Wireless Networks
(Reti Wireless)

Prof. Claudio Palazzi
cpalazzi@math.unipd.it

Università degli Studi di Padova

Case Study

Web^2 and PATH 2.0

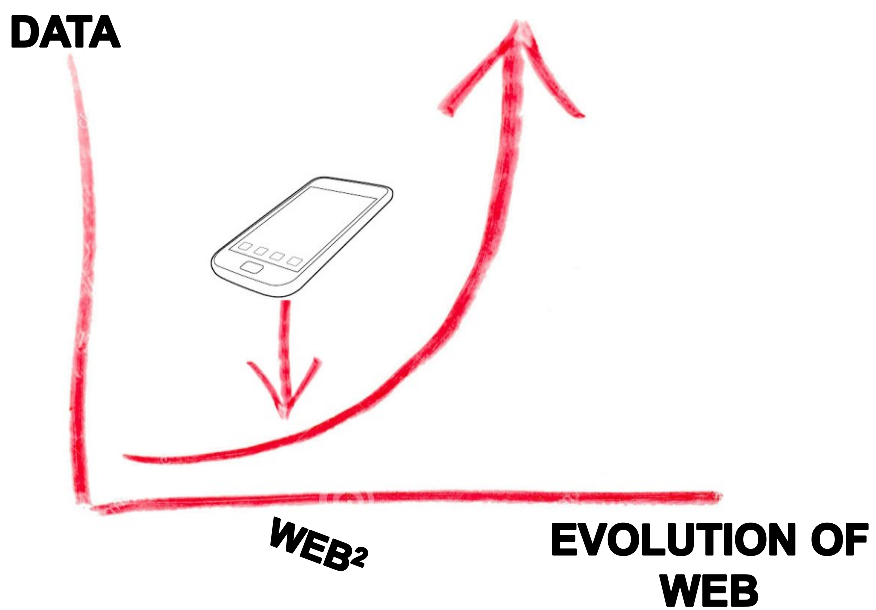
A PARTICIPATORY SYSTEM TO GENERATE ROUTES BASED ON SENSED DATA

Smartphone Sensor Networks

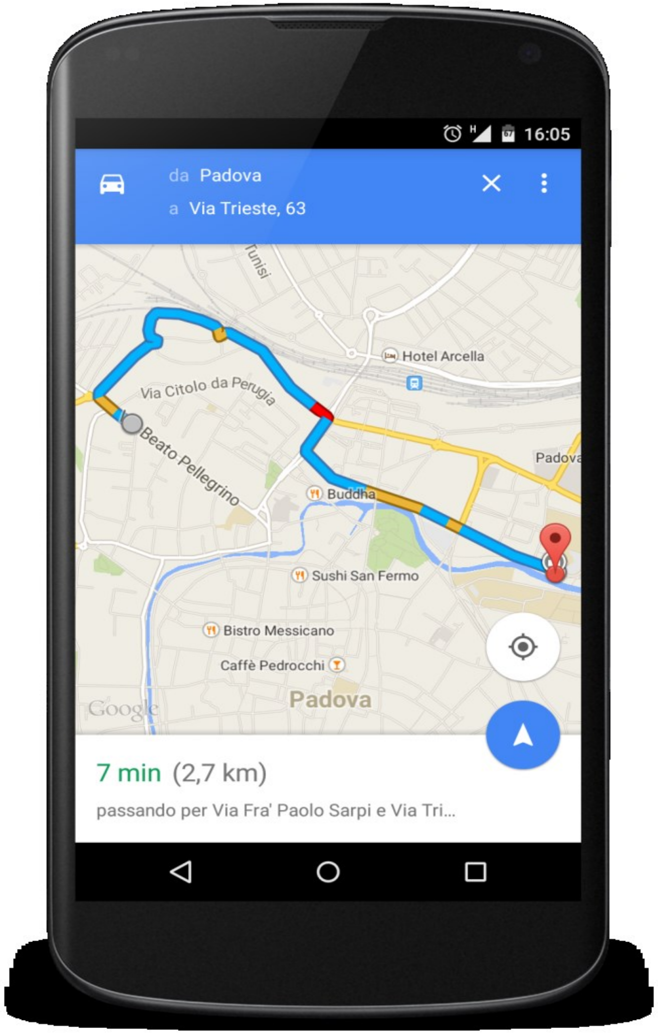
- Sensor Networks can now become widely common thanks to the smartphone revolution

Web^2 (Web Squared)

- A huge amount of data could be automatically gathered from smartphone sensors
- Appropriately rerouting/combining/processing them could generate innovative services
- From Web 2.0 to Web^2
  – Tim O'Reilly



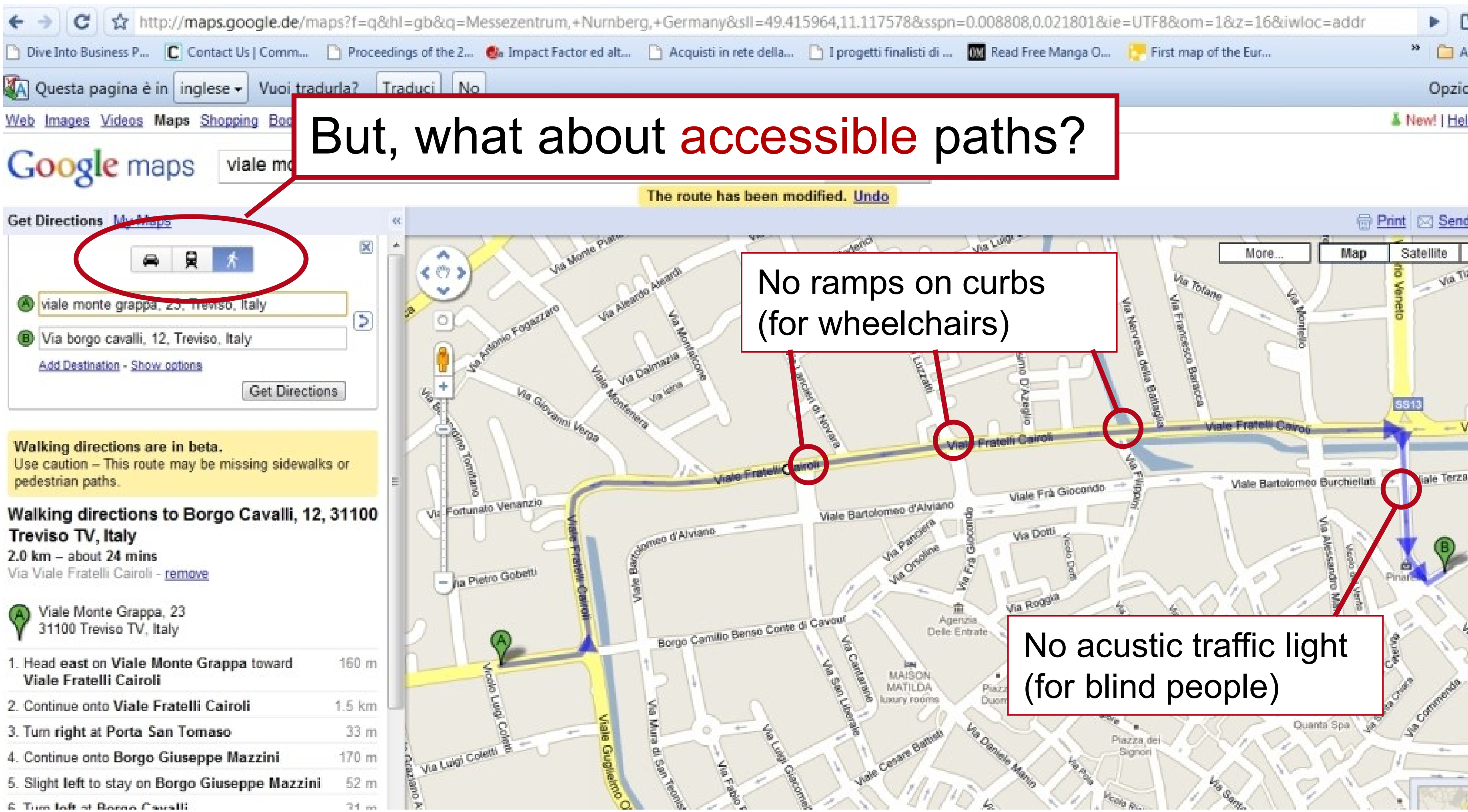
Web^2: An Example



- Google Maps automatically reports roads with slowing down traffic
- Senses users movement speed
- Does not require any interaction with the device
- Estimates based on gathered data for that path

Web 2.0 Services vs. Accessibility

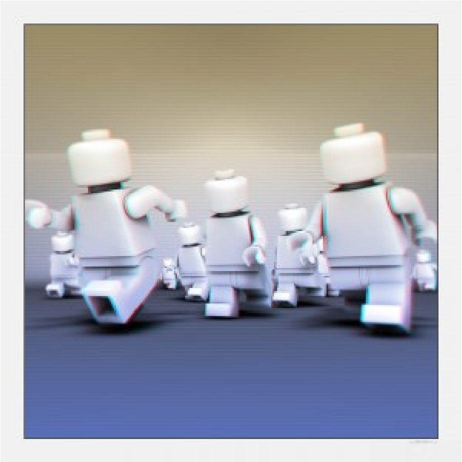
Today it's easy to find a path in a town (by car, public transportation, on foot)



Problem Statement

- To generate accessible paths we need an up to date database of information related to road's accessibility
  - For every town
  - We cannot pay a million of employees to test the accessibility of roads in every town!
- Serious games?
- Or… can we do this automatically?

Solution Sketch



CROWDSOURCING + SMART PHONES

- We can use sensors mounted on current smart phones to automatically detect accessible track

Assumption

- A person with a certain disability frequently (e.g., daily or weekly) moves along certain routes
  - To go to work or shopping, or back home…
  - Frequent routes belong to urban areas well known by that person
- She/he probably judges those routes as sufficiently accessible

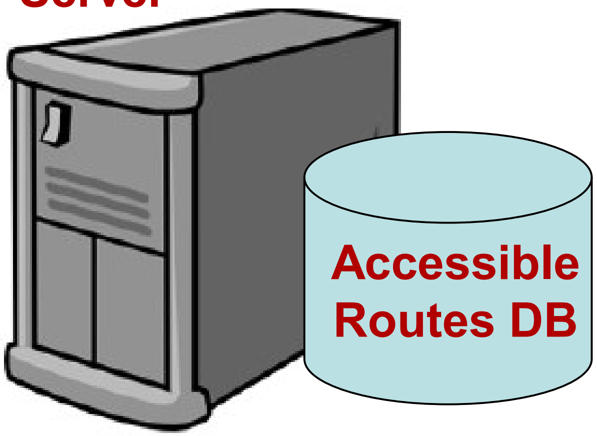
Idea

- If we could catch those frequent routes, we could suggest other people with the same disability to use them when eventually coming in the area
- We can use a smart phone application utilizing the GPS information to track those frequent routes down and anonimally forward them toward a remote server
- Then, all those accessible routes could be exploited and combined when needed to compute other accessible paths.

PATH 2.0 Data Gathering

GPS

Server



Android based smart phone

We tested it on real HTC Magic and Hero

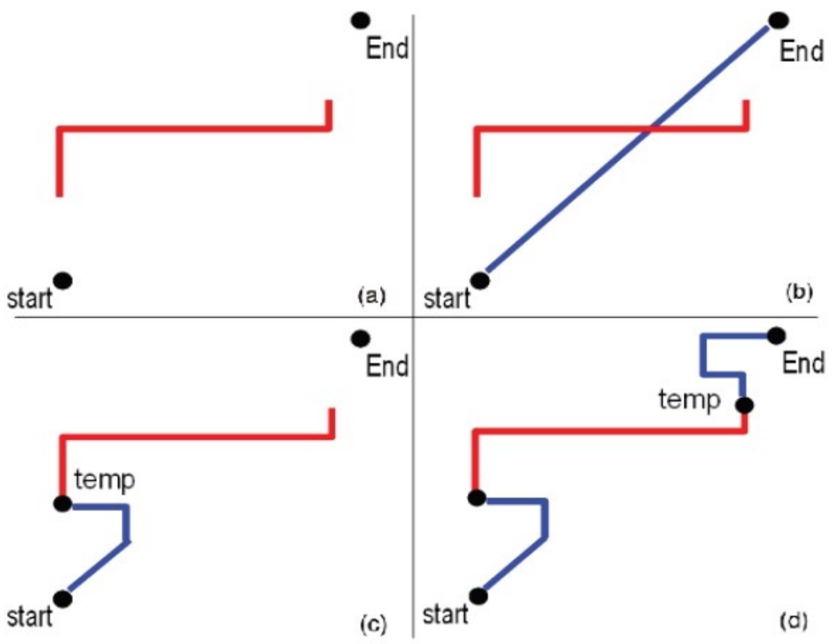
CROWDSOURCING:

The more people use the system, the better its reliability becomes

PATH 2.0: New Routes Generation

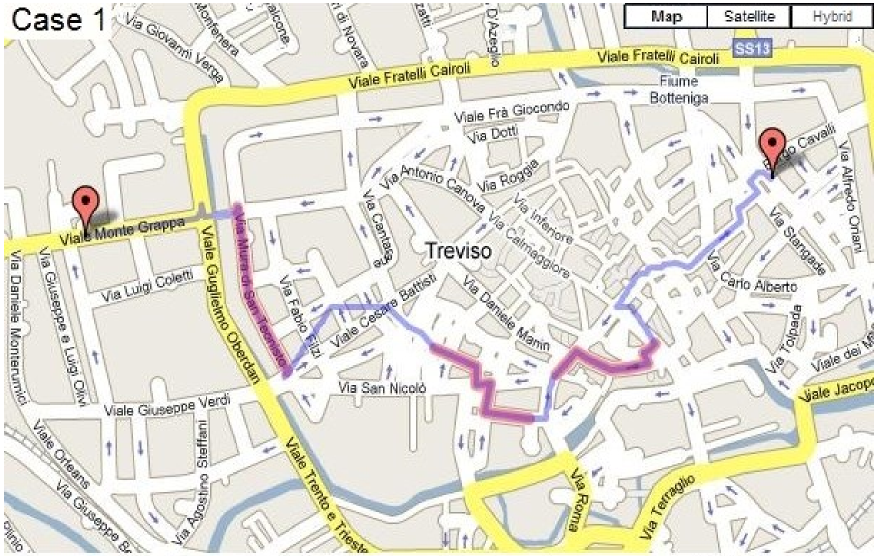
- When a user with a certain disability looks for a path in an area she/he does not know, new routes are generated considering the accessible routes DB

- The user inputs “start” and “end” points
- Google Maps would suggest a fast route (blue line)
- There is an accessible path in the DB (red line) which is between “start” and “end”
- PATH 2.0 server sends a query to Google maps for the best path between “start” and the accessible path and between the accessible path to “end”
- A new path (with better accessibility) is computed with the available information



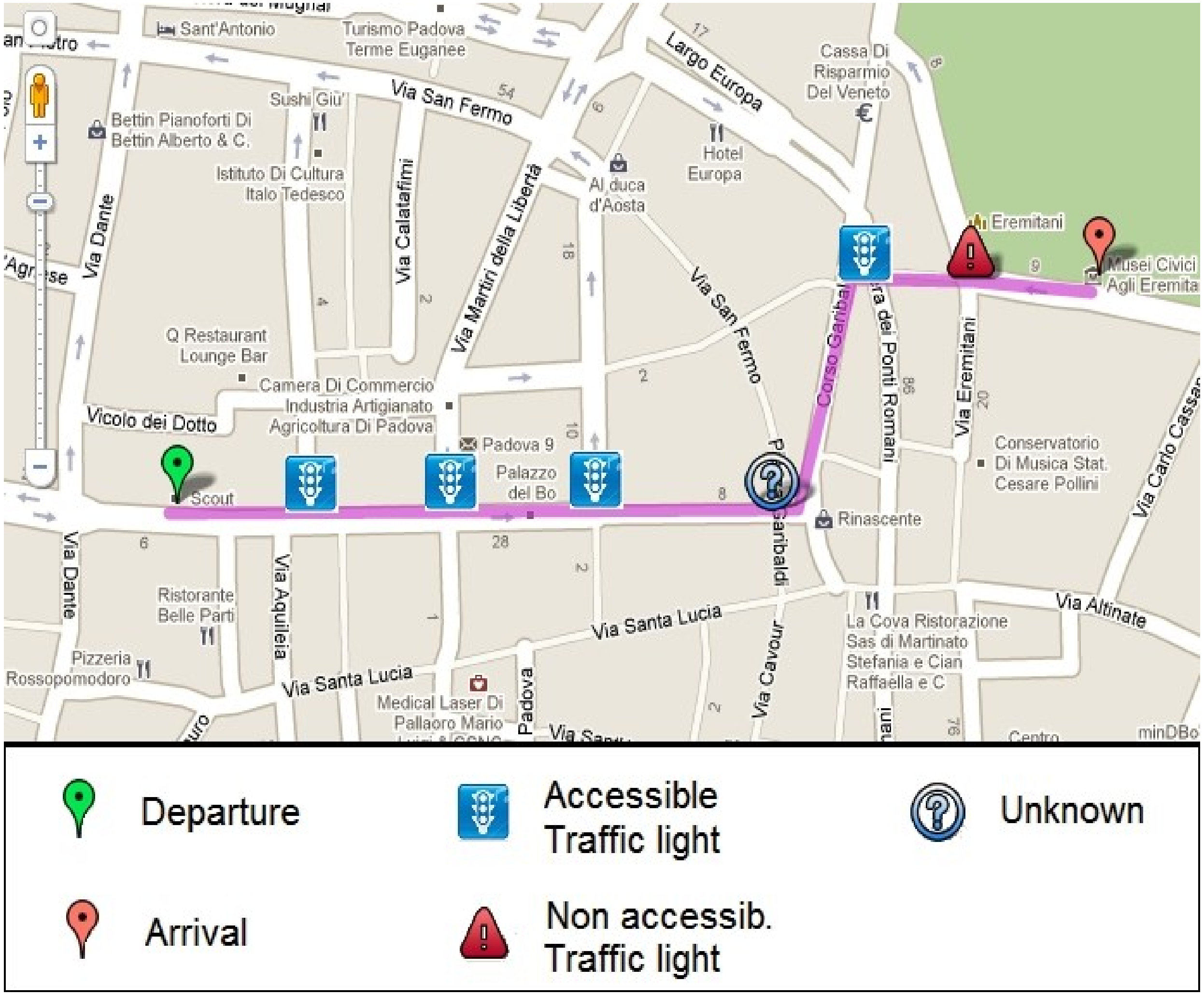
PATH 2.0: Routes Visualization

- Two examples of paths generated with PATH 2.0
- Accessible paths present in the database are highlighted in purple



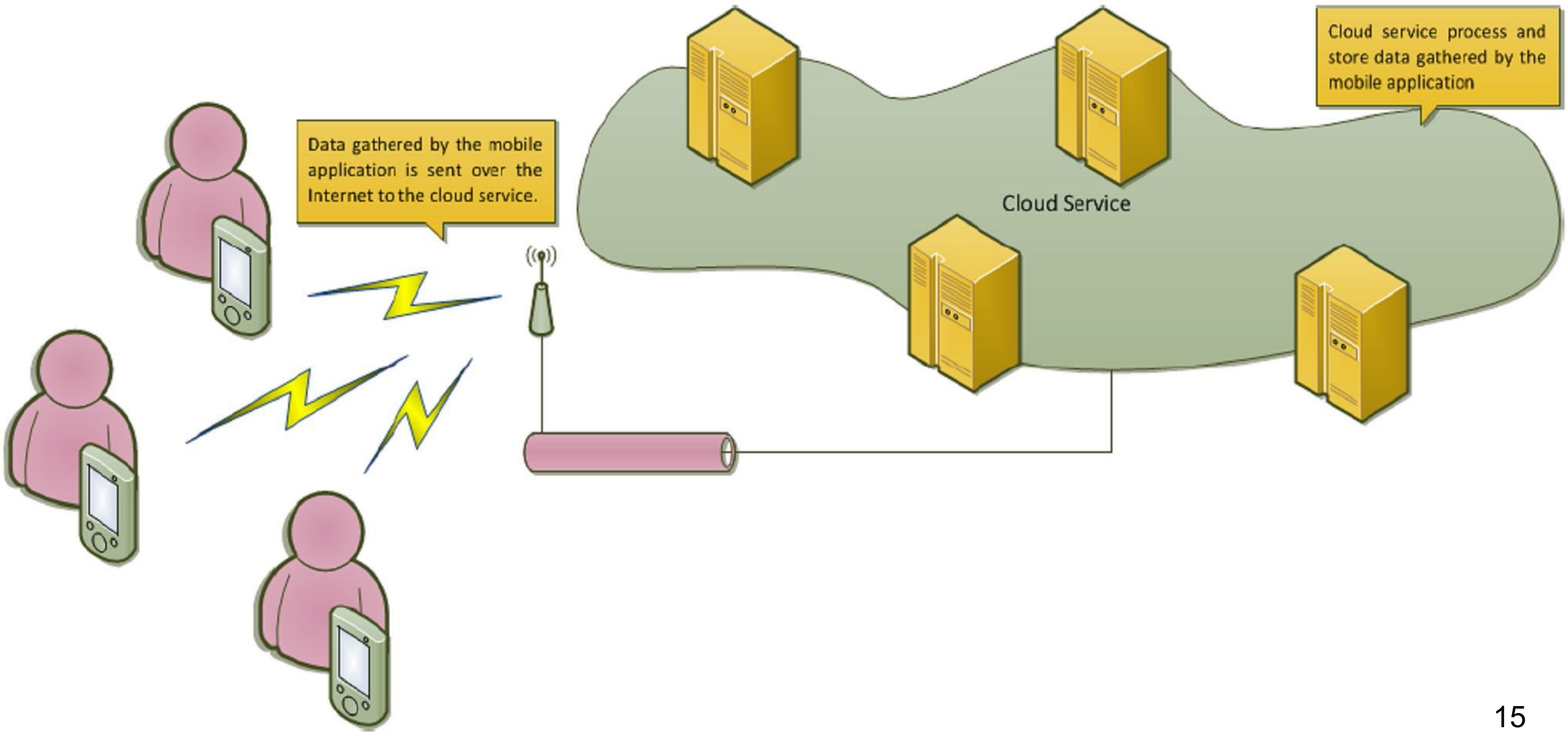
How about Accessible Traffic Lights?

Again: need a huge database with accessibility data



Proposed Solution

- Use smartphones' microphone to record audible signals when the user is near a traffic light
- Use remote servers to detect whether there is a signal for blind people in the audio file



In Practice

Problem Statement:

- Detect accessible traffic lights so as to mark them as accessible in a remote database used then to compose routes for users

Possible Approaches:

- Actively involving users
- Automatically and passively gathering data

Active Approach

- Actively involving users in providing information
  - For instance through social platforms or serious games

- Serious game approach
  - Ask users to identify accessible traffic lights
  - Could also be done by asking them to send to a server pictures or audio that are then processed by algorithms to detect whether the traffic light is actually accessible
  - Points are given when the information is correct
  - More points are given when the traffic light has not been already referred to by other users (to privilege the expansion of the database)
  - Points are deducted when the information is wrong
  - Consensus?
  - What can users do with points?

Passive Approach

- Automatically detect proximity to a traffic light so as to start recording a file when it transits from red light to green light

Chosen Approach:

- Exploit accelerometer to identify user’s movement pattern when stopping at a pedestrian traffic light
  - start recording audio for a few seconds
  - Attach GPS coordinates to the audio file and transmit it to the server

Pattern Identification

Step 1:

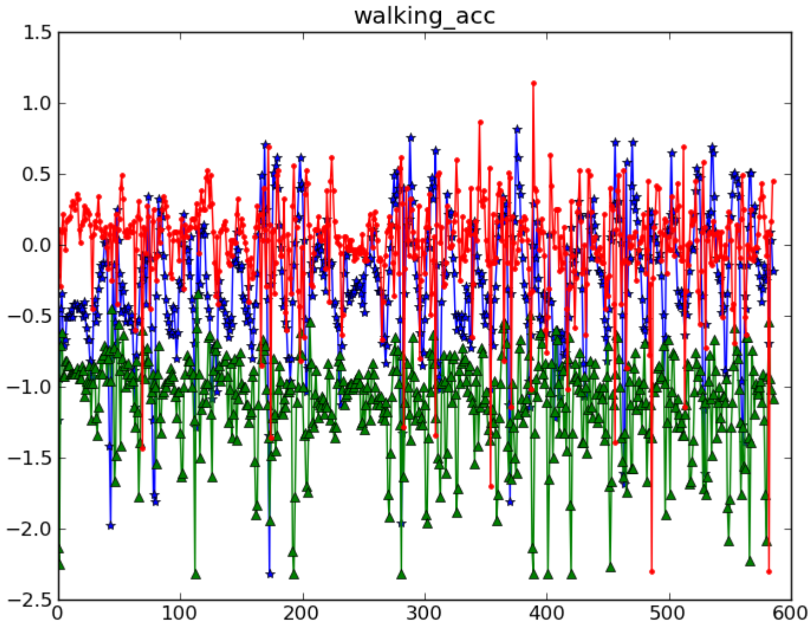
- Elaborate XYZ-axis accelerometer data to focus only on the movement's magnitude

Step 2:

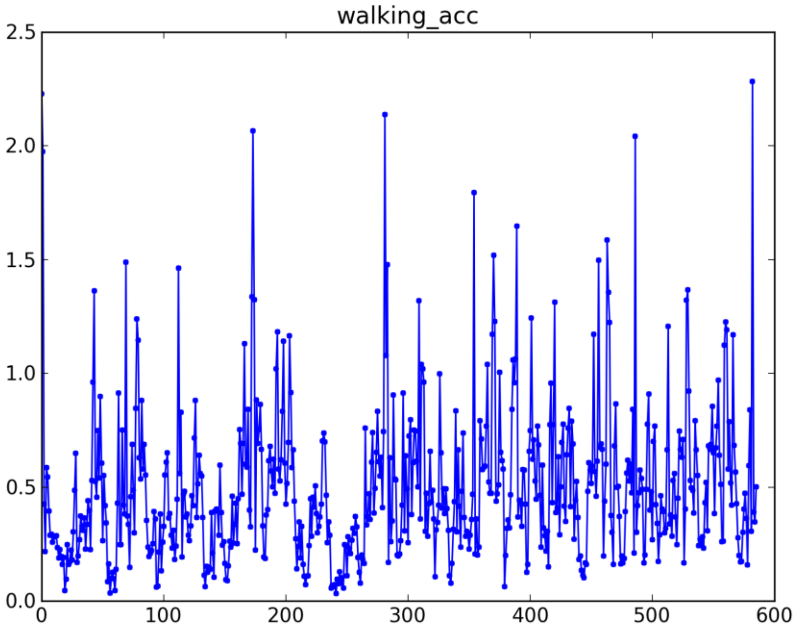
- Identify crossing road pattern:
    1. User walking
    2. User stopping at traffic light
    3. User traversing road at higher speed
    4. User resuming regular speed

User Walking Test (Step 1)

Appying high-pass filter to walking data and compute magnitude as: $\text{magnitude}(v) = |v| = \sqrt{v.x^2 + v.y^2 + v.z^2}$

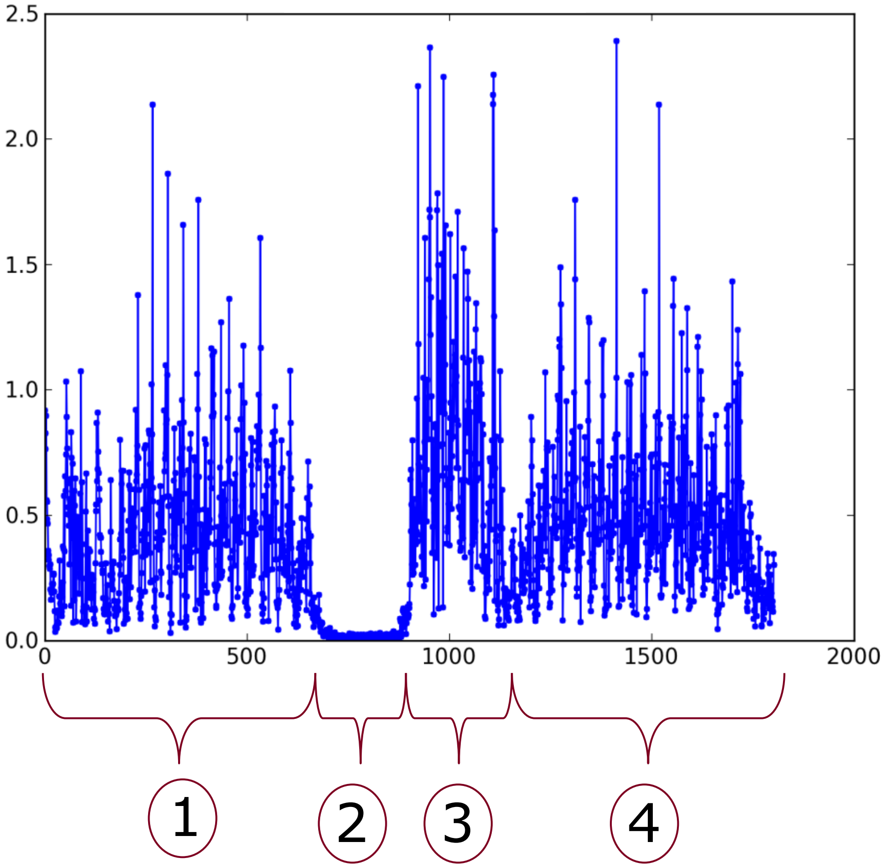


data before elaboration



data after elaboration

User Walking Test (Step 2)

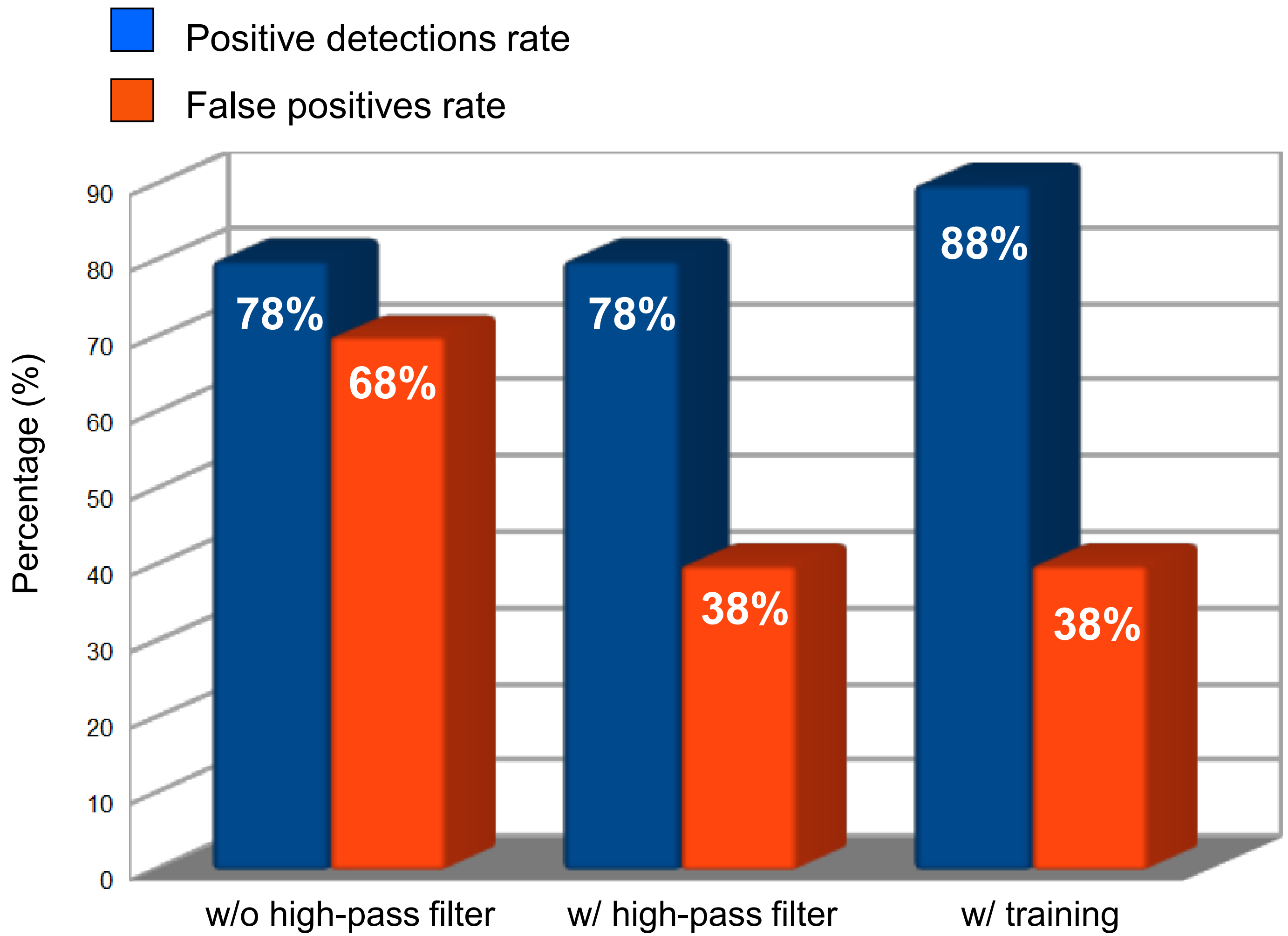


Identify user's pattern at red traffic light:

1. Regular walk
2. Stop at red traffic light
3. Crossing road fast
4. Resuming regular speed

Results can be slightly improved by adapting the thresholds individually through a preliminar training phase or by using other techniques (e.g., Support Vector Machine?)

Final Results



Conclusion and Future Works

- We discussed a system for pervasive and automatic collection of data that are then intelligently combined by a server to generate routes that are accessible for people with disabilities
  - Practically improve the inclusion of users with disabilities in the society and their access to Web 2.0 advantages.

- Future directions:
  - extend with information about accessible stores
  - improve the algorithm computing accessible routes
  - allow users to actively evaluate their regular paths if they wish
  - study energy optimization solutions
  - perform extensive field testing

The **Case Study** continue with…

Enhancing Geographical Maps with Environmental Sensed Data

Pervasive Urban Sensing

Context

- Web Squared for health and well being

Objective

- Collect unbiased environmental data through commercial off-the-shelf devices
  - Brightness and noise levels

- Enhanced pedestrian navigation system contemplating in the routing criteria other environmental sensed data (and not just length/time)
  - Useful in case of:
    - Photophobia
    - UV related skin illnesses
    - Noise related stress
    - etc.

- Other uses (e.g., Smart Cities)
  - Show heatmaps and historical trends on Web pages

Main Challenge

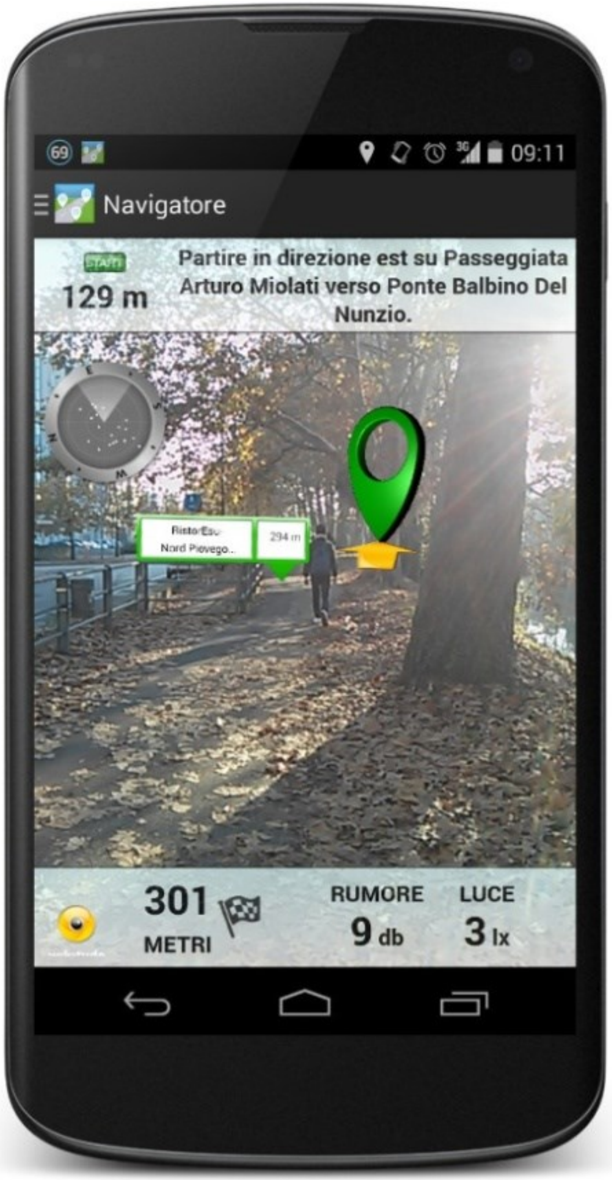
How to get environment data (noise, light) from a device that is usually in our pockets or in a purse?

Data would be affected by the close and dark surrounding

We need to make sure that the smartphone is out in the user's hands when collecting data

We need to use a trick: data are collected only when a certain application is run

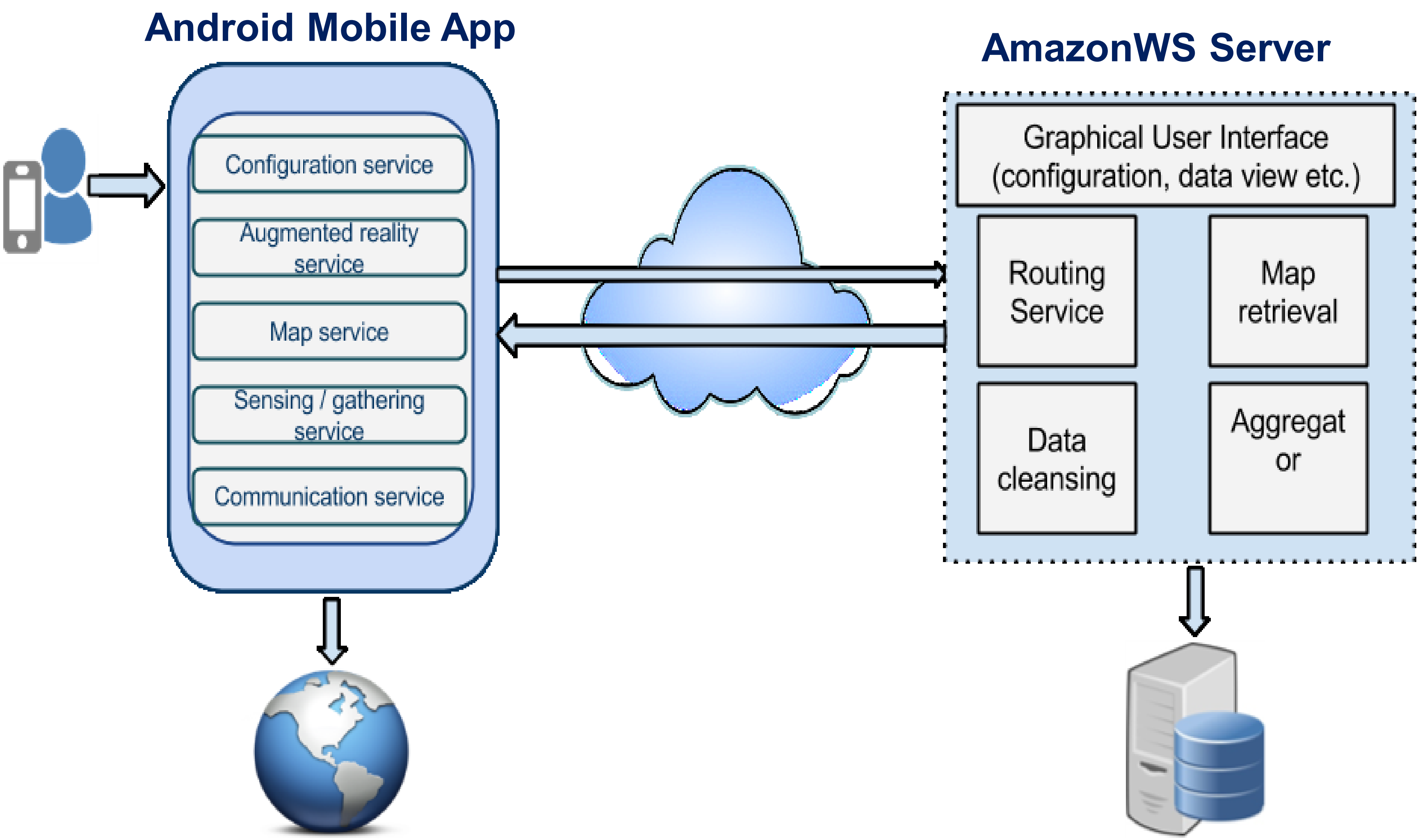
AR-based Pedestrian Navigation



Technical Challenges

- Access to raw geographical data
  - Fine control over sample / actual position association
  - Mostly, needed by the navigation system for routing decision
- Environmental data collection
  - Data quality (measurement precision)
  - Necessary critical mass to gather/produce useful information
  - Resource constraints on the client side (e.g., battery, communication costs)
- Produce information based upon this raw data
  - Measurements taken from heterogeneous users/devices with different behaviours

Approach: Architecture Overview



Approach: Tackling the Issues

- OpenStreetMap as a geographical data source
  - Provides access to raw data and fine control over the matching process between data and street

- Data collection
  - To keep users attracted while collecting unbiased data in the open we designed an app for augmented reality navigation of pedestrians
  - Performed a calibration phase on the client-side to increase sensing measurement precision (e.g., brightness)
  - Used a Delay Tolerant Networking (DTN) approach for transferring sensed data

- Generate reliable information elaborating collected raw data
  - Data filtering techniques (e.g., remove outliers)
  - Employ S.T. Map Matching (Lou *et. al.*) technique to reconstruct the users measurement path

Mobile App

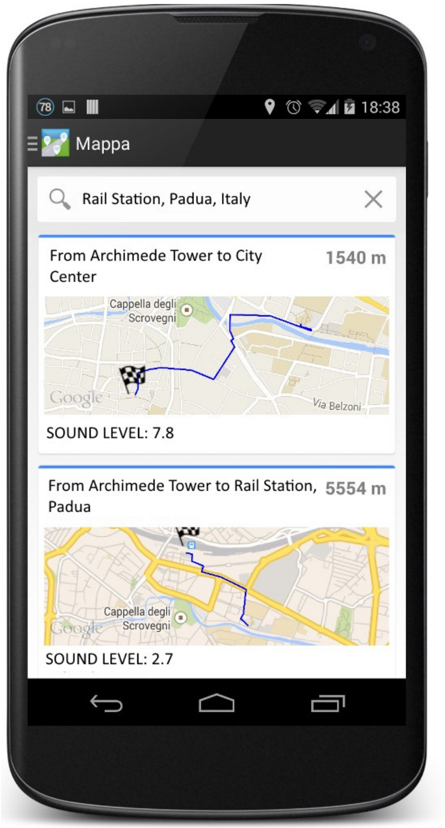
- Measure brightness and noise levels
  - Data sensed either periodically or distance-based

- Google maps as the service view
  1. Users search for a certain destination on their phone
  2. A series of possible Google maps-like paths is shown
  3. Users select one of the paths and can start augmented reality overlay showing direction and other information on the screen (see next slide)

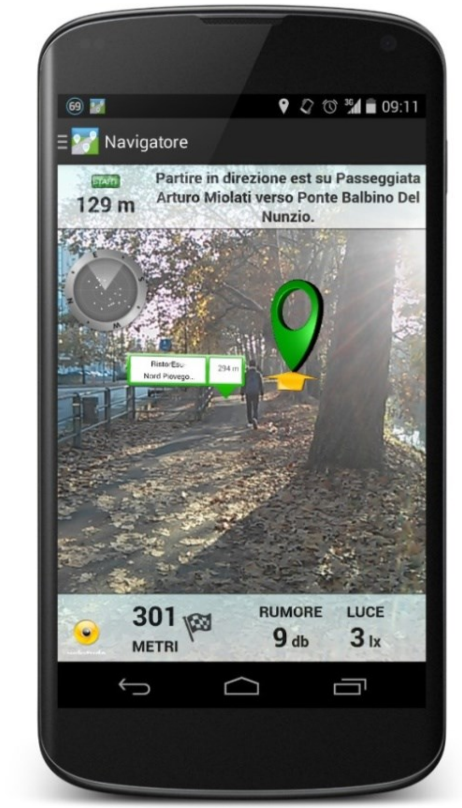
- Server data communication
  - Measurement tagged with position coordinates
  - Sensed data and GPS coordinates transferred to the server in an anonymous way using GeoJson as standard data format

Mobile App: Maps and AR Navigation

Destination selection and map



AR navigation example

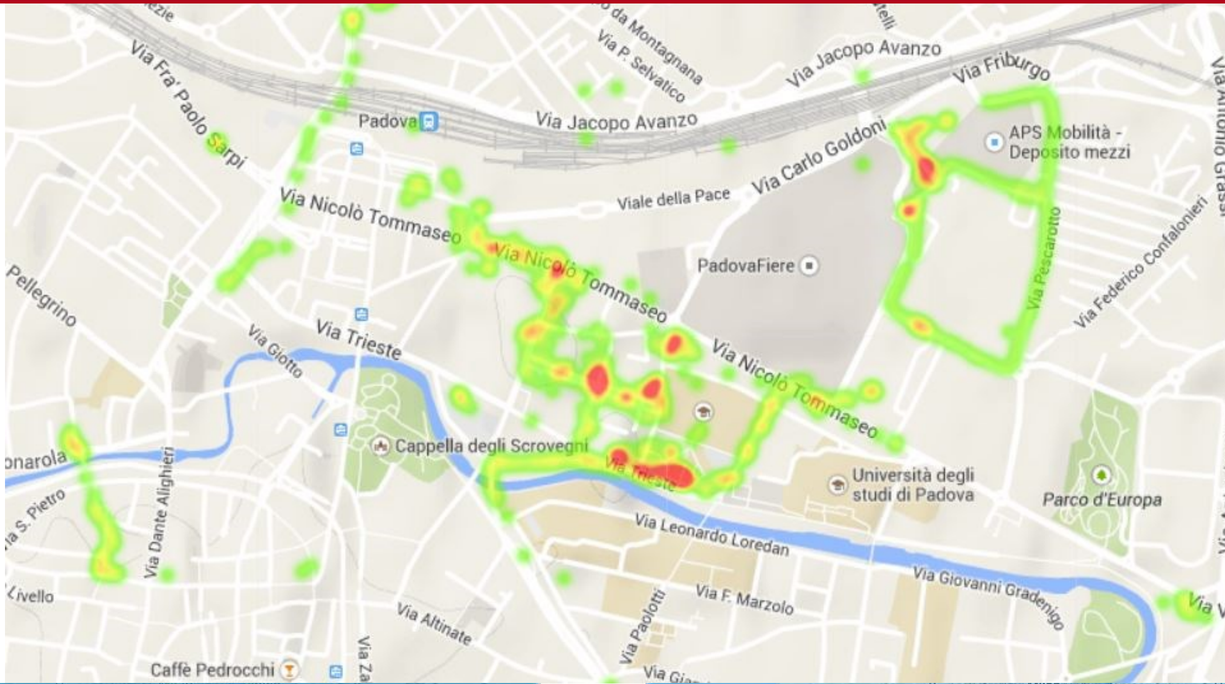


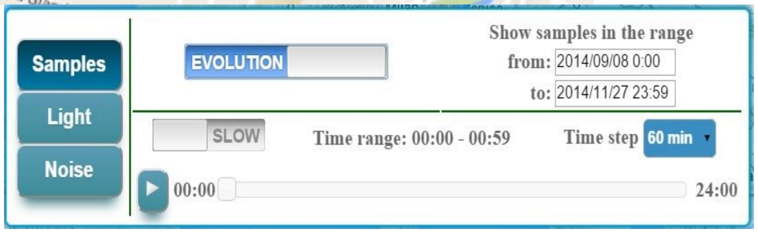
Augmented reality view as overlay incentivizing users to collect unbiased date in the open

Server Side

- Routing Service taking into account brightness / noise
  - Path generation algorithm also based on brightness / noise
  - Weight of an edge includes sensed data values beside edge length
- Data Cleansing
  - Associate a sample to its actual position on the map considering the actual roads and not just the GPS coordinates (may have errors)
- Generating periodic aggregated data view
  - Average brightness / noise levels are generated from users' samples with a hour / day basis granularity

Server Side: Data Map View



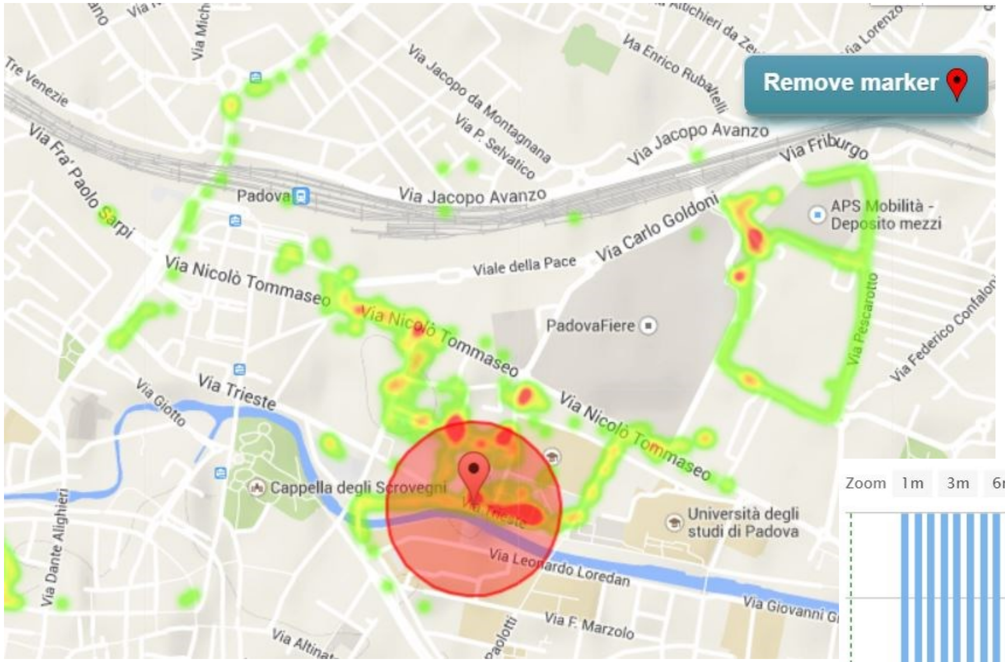


Control Panel - 1 **Control Panel - 2**

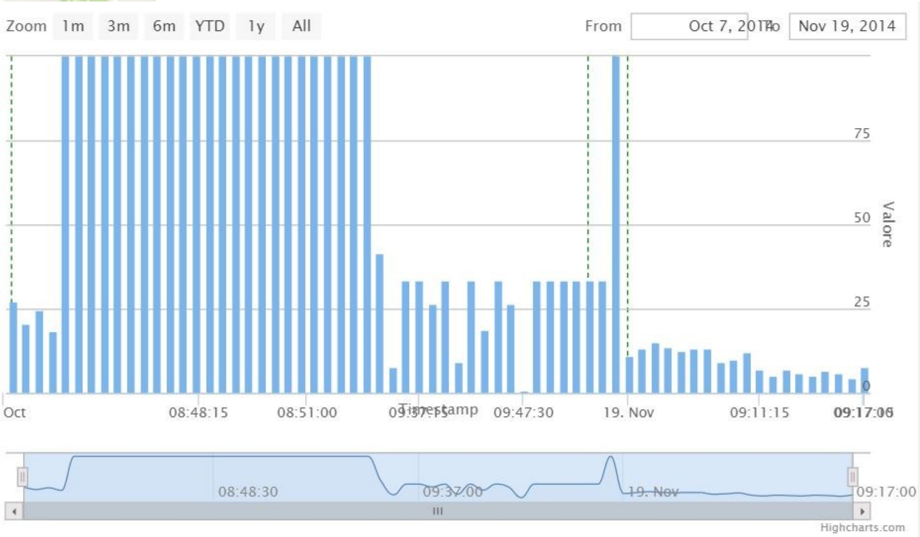
On a Web heatmap we can see information regarding a certain hour of a certain day, or its evolution during a day or a longer period: <u>number of samples, brightness, noise</u>

Server Side: Data Statistics

Analysis can be limited to sensed samples inside the bounded region



Brightness statistics along the day in the selected region



Conclusion and Future Work

- Collecting unbiased environmental data
    - Augmented reality app incentivizing unbiased collection of data
    - Calibration phase to augment measurement precision
- Case Study: Pedestrian route guidance system
    - Currently we consider brightness / noise levels
    - Shortest path based on one or more weighted labels
- Yet, there is much to be done
    - We need more data to provide coverage and extract information
    - System scalability on the server side due to the necessary choice to replicate and tag geographical maps

Little Bibliography

- A. Bujari, M. Ciman, O. Gaggi, G. Marfia, C. E. Palazzi, “PathS: Enhancing Geographical Maps with Environmental Sensed Data”, in Proc. of ACM MobileHealth 2015/MobiHoc 2015, Hangzhou, China, Jun 2015

- A. Bujari, B. Licar, C. E. Palazzi, “Movement Pattern Recognition through Smartphone’s Accelerometer”, in Proc. of the 9th IEEE Communications and Networking Conference (CCNC 2012), Las Vegas, NV, USA, Jan 2012

- C. E. Palazzi, L. Teodori, M. Roccetti, “PATH 2.0: A Participatory System for the Generation of Accessible Routes”, in Proc. of the IEEE International Conference on Multimedia and Expo (ICME'10), IEEE, Singapore, Jul 2010