A map of The Hague, Netherlands, showing the locations of numerous Netatmo weather stations. The stations are marked with red pin icons, each containing a number representing the AQI. The map includes various neighborhood names such as Duindorp, Bomenbuurt, Valkenboskwartier, Voorburg, and Leidschendamse. Major roads like the A12 and S100 are also visible. The text is overlaid on a semi-transparent white box in the center of the map.

# Improving location accuracy of a crowdsourced weather station by using a point cloud use case base Netatmo on the Hague

Yixin Xu

Mentors:

Ken Arroyo Ochori

Alexander Wandl

Agung Indrajit

# Introduction

## 1. Urban Heat Island (UHI)

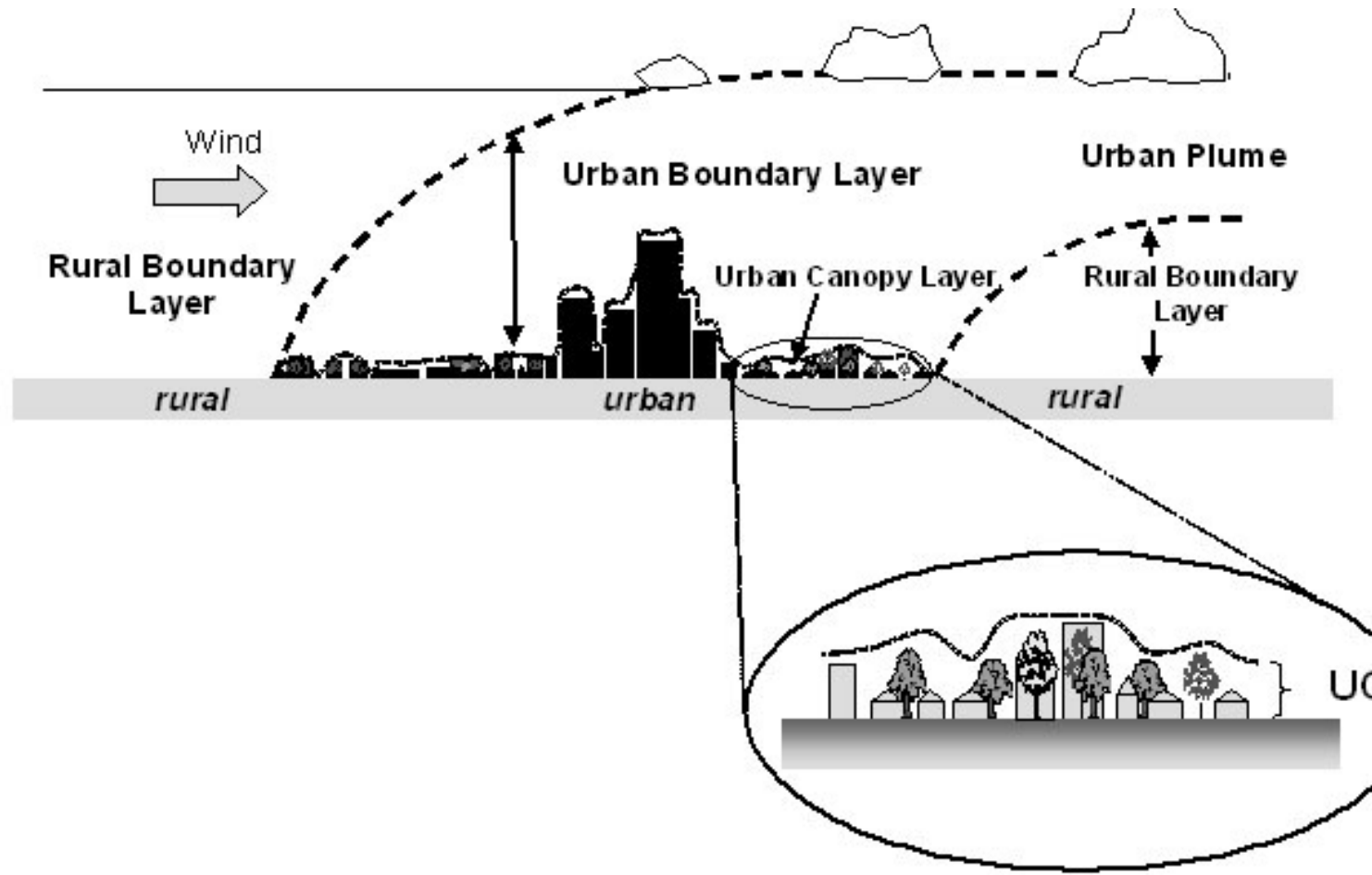
- UHI is an urban area that is significantly warmer than the surroundings due to human activities.
- UHI leads to health issues, higher energy demands, economic losses
  - Increasing heat mortality in Shanghai and UHI has been proved to be directly responsible for it.
  - Higher temperature causes 10% of urban electric consumption for air conditioner in America.



# Introduction

## 1. Urban Heat Island (UHI)

- Three types of heat islands:
  - canopy layer heat island (CLHI)
  - boundary layer heat island (BLHI)
  - surface heat island (SHI)
- Different acquisition method
  - Remote sensing
  - Thermometer



# Introduction

## 2. Problem statement

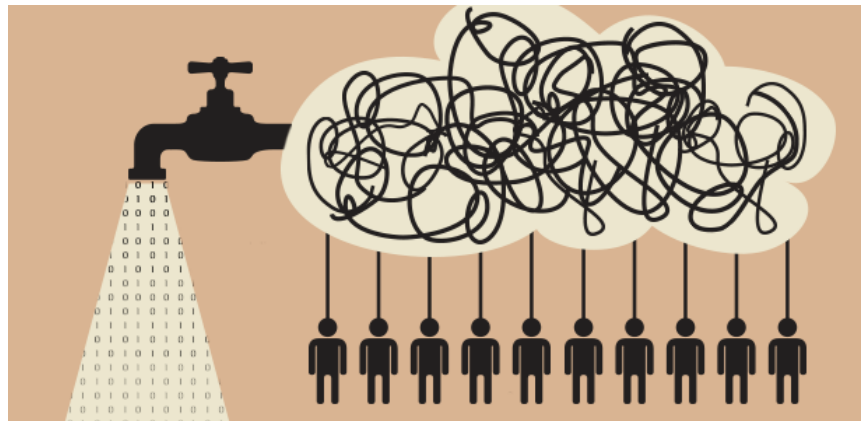
- Canopy layer heat island is most related to human living, this layer focus on **air temperature above the surface**.
- Traditional approach to detect air temperature: ground-based thermometer
  - Pros: highly accurate
  - Cons: less spatial resolution
- Now more and more crowdsourced weather stations come to use, can we use this station in UHI research?
  - Most of previous researches stress concerns about the accuracy in scientific use.



# Introduction

## 3. Crowdsourced data

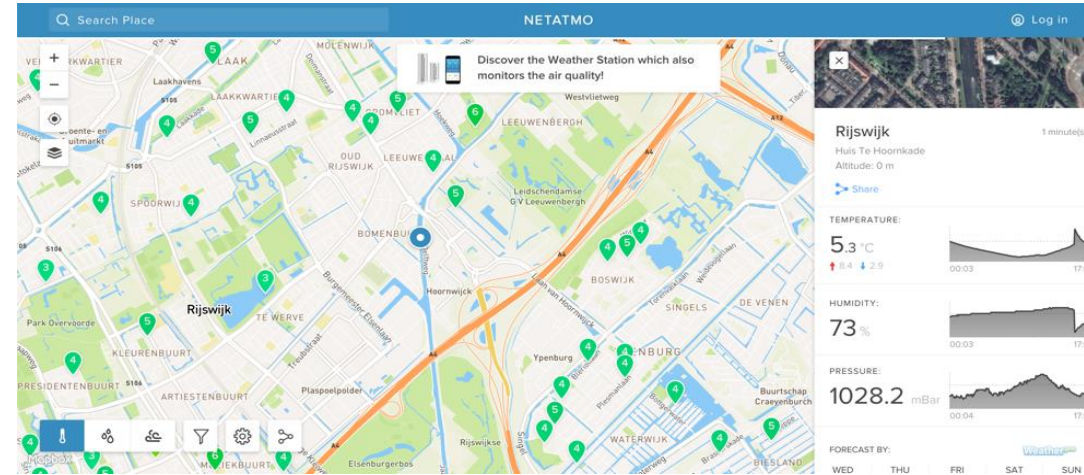
- Crowdsourced is used to describe the process of getting work from a large group of people in an online setting.
- Advantages of using crowdsourced include improved costs, speed, flexibility, scalability.
- The use of crowdsourced data in the atmospheric sciences is very limited, the main reason is the difficulty in obtain an accurate observation.



# Introduction

## 4. Netatmo weather station

- Pros:
  - Easy to use
  - Time and spatial resolution is better
  - Provide API to public
- Cons:
  - Raw data is noisy (system error)
  - Can not block solar radiance (influence air temperature detection)
  - Position of sensors are not accurate (upload by smartphone GPS)



# Introduction

## 4. Netatmo weather station

- Netatmo records are stored in a TU Delft sever, update every 15minutes
- Around 300 Netatmo sensors in the Hague
- But the quality of the observations and location accuracy is unclear.



- How to locate a more accurate position of NETATMO sensors?
  - How to find potential locations for each sensor?
  - How to know if a sensor's record is higher than it should be?
  - For a certain area, how to know when it receives direct solar radiance?
  - How to compare station records with solar simulation?

The reason why I choose location as topic:

1. Temperature inside a city could be complex and various from place to place
2. Many location related factors e.g. NDVI, land cover influence temperature modeling



## General idea:

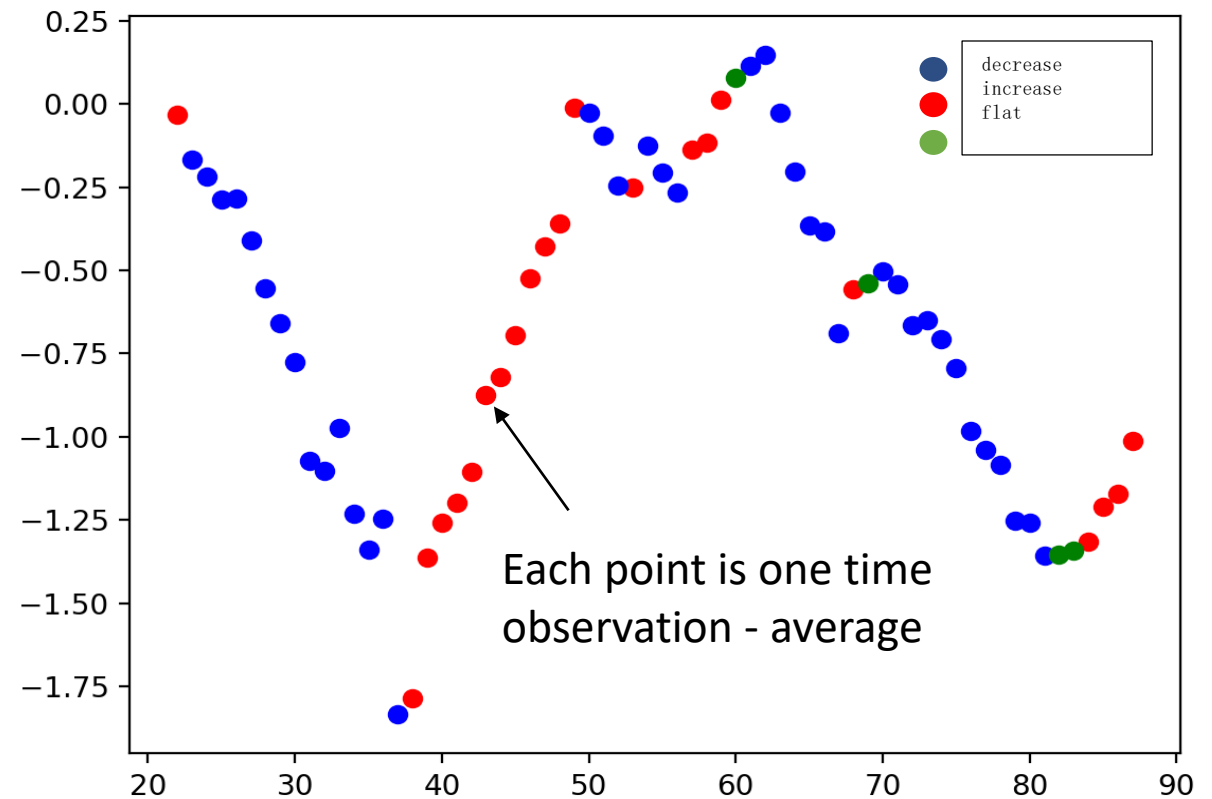
**create candidate location and compare them with solar simulation**

- Data pre-process
- Detecting higher temperature time
- Generating potential location of Netatmo stations
- Computing sky view (dome) and solar parameter
- Finding the most likely horizontal location of the station
- Assigning height value to points

- One-day data process
  - Delete sensors with strange behavior (e.g. obvious system error)
  - Clip sensors and ensure all sensors inside the Hague
- Seven-day data process (2018, May, 22 -28 with good weather)
  - Station daily performance is not stable (e.g. normal on one day but not on the next day)
  - Only use one-day filter is not “justice” enough for all sensors
  - Sensors who work normally at least 5 days in the week are saved

## 2 Detect higher temperature (influenced by sun)time

- How to know real air temperature around a station?
  - True value is unknown without massive field experiment.
  - Average temperature to replace real temperature.
  - Temperature changing pattern to replace absolute value.
- Use record data minus average data, follow increase check algorithm
  - Gradient of near points.

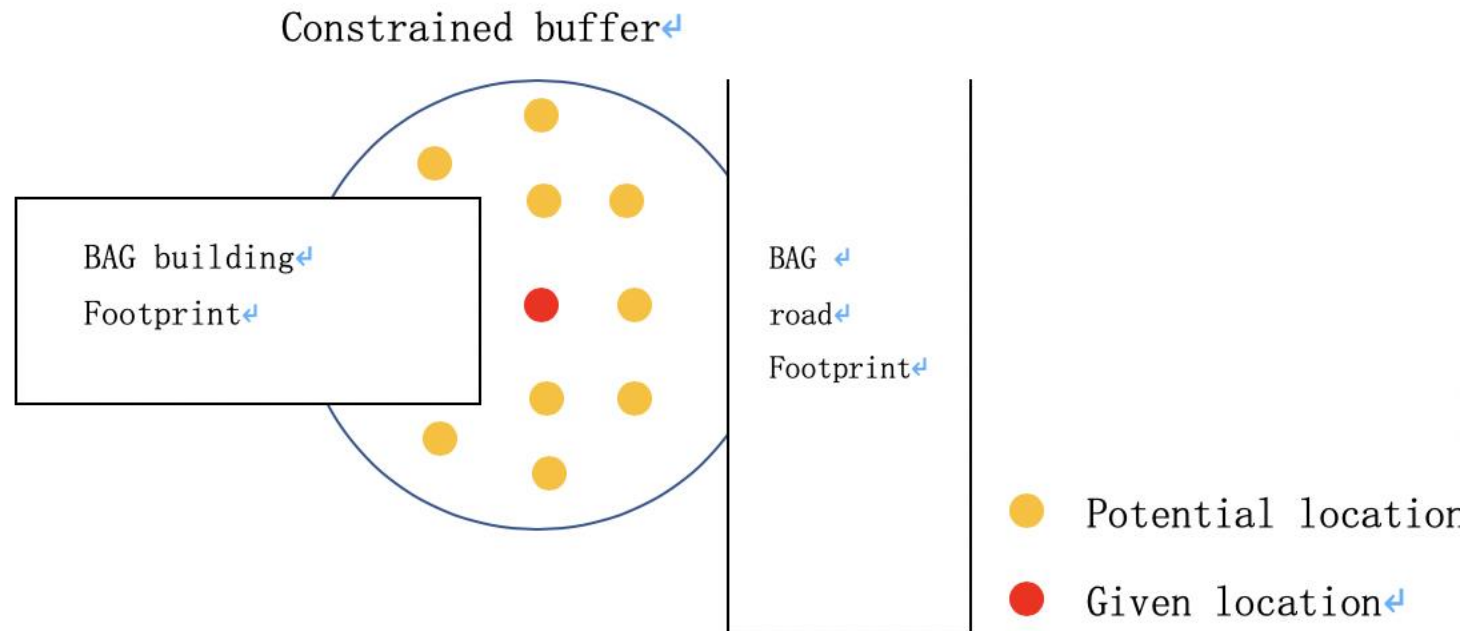


- Each sensor, the dataset store at least 5 days data. How combine them?
  - introduce a concept called "increase possibility"
  - Example: one sensor at 1:00 pm, increase algorithm result in 7 days :

[05-22: increase ;05-23: increase;05-24: increase;05-25: increase;05-26: decrease;05-27: decrease;05-28: decrease]

- increase possibility at 1:00 pm =  $4/7 = 0.57$
- Only increase possibility bigger than 0.5 will be regarded as "solar influenced"

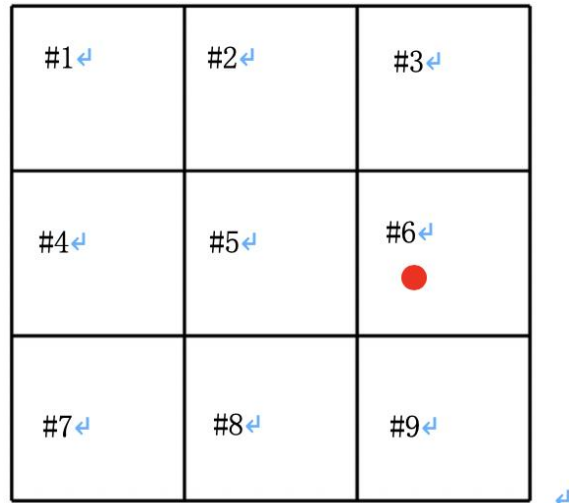
- Why generate potential location?
  - Sensors location online are not accurate
  - Create “candidates ”and find who is most suitable
  - Potential location will be inside the constrained buffer



Idea:

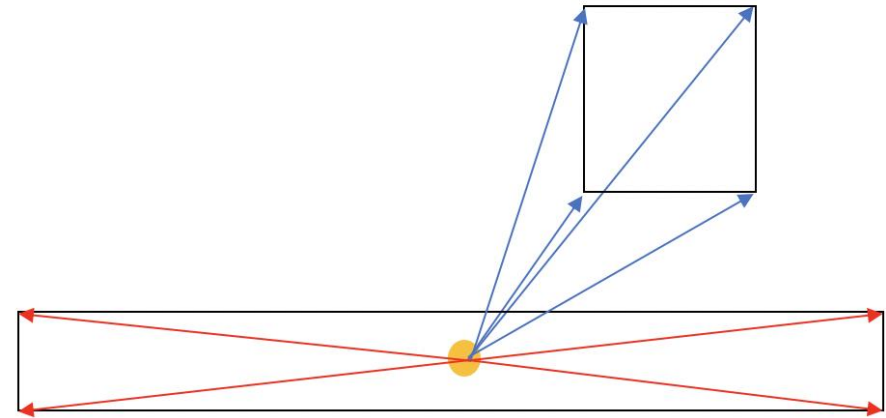
- Evenly generate points inside a circle
- For each scatter point, check if it is inside any polygon

Two solution to speed up program: tiling, finding nearest polygons

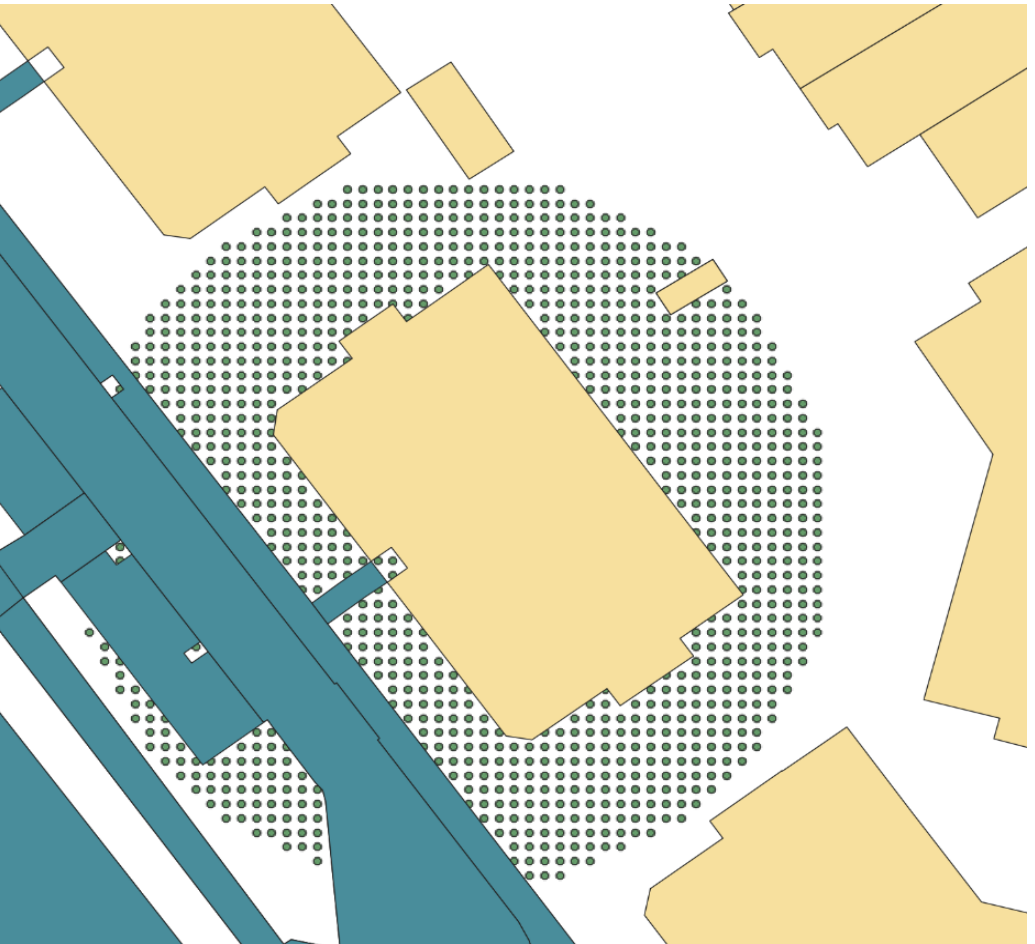


● Given location

divide Hague into several parts



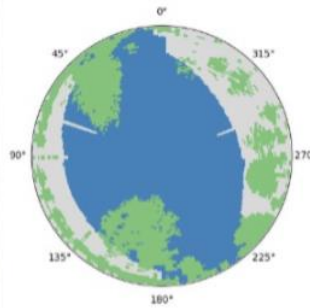
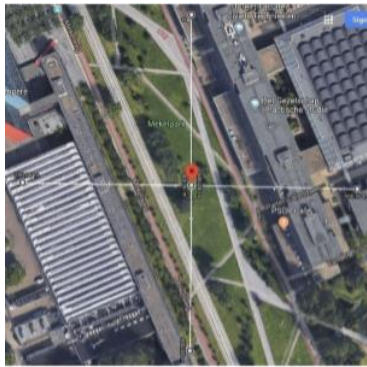
define a “point to polygon” distance by average the distance of the point to each vertex of a polygon



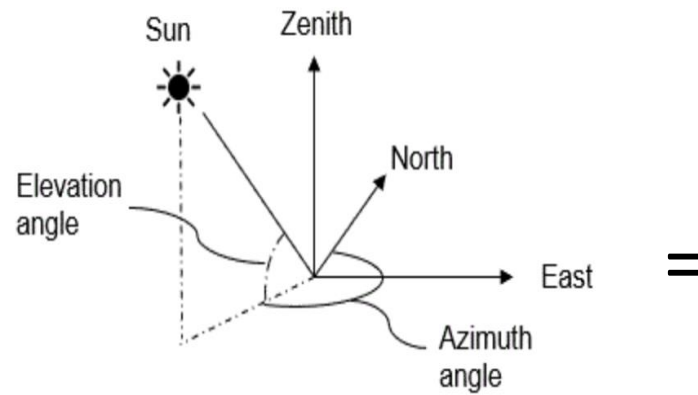
### 3 Generating potential location

# 4 Computing sky view and sun parameter

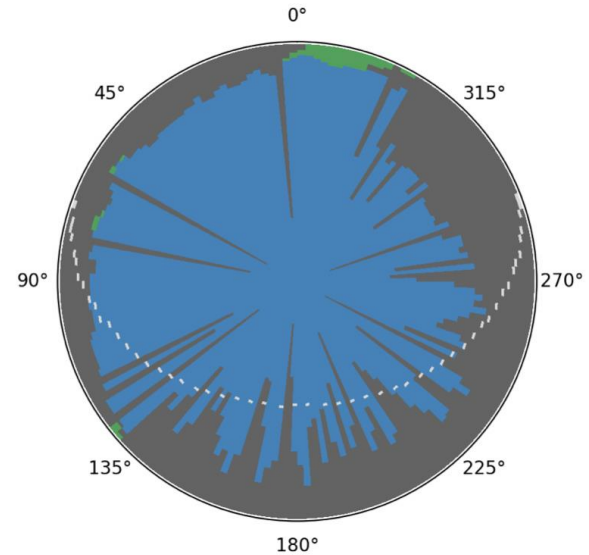
- In order to know, for each potential location, when it receives solar radiance, sky view and sun position will be computed.
  - Doom reconstruction is done by Urban Horizon project (previous Q4 thesis project)
  - Calculate sun position
  - When it receives direct solar radiance



+



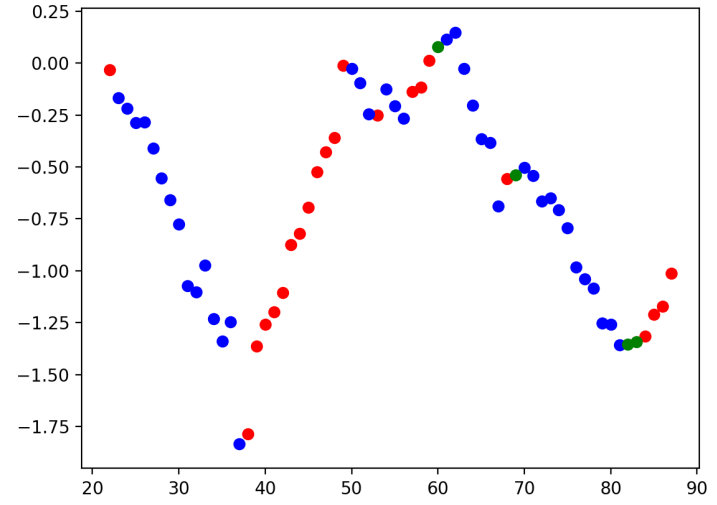
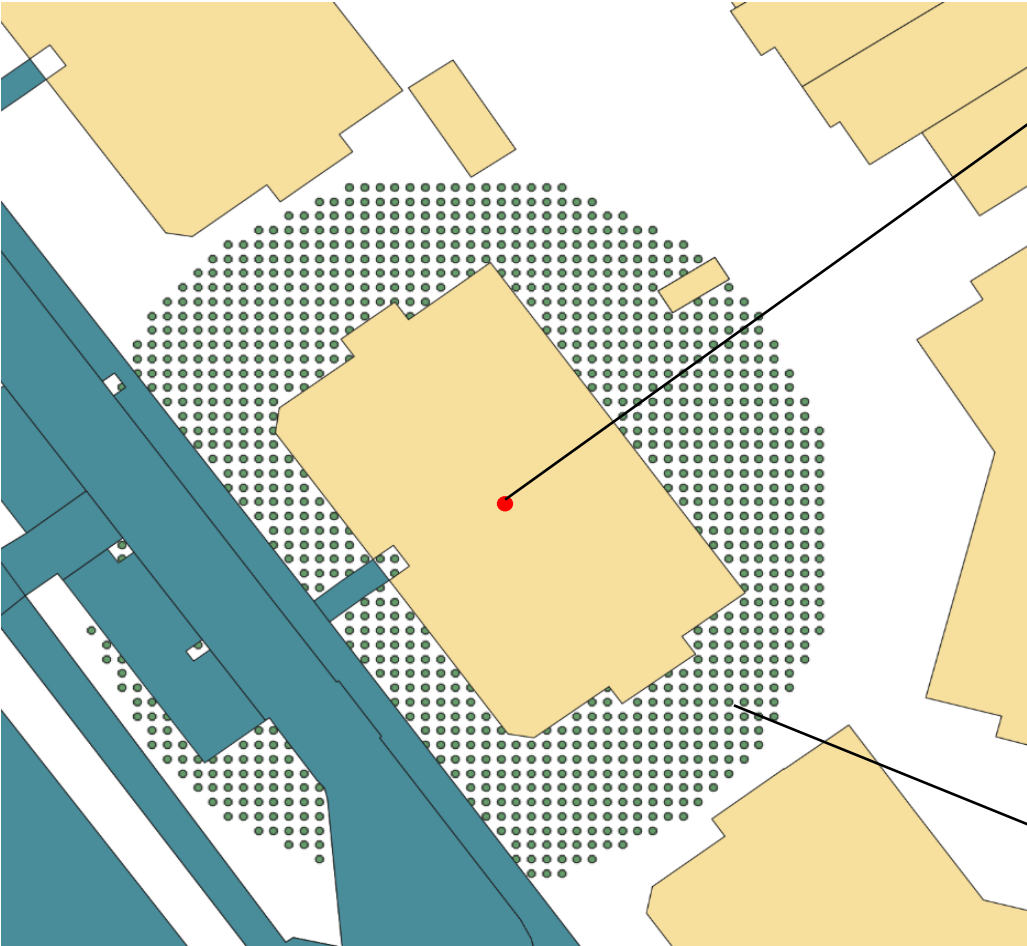
=



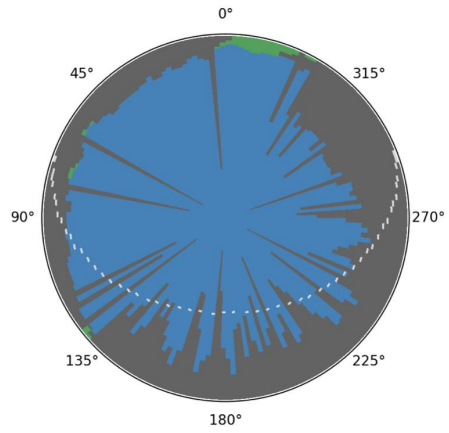
azimuth angle: north=0, east=90, south=180, west=270 degree



# 5 Finding the horizontal locations



How quantify similarity here?



The project define a formula to quantify similarity of

- temperature observation
- solar simulation

example shows below:

**Sky view result from one potential locations  $\alpha$  of sky view simulation is:**

- $\alpha$  11:15: 'influenced', 11:30: 'influenced', 11:45: 'not influenced'

**Result from 7-days increase possibility is:**

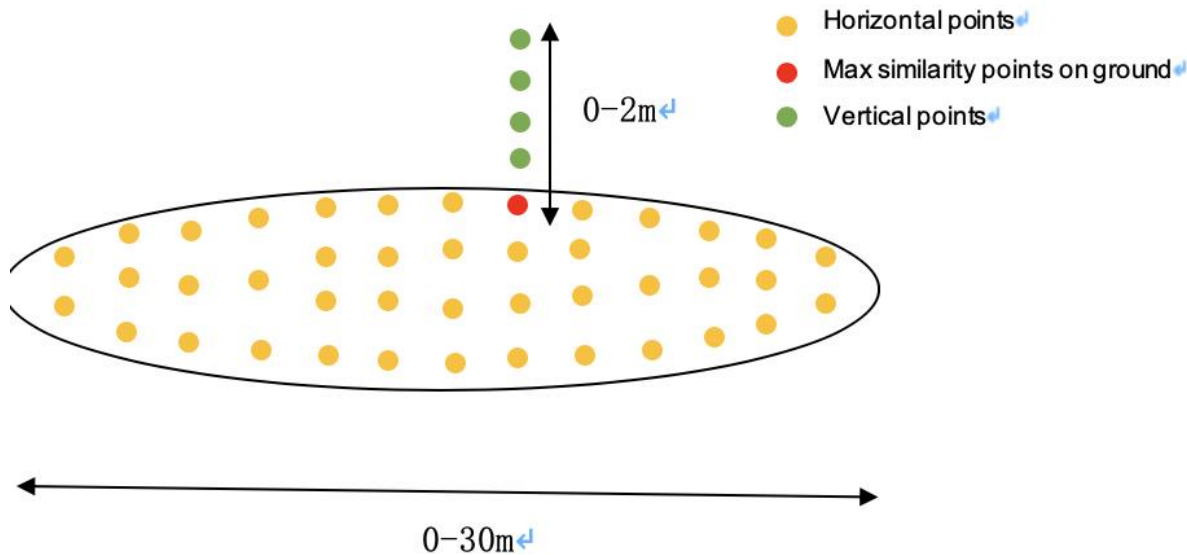
- 11:15: '0.7', 11:30: '1', 11:45: '0.6'

**Then**

- $S(\alpha)_{\text{similarity}} = (0.7+1)/(1+1+1)=0.56$

10 potential horizontal points with max similarity will be returned to create 3D points

- Result so far is 2D points.
- However, it is possible that users place their sensors above the ground.



### Approach:

Vertically generating potential points for these 10 points and then calculate similarity, and only one max similarity 3D point will be returned.

- Result
  - Similarity interpolation
  - Relocation result
- Validation
  - Comparison of result from two period of 7-days
  - Experiment by sample Netatmo weather station

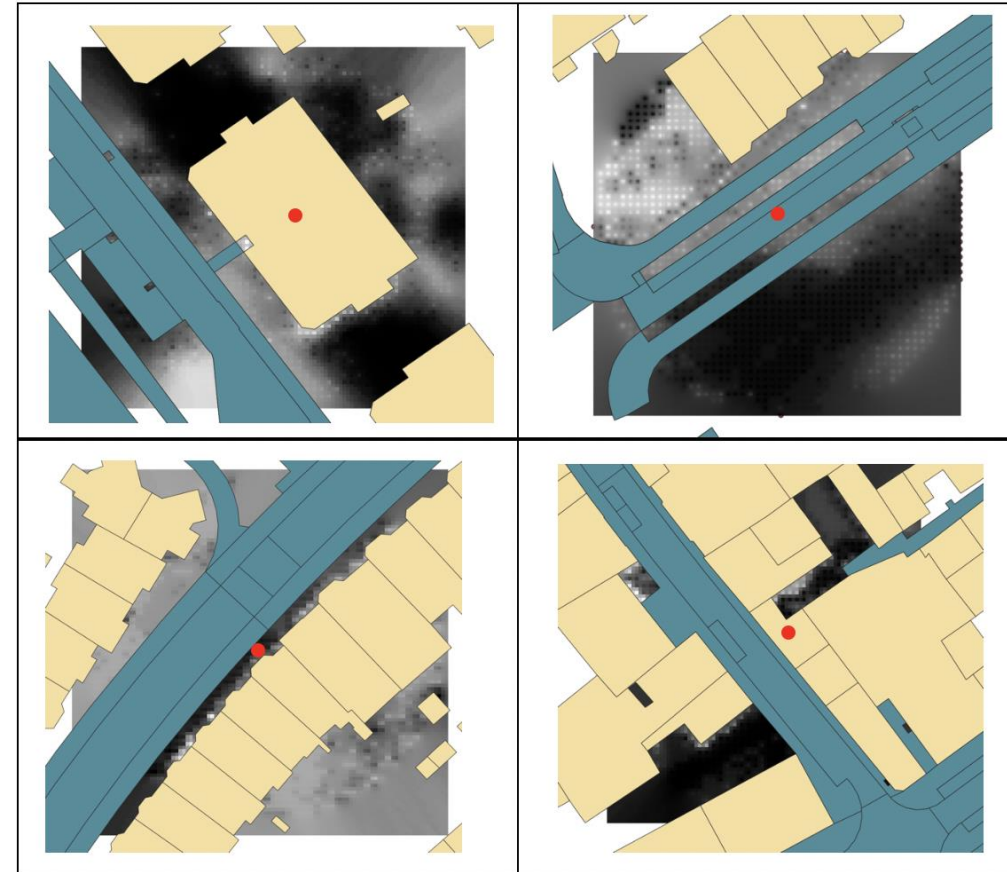
## Similarity result is not a nature phenomenon

- based on sky view and temperature records
- they all are highly related to geographic surface
- validated similarity interpolation result should shows smooth changing pattern.

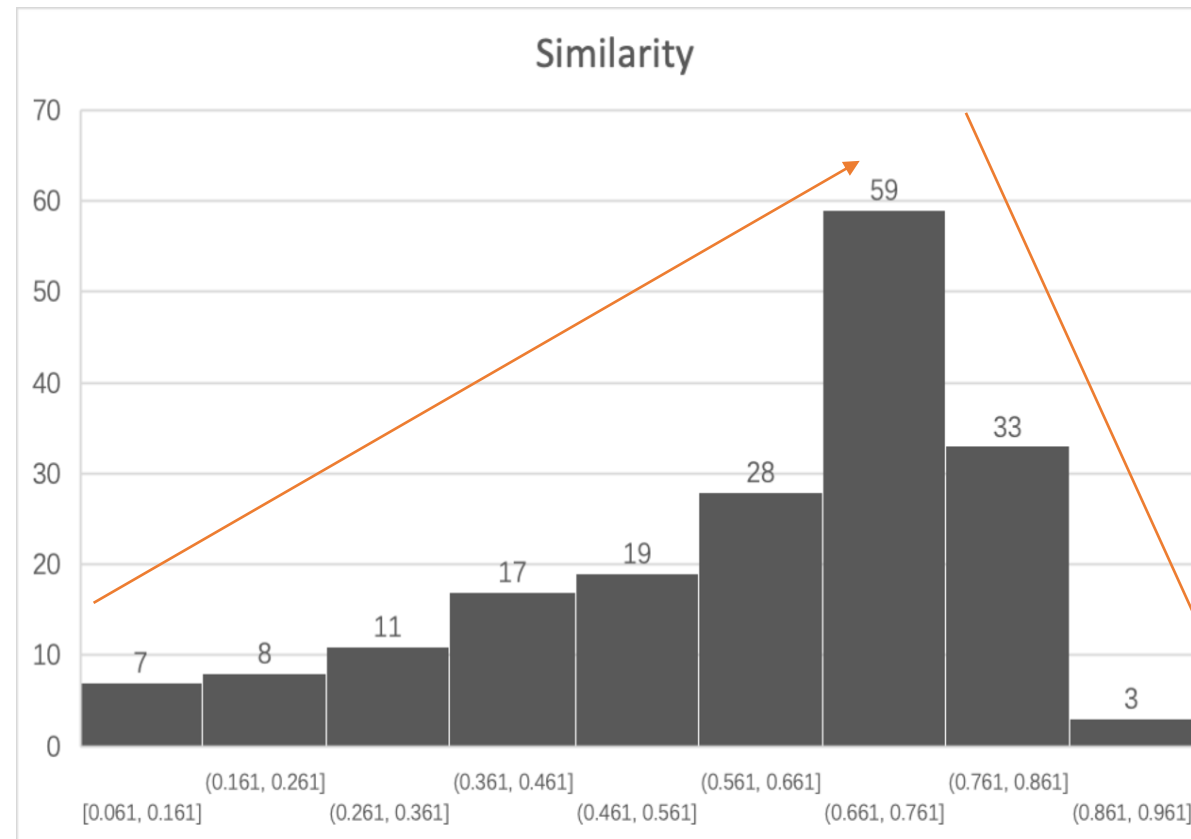
## What we can see in the interpolation map:

1. High&low possibility blocks are not totally random
2. high possibility pixels are next to edge of building

High similarity is white



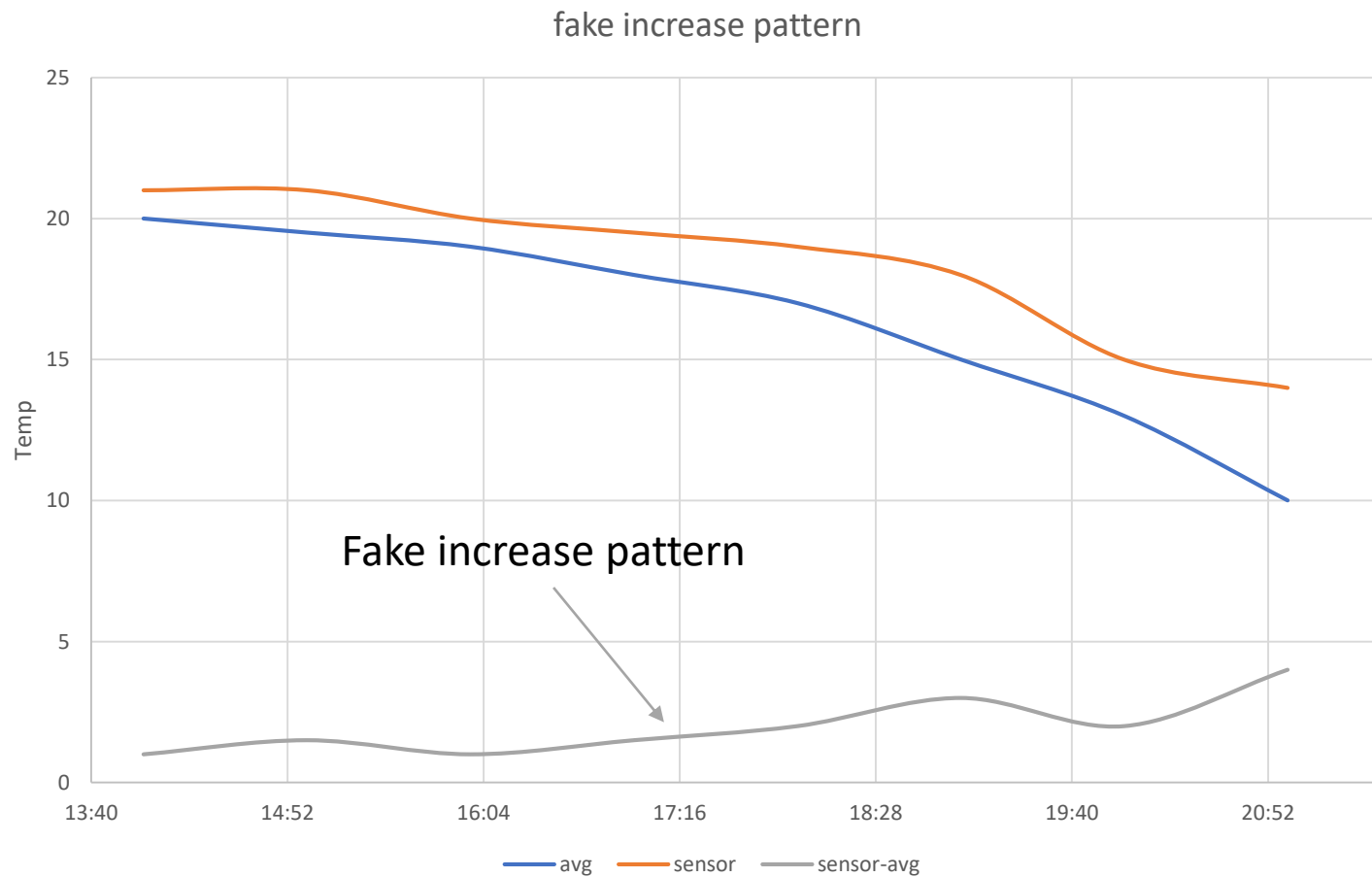
The statistics distribution of similarity result of all 185 sensors



# Similarity result

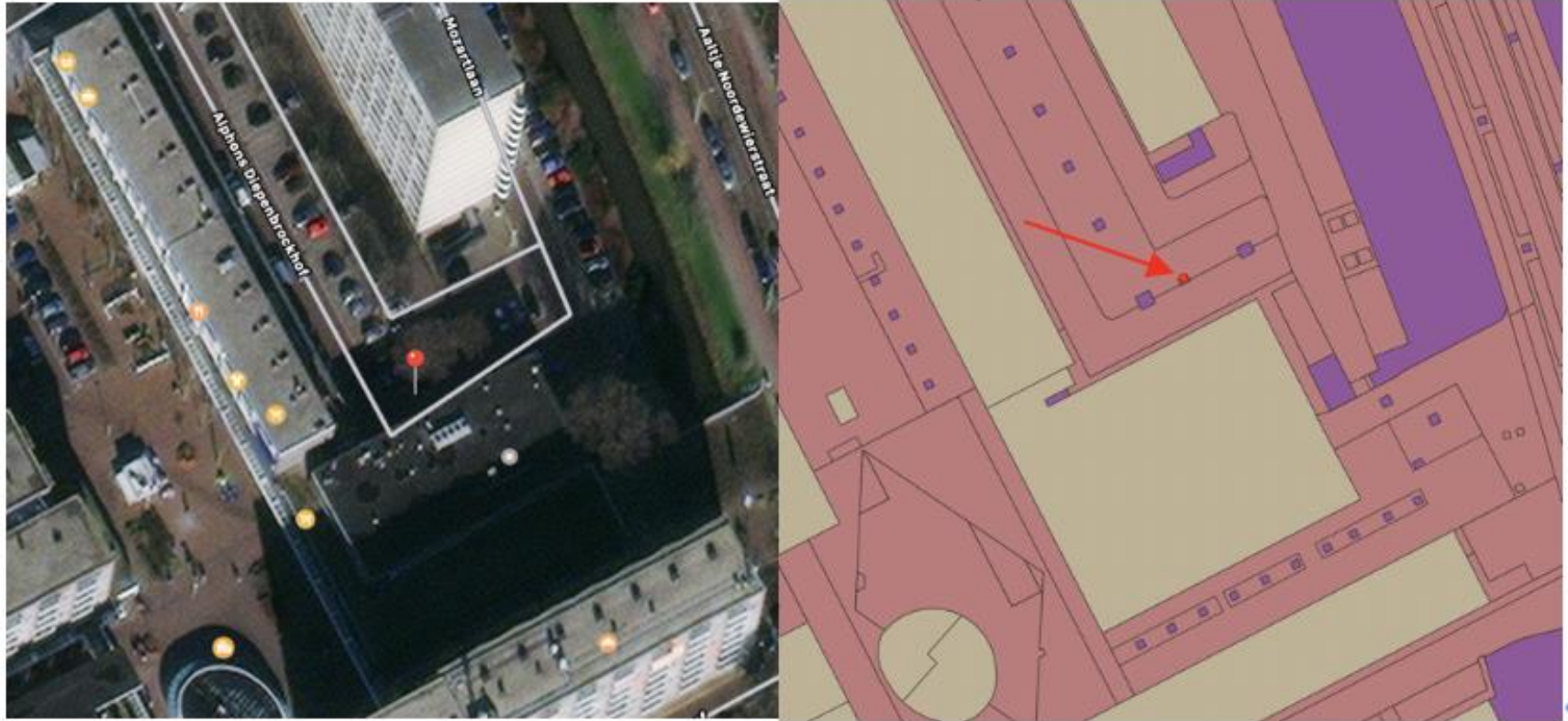
## Why low similarity

- Explanation 1 : some stations stay in shadow but their records don't accord with average temperature pattern, so the fake increase temperature pattern will appear in calculation process. If this kind of fake pattern cannot fit solar simulation result, then the similarity will accordingly be very low.



# Similarity result

## Why lo similarity



Explanation 2 : potential location failed to cover the real location of stations. Usually, stations are placed in the garden behind their house. Methodology could be wrong if users live in an apartment where users putting stations in their balcony. Because balcony is inside building polygons and will be removed.



- **Idea**

- every consecutive seven-day is a control group.
- Result of location from different periods should be more or less same.

- **The validation period is 17<sup>th</sup>, April 2018 to 23<sup>th</sup>, April 2018.**

- good weather condition in general
- not far from research period
  1. city environment won't change too much
  2. all working sensors are almost identical

# Validation

## 1. Comparison of two periods

horizontal distance (m)	Amount	height distance (m)	Amount
<5 (high quality)	130	0	95
		0.5	21
		1	8
		1.5	4
		2	2
average (m)	1.451	average (m)	0.219
5-10 (mid quality)	25	0	14
		0.5	5
		1	1
		1.5	4
		2	1
average (m)	7.427	average (m)	0.460
10-15 (low quality)	8	0	3
		0.5	3
		1	0
		1.5	1
		2	1
average (m)	12.020	average (m)	0.563
>15 (no quality)	24	0	7
		0.5	10
		1	2
		1.5	2
		2	1
average (m)	21.799	average (m)	0.545

- **4 quality levels**

The project divides result's quality into 4 levels based on horizontal distance. Over 70% stations show high quality and the average horizontal distance of them is less than 1.5m.

- **Quality interpolation (black = high)**



## Why low quality of some stations ?

Case one (blue=given location, red=research period, yellow=validation period )



- Horizontal distance (quality) is about 20 meters.
- Explanation: public green land is not suitable for placing potential location but it is not being removed in the shapefile map.

- Why low quality of some stations ?

Case two (blue=given location, red=research period, yellow=validation period )



- Horizontal distance (quality) is 42 meters.
- Explanation: objects make the environment very complex. Even for human eyes, determine where the sensor might be is difficult.

# Validation

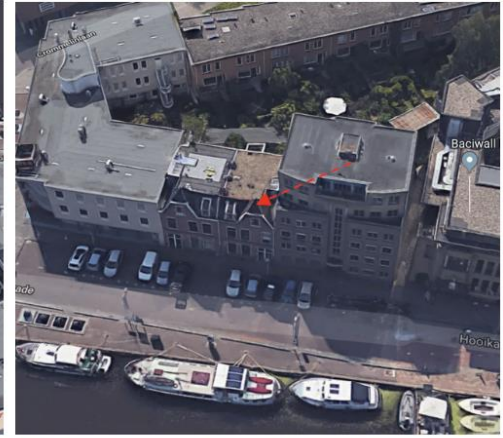
## 2. Netatmo experiment

### Idea:

Check is the program useful in the real situation

### Experiment condition:

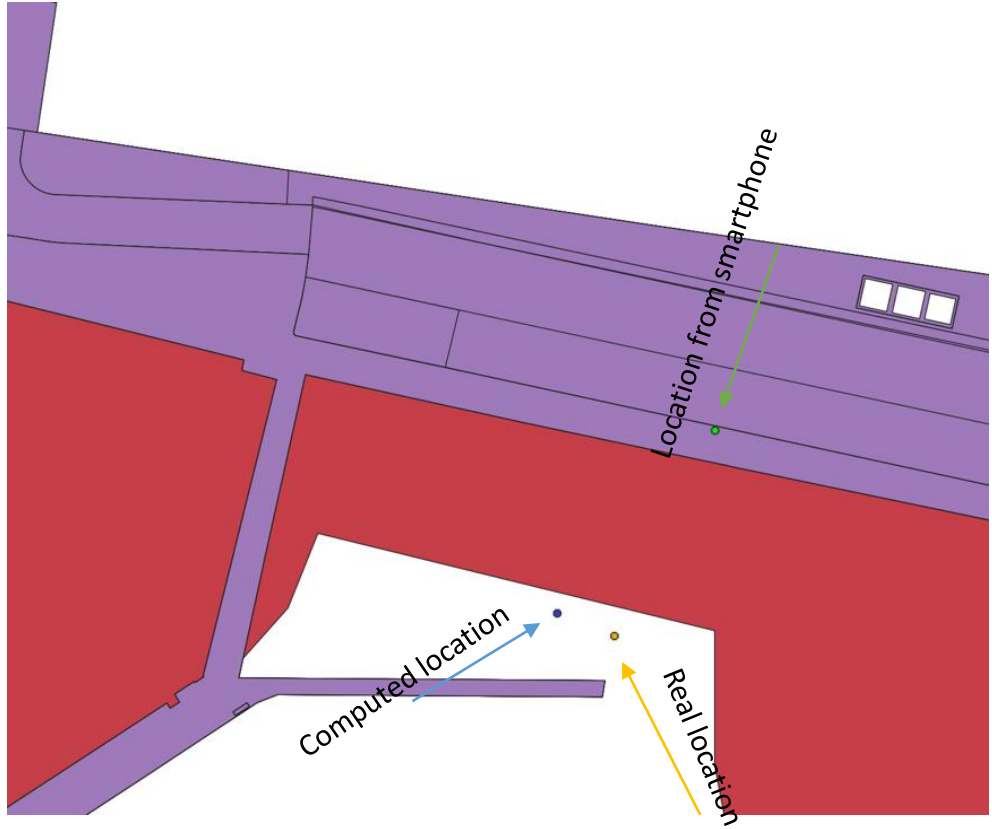
- Hooikade 26, 2627 AB, Delft
- On a 0.5 meter-height table in the garden
- Without any radiance protection
- started on 13<sup>th</sup>, May until 19<sup>th</sup>, May 2019.



# Validation

## 2. Netatmo experiment

- 3 locations:
  - Blue: computed location
  - Green: default location
  - Orange: real location
- Location difference reduce from 16 meter to 4 meter



- The results proved feasibility and rationality of the methodology
  - Stations are likely to be placed next to the wall of buildings
  - Records totally mismatched with solar simulation are not many
  - Two period comparison shows that 70% new location has a high quality
  - Program largely improve location accuracy of the experimental station
- Unstable stations performance and spatial information limits result quality
  - Irregular temperature make it hard to distinguish when and whether station is exposed to solar radiance
  - Complicated shapefile leads to some new locations appear at strange places

- Cons of the methodology
  - “average temperature” to represent real temperature still not ideal
    1. Cause fake increasing pattern
    2. Cannot find stations who stay in shadow all the time
  - Limited utilization of spatial information
    1. Cloudscape will influence solar radiance
    2. High-rise apartment users who put sensors at balcony
    3. More restrictions for potential location e.g. green land, water area



## Crowdsourced data doesn't destine to serve civilian use only.

- Future work
  1. More experiments about Netatmo weather station mechanism and working principle, a baseline test for influence from direct solar radiance.
  2. solar simulation quantify, amount of radiance that accumulates at station.
  3. Combine potential location with 3D building model.

