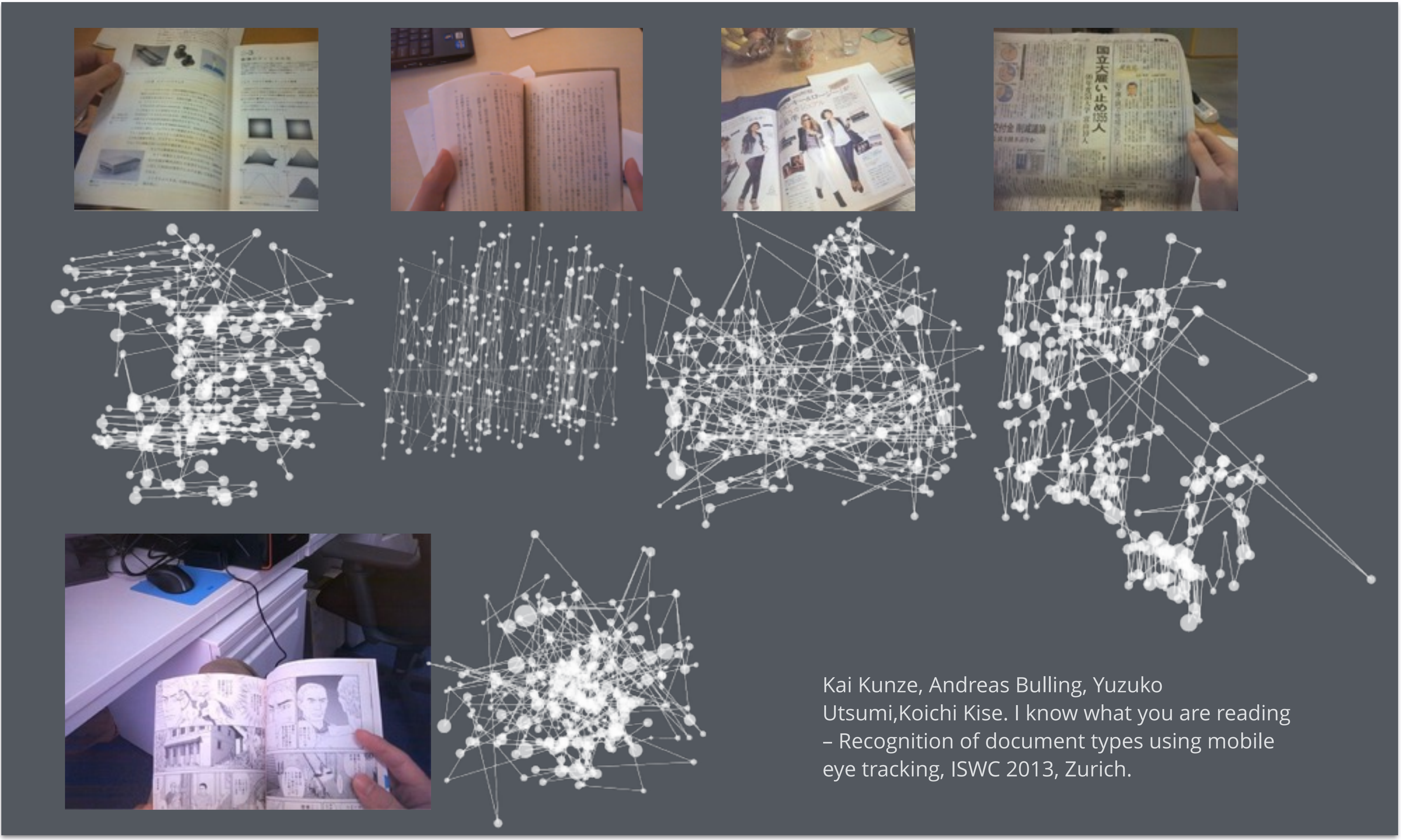
Manga

Second Case Study: Document Type Recognition





Second Case Study: Document Type Recognition



Kai Kunze, Andreas Bulling, Yuzuko Utsumi,Koichi Kise. I know what you are reading – Recognition of document types using mobile eye tracking, ISWC 2013, Zurich.