D7.6: Report on hackAIR updated support services and methodologies

WP7 – Pilot operation and evaluation

Document Information

Grant Agreement Number	688363	**Acronym**	hackAIR	
Full Title	Collective awareness platform for outdoor air pollution			
Start Date	1st January 2016	**Duration**	36 months	
Project URL	www.hackAIR.eu			
Deliverable	D7.6 – Report on hackAIR updated support services and methodologies			
Work Package	WP7 – Pilot operation and evaluation			
Date of Delivery	**Contractual**	31th December 2018	**Actual**	27th December 2018
Nature	Report	**Dissemination Level**	Public	
Lead Beneficiary	DRAXIS Environmental SA			
Responsible Author	Christodoulos Keratidis, Panagiota Syropoulou, Irene Zyrichidou (DRAXIS)			
Contributions from	Anastasia Moumtzidou (CERTH), Polychronis Charitidis (CERTH), Marina Riga (CERTH), Symeon Papadopoulos (CERTH), Stefanos Vrochidis (CERTH), Markos Zampoglou (CERTH), Ioannis Kompatsiaris (CERTH), Manolis Krasanakis (CERTH), Philipp Schneider (NILU), Ilias Stavrakas (TEI), George Hloupis (TEI), Carina Veeckman (VUB)			

Document History

Version	Issue Date	Stage	Description	Contributor
1.0	20/11/2018	Draft	Request input from partners	Christodoulos Keratidis, Panagiota Syropoulou, Irene Zyrichidou
2.0	21/12/2018	Draft	Input on updates on each hackAIR service/ methodology	Anastasia Moumtzidou (CERTH), Polychronis Charitidis (CERTH), Marina Riga (CERTH), Symeon Papadopoulos (CERTH), Stefanos Vrochidis (CERTH), Markos Zampoglou (CERTH), Ioannis Kompatsiaris (CERTH), Manolis Krasanakis (CERTH), Philipp Schneider (NILU), Ilias Stavrakas (TEI), George Hloupis (TEI), Carina Veeckman (VUB)
3.0	24/12/2018	Draft	Draft version available for internal review	Panagiota Syropoulou
4.0	26/12/2018	Draft	Comments from internal review received	Carina Veeckman (VUB)
5.0	27/12/2018	Final	Integration of the comments	Irene Zyrichidou

Disclaimer

Any dissemination of results reflects only the author's view and the European Commission is not responsible for any use that may be made of the information it contains.

Copyright message

© hackAIR Consortium, 2016
This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both. Reproduction is authorised provided the source is acknowledged.

Table of Contents

1 Executive summary

The scope of the hackAIR project was to deliver an open platform to raise public awareness about the problem of air pollution by enabling citizens to access to air quality data and participating in air pollution monitoring on their own.

Within the hackAIR Horizon2020 project a first version of the hackAIR platform was produced, a feasibility study with approximately 3,000 users all around Europe was implemented, and a final version of the platform was delivered based on findings from the pilots. The purpose of the feasibility study was to act as a test bed for the potential wider implementation of the hackAIR platform, and to reveal the interest of citizens and communities to adopt a solution such hackAIR. More information about the hackAIR pilot implementations and their impact can be found in D7.5-Pilot implementation report and D7.7-Final evaluation report respectively.

The current document summarizes the updates performed in the provided methodologies according to the user needs and the modifications implemented in the relevant methodologies based on the findings from their application in real-life conditions. Specifically, it describes the new air quality data sources added in the platform, updates on the air quality estimation methodology from social media posts and on the image analysis service setup. The design of the hackAIR low-cost sensors was updated so that it is easier for non-familiar citizens to assemble and use them. The present document describes not only these modifications, but also validation results that show that these changes resulted in more credible air pollution measurements. Another important update had to do with the better granularity of the data fusion results, a modification that was only tested for the city of Oslo, Norway where substantial physical modelling information was available. The hackAIR knowledge base and the decision support service for the provision of personalized recommendations were also updated to make the recommendations more meaningful and avoid misunderstandings (e.g. the reduction of outdoor workout in case the hackAIR platform shows high levels of air pollution). As regards the citizens' engagement strategy, the deviations from the initial plan are presented along with the reasons why the pilot partners implemented or not each of the proposed tactic. For the further support of the engagement strategy, updates on the discovery of relevant social media accounts have been performed. Finally, as the final version of the hackAIR platform was only recently delivered and described in deliverable D5.3 (M34), in the current document a summary of deliverable D5.3 is presented.

2 Updates on services and methodologies

The following sections describe the updates performed on each of the developed service and methodology.

2.1 The hackAIR air quality sensing component

In this section, we summarise all the technological updates, enrichments and extensions that have been implemented for supporting additional features, improvements and functionalities of the hackAIR component that supports the sensing of air quality measurements using two basic means of collective sensing approaches, i.e. measurements of existing stations and user-generated images. These updates are associated to WP3 tasks 3.1 "Indexing of environmental nodes" and 3.2 "Acquisition and processing of user-generated images for air quality prediction" and include the following:

1. Considering and incorporating the Luftdaten open initiative as a source of collective measurements into the hackAIR platform;
2. Implementing the Flickr images service for increasing the number of user-generated sky-depicting images collected by incorporating non-geotagged images (performing location estimates);
3. Improving the framework for collecting and geotagging air quality-related Twitter posts by applying transfer learning to enable air quality estimations for unmonitored cities using data from monitored nearby cities and
4. Improving the Image Analysis Service in terms of response time and reliability;

The updates involved were made on the basis of the pilots' evaluation results and users' feedback.

More details are given in the following subsections.

2.1.1 Incorporation of the Luftdaten source

To increase the number of air quality measurements considered by the hackAIR platform, and given that several air quality monitoring initiatives have emerged the last years that promote the establishment of personal environmental stations by citizens, we decided to incorporate an additional source, namely the Luftdaten initiative[1], an open initiative that promotes the collection of air quality measurements (specifically PM10 and PM2.5) from low-cost and relatively easy-to-use hardware sensors that are owned by citizens around the world, then stores the received data and eventually publishes them via a continuously updated particulate matter map or a REST based API. The initiative is supported by "The OK Lab Stuttgart"[2], which is dedicated to the fine dust measurement with the Citizen Science project luftdaten.info and is part of the Code for Germany program of the Open Knowledge Foundation Germany. The goal of the lab is to promote transparency development, open data and citizen science by developing apps that inform, positively shape and support society, and make work of governments more transparent. The sensors provided by the lab are the SDS011 and the DHT22 which are used for measuring Particulate Matters and temperature and air humidity respectively. Regarding the SDS011 sensor that measures PM 2.5 and PM 10, it is considered as a favorable device in terms of price-performance ratio according to conducted experiments by citizen scientists.

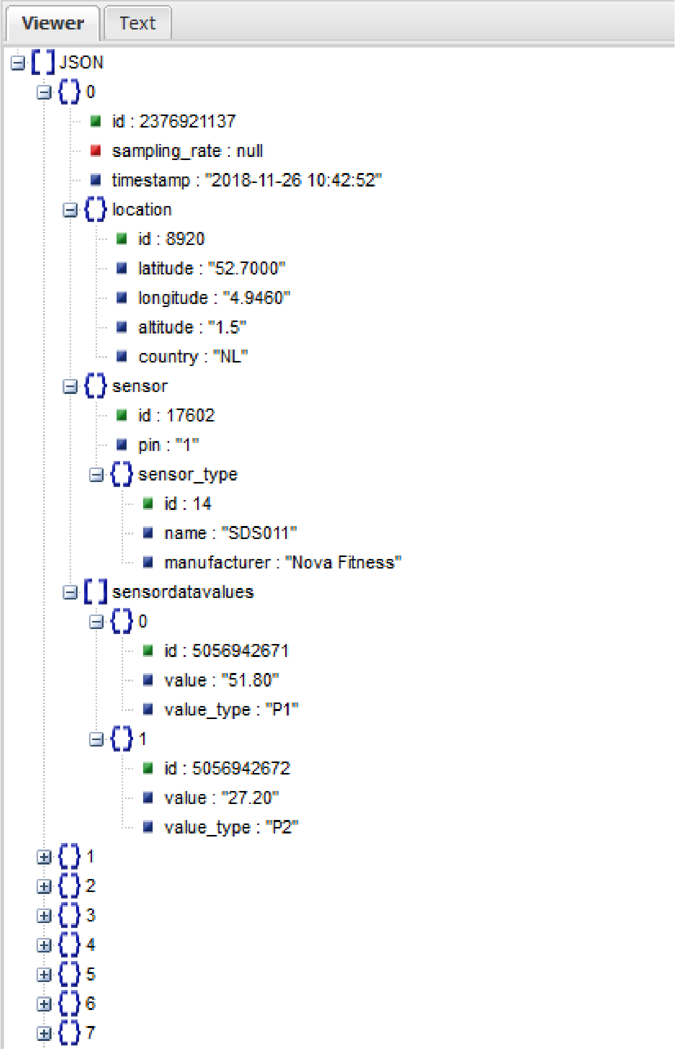
The platform offers the data as they are received from their originating sources, without performing any kind of transformations and it checks each data source for updated information every 10 minutes. Apart from the original data, the Luftdaten API provides the average value per hour and per 24 hours for each sensor.

[1] https://luftdaten.info/

[2] https://codefor.de/stuttgart/

With respect to geographical coverage, the platform collects measurements from 10,840 locations in 65 countries European and non-European countries. **Error! Reference source not found.**Table 1 shows the top 10 European countries with the most Luftdaten sensors. It is worth-mentioning that Luftdaten uses the same PM sensors as hackAIR does, meaning the SDS011 sensor. Moreover, we observe that there are more sensors measuring PM10 in each country compared to PM2.5 data. We also observe that as far as the countries of the pilots are concerned, Germany is very well represented, as there are 5,341 PM10 and 4,954 PM2.5 sensors available, while Norway has significantly less sensors (i.e. 65 and 55 respectively), which is nonetheless expected given that Luftdaten started from Germany.

Based on the aforementioned information, Luftdaten is considered an ideal source of PM10 and PM2.5 sensor measurements for the hackAIR platform and it was incorporated into the platform. In order to retrieve the data from the provided API (http://api.luftdaten.info/static/v1/data.json), a specialized data collection framework was developed. The API provides all measurements of the last 5 minutes for all sensors and it queries for new data once per hour. Figure 1 shows part of the response of the Luftdaten API. We notice that the response contains all the required information for each measurement, i.e. exact geolocation and time, value and unit.



```
⊟ [ ] JSON
    ⊟ { } 0
        id : 2376921137
        sampling_rate : null
        timestamp : "2018-11-26 10:42:52"
        ⊟ { } location
            id : 8920
            latitude : "52.7000"
            longitude : "4.9460"
            altitude : "1.5"
            country : "NL"
        ⊟ { } sensor
            id : 17602
            pin : "1"
            ⊟ { } sensor_type
                id : 14
                name : "SDS011"
                manufacturer : "Nova Fitness"
        ⊟ [ ] sensordatavalues
            ⊟ { } 0
                id : 5056942671
                value : "51.80"
                value_type : "P1"
            ⊟ { } 1
                id : 5056942672
                value : "27.20"
                value_type : "P2"
    ⊞ { } 1
    ⊞ { } 2
    ⊞ { } 3
    ⊞ { } 4
    ⊞ { } 5
    ⊞ { } 6
    ⊞ { } 7
```

Figure 1. Example data.json retrieved from the Luftdaten API.

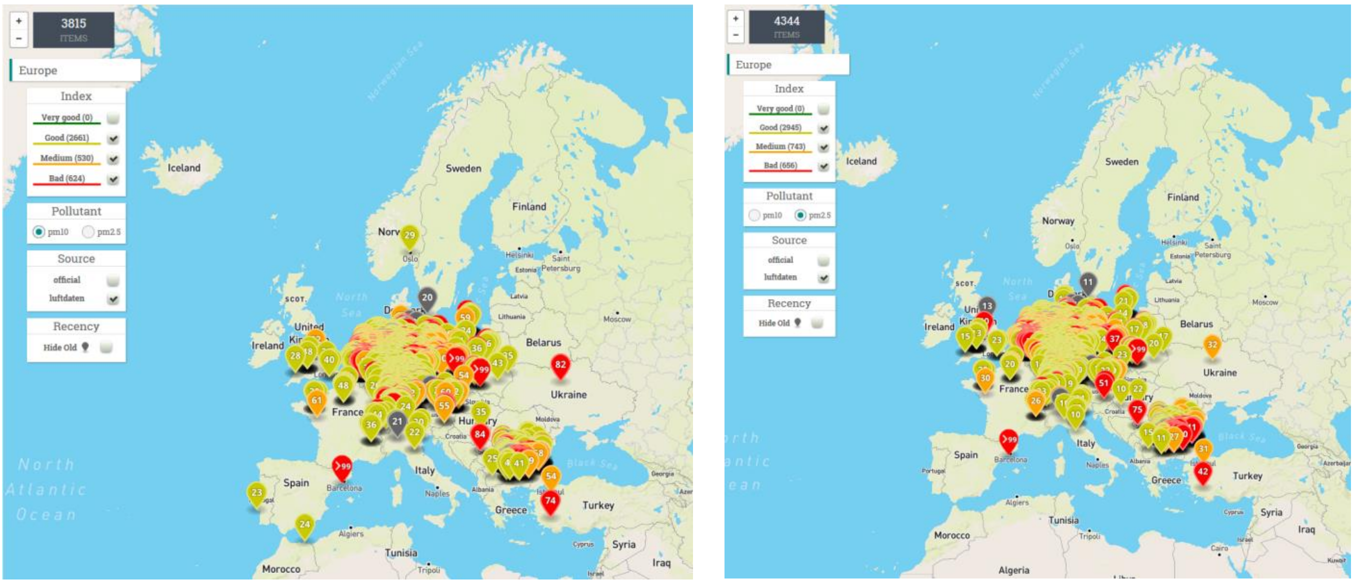


Figure 2. Screenshots of the Luftdaten sensor data collection visualization web application (Available on: *http://hackair-mklab.iti.gr/sensors/)*

Table 1. Number of Luftdaten sensors for top – 10 countries

	Sensors number	
Countries	**PM10**	**PM2.5**
Germany	5341	4924
France	831	663
Bulgaria	663	628
Belgium	627	392
Spain	519	286
Austria	416	216
Netherlands	316	201
Poland	268	197
Sweden	206	174
Italy	189	114
Norway	68	55
Other Countries	958	690
Total	**10402**	**8540**

The Luftdaten environmental data retrieval framework was deployed on 1/12/2017 and has been continuously collecting data since then. As a result, we have currently collected more than 1 million measurements in total. To facilitate the inspection of the collected data, their geographical distribution, and the current air quality conditions in Europe (in terms of PM10 and PM2.5) we built a web application[3] (Figure 2) that displays the latest PM10 and PM2.5 measurements with appropriate markers on a map.

[3] http://hackair-mklab.iti.gr/sensors/

2.1.2 Flickr images service updates

To increase the number of user-generated images collected by the hackAIR platform and given that the Flickr geographical queries for Europe return only about 5,000 geotagged images per day on average, we worked on expanding the set of retrieved sky depicting images by also utilizing non-geotagged Flickr images. To this end, we incorporated into the hackAIR platform a module described in Deliverable 3.2 (hackAIR 2017, D3.2) that estimates the capture location of the images based on textual metadata such as image tags, title and description. This module attaches geolocation information to the non-geotagged Flickr images that make up almost 97% of the total volume of images uploaded to the platform (number of total images for a month is 50,084,562) and it would therefore increase significantly the pool of images utilized for air quality estimation. The tag used for retrieving the non-geotagged Flickr images was the word sky. Table 2 contains the number of geotagged versus non-geotagged images retrieved from Flickr for a period of five days. It is worth noting that while the number of geotagged images returned by the collector is significantly higher which can be explained by the fact that only a single tag (i.e. *sky*) is used for retrieving non-geotagged images, the percentage of images containing sky over the total amount of collected images is significantly higher in the case of the Flickr-textual collector rather than the Flickr collector.

Table 2. Number of geotagged and non-geotagged Flickr Images collected for 5 days period.

	All images			Usable images		
Date	**Flickr**	**Flickr-textual**	**Total**	**Flickr**	**Flickr-textual**	**Total**
25/11/2018	6890	18	6908	630	6	636
26/11/2018	2078	25	2103	185	5	190
27/11/2018	1997	51	2048	172	5	177
28/11/2018	2084	414	2498	162	98	260
29/11/2018	638	395	1033	50	64	114
Sum	**13687**	**903**	**14590**	**1199**	**178**	**1377**

2.1.3 Updates on Air Quality Estimation from Social Media Posts

In this subsection, we present all the updates on our initial experimental work described in D3.2 (hackAIR 2017) that investigated the possibility of making air quality estimations based on the analysis of content posted on Twitter. In our updated work, we aim to create a more robust model for making air quality estimations.

2.1.3.1 Methodology Updates

Our work aims at producing estimates of current air quality conditions for cities without air quality monitoring infrastructure based on Twitter activity and measurements from nearby cities. To simplify our analysis, we focus on estimating $PM_{2.5}$ but it is straightforward to extend the approach to other pollutants like PM_{10} that was used in the previous work.

Spatial prediction deals with the problem of estimating a quantity of interest in a set of locations, on which the quantity is not measured, based on measurements of the quantity in another set of monitored locations. In this context, the quantity of interest is city-wise average $PM_{2.5.}$ We use C_M/C_U to denote monitored/ unmonitored cities respectively. Due to the high correlation between $PM_{2.5}$ values in nearby locations, the problem can be tackled using spatial interpolation techniques such as Inverse Distance Weighting (IDW), which estimates $PM_{2.5}$ in an unmonitored city as a weighted average of the observed $PM_{2.5}$ in nearby cities. In this work we follow a model-based approach. For each city

$c_j \in C_M$ we construct a set of N training examples $D_{cj} = \{(x^1, y^1), \ldots, (x^N, y^N)\}$ where $x^i \in R^D$is a d dimensional input vector that provides an informative summary of the tweets referring to c_j during the i-th temporal bin and $y^i \in R$ is the average PM2.5 concentration in c_j during the same temporal bin. For cities $c_q \in C_U$, only x_i are available and our aim is to build a model $h_{c_q} : \boldsymbol{X} \rightarrow \boldsymbol{Y}$ for each $c_q \in C_U$ in order to estimate the unknown y^i. The problem at hand can be considered as a special type of transfer learning (Pan et al., 2010) where there are multiple target learning tasks $h_{c_q}, c_q \in C_U$ for which labelled data are completely unavailable (i.e. unmonitored cities) while there are plenty of training data for a number of auxiliary tasks (nearby cities with air quality measurements) $h_{c_j}, c_j \in C_M$ that are related to the target tasks.

Assuming that air pollution exhibits a similar statistical dependence with Twitter activity in cities that share common characteristics (i.e.$P(Y^{c_j}|X^{c_j}) \approx P(Y^{c_i}|X^{c_i})$ as $sim(c_j, c_i) \approx 1$ in case c_j, c_i belong to the same country) we follow a data pooling approach and train a regression model h on $D = \cup_{c_j \in C_M} D_{c_j}$ that learns to simultaneously minimize the prediction error on all monitored cities and we therefore expect it to yield accurate predictions for the unmonitored cities as well.

Besides data pooling, we also apply explicit feature selection to ensure that the learned model will be constrained to a subset of the Twitter-based features that are highly correlated with PM2.5 in all cities. To this end, we compute the Pearson correlation[4] coefficient between each feature X_i and the target Y and keep the k features that exhibit the highest correlation. This lower dimensional feature representation is expected to facilitate learning a more robust, city-invariant model.

Additionally, under the assumption that the smaller the distance $d(c_j, c_i)$ between two cities, the higher the similarity of their conditional distributions, we develop a weighted data pooling variant where each training example gets a weight that is inversely proportional to the distance between the city it belongs to and the target city. In other words, our model counts more on data from nearest cities to make its predictions rather than the further ones.

To achieve even better results we also adopt a late fusion scheme that combines our Twitter-based estimates with the IDW estimates by learning a meta-model that uses two features: a) IDW estimates for the training cities, b) Twitter-based estimates for the training cities (obtained through inner cross-validation).

2.1.3.2 Data Collection Updates

We follow almost the same data collection scheme presented in D3.2 (hackAIR 2017) for both Twitter Data Collection and Collection of Ground Station Measurements with some small updates.

The location estimation method of Kordopatis-Zilos et al., 2017 is now applied as follows. Since we are actually interested in the location that the tweet refers to instead of the upload location, we first check if a location can be estimated with high confidence (≥ 0.8) based on the tweet content and in case it does we use it as the tweet location. Otherwise, similarly to previous works, we use the account's declared location as the tweet location. However, instead of relying on simple text matching, we again perform location estimation using the account's location description as input.

Also, we expand the Twitter data collection of five English cities (London, Birmingham, Liverpool, Leeds and Manchester), by including five American cities (New York, Baltimore, Philadelphia, Boston and Pittsburgh). The Twitter data alongside with Ground Station Measurements are collected for a time period spanning almost a whole year (8/2/2017-18/1/2018). As a result, this dataset is significantly larger compared to the one used in D3.2 (hackAIR 2017).

[4] https://en.wikipedia.org/wiki/Pearson_correlation_coefficient

2.1.3.3 Updates on Feature Extraction

Our goal is to generate features that provide a descriptive summary of the tweets posted around a specific city during a fixed period of time (timestep), and at the same time exhibit strong correlations with the target variable (PM2.5). As in D3.2 (hackAIR 2017), each day is divided into four 6-hour timesteps and features are computed over all tweets posted during the respective timestep to generate the feature vector. We keep the best 3 features (those that yield the best results in terms of RMSE) listed in Table 10 in D3.2 (hackAIR 2017). These features are tweetsNo, AQ_S and high. In our new approach, alongside these three features we also test new, more predictive features by utilizing tweet information in a more explicit manner.

To generate a descriptive representation of the tweets assigned to a city c during a temporal bin t (i.e. spatiotemporal bin (c, t)), we use a BoW scheme. First, all tweets are preprocessed by applying tokenization, lowercasing and stopword removal. Then, we create a vocabulary $W = \{w_1, \dots, w_n\}$ that consists of the $n = 10000$ most frequently occurring words in a random 1 million sample of the collected tweets. Using this vocabulary, a BoW vector $x = [x_1, \dots, x_n]$ is generated to represent all tweets in (c, t), where x_i denotes the number of tweets containing w_i divided by the total number of tweets in (c, t). Each element of the vector corresponds to a single word and the value is the number of tweets that contain this particular word divided by the number of tweets in a temporal bin.

In addition to this “current” BoW representation, we also generate lagged BoW representations (denoted as BoW^{-j}), where instead of considering only the tweets posted during the current temporal bin t we also consider the tweets of the j previous bins.

2.1.3.4 Updated Air Quality Estimation Experiments

2.1.3.4.1 Experimental setup

Each city is in turn treated as the test city (hypothetically without air quality measurements) and all the remaining neighbouring cities are used for training. For each city, we train and evaluate models able to perform predictions at three different temporal granularities: 6, 12 and 24 hours. This is accomplished by grouping the hourly $PM_{2.5}$ observations into correspondingly sized temporal bins and calculating a single ground truth $PM_{2.5}$ value for each bin as the average of the hourly values. In some cases measurements from ground truth stations are missing. We deal with this by ignoring the missing values and we average only the valid measurements. If there are no valid measurements in a temporal step then we remove this from the dataset. Prediction accuracy for each city and temporal granularity is measured in terms of Root Mean Squared Error (RMSE) and macro averaging is applied to calculate country-wise or overall performance (denoted as aRMSE). In all our experiments we use Gradient Tree Boosting as the regression algorithm, since it is recognized as one of the best off-the-shelf supervised learning algorithms (Friedman et al., 2001) and was found to perform equally good or better compared to other algorithms in a set of preliminary experiments.

The implementation of the solution along with the necessary datasets is available as an open-source project on GitHub: https://github.com/MKLab-ITI/twitter-aq.

2.1.3.4.2 Baseline performance

Figure 3 shows a scatter plot of the distances and the Pearson r between average daily $PM_{2.5}$ concentrations for all distinct city pairs in UK and US. Clearly, the smaller the distance between two cities, the higher the correlation between their average daily $PM_{2.5}$ concentrations. Given this high spatial dependence, it is not surprising that spatial interpolation methods such as IDW yield accurate estimates as shown in Table 3. Moreover, we see that a baseline that always predicts the mean $PM_{2.5}$ value per city, although having worse performance than IDW, still has a relatively small error, which is due to the fact that $PM_{2.5}$ levels in the studied cities are generally low and exhibit small variability. Another thing that we can observe from Table 3 is that in larger temporal granularities the error tends to become smaller. This behaviour can be explained because in larger granularities there are more averaged data and make the prediction task less strict.

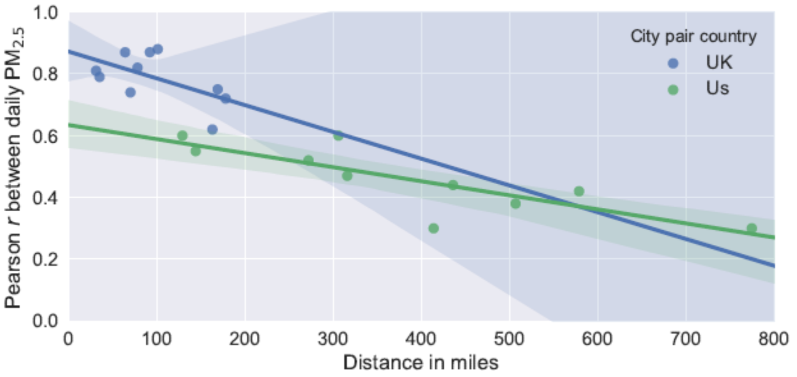


Figure 3. Scatter plot of distances and Pearson r between average daily $PM_{2.5}$

Table 3. aRMSE of baseline method

	UK			US			Overall		
	6h	12h	24h	6h	12h	24h	6h	12h	24h
IDW	3.79	3.34	3.09	4.12	3.73	3.73	3.96	3.54	3.25
Mean	7.00	6.11	6.36	4.60	4.26	4.02	5.80	5.46	5.19

2.1.3.4.3 Within-city predictions

Before evaluating spatial $PM_{2.5}$ prediction using our transfer learning approach, we first evaluate the predictability of $PM_{2.5}$ in each city using a model trained on Twitter and ground truth data of the city. As already discussed, this represents an unrealistic setup because ground truth data is not available for unmonitored cities. However, it is suitable for assessing the effectiveness of different Twitter features. In this set of experiments, data from each city is split based on time, using odd months for training and even months for testing.

Table 4. aRMSE of different Twitter features

	#tw	#aqs	#high	All	Bow	Bow-1	Bow-2
6h	5.96	5.93	5.98	5.84	5.15	4.99	4.97
12h	6.17	5.98	6.02	5.77	4.96	4.84	5.16
24h	5.83	6.11	5.82	5.52	4.65	4.96	5.16

Table 4 shows the results obtained using models trained on “current” and lagged BoW features, as well as four simpler Twitter feature. ‘#tw’ (total number of tweets in each spatiotemporal bin formerly denoted as tweetsNo), ‘#aqs’ (number of tweets that provide information on current air quality formerly denoted as AQ_S), ‘#high’ (number of tweets that refer to high air pollution levels) and ‘all’ (the concatenation of ‘#tw’, ‘#aqs’ and ‘#high’). We notice that for all temporal granularities, ‘all’ leads to better accuracy than ‘#tw’, ‘#aqs’ and ‘#high’, suggesting that these features capture complementary information about current air quality. However, we see that the best performance for each temporal granularity is obtained by a BoW variant and, interestingly, we notice that for finer temporal granularities it is beneficial to use lagged BoW features (BoW-2 and BoW-1 for the 6- and the 12-hour temporal granularity, respectively). Based on these results, subsequent experiments employ the best performing BoW representation for each temporal granularity.

2.1.3.4.4 Cross-city predictions

Table 5 shows the results obtained when using full-dimensional BoW vectors ('full' column) as well as vectors where only the top-k most correlated features are kept, with (w=1) and without (w=0) sample weighting[5]. First, we observe that the performance of full-dimensional BoW is considerably worse compared to the within-city setup. As expected, the absence of city-specific training data makes the learning task more difficult. With respect to the different transfer learning setups, we see that joint feature selection results in important performance gains in all temporal granularities, with the best results obtained when the top 50 or 100 features are used. As we mentioned in the top features are selected by calculating their Pearson correlation with ground truth $PM_{2.5}$ measurements. This will ensure that the most descriptive features will be used to train the regression models. Sample weighting, on the other hand, has a less pronounced but consistently positive effect.

Table 5. Cross-city aRMSE with different transfer learning setups

		full	k=10	k=20	k=50	k=100	k=200	k=500
W=0	6h	5.36	5.48	5.28	5.21	5.24	5.29	5.31
	12h	5.21	5.29	5.18	5.12	5.09	5.11	5.15
	24h	4.97	4.89	4.78	4.78	4.75	4.79	4.86
W=1	6h	5.35	5.47	5.27	5.21	5.24	5.29	5.16
	12h	5.21	5.26	5.18	5.11	5.08	5.11	5.16
	24h	4.95	4.85	4.77	4.76	4.73	4.77	4.84

Comparing the performance of our Twitter-based estimates with those of IDW, we notice that they do not perform on par. We believe that this result should be largely attributed to the fact that the studied cities exhibited very good air quality conditions for an overwhelming part of the studied period which makes it less likely for people to express their feelings about air quality on Twitter. Our findings match those reported in (Mei et al., 2014) where IDW was also found more accurate than the proposed approach under good air quality conditions.

We notice the same trend that in larger temporal granularities the error tends to be smaller. This can be explained if we take into consideration that some of the 6-hour temporal bins correspond to late night hours where Twitter usage is restricted. This means that for these timesteps, predictions are not going to be accurate. Instead, in 24-hour bins, daily twitter information is aggregated leading to more normalized and accurate results.

Figure 4 depicts the prediction of $PM_{2.5}$ values with sample weighting in the city of London. Actual corresponds to $PM_{2.5}$ ground truth measurements. IDW is the baseline prediction and Twitter is the combinations of Twitter features with spatial interpolation. The prediction period on this figure covers approximately two and a half months.

[5] Each sample is weighted based on the normalized Inverse Distance Weight (IDW), meaning that samples from nearby cities affect the model more. An example of an SVM Classifier using the sklearn implementation of sample weights, can be found in this link: https://scikit-learn.org/stable/auto_examples/svm/plot_weighted_samples.html

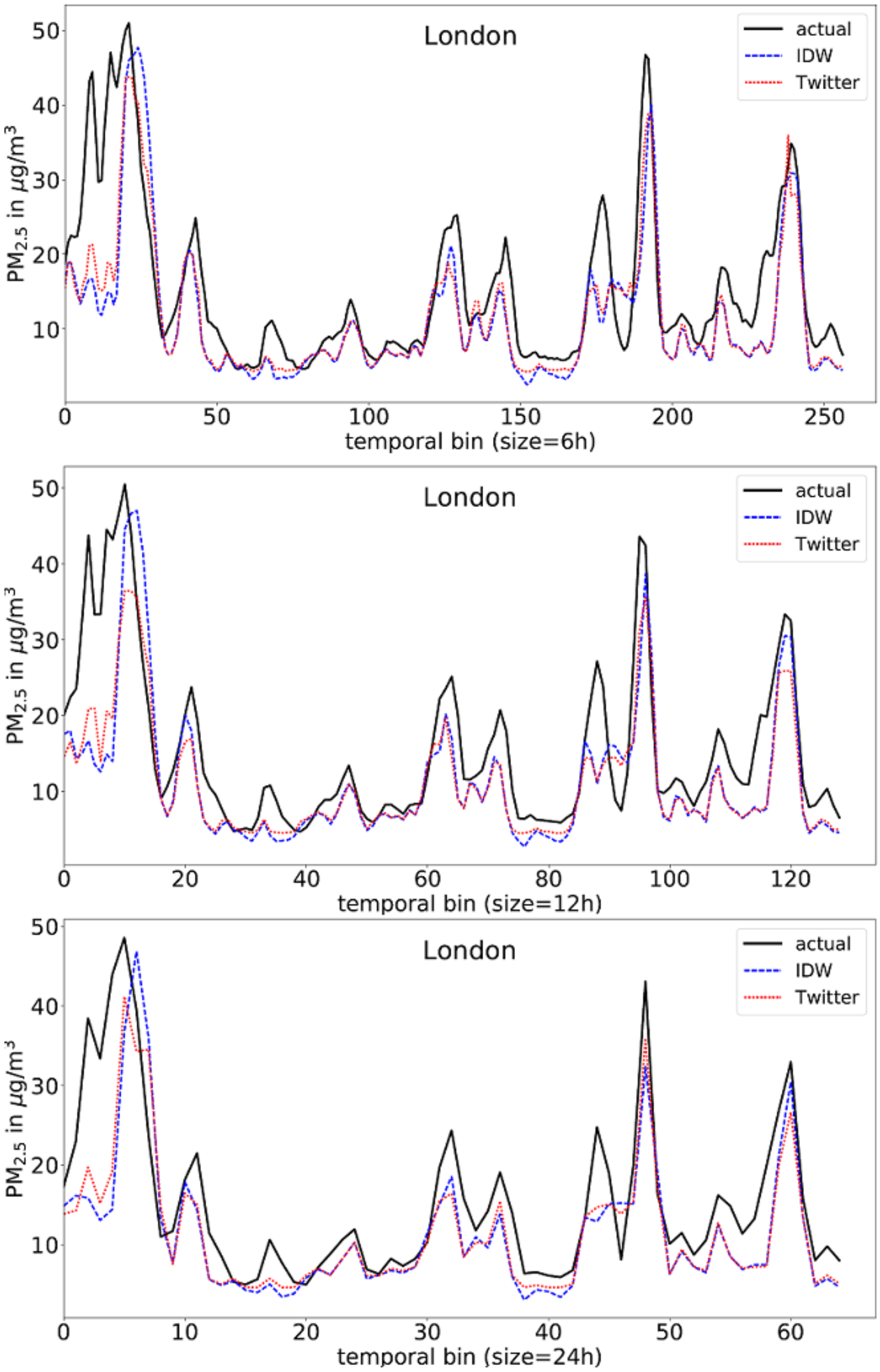


Figure 4. Prediction results on London for 6h, 12h and 24h timesteps

Despite the fact that IDW yields better results, we notice that Twitter-based estimates carry considerable predictive power as they manage to obtain significantly lower error than the mean baseline. Motivated by that, we evaluated a late fusion scheme that combines our Twitter-based estimates with the IDW estimates by learning a meta-model that uses two features: a) IDW estimates for the training cities, b) Twitter-based estimates for the training cities (obtained through inner cross-validation). This model obtains an aRMSE of 4.15, 4.00 and 3.63 for the temporal granularities of 6, 12 and 24 hours respectively. Although its performance is still worse on average compared to IDW, Table 6 shows that it performs better than IDW in 3 out of 10 cities: Boston, London and Pittsburgh.

Table 6. Per city results

City	IDW error	Twitter with IDW
Baltimore	3.12	3.85
Birmingham	2.28	2.95
Boston	3.34	3.02
Leeds	3.32	3.36
Liverpool	3.65	4.01
London	4.66	4.25
Manchester	3.12	4.01
New York	3.14	3.37
Philadelphia	3.98	4.2
Pittsburgh	5.16	4.88

2.1.4 Image Analysis Service Setup

The image analysis (IA) service incorporates all the operations required for the extraction of R/G and G/B ratios from sky-depicting images. It is a Java web service that accepts image analysis requests, carries out all the required processing, and responds with the results of the image analysis. The service accepts HTTP post requests that specify either a set of local paths that correspond to images already downloaded on the server through one of the image collectors (Flickr or webcams), or a set of image URLs.

The IA service uses internally the following three components, concept detection (CD), sky localization (SL) and ratio computation (RC). The CD component applies concept detection on each input image and returns a set of scores that represent the algorithm's confidence that the sky concept appears in it. The SL component implements the (FCN) Fully Convolutional Networks-based sky localization framework, which is a computationally heavy processing step that is carried out on the GPU, and returns the sky mask of each input image. The RC component takes the sky masks computed by the Fully Convolutional Networks approach as input, refines them by applying a heuristic approach D3.2 (hackAIR 2017) and computes the R/G and G/B ratios of each image. More information on the three components can be found in the deliverables D3.1 (hackAIR 2016) and D3.2 (hackAIR 2017).

The overall architecture of the service can be summarized in the following steps and is visualized in Figure 26 of D3.2 (hackAIR 2017):

- IA receives a request from an image collector (Flickr or webcams)
- IA sends a request to CD component
- IA parse the response send by the CD component to check which images are the most likely to depict sky.
- IA service sends a request to the SL component
- IA service receives the response from the SL component that is the sky mask of the request image
- IA sends a request to the RC component
- IA service parses the response of the RC component
- IA combines the results of all processing steps to synthesize the IA response

In the initial setup, IA service was distributed among three different machines; a practice that was realized because the CD and SL components involved were computationally heavy and because the CD component initially required a Windows machine, while the SL component a Linux machine. Therefore, initial setup comprised the following servers:

- Windows server 2016 provided by DRAXIS with the following specifications: Intel(R) Xeon(R) CPU E5-2630 v4 @2.20GHz, 64bit, 50GB RAM
- Linux machine owned by CERTH with the following specifications: Ubuntu 16.04.5, Intel(R) Xeon(R) CPU E5-2620 2.0GHz, 125GB RAM, 24 Cores, GeForce GTX 1070
- Linux machine owned by CERTH with the following specifications: Ubuntu 16.04.4, Intel(R) Core(TM) i7-3770K 3.5GHz, 15GB RAM, 8 Cores, GeForce GTX 1070

Specifically, the Windows server hosted the IA service itself, the Image Collectors, the relatively lightweight RC component, the CD component and a Mongo Repository that stored the information produced by the above modules. The improved CD component that was implemented based on the TensorFlow[6] deep learning framework was hosted in the i7 - Linux machine. Finally, the e5 - Linux machine hosted the SL component that was implemented based on the Caffe[7] deep learning framework. It should be noted that the deployment of CD and SL components in different servers was decided to make sure that sufficient computational resources were available.

However, this setup introduced delays because both the image information as well the images themselves had to be downloaded to each machine in order to be processed by each component. Thus, a new deployment was realized that involved a single Linux machine that hosted all the aforementioned processes. The new Linux server was provided by DRAXIS and had the following specifications: Ubuntu 16.04.4, Intel Core i7-3930K 3.2GHz, 12 Cores, 64GB RAM 200GB HDD, GeForce GTX 1070. The new deployment reduced significantly the runtime of the IA service as shown in Table 7. Figure 5 depicts the old service deployment versus the new one for two samples of 10 and 100 images.

Table 7. Response time of IA service deployments

	Response Time (10 images)	Response Time (100 images)
Old IA service deployment	25,6 sec	98,5 sec
New IA service deployment	7 sec	35,1 sec

It should be noted that during the deployment of the IA service and its components in single machine, several issues arose, including the difficulties of TensoFlow and Caffe coexistence that involved not only the frameworks themselves but also the CUDA versions installed for running the CD and SL components with GPU.

Moreover, apart from the deployment of the IA service in a new machine, updates on the service itself were realized that aimed at:

- handling problematic images that caused repeated requests by the orchestrator which eventually led to bottlenecks in the hackAIR pipeline;
- responding to updated requests by the hackAIR orchestrator;
- authentication mechanisms were applied for securing both the tomcat server and the mongoDB.

[6] https://www.tensorflow.org/

[7] http://caffe.berkeleyvision.org/

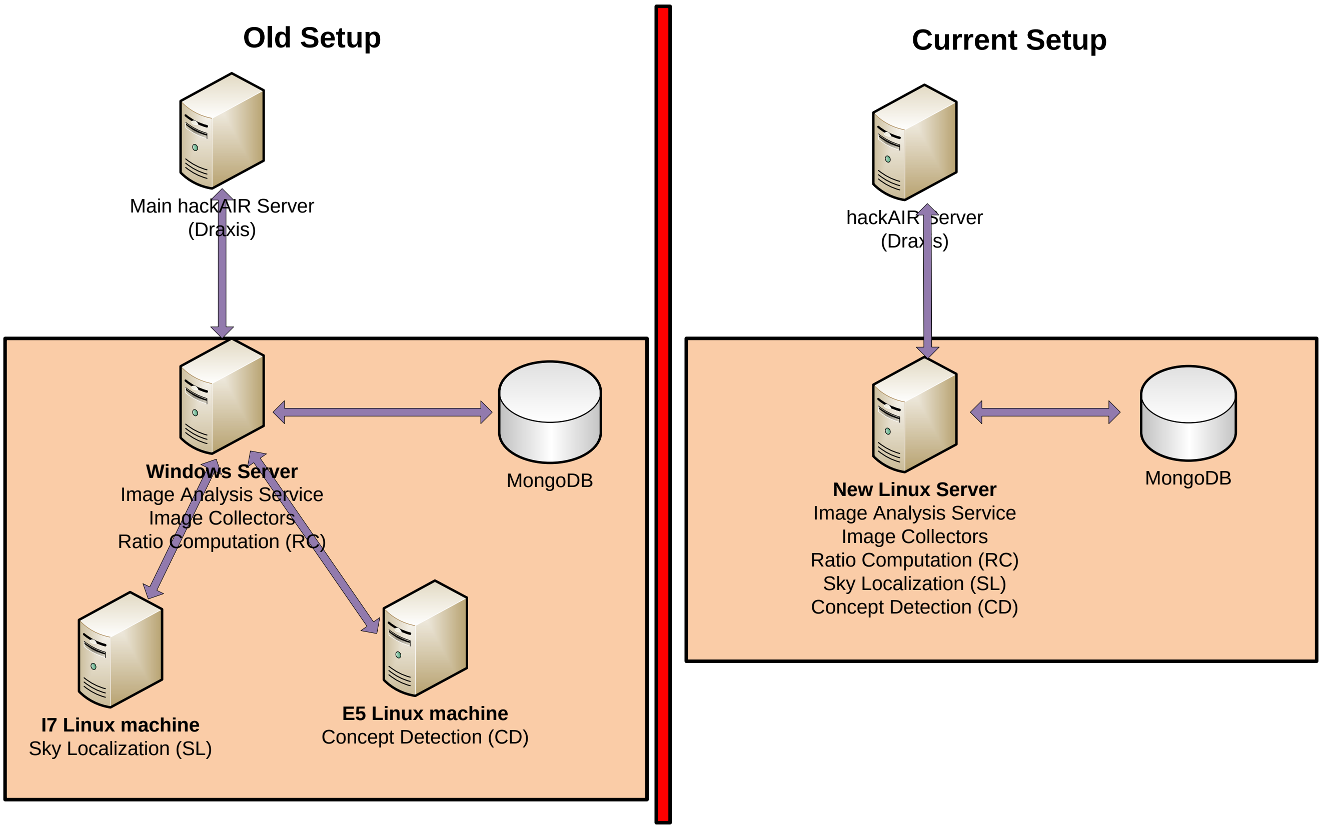


Figure 5. Image Analysis service deployment

The three image collectors, i.e. the Flickr collector, the webcams.travel collector, and the webcams collector that contains two webcam sources from the meteo[8] site have been collecting images from the beginning of 2017. During this period and until the end of November 2,625,437 images had been collected in total across the whole Europe from all sources. Figure 6 shows the number of images collected daily from each source. The number of images collected per day by the two webcam image sources is almost stable. In particular, 1,892 webcams from webcam.travel and the 2 meteo webcams are visited four times per day and, as a result, about 4,000 images, and respectively, 8 images are collected daily from these sources. On the other hand, as far as the number of images collected daily from Flickr is concerned, it exhibits a large variability since it depends on the number of geotagged images (in Europe) that are uploaded daily by Flickr users. Figure 7 depicts the total number of images collected per day from all sources (average around 11,000 images) while Figure 8 depicts the images that can be actually used for air quality estimation which is roughly 1,030 on a daily basis.

[8] http://meteo.camera/

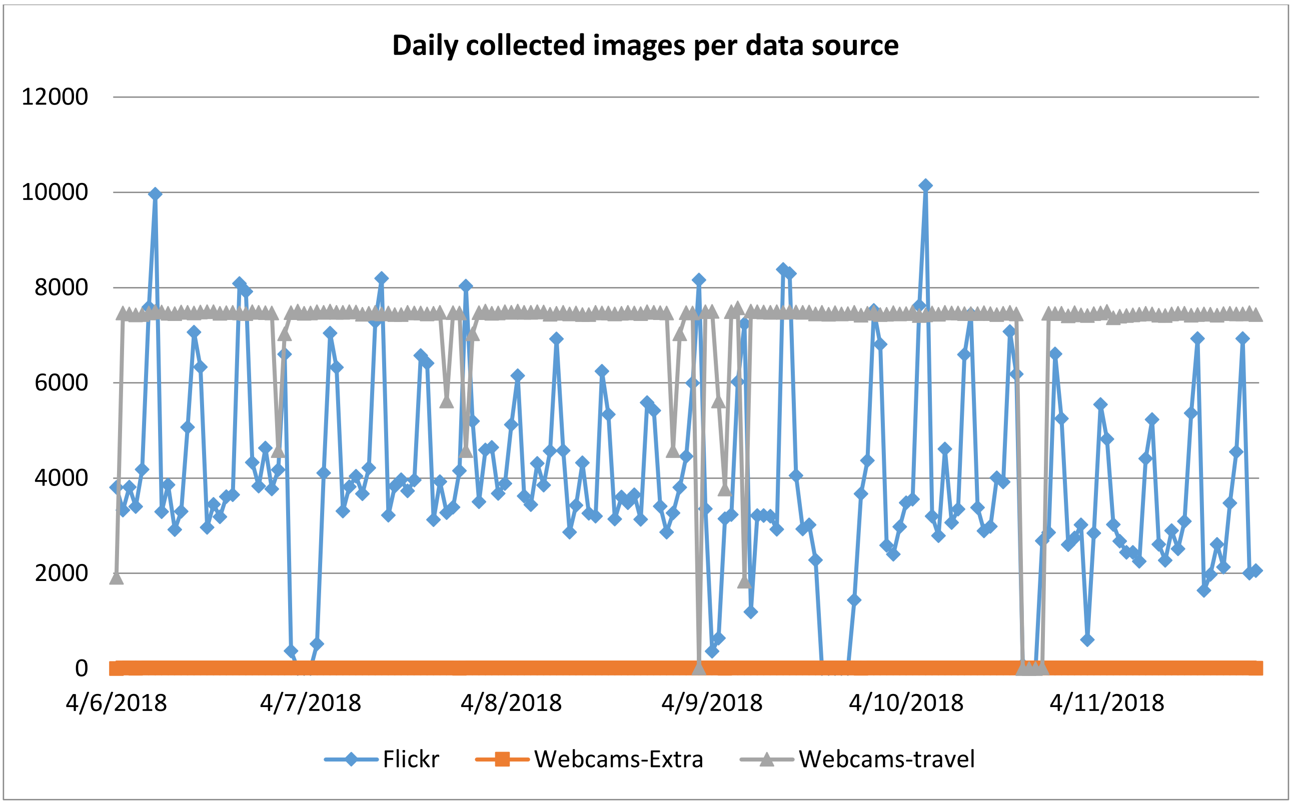


Figure 6. Daily collected images per data source

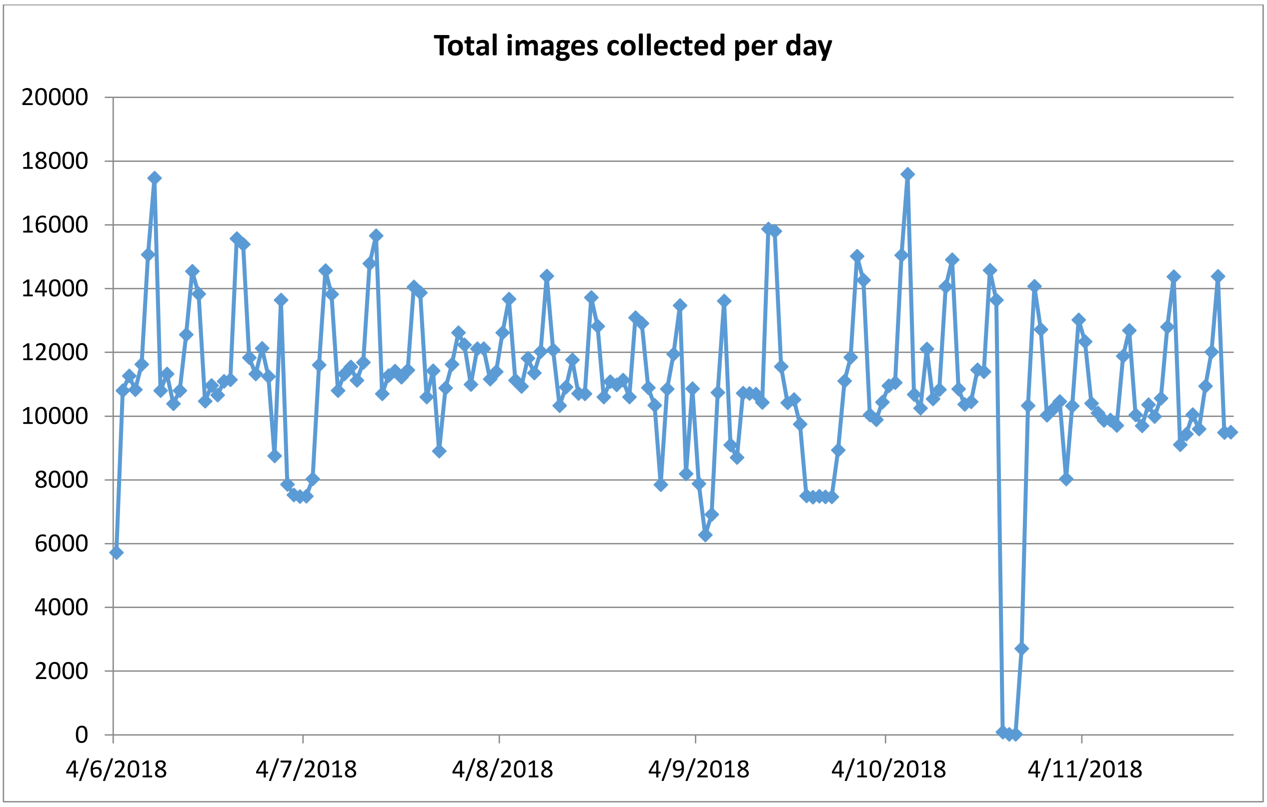


Figure 7. Total images collected per day

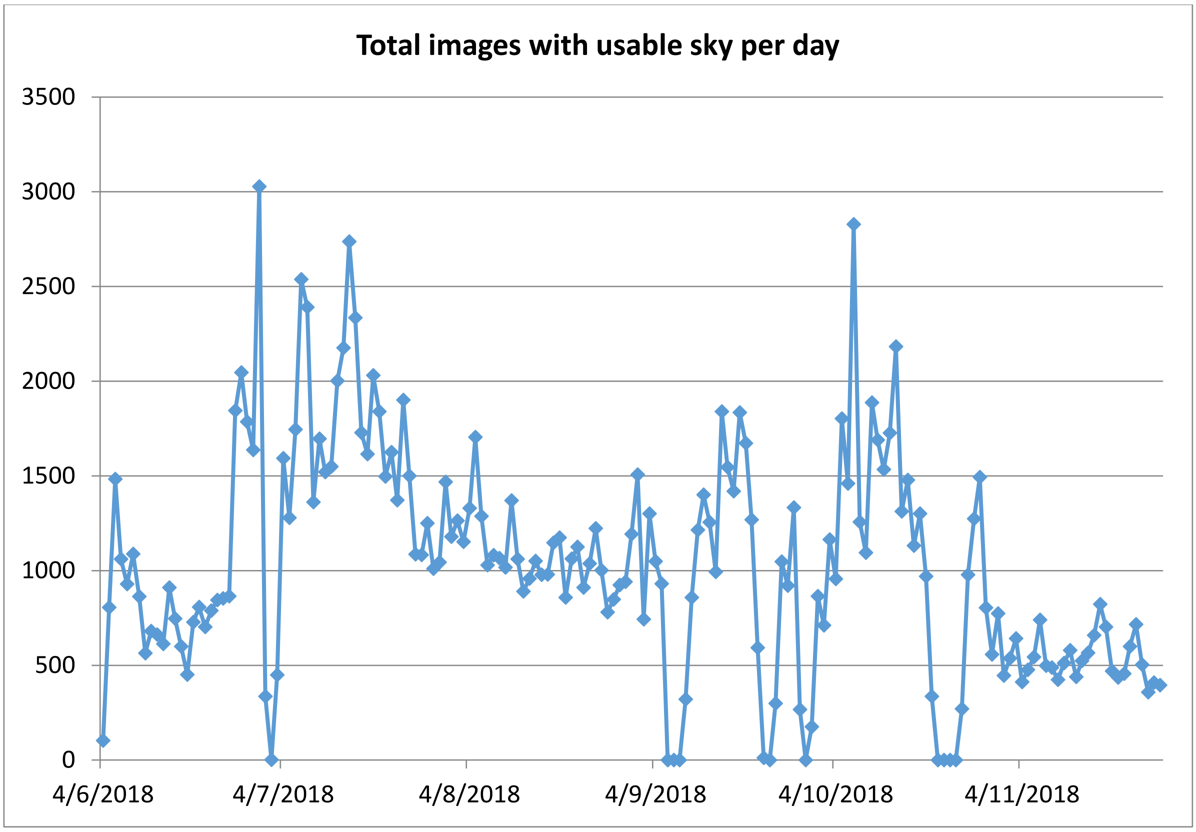


Figure 8. Total images with usable sky per day

2.2 Open hardware specifications

2.2.1 Improvements in Arduino Code to adopt humidity correction

Air borne particulate matter (PM) absorbs humidity from the air. Therefore, the measured values of PM sensors systematically increase when the relative humidity is high. The problem is more intensive when using low-cost sensors. Depending on both humidity and the particle composition, a certain amount of water condenses on the particulate matter. As a result, the particle's effective cross section increases leading to an increased likelihood of scattering of the laser in the detector. This can lead to the effect that particles which wouldn't have usually been detected during low humidity, might well be measured in high humidity conditions.

This effect cannot be modeled using a simple equation because it’s mainly determined by the composition of the particle which is usually unknown. All the efforts to find a successful equation to describe this effect empirically are under development. The common point to all efforts is that the phenomenon is not linear and is especially noticeable above 70% humidity. This can be easily understood from the fact that the data sheet of the SDS011 specifies the work environment for the sensor at max 70%.

In practice, this means that all measurements taken at a humidity level of more than 70% will contain a significant amount of uncertainty and will be characterized as “questionable validity”. Following previous studies, we attempt to partially minimize this effect with a correction formula by using a real time calculation of (humidity effect) readings.

The design team used empirical formulas found in the literature and concluded (after conducting tests that involve comparative measurements) that the best fit results are those reported by Zbyszek Kiliański & Piotr Paul (// https://github.com/piotrkpaul/esp8266-sds011). Thus, adopted the proposed formulas as follow:

```
 data.pm25 = data.pm25 / (1.0 + 0.48756 * pow((humidity / 100.0), 8.60068));
data.pm10 = data.pm10 / (1.0 + 0.81559 * pow((humidity / 100.0), 5.83411));
```

The versioning of this implementation (recorded in GitHub) is presented below:

- Add option for humidity compensation ,sakisds committed on Nov 25, 2017
- Change humidity normalization function to something more tested, sakisds committed on Nov 29, 2017
- Add `humidityCompensation()` into the hackAIR class ,sakisds committed on Mar 12 , 2018

If the humidity sensor fails do not use humidity compensation , sakisds committed on May 6 , 2018.

2.2.2 WeMoS implementation

It was experienced that some of the user were facing problems building the WiFi shield for the Arduino hackAIR node. Thus, WeMos D1 mini was selected to be the core module of hackAIR home sensor v2. Such a decision provides significant benefits for the hackAIR users and development team. Specifically,

- Includes WiFi support, no need for a shield
- One-step programming with the Arduino IDE
- Set API key during WiFi setup, no need to modify code for a basic node
- Smaller board, optional Humidity sensor shield (DHT11, DHT22 and AM2302 supported – users obviously can add any other that has Arduino library)

The steps to build and setup a WeMos based home sensor is fully described in: http://www.hackair.eu/hackair-home-v2/

2.2.3 New sensor & boards support, improvements to libraries

2.2.3.1 For PSoC

During the project the PSoC's company Cypress issue an End-of-Life (EOL) warning for the CY5671 module. The design team produced new .hex files for another common PSoC module, the CY5676. The .hex files were built for the two common sensors used with PSoC's:

1. SDA011
https://github.com/hackair-project/hackAIR-PSoC/blob/master/SDS011%20-%20Laser%20Dust%20Sensor/PSoC%20Firmware/PSoC%20Programmer%20Files/CY5676/SerialLaserSensor_SDS011.hex
2. SEN0177
https://github.com/hackair-project/hackAIR-PSoC/blob/master/SEN0177%20-%20Laser%20Dust%20Sensor/PSoC%20Firmware/PSoC%20Programmer%20Files/CY5676/SerialLaserSensor_SEN0177.hex)

Detailed GitHub activity reports are available at:

https://github.com/hackair-project/hackAIR-PSoC/commits/master

2.2.3.2 For WeMoS & Arduino

For WeMoS and Arduino implementation boards, several improvements were embedded in the hackAIR library. Specifically:

- Low resolution (DHT11) and high resolution (DHT22, AM2302) temperature sensors are fully supported in WeMos sketches
- Users can run directly from Arduino IDE the sketches for WeMos module
- Add delay after sensor wakeup (SDS011)
  - Laser diodes have an expected lifetime around 8000hrs. In order to avoid a short term termination, the sensors operate in non-continuous mode. More specifically, when sensing devices are not taking measurements, sensor falls into sleep mode. When the sensor wakes-up for the next measurement cycle it is set in a "stabilization" mode by adding a delay of several seconds in order to have the laser diode stabilize its output.

For all the above changes, detailed GitHub commit activities can be found at: https://github.com/hackair-project/hackAir-Arduino/commits/master and https://github.com/hackair-project/hackair-v2-advanced/commits/master.

2.2.4 Cardboard Sensor

The Cardboard sensor is an innovative scheme for the qualitative estimation of the atmospheric concentration of particulate matter with diameter equal or smaller of 10μm (PM10) by using Commercial Off the Shelf (COTS) materials. The proposed sensor incorporates well-established computer vision techniques to calculate the dimensions of the microparticles on the surface of the sensor probe, which is coated with petroleum products. It is desirable for the users of the proposed sensor to assess air quality using solely COTS products.

The sensor was tested for its ability to understand the lesions in the petroleum jelly layer before and after the exposure of the sensor probe for 24 hours to the ambient air. Subsequently, measurements of the proposed sensor were made in comparison with the commercially available Dylos DC1100 Pro air-particles concentration measurement system in rural and urban environments to investigate the sensor's ability to perceive the differences in particle concentration levels between different regions. The final part of the experimental process provided for a complete simulation of measurements in one area, compared with official measurements by the Greek Ministry of Environment and Energy, aiming at assessing the reliability of the proposed sensor, optimizing certain parameters of its construction and identifying any weaknesses which should be resolved before the sensor is given for use by the general public.

The sensor probe uses as a collecting surface for the air-particles, the aluminum-coated side of a Tetra Pak food package. which was decided to remain with its original silver colour. Once the food packaging has been cleaned and the user is sure that there are no food residuals or any other stains on the surface with the aluminum coating, a square piece of 5cm x 5cm is cut. This piece is the base of the sensor probe. Along a diagonal axis of the user's choice and close to the center of the piece, two small dots are drawn using a propelling pencil of a known diameter, in the case of this work the tip's diameter was 0.7mm. The two dots act as reference points, on the sensor probe, creating a plane of interest on which the user's mobile phone camera will focus. Subsequently, on the aluminium coating side of the sensor probe, a thin layer of petroleum jelly is applied using a butter knife. The petroleum jelly serves as the capturing substance for the air-particles, thus creating the particulate matter (PM) entrapment surface. At this point, the cardboard sensor probe is ready to be exposed to the ambient air for 24 hours in order to collect air-particles. Upon completion of the 24 hours exposure, the user retrieves the sensor probe and receives a set of five photos of the plane of interest on the PM entrapment surface.

The final step of using the cardboard sensor is to run the Particulate matter (PM) concentration estimation algorithm at the set of photos the user captured from the PM entrapment surface.

It is emphasized that in the present work the photographs from which the results were obtained and presented in the corresponding chapter, were taken under the same lighting conditions, using the Xiaomi Redmi Note 4X mobile phone which carries a Sony IMX258 camera sensor set to the highest resolution at 15MP, on which the SmartMicroOptics: Blips Lens magnifier has been adjusted at x10 magnification.

2.2.4.1 Particulate matter (PM) concentration estimation algorithm

The proposed algorithm for calculating the number of the trapped air particles in the petroleum jelly layer of the test surface is shown at Figure 9.

As an input image, the proposed algorithm receives the pictures that were captured by the user using their mobile phone from the PM entrapment surface of the cardboard sensor. These images are two-dimensional RBG and need to be converted into grey-scale format. Once the input image is received, the algorithm is converted by RBG to grey level. It is noted that at this implementation stage the algorithm receives a single RGB image each time as an input. The first action taken to the grayscale image is to increase its contrast, using the CLAHE algorithm (Pisano E.D. et al., 1998) which makes the hidden features of the image more visible, helping to distinguish the objects of interest from the area in which they are located. In the case of the hackAIR cardboard sensor, objects of interest are the suspended particles trapped within the petrol jelly layer.

The image is, then, binarised using the Otsu thresholding method (Otsu N. A., 1979), which can achieve good results when the histogram of the original image has two distinct regions, one belonging to the background and one belonging to the foreground or the signal. In our case, the air-particles, within the petroleum jelly layer, are considered to be foreground objects and the neutral coloured aluminium layer as the background. Binarising the input image is a necessary step as the following morphological algorithms only accept binary images as input vectors. The pre-processing of the input image is completed by using the region filling algorithm (Gonzalez R.C. and Woods R.E., 2002), which fills unwanted pixels of the background within the objects of interest, helping to completely separate foreground objects from the background.

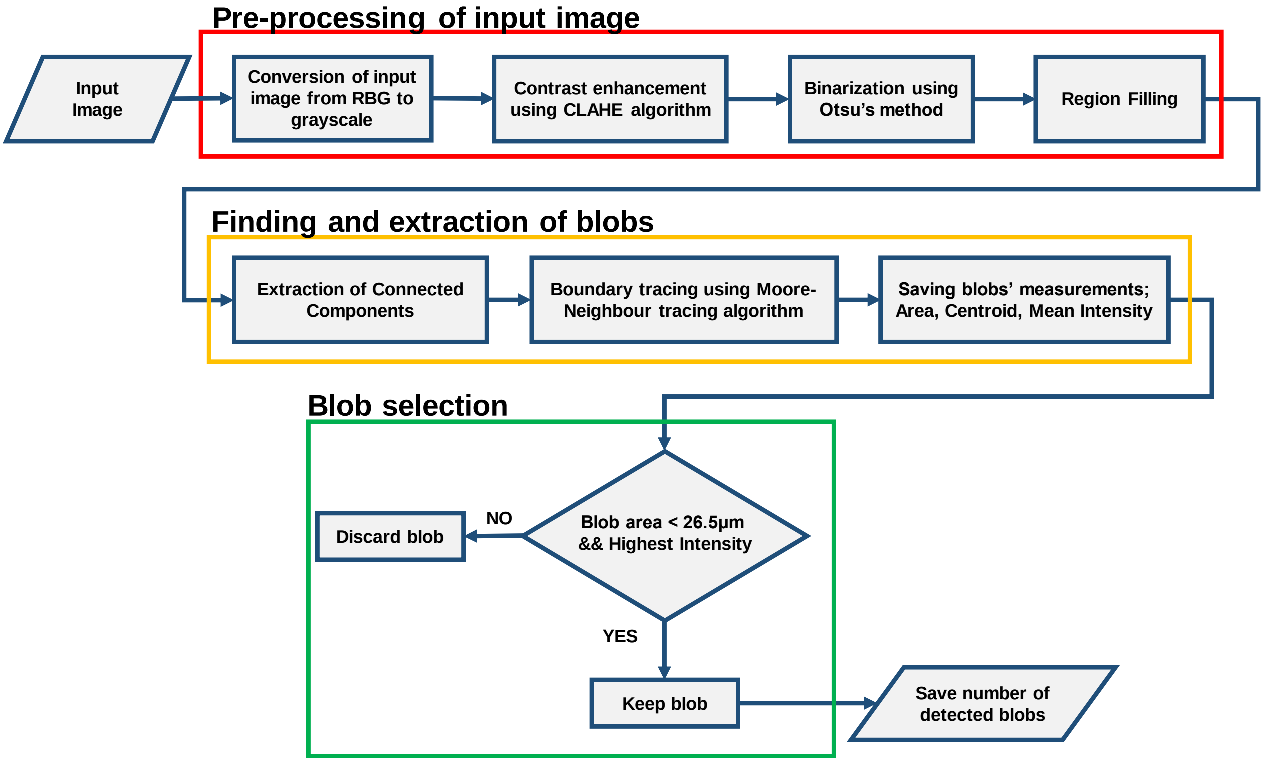

Pre-processing of input image
Input Image
Conversion of input image from RBG to grayscale
Contrast enhancement using CLAHE algorithm
Binarization using Otsu's method
Region Filling
Finding and extraction of blobs
Extraction of Connected Components
Boundary tracing using Moore-Neighbour tracing algorithm
Saving blobs' measurements; Area, Centroid, Mean Intensity
Blob selection
Blob area < 26.5μm && Highest Intensity
NO
Discard blob
YES
Keep blob
Save number of detected blobs

Figure 9: Proposed algorithm for the estimation of the PM concentration

At this point, the algorithm is required to identify and organize the individual blobs, taking into account the relationships of their pixels, for example, which neighbourhood are characterized by or and which are connected to each other. This can be accomplished via the connected component extraction algorithm (Foltz M.A, 1997) and the Moore Neighbourhood Algorithm (Ren M. et al., 2002). Thus, in each blob, a tag is given which contains stored values of some of its properties. In this case, in accordance with the application's objectives and equipment constraints, it was chosen to store the average intensity values and the total area occupied by each blob.

Entering the final stage of the proposed algorithm, two classification criteria are applied to determine the number of blobs whose change may coincide with the change in the concentration of particles with a radius greater than 10μm.

Firstly, they occupy a total area of more than 26.5μm and have the highest average intensity from their neighbouring pixels. The value of 26.5mm found through direct experimentation and is the smallest possible size of blobs, the algorithm can detect using SmartMicroOptics: Blips Lens magnification x10. The choice of the highest intensity blobs is aimed at separating the externally attached particles from their environment, which may include lesions due to uneven application of petroleum jelly, scratches on the surface of the specimen etc.

In order to better address the randomness and dispersion of the sample received by the cardboard sensor probe after the exposure to the ambient air, it was considered preferable instead of only obtaining one picture of the PM entrapment surface and detect the blobs only in that particular picture, to capture five pictures of the same the PM entrapment surface and detect in each of those individual pictures the total amount of blobs and to be stored in a separate vector.

As a result, through the above-mentioned procedure, a vector of five values is obtained, which correspond to the blobs detected in each image.

The air quality assessment of the area in which the measurement was performed is the average of the values of the blobs detected in each photo after the maximum and minimum detected values were removed from the vector. The averaging procedure is shown inFigure 10. Note that the more photos are captured from the same sample, the better the air quality estimate will be, since after a large number of photos the results start to converge.

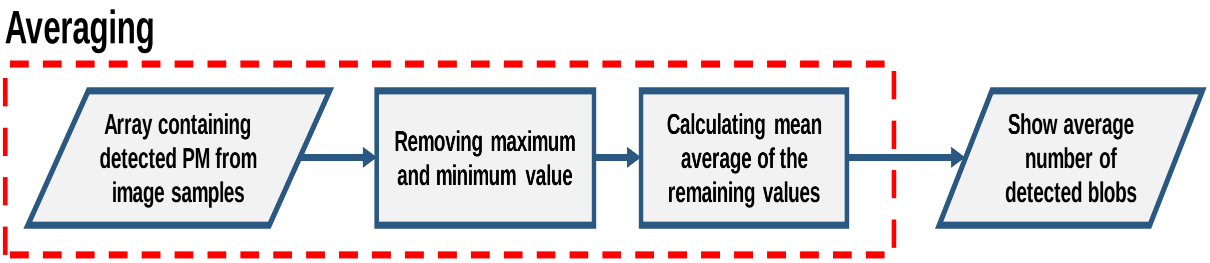

Averaging
Array containing detected PM from image samples
Removing maximum and minimum value
Calculating mean average of the remaining values
Show average number of detected blobs

Figure 10: Averaging algorithm for the PM concertation estimation

2.2.4.2 Comparative results

At this point, it is appropriate to discuss the credibility of the cardboard sensor in comparison with commercially available air-particles concentration measurement systems such as the Dylos DC1100 Pro and the official PM10 concertation measurements by the Ministry of the Environment and Energy.

As discussed further below, a first concern in constructing the cardboard sensor was to verify the ability of the PM concentration estimation algorithm to perceive alterations in the surface of the petroleum jelly layer that are occurring during the 24h exposure of the sensor probe to atmospheric air.

Moreover, the cardboard sensor had to be able to grasp the difference between measurements in areas where PM levels are considerably different, as in the case of measurements carried out in urban and rural environments. This fact is shown by the difference in the averages of the detected blobs of the two regions.

During the final measurements, the averaging algorithm was introduced resulting to the proposed exhibiting a response curve, similar to the corresponding response curve of the official measurements of the PM10 'Aristotle' station, which were used as reference measurements for the cardboard sensor measurements.

However, the proposed sensor measurements remain qualitive and for the user to assess the quality of the atmosphere in the area, the measurements are held, high number of data is required in depth of weeks, in order to obtain an estimate of the order of magnitude of the measurements that occur in the user's habitat and then attempt to estimate the PM concertation and how its levels are changing. The process of obtaining the final measurements is shown in Figure 11. The collection and replacement of the specimen was carried out every day at 2 pm so that it corresponds to the Ministry of Environment and Energy measurement cycle.

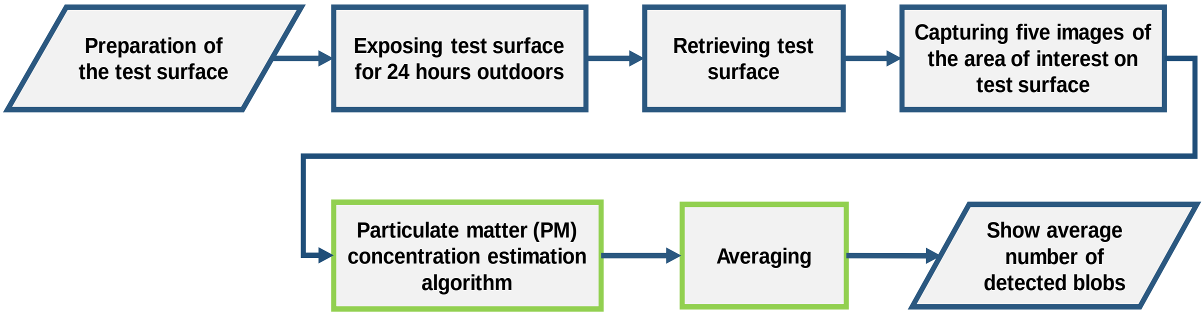

Preparation of the test surface
Exposing test surface for 24 hours outdoors
Retrieving test surface
Capturing five images of the area of interest on test surface
Particulate matter (PM) concentration estimation algorithm
Averaging
Show average number of detected blobs

Figure 11: Final workflow

2.2.4.3 Measurements before and after exposure of the specimen to the atmosphere

In the early stages of implementing the PM concentration estimation algorithm for the cardboard sensor, pictures of the PM entrapment surface of the same specimen were taken before and after exposing the specimen to the ambient air for 24 hours.

This series of measurements was designed to verify whether the PM concentration estimation algorithm can identify the lesions on the surface of the petrolatum layer caused by the deposition of pollutants at the particle collection surface upon exposure of the specimen in the atmosphere. The algorithm should be able to capture points and plan the contours of the blobs on the original photos.

In Figure 4, the detected blob's nuclei in the early stages of the PM concentration estimation algorithm appear in red. At this stage of implementation, the algorithm has already introduced the selection criteria for the size of blobs and the average intensity values of their pixels.

It was also attempted to illustrate the contours of the blobs overlaying upon the original image, which was not possible, due to the limitations imposed by the mobile phone's magnifying lens used, at that moment.

As a consequence, in the case of larger blobs the contour of the blobs appeared as an irregular mass perimetrically to the core, which, often because of its proximity to other adjacent blobs, covered smaller blobs. so that the individual blobs are not properly distinguished. While in the case of smaller blobs, the contour was not depicted at all, although the area covered by each blob was calculated which was significantly increasing the computational execution time of the PM concentration estimation algorithm. Thus, it was decided to omit the outline of the contour overlaying upon the original picture samples.

Figure 4 shows a change in the distribution of blobs before and after exposure of the specimen to the atmosphere for 24 hours. Pictures of the PM entrapment surface were captured using MoKo Universal Camera Lens x6 Macro Lens.

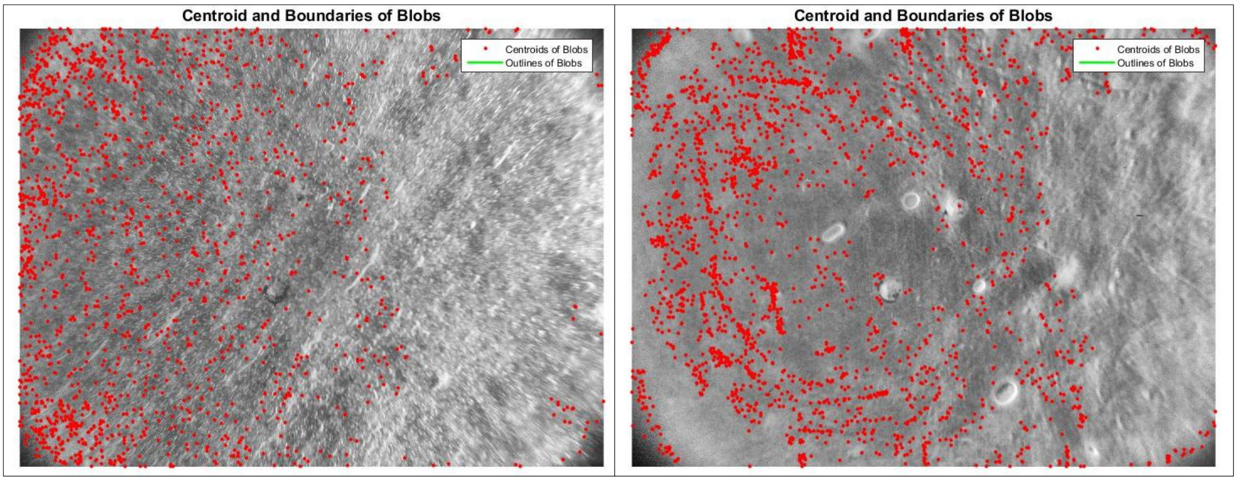


Figure 12: Pictures showing the alterations in the distribution of the detected blobs in the PM entrapment surface before and after the 24h exposure to the ambient air.

2.2.4.4 Measurements in urban and rural environments

Having ensured the ability of the agglomeration finding and extraction algorithm to respond to and detect lesions on the surface of the petroleum jelly layer, the next step was to carry out exploratory measurements.

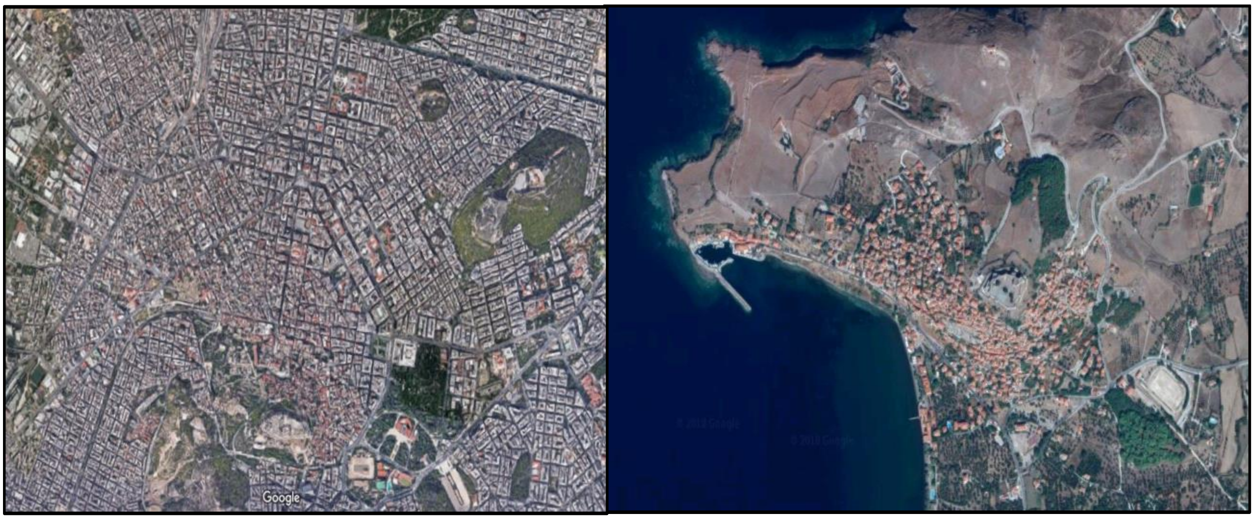


Figure 13: Residential relief of downtown Athens (left) and Mithymna (right)

The measurement procedure was similar to the one already presented above, apart from the averaging algorithm that was not introduced in this case, and instead the estimation was carried out taking only one image of the surface of the specimen, resulting in great uncertainty in the experiment, drastically reducing the reliability of the attempted qualitative assessment of the PM concertation.

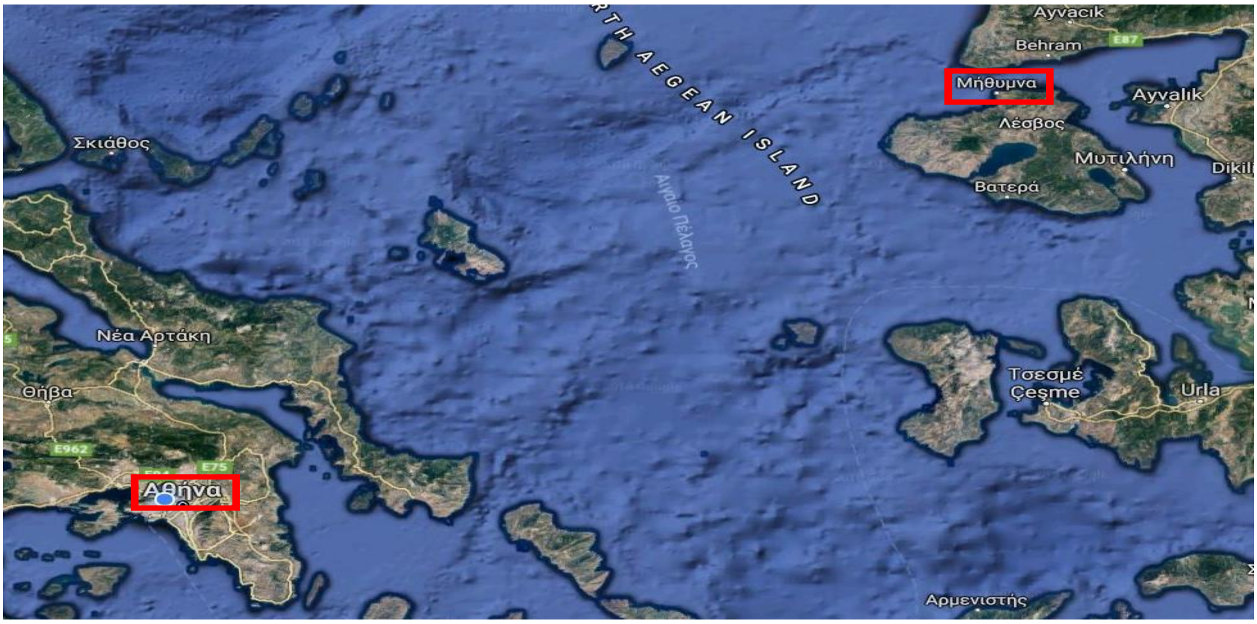


Figure 14: The geographic location of the two measuring areas

At an early stage of the measurements, it was observed a fluctuation of the detected blobs in the pictures from the PM entrapment surface of the specimen in accordance of the levels of pollution, so it was decided to carry out measurements in environments with very different levels of air pollution.

This experiment was designed to investigate the ability of the proposed cardboard sensor to detect - in order of magnitude - the different levels of air pollution from one area to another at a depth of 20 days during which the Dylos DC 1100 Pro sensor was put into a steady state, while a protective enclosure surrounding had been fitted to protect it from direct sunlight and weathering. Thus, the measurements presented below are divided into two sets.

The first set of measurements refers to data collected in Metaxourgeio, a typical neighbourhood of the historic centre of Athens from 14/9/2017 to 29/10/2017, with high levels of air pollution. At the opposite end, the second set of measurements took place in Mithymna from 4/7/2017 to 3/9/2017, an outlying village of the Aegean Sea in Lesvos with a low atmospheric charge but with high humidity due to the proximity of the settlement to sea. Figure 14 shows the geographic location of the two measuring areas and the relative distance between them.

Figure 15 shows the residential relief of the two areas where the measurements were carried out. One can observe the very different building conditions existing in the two measurement areas, with the strong urban character that governs the centre of Athens, with dense construction, intense and continuous flow of vehicles and a total lack of green spaces, factors that contribute decisively to the increase in the levels of particulate matter concentration and the overall atmospheric burden.

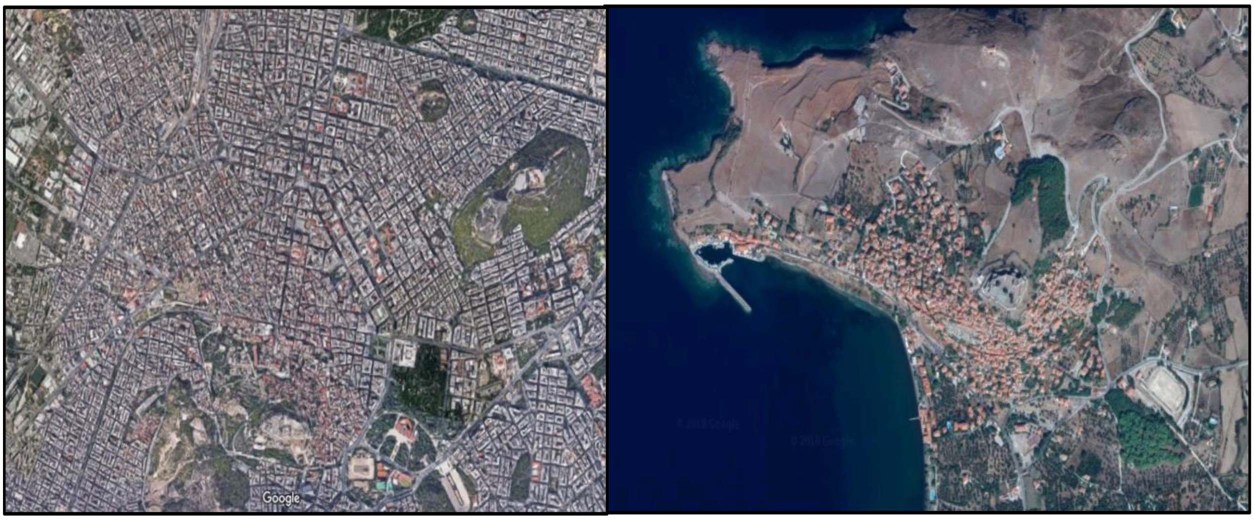


Figure 15: Residential relief of downtown Athens (left) and Mithymna (right)

Figure 16a shows the measurements of the cardboard sensor in Athens from 14/9/2017 to 29/10/2017 and in Mithymna from 24/7/2017 to 3/9/2017. Figure 16b shows the measurements of the Dylos DC1100 Pro system, for the same time period in both areas. According to Figure, that the proposed hackAIR board sensor fails to fully follow the changes in the PM concentration values detected by the Dylos DC1100 Pro system. A fact that was solved to some extent with the introduction of the averaging algorithm. However, the cardboard sensor manages to perceive the transition from one area with increased levels of PM, to an area with PM concentration. This is evidenced by the fact that almost all of the lower number of detected blobs were obtained from measurements made in Mithymna.

More specifically, in the test period from 24/7/2017 to 3/9/2017, the average of the daily detected obtained from measurements in Mithymna was 1894.95 days of detected blobs. Whereas, for the measurements made in Athens from 14/9/2017 to 29/10/2017, the average of daily detected blobs was calculated at 2891.3. A similar situation prevails in the PM10 concentration measurements of the commercial system Dylos DC1100 pro. In Athens, for the period from 14/9/2017 to 29/10/2017, the average PM10 concentrations were 34.75 μg/m³. While for Mithymna from 24/7/2017 to 3/9/2017 were at 16.6 μg/m³.

Taking into account the above, this measurement cycle may be judged to be partially successful. As the cardboard sensor is able to perceive and capture in its measurements the difference between regions with different PM10 concentration levels, however the sensor was not able to follow the daily changes in the concentration of air particles in the measuring ranges as recorded by the Dylos DC1100 Pro sensor

This phenomenon is attributed to the original choice of having only one image taken from the PM entrapment surface of the cardboard sensor specimen. This action led to great uncertainty in the experiment and drastically reduced the reliability of the attempted qualitative assessment of the inhalation air of the measuring ranges.

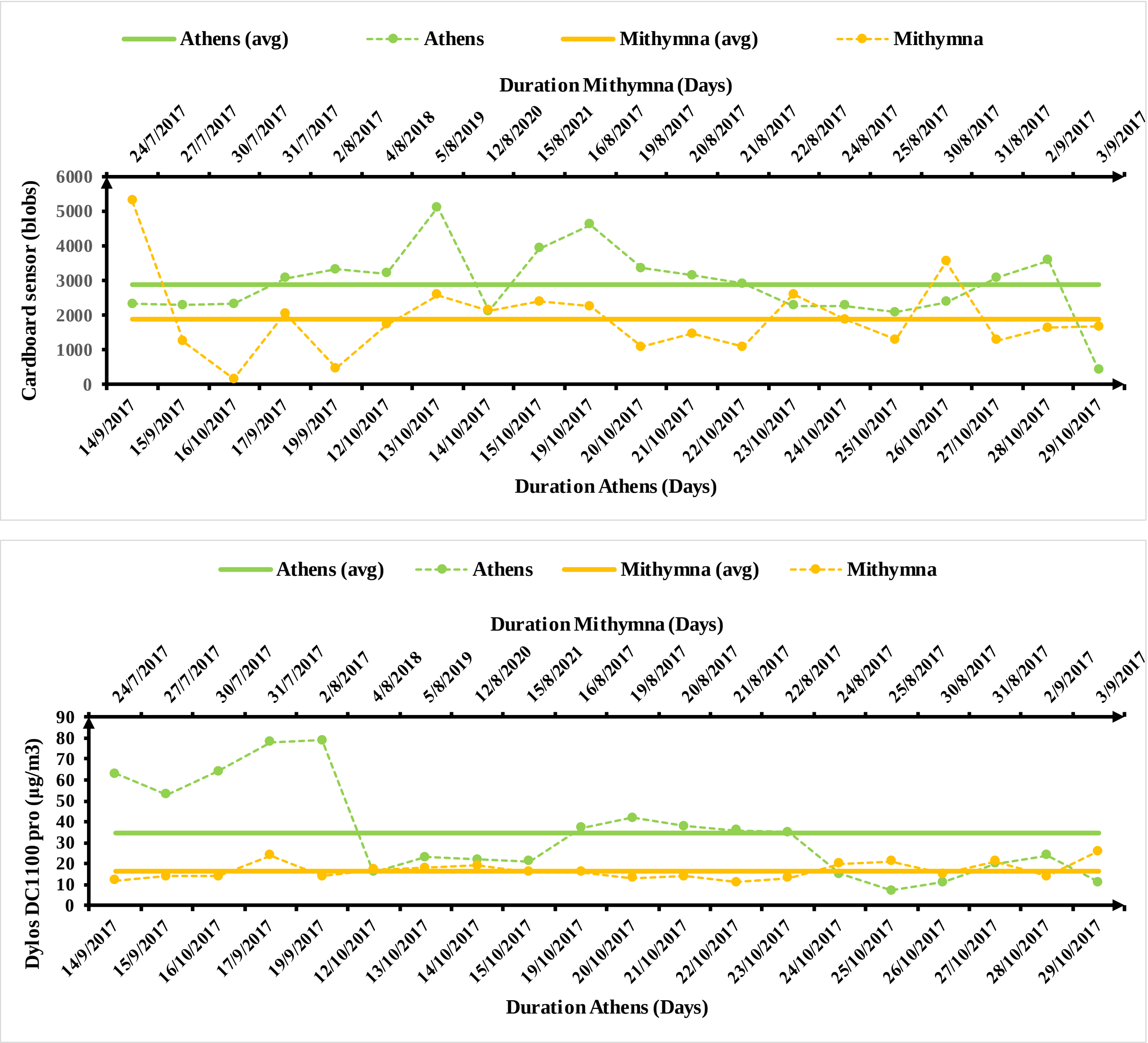


Figure 16: Graphic representation of measurements collected in rural and urban environments. a) Cardboard sensor, b) Dylos DC 1100 Pro trading data

2.2.4.5 Final measurements

The final evaluation stage of the cardboard sensor concerned the full simulation of data capture for the qualitative characterization of the air pollution of a measurement area by the average user. The primary objective of these measurements was to investigate the ability of the sensor to provide reliable daily qualitative estimates of PM10 concentration levels in the selected area, compared with official daily measurements by the Greek Ministry of Environment and Energy which would be used as reference measurements, for the final evaluation of the proposed sensor.

As a measuring area, Metaxourgeio was selected due to the intense urban environment that characterizes it with the high levels of particulate concentrations makes it ideal for conducting exploratory measurements for the performance of the cardboard sensor. Another advantage that contributed to the selection of Metaxourgeio as a measuring area was the familiarization with the area obtained from the second experimental procedure, which provided an earlier knowledge of the weather phenomena that would occur during the measurements. The station of the Ministry of Environment and Energy selected is located in Aristotle Street. This measurement station was selected because of its proximity to the measuring point of the cardboard sensor as the distance between them in straight line and through satellite imagery of Google Maps was estimated at 1.07km (Figure 17). In particular, if one takes into account the common features between the two mounting points, such as dense construction and heavy traffic, it can be assumed that the two points are common air pollution.

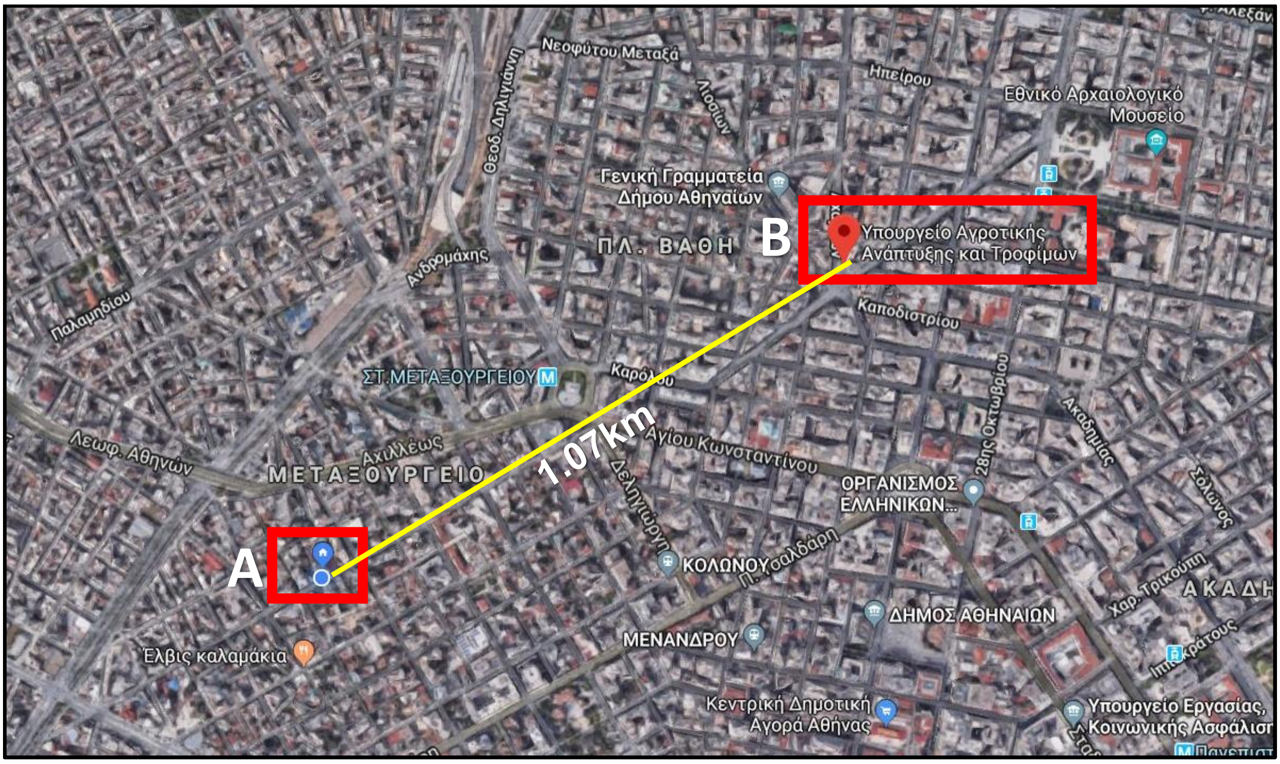


Figure 17: Distance between the two measuring points

The experimental procedure lasted a total of 50 days from 8/11/2017 to 25/6/2018. The retrieval and replacement of the specimen took place every day at 2 pm to match the Ministry of Environment and Energy measurement cycle. Figure 18 shows the measurements of the cardboard sensor in comparison to the corresponding measurements PM10 concentration measurement of the official "Aristotle" monitoring station in the Metaxourgeio area from 8/11/2017 to 25/6/2018.

This is particularly evident in days where PM10 concentration values exhibit a continuous upward or downward trend, which culminates with an extreme value. However, by looking more closely at the graphic representation, a series of conclusions emerge that best understand the proposed sensor performance. The proposed sensor has the ability to better monitor the reference value transitions on days around extreme PM10 concentrations, whether very high or very low. In other words, the proposed cardboard sensor can perceive the peaks of the reference measurement curve and the transitions that lead to them. Indicative examples of such transitions are the measurements in the periods from 14/12/2017 to 19/12/2017, from 19/12/2017 to 28/12/2017 and from 12/1/2018 to 7/2/2007, 2018.

Continuing, there are days when the proposed sensor partially fails to keep up with the changes in the daily PM10 particulate matter concentration recorded by the station "Aristotle". During these days, there is a continuous variation in the values of PM10 concentrations in μg/m³ that are not reflected by the corresponding changes in the number of detected blobs calculated by the cardboard sensor. This is evident during the days, from 8/11/2017 to 10/11/2017, from 12/12/2017 until 13/12/2017 and from 27/4/2018 to 2/5/2018.

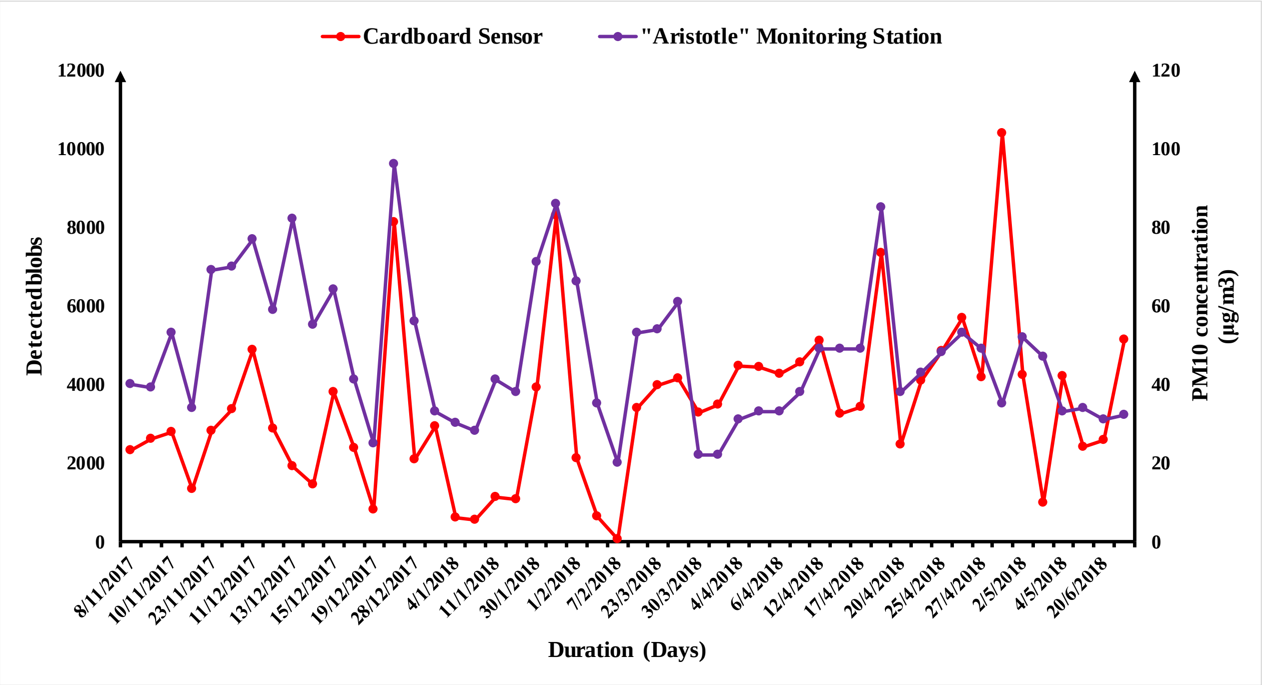


Figure 18: Comparative results of final measurements between the proposed Cardboard sensor and the PM10 the "Aristotle" station of the Ministry of the Environment and Energy

One of the main drawbacks of the proposed sensor in the current development phase is its failure to give natural meaning to its detected aggregates, thus correlating the number of detected blobs with the concentration of the suspended particles at μg/m³. This in Figure 18 is plotted as days intervals between the peaks of the curves, which means that large number of daily detected blobs is correlated with small particle concentration values and vice versa. A typical example of this is the comparison of the interval from 8/11/2017 to 14/12/2017 with the interval from 30/3/2018 to 12/4/2018.

As a result, the number of detected blobs is not sufficient on its own to the required qualitative characterization of the atmospheric charge is performed in an area when averaged concentrations of particulate matter prevail and the need for consistency with reference measurements such as in this case the "Aristotle" station in Athens.

In order to address this phenomenon, sensitivity should be increased either by purchasing new magnifiers or by introducing stricter criteria for the selection of blobs calculated by the PM concentration estimation algorithm. The non-calibration of the proposed carton sensor at its current stage of development was initially foreseen when designing its development and manufacturing schedule.

2.3 The hackAIR data fusion algorithm

One of the principal improvements made in WP4 as a results of pilot user feedback is related to the spatial scale at which the data fusion is carried out. The hackAIR project (Kosmidis et al., 2018) collects data on air quality from various sources, including open hardware sensors and online as well as user-provided images. This data, which is primarily available at the point-level, is then used to provide the interested public with targeted information on air quality in the ambient environment at specific locations of interest as well as personalized recommendations for outdoor activities given the current conditions. In order to do this, it is necessary to be able to provide information on air quality at any location within the study domain, even if no measurements have been made there. This can be accomplished by combining the various sources of information with each other as well as with additional output from an operational air quality model. Data assimilation (Lahoz and Schneider, 2014) and data fusion as its subset provide an efficient and mathematically objective way to carry this out and have been used within the hackAIR project to provide country-scale maps of air quality based on the collected hackAIR data. These maps were produced at a spatial resolution of 5 km (this resolution was primarily determined by the used model information), which is sufficient to provide average air quality information at the city-level and is thus useful for providing personalized recommendations but is not capable of providing air quality information at the street scale or individual neighbourhoods. During the course of the project user feedback indicated that also the latter information would be very valuable. While it was not possible to carry out local-scale data fusion for the entire area of interest of the hackAIR project, as a proof-of-concept WP4 carried out some initial experiments of using the information collected by the hackAIR open hardware sensors for urban-scale data fusion in the city of Oslo, Norway, where substantial physical modelling information is available.

A methodology was developed for combining observations from a network of low-cost air quality monitoring devices (hackAIR open hardware sensors) at fixed locations together with data from official air quality monitoring stations equipped with reference instrumentation with long-term average information from a high-resolution urban-scale air quality model. The result of the data fusion process, which is based on geostatistical techniques, is a new value-added map representing the best-guess concentration field at the time at which the observations were made. This concentration field inherits properties from both input datasets, i.e. it inherits the overall spatial patterns shown by the time-invariant modeled long-term average concentration field and at the same it inherits the absolute values provided by the instruments deployed within the sensor network. Figure 19 shows an example of how much spatial detail the data fusion system is capable of providing at the local scale.

The method opens up a wide variety of applications related to personalized air quality and exposure estimates for individual persons, based on their location and traveling patterns. It further allows for personalized activity recommendations as carried out within the hackAIR project at the country level but at much finer spatial granularity. A variety of issues currently limit the method to experimental rather than operational use, including but not limited to the generally low number of deployed sensors within a small urban area, and the requirement for an accurate urban-scale air quality model that can be used to provide long-term average information of the typical spatial patterns of air pollution in a city. Nonetheless, such datasets are becoming more and more available and networks of low-cost microsensors for air quality are expected to become significantly denser over the course of the next few years, thus increasing the value of this methodology.

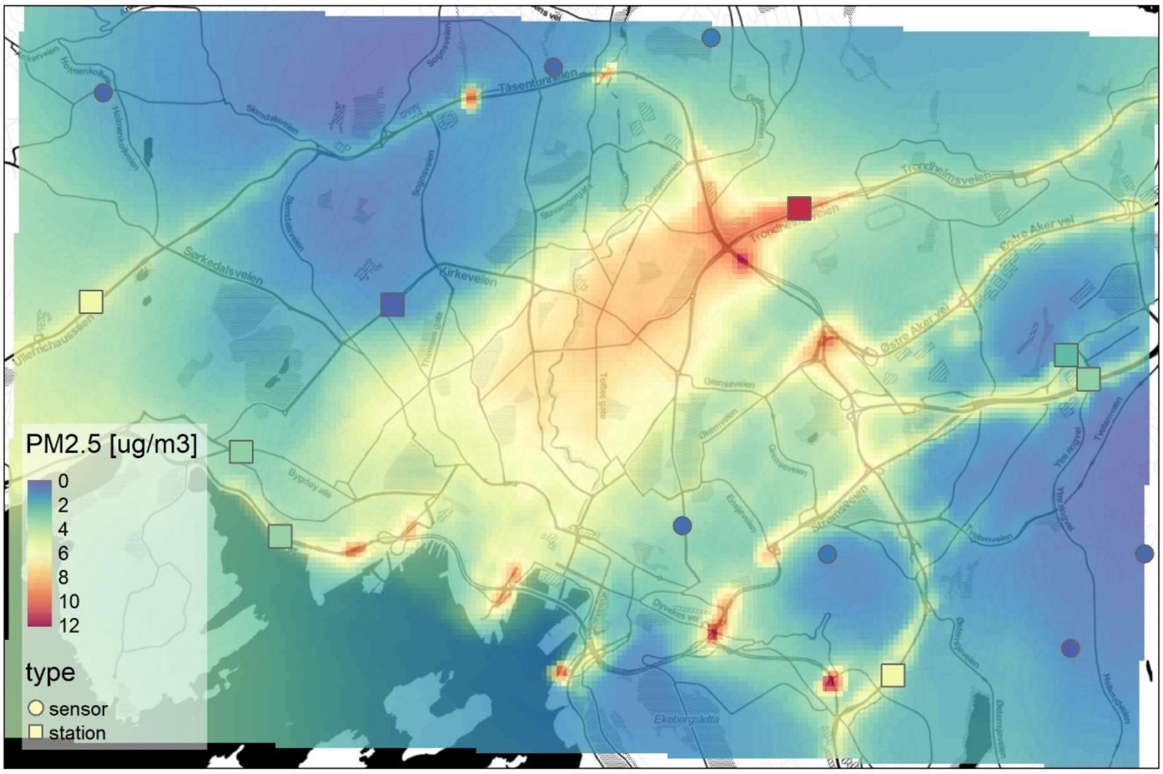


Figure 19: Example of urban-scale data fusion for PM2.5 as carried out for Oslo Norway, using the hackAIR sensor network as well as monitoring stations, shown here for 23 August 2018 at 14:00 UTC.

2.4 Semantic integration and reasoning of environmental and user-specific data

In this section, we summarise all the technological updates, enrichments and extensions that have been implemented for supporting additional features and functionalities of the two main hackAIR components responsible for the semantic integration and reasoning of environmental and user-specific data, i.e.: (i) the hackAIR knowledge base (KB) and reasoning framework, and (ii) the hackAIR decision support (DS) service for personalised recommendation provision. Such updates were made on the basis of the following:

- through the continuous integration of the hackAIR KB and the DS module with the hackAIR user interface (UI)[9] and on the basis of a constant internal evaluation of their efficiency as the system evolved; and
- the pilots' evaluation-results and users' feedback.

More details are given in the following subsections.

2.4.1 The hackAIR KB Framework

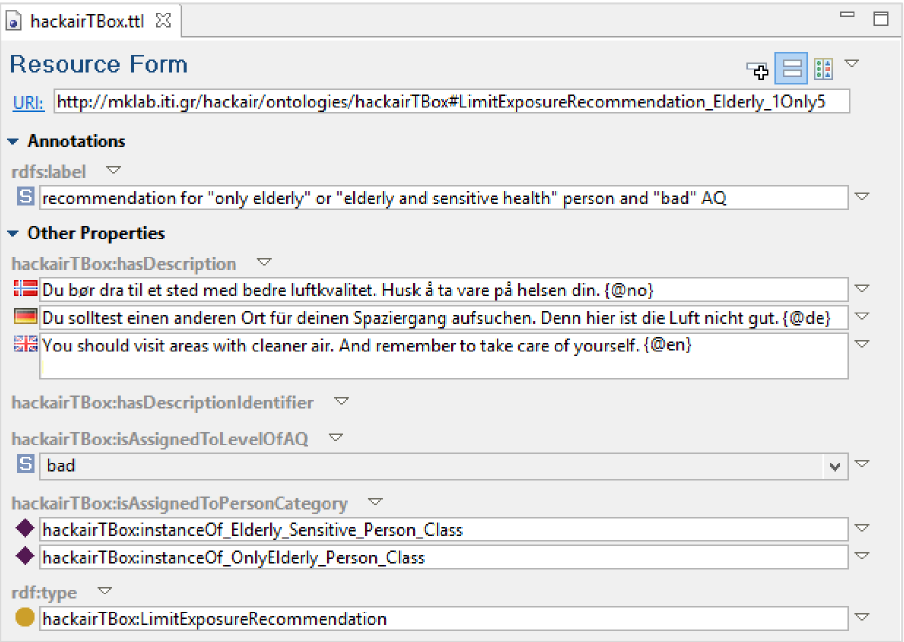
Since the official release of hackAIR D4.2 (hackAIR 2017, D4.2), several updates have taken place in the hackAIR KB and the reasoning mechanism. The first component involves the ontology-based module which poses the main backbone of the knowledge representation of the domain of interest, while the latter is the inference module operating behind the provision of two different types of recommendations defined within the context of the project: (i) the tips of the day,

[9] hackAIR UI refers to both the hackAIR mobile application and the hackAIR web platform.

i.e. general advice on alternative ways of living for keeping air pollution exposure/production at a minimum level, and (ii) the personalised recommendations, i.e. activity-related suggestions with respect to the user profile characteristics/needs and the estimated existing air quality conditions.

Initially, through a thorough collaboration among all project partners, the content (actual messages) of tips and personalised recommendation were revised in order to avoid any potential negative impact of the proposed activity-related advice to the people, especially in cases of severe air quality conditions. The rephrased messages ought to have the following characteristics: (i) be more-informative and less order-like, and (ii) carry a sense of encouragement for alternative outdoor activities, that limit the exposure to poor outdoor air quality, ensuring at the same time that no restriction in performing activities is promoted to individuals/users of the hackAIR system. An example message that could potentially cause decrease of physical activity is: "You should reduce prolonged outdoor activity", while a message that promotes alternative action is: "You had better go for a walk in an area with cleaner air. Check the hackAIR map to discover clean areas!".

For making the multi-lingual support of end-users feasible, all defined English tips and recommendations were translated in two additional languages: Norwegian and German. By taking advantage of the feasibility and adaptability of the implemented ontology-based representation, the translated messages were efficiently and easily stored in the latest version of the hackAIR ontology, as simple text asserted in the hackairTBox:hasDescription property of each relevant instance of type hackairTBox:Recommendation, with different language annotations, as it can be seen in Figure 20. As the multi-lingual definitions had also to be supported by the reasoning mechanism, in order to provide results (recommendations) on the basis of user's preferred language, the relevant SPIN rules were enriched accordingly.



hackairTBox.ttl

Resource Form

URI: http://mklab.iti.gr/hackair/ontologies/hackairTBox#LimitExposureRecommendation_Elderly_1Only5

Annotations

rdfs:label

recommendation for "only elderly" or "elderly and sensitive health" person and "bad" AQ

Other Properties

hackairTBox:hasDescription

Du bør dra til et sted med bedre luftkvalitet. Husk å ta vare på helsen din. {@no}

Du solltest einen anderen Ort für deinen Spaziergang aufsuchen. Denn hier ist die Luft nicht gut. {@de}

You should visit areas with cleaner air. And remember to take care of yourself. {@en}

hackairTBox:hasDescriptionIdentifier

hackairTBox:isAssignedToLevelOfAQ

bad

hackairTBox:isAssignedToPersonCategory

hackairTBox:instanceOf_Elderly_Sensitive_Person_Class

hackairTBox:instanceOf_OnlyElderly_Person_Class

rdf:type

hackairTBox:LimitExposureRecommendation

*Figure 20: Example instance of **hackairTBox:Recommendation** type with declaration of its message in three different languages (screenshot taken from TopBraid Composer[10])*

Another issue that was handled was the need to manipulate different environmental measurements within the context of the recommendation mechanism in a unified/universal way. In the initial version of the hackAIR KB, the general concept of `hackairTBox:EnvironmentalData` was included for covering the representation of any type of environmental data, while the `hackairTBox:AODEnvironmentalData` was its only subclass for representing the aerosol optical depth (AOD) numerical measurements. Relevant rules were handling the conversion of numerical into

[10] A visual modelling environment for creating and managing ontologies. Available at: https://www.topquadrant.com/tools/IDE-topbraid-composer-maestro-edition/

nominal values (*bad*, *medium*, *good*, *very good*), i.e. the four air quality (AQ) levels defined within the hackAIR context. In the latest version of the hackAIR KB, three additional environmental data types (PM_{10}, $PM_{2.5}$, hAQI[11]) were added to the ontology without affecting the main schema but only enriching its semantic content, while rules were adjusted correspondingly in order to convert the additional numerical definition (specific values) into general AQ levels (nominal values) on the basis of different range definitions. It becomes evident that the proposed ontology-based implementation can be easily extended to any other types of environmental data, given the fact that the model and the overall rule-based recommendation mechanism is relied on a general, post-calculated AQ level, regardless of the type of the environmental measurement which was converted into a nominal value, for providing relevant recommendations to the user with respect to the existing AQ.

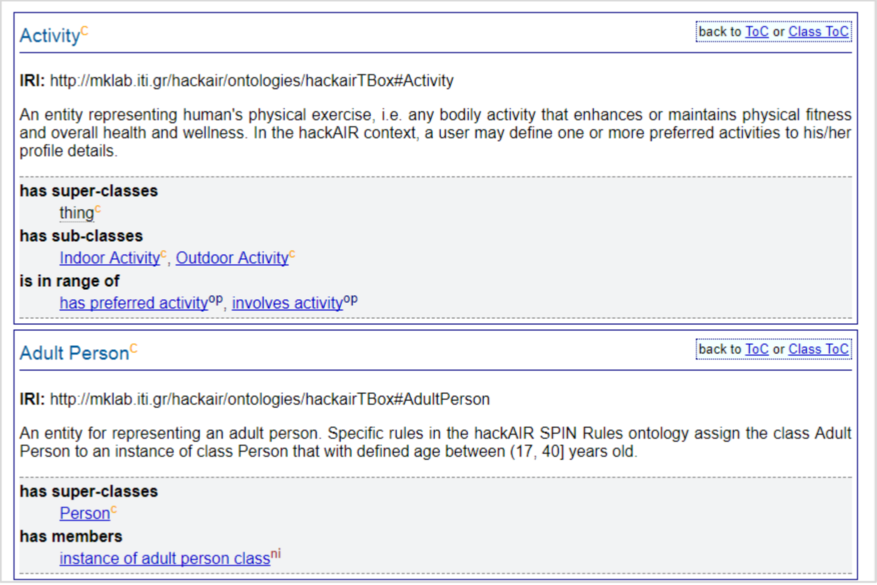
While focusing on the intermediate pilots' evaluation results related to the hackAIR recommendation module (hackAIR 2018, D7.4), overall it was acknowledged that the perceived usefulness of the personalised recommendations is well received by the respondents; users mostly think that the personalised recommendations are efficiently reflecting their profile details, are informative and are describing their information needs. From the implementation perspective, these results can be positively interpreted, in a way that the different user profiles provided by the hackAIR KB were efficiently defined/selected throughout the modelling process. However, a suggestion for improvement was reported the project reviewers, stating that: *"The recommendations (actual messages) seem most of the time the same"*. This issue was *easily* but *merely* handled: (a) easily, since the implemented ontology-based solution supports the adaption and extension of the KB and the recommendation model with different messages per use case, and deliver it to the user. Some indicative messages were added per case, and SPIN rules were enriched so as to randomly select one message from a "bag-of-recommendations" defined per case, i.e. for example for *elderly users* when *AQ is bad*, increasing this way the usefulness of the service and reducing the repeatability of the same messages appearing to the user; (b) merely, because it requires a big effort to define many new recommendation messages for each single use case. Thus, the extensive integration of additional user categories, activities or messages per case can be considered as future work, given the fact that the existing implementation already supports such actions.

Finally, and for enabling the dissemination of the proposed ontology-based solution and its adoption from third-parties, we proceeded with the following actions: (a) we created a detailed HTML-formatted documentation of the final version of the hackAIR ontology, with the use of LODE service[12] (Figure 21). The latter requires the assertion of specific properties (`rdfs:comment`, `rdfs:label`, `dc:description`, etc.) in our defined classes, relations and rules; and (b) we additionally created a public repository that hosts the source files of the final hackAIR ontology. The aforementioned dissemination material can be found in: http://mklab.iti.gr/project/hackair-ontologies. The aforementioned material is distributed under the Apache License (v2.0)[13].

[11] The *hAQI* (hackAIR AQI) corresponds to the numerical concentrations calculated from the hackAIR Data Fusion module.

[12] LODE (Live OWL Documentation Environment) is available at: http://www.essepuntato.it/lode

[13] https://www.apache.org/licenses/LICENSE-2.0



Activity^{c} back to ToC or Class ToC

IRI: http://mklab.iti.gr/hackair/ontologies/hackairTBox#Activity

An entity representing human's physical exercise, i.e. any bodily activity that enhances or maintains physical fitness and overall health and wellness. In the hackAIR context, a user may define one or more preferred activities to his/her profile details.

has super-classes
thing^{c}
has sub-classes
Indoor Activity^{c}, Outdoor Activity^{c}
is in range of
has preferred activity^{op}, involves activity^{op}

Adult Person^{c} back to ToC or Class ToC

IRI: http://mklab.iti.gr/hackair/ontologies/hackairTBox#AdultPerson

An entity for representing an adult person. Specific rules in the hackAIR SPIN Rules ontology assign the class Adult Person to an instance of class Person that with defined age between (17, 40] years old.

has super-classes
Person^{c}
has members
instance of adult person class^{ni}

Figure 21: An excerpt of the HTML-based documentation of the hackAIR ontology

2.4.2 The hackAIR DS Service

The main contribution conducted after the completion of the ontology-based hackAIR KB and reasoning mechanism reported in D4.2, was the implementation of a RESTful web-service responsible for the integration and communication between the hackAIR UI and the recommendation framework. More specifically, the functionality of the developed application programming interface (API) can be described with the following step-by-step process:

1. The service receives a JSON-formatted request for decision support through an HTTP POST request; the JSON object contains all the necessary information for decision support, i.e.: the user profile details (*age*, *location*, *health sensitivities*) and preferences (*preferred outdoor activities*) relayed from the hackAIR UI, as well as the *existing AQ conditions* derived from the Data Fusion module.
2. The available data are automatically populated in the hackAIR KB, on the basis of the already defined problem description language (PDL) (hackAIR 2017, D4.2).
3. The rule-based recommendation mechanism is triggered. New data and relations are automatically inferred.
4. The derived personalised recommendations are returned in JSON format for proper visualisation to the user through the hackAIR UI.

The web-service receives POST requests under the following URI:

```
<http://{BASE_URL}/hackAIR_project/api/requestRecommendation>
```

It is distributed under Apache License v2.0 and its documentation (examples of use, installation instructions, dependencies, etc.) together with the source files of the API are publicly available on the following GitHub repositories: https://github.com/MKLab-ITI/hackair-decision-support-api and https://github.com/hackair-project/recommendations-decision-support-module.

When initially developed, both the static and dynamic parts of the recommendation system (RS) were hosted on a local Glassfish server. The migration of the RS to a remote, public server, dedicated for the hackAIR project, was proved to be crucial for its efficient and stable operation. DRAXIS gave access to CERTH on a Docker installation (i.e. a stand-alone container image), hosted on a DRAXIS server. Proper installations (setting up a Glassfish server, scheduling cron jobs, etc.) and parameterization issues were handled successfully by CERTH. Moreover, several upgrades were made to the container (setting up an Apache server, define htaccess, etc.), in order to efficiently serve the hackAIR ontologies from the same container. The latter action achieved reduced response times due to the fact that the static and the dynamic parts of the RS communicate from the same source without any intermediates.

Throughout the integration process of the recommendation service to the hackAIR UI and as the overall system evolved, several additions had to be examined and implemented. We enriched the DS service to handle the multi-lingual aspect of the platform as well as the different environmental definitions (PM_{10}, $PM_{2.5}$, PM_AOD and hAQI); requests can now be handled for any of the supported languages and any of the aforementioned AQ values.

As the number of actual users increased, the need was to ensure the stability and efficiency of the recommendation service in terms of performance, i.e. to be able to handle multiple requests simultaneously, within reasonable response times. We overcame those high-risk blocks of code by ordering the requests that arrive in the system on a first-come first served basis. Even though we achieved a thread-safe service, inevitably the response time increases correspondingly. Considering an example where two requests, R_1 and R_2 arrive at different times, i.e. the API is available to serve their requests, and R_1 takes t_1 sec to be served, while request R_2 takes t_2. If those same requests arrived almost simultaneously (R_2 after R_1) then R_1 will still need t_1 sec to be served, while R_2 will need t_1+t_2 sec, since R_2 waits to be handled until R_1 is completely served. In practice, the additional response times are acceptable – every single request gets 1.5-3.5 sec to be served, something that is significantly smaller than the KPI (Key Performance Indicator) defined within the hackAIR project.

Finally, for dissemination purposes, two relevant papers have been published: (a) a conference paper (Riga et al., 2018) describing the overall work carried out in WP4 – T4.2, focusing on the architecture, functionality and advantages of the implemented ontology-, rule-based recommendation module, and (b) a journal paper (Kosmidis et al., 2018) describing the overall technical work (scope, implementation and communication of different hackAIR modules) that has been carried out within the context of the hackAIR project.

2.4.3 Future perspectives

The potential of delivering alternative routes or different content on the basis of the user's existing AQ exposure, defined position or device cannot be applicable on the basis of the existing plan of the project; the available granularity of hackAIR spatiotemporal data cannot efficiently cover the aforementioned scenarios. However, if such data were available in future extensions of the system, an ontology-based solution for path planning, i.e. delivering spatiotemporarily targeted content to the user, could be potentially examined. A number of relevant promising works existing in literature (Belouaer et al., 2010; Codescu et al., 2012; Provine et al., 2004) can strengthen and support such an implementation.

2.5 Updated engagement strategy

The engagement strategy of hackAIR was developed in D6.1. "Engagement strategy for hackAIR community development".

The engagement strategy included a set of online and offline tactics that the hackAIR pilots could implement and tailor towards their specific context. Each tactic included a general description, envisaged target group and expected outcome. The selected tactics were based on empirical evidence collected from literature, and through practical experience from pre-existing initiatives that dealt with citizen engagement for air quality monitoring. The conceptual framework of the engagement strategy was developed upon the 7E-model of Bambust (Bambust, 2015), which comprises a set of 'leverage points' (that all start with letter E) to reach optimal engagement and eventual behavior change by impacting user motivations and different levels of interaction:

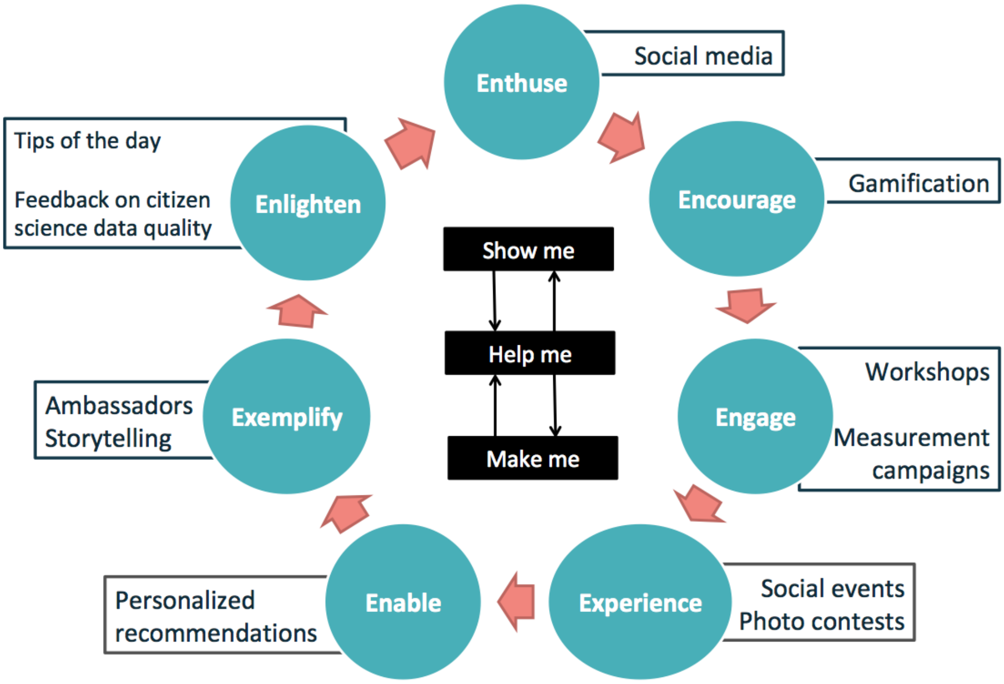


Figure 22: Engagement strategy hackAIR (More information: D6.1)

A preliminary engagement strategy was designed in M20 and it consisted an initial plan for the pilot operations. Each pilot partner evaluated whether the proposed tactics cover their needs and decided to fully or partially implement them or reject them.

This section describes the implementation status of these different tactics.

2.5.1 Online engagement tactics

Tactic	Description	Aim	Status
Tips of the day	When users access the hackAIR mobile application or the hackAIR web app, a daily tip is provided to them about how they can contribute towards a cleaner air.	Enlighten & encourage: better inform citizens about guidelines on how to reduce the emission of pollutants in the atmosphere, accompanied with actions which can be taken in order to avoid hazardous effects to their health.	Implemented. The tips of the day are displayed in the hackAIR platform. Some modifications were done to improve the visibility of it, as the initial usability and user satisfaction survey showed that users hardly noticed the "tips of the day" on the screen.
Personalized recommendations	In the general settings, a user of hackAIR can, when setting up his/her account, define to be interested in information related to 1) respiratory/ cardiac problems 2) pregnancy 3) outdoor sports 4) toddlers, ... The user	Enlighten & encourage: inform citizens of specific risks related to their profile.	Implemented. The personalized recommendations are displayed in the hackAIR platform. Some modifications to the personal settings were done to align with privacy regulations.

	receives personalized recommendations based on his/her interests.		
Gamification module	A user can be involved in the gamification components of hackAIR: points can be collected and badges can be collected throughout the different phases of usage of the hackAIR application.	Enable & encourage: the gamification module targets the user's curiosity and extrinsic motivations of collecting points and badges	Implemented. Citizens are mostly involved in air quality monitoring out of intrinsic motivations, the gamification module was therefore less popular among the community. The gamification module did not lead towards an increased usage of the platform, neither towards behavior changes.
Feedback on citizen science data quality	A citizen who uploads data to the platform (environmental data gathered via DIY sensors or uploaded pictures) gets informed on the quality of the data s/he provides and the value of the data for optimising the data fusion map.	Enlighten & encourage: give feedback to citizen's participation in the project and possibility to start a conversation about the potential of an "ask an expert" support function.	Implemented. A user can check whether or not the uploaded picture is taking into account in the data fusion map (through a check mark).
Active and responsive social media presence	The social media accounts of hackAIR should tweet or post on a frequent basis about the outcomes of the pilot implementation periods. Furthermore, the tweets and posts should go hand in hand with the other types of engagement tactics, e.g. when organising a workshop, this can be posted on this account. The storytelling (see next tactic) can provide a lot of content for social media presence.	Enthuse & experience	Implemented. hackAIR was very active on social media (Facebook, Twitter and Instagram)
Story telling	The usage of narratives/ user testimonials as part of the hackAIR	Enlighten & exemplify	Implemented. Several user stories were published on the hackAIR project

	communication strategy gives substance to differing degrees of participation across pilot locations. By tracking and making the very stories and experiences of citizens involved in hackAIR visible, the goal is to focus on the human aspects of participatory sensing.		website, and on hackAIR social media. http://www.hackair.eu/news/
Updated list of events	The hackAIR website integrates the schedules of hackAIR workshops at all pilot locations, so that hackAIR's offline activities are as accessible as possible for citizens and interested parties.	Enlighten	Implemented. A list of events was published on the project website and in the newsletters of hackAIR.

2.5.2 Offline engagements tactics

Tactic	Description	Aim	Status
Workshops	A very successful engagement tactic was the organisation of workshops across both pilots. The goal of these activities was to reach a community of interested citizens to build awareness about the project, its benefits and results, to get familiar with hackAIR, and to start using the different tools. The workshop toolkit developed by ON:SUB was used for organising the workshops.	Experience, enlighten & engage	Implemented. This was a very successful engagement tactic, with 562 workshop participants during 21 organised workshops in the pilot countries. More evaluation results about the workshops can be found in D7.7.
Measurements campaigns	A measurement campaign can be a one-day activity or a longer time period in which citizens are asked to upload as many contributions as possible	Experience, enlighten & engage	Not implemented. Pilots investigated the potential of organising measurement campaigns but decided not to dedicate any efforts to this task.

	to the hackAIR platform, around a specific outlined task (e.g. measure the air quality during New Year's Eve, measure the air quality at the start of winter, etc.).		A measurement campaign seemed to have potential for BUND during wintertime. However, unfortunately, the timing of the pilot trials did not allow this.
Active and responsive pilot presence	Pilots are responsive to questions from participants, and provide support where necessary. This may require that they have an open channel of communication with technical partners (i.e. via a mailing list) so that they can respond in time to technical requests.	Exemplify, enable & encourage	Implemented. The pilots were pro-active on answering questions from users, especially about the technical support for sensors. This was mostly organised via email or face-to-face.
Awareness raising tactics	Ad-hoc demonstration, meetings and appearances from pilot partners can all be considered under this category (e.g. seminars, conferences, or other organised events)	Enlighten, enable, enthuse	Implemented. Several seminars (e.g. breakfast or lunch seminars) were organised by the pilot partners, as well as presence and demonstrations during conference.
Photo contests and social events	The goal of photo contests and social events is to let citizen's experience behaviour in a positive way and experience hackAIR in a fun and enjoyable way.	Engage, experience	Partially implemented. A photo contest was organised during the summer period by ON:SUB. The photo contest was very successful and led to more active users and uploads of photos during the contest period. Social events were not implemented.
Ambassadors and leaderships tactics	An ambassador is a participant in the hackAIR project that is an early adopter in the first stage. An ambassador most likely has a high level of awareness, and already takes current actions to reduce the individual	Exemplify	Partially implemented. This tactic was implemented by BUND, whereby a specific user provided support to others through a dedicated mailing list and budget.

	source of air pollution. Ambassadors can link to both the platform functions (skills-based) or to the engagement activities (network-based) in Germany and Norway. By also making these figures visible (through a t-shirt, token or in-app badge), ambassadors become users or figures that can be approached by other individuals and create trust between citizens and the technologies available in hackAIR.		

2.6 The hackAIR social media monitoring tools

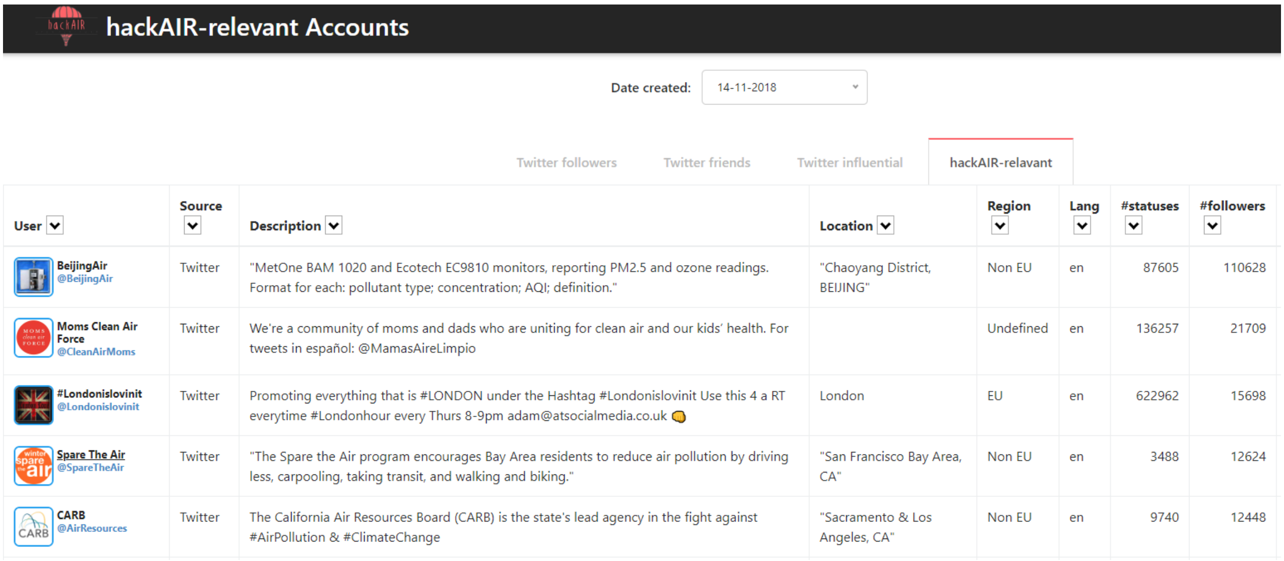
In this section, we summarize all the technical advancements and extensions that have been implemented for supporting additional features and functionalities concerning social media monitoring tools, corresponding to Task 6.3 Social media monitoring tools for assessing and supporting the engagement strategies. These updates were designed to support the engagement strategies and provide useful and easy-to-use tools.

2.6.1 Updated Discovery of Relevant Social Media Accounts

In D6.3 (hackAIR 2017) section 4, we thoroughly described the procedure for discovering relevant social media accounts. One major change from the previous work is relevant with the social media retrieval. Apart from the fact that we stopped using the Google+ API due to very limited relevant information, support for Facebook was also stopped due to Facebook's breaking changes[14] on 4/4/2018, which introduced restrictions to some endpoints that were critical for the application. One example is the /search[15] endpoint of the Facebook API which stopped supporting page and user object types based on keywords.

Furthermore we built an automated service written in Python, periodically running the discovery of relevant social media accounts pipeline. In a newly developed hackAIR-relevant accounts user interface[16], the user can inspect the results of the discovery process. Alongside the relevant social media accounts there is additional information about the region of the account (EU/non-EU), the language and some social media specific information about the activity and popularity of the account. This process is updated on a weekly basis. Historical information is also available through a dropdown list of previous dates. Figure 23 depicts the discovery of relevant social media accounts service.

[14] https://developers.facebook.com/docs/graph-api/changelog/breaking-changes/
[15] https://developers.facebook.com/docs/pages/searching/
[16] http://hackair-mklab.iti.gr/table/



Date created: 14-11-2018

User	Source	Description	Location	Region	Lang	#statuses	#followers
BeijingAir @BeijingAir	Twitter	"MetOne BAM 1020 and Ecotech EC9810 monitors, reporting PM2.5 and ozone readings. Format for each: pollutant type; concentration; AQI; definition."	"Chaoyang District, BEIJING"	Non EU	en	87605	110628
Moms Clean Air Force @CleanAirMoms	Twitter	We're a community of moms and dads who are uniting for clean air and our kids' health. For tweets in español: @MamasAireLimpio		Undefined	en	136257	21709
#Londonislovinit @Londonislovinit	Twitter	Promoting everything that is #LONDON under the Hashtag #Londonislovinit Use this 4 a RT everytime #Londonhour every Thurs 8-9pm adam@atsocialmedia.co.uk	London	EU	en	622962	15698
Spare The Air @SpareTheAir	Twitter	"The Spare the Air program encourages Bay Area residents to reduce air pollution by driving less, carpooling, taking transit, and walking and biking."	"San Francisco Bay Area, CA"	Non EU	en	3488	12624
CARB @AirResources	Twitter	The California Air Resources Board (CARB) is the state's lead agency in the fight against #AirPollution & #ClimateChange	"Sacramento & Los Angeles, CA"	Non EU	en	9740	12448

Figure 23. Discovery of relevant social media accounts service (available on: http://hackair-mklab.iti.gr/table/)

2.6.2 Updates on Advanced audience analysis

To further support identification of hackAIR ambassadors there was a need to automate the tools we described in D6.3 (hackAIR 2017), section 5.1. Community detection through follower graph construction was found to be a very valuable tool for identifying influential users among communities in the @hack_air[17] Twitter graph. Additionally, this task promotes the engagement via following relevant accounts or direct communication with relevant users and works in a complementary manner with the service discussed in 2.6.1. We created a Python service to automate the following tasks:

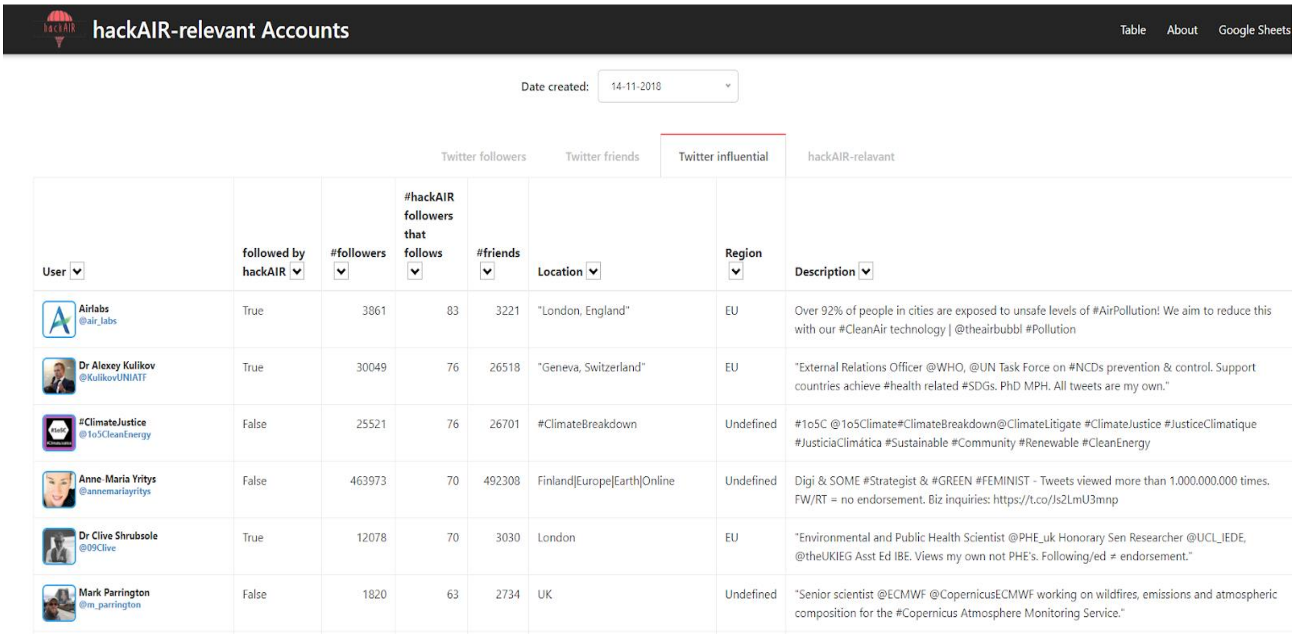
- Construct the follower graph as explained in D6.3 (hackAIR 2017), section 5.1.1
- Perform community detection using a Python implementation[18] of the Louvain algorithm
- Name the discovered communities using names of communities defined in D6.3 (hackAIR 2017) section 5.1.2. To accomplish this we make a list for the top 50 accounts (based on the incoming degree in the follower graph) for every community in the initial graph created for (hackAIR 2017, D6.3). For every new graph we inspect the mapping of these accounts to new communities. If the largest proportion of these accounts is mapped to one certain community then this community is named after the community of the top accounts. If more than one community accounts mapped to one new community then the new community is named after more than one names. In Figure 25 we can see that some communities formed in November containing nodes from more than one former communities and as a result they appeared merged in the new graph. For example Green-smart cities and Greek accounts communities from February have formed one unified community in November.
- Identify the influential and relevant users in the graph. To find the influential users we check the number of incoming edges in the graph or the number of Twitter follower each user has. To discover relevant users we

[17] https://twitter.com/hack_air
[18] https://github.com/taynaud/python-louvain

make the assumption that the more @hack_air followers an account follows the more likely it is to be relevant. Based on that, we sort users based on the number of @hack_air followers they follow.

- Provide a visualization service[19] of network graph and communities

The service[20] that provides the list of influential users discovered with the above procedure is shown in Twitter influential tab. Figure 24 shows the influential accounts discovery that utilizes the @hack_air follower graph. The user can sort the accounts based on their number of followers (influential) or based on how many @hack_air followers it follows (relevant). There is also information about whether this account already follows the @hack_air account or not. This process is executed weekly and can further assist pilot coordinators and local communication managers to identify ambassadors on a regular basis.



Date created: 14-11-2018

User	followed by hackAIR	#followers	#hackAIR followers that follows	#friends	Location	Region	Description
Airlabs @air_labs	True	3861	83	3221	"London, England"	EU	Over 92% of people in cities are exposed to unsafe levels of #AirPollution! We aim to reduce this with our #CleanAir technology | @theairbubbl #Pollution
Dr Alexey Kulikov @KulikovUNIATF	True	30049	76	26518	"Geneva, Switzerland"	EU	"External Relations Officer @WHO, @UN Task Force on #NCDs prevention & control. Support countries achieve #health related #SDGs. PhD MPH. All tweets are my own."
#ClimateJustice @1o5CleanEnergy	False	25521	76	26701	#ClimateBreakdown	Undefined	#1o5C @1o5Climate#ClimateBreakdown@ClimateLitigate #ClimateJustice #JusticeClimatique #JusticiaClimática #Sustainable #Community #Renewable #CleanEnergy
Anne-Maria Yritys @annemariayritys	False	463973	70	492308	Finland|Europe|Earth|Online	Undefined	Digi & SOME #Strategist & #GREEN #FEMINIST - Tweets viewed more than 1.000.000.000 times. FW/RT = no endorsement. Biz inquiries: https://t.co/Js2LmU3mnp
Dr Clive Shrubsole @09Clive	True	12078	70	3030	London	EU	"Environmental and Public Health Scientist @PHE_uk Honorary Sen Researcher @UCL_IEDE, @theUKIEG Asst Ed IBE. Views my own not PHE's. Following/ed ≠ endorsement."
Mark Parrington @m_parrington	False	1820	63	2734	UK	Undefined	"Senior scientist @ECMWF @CopernicusECMWF working on wildfires, emissions and atmospheric composition for the #Copernicus Atmosphere Monitoring Service."

Figure 24. Influential accounts discovery service in the hackAIR follower graph

As we stated before, we also built a visualization service for the network and the communities around the hackAIR Twitter account. This service can enable users to locate key accounts in each of the communities, get Twitter insights and check the graph progress through time. Figure 25 illustrates the visualization service. Users can click on the nodes of the graph and inspect the account information, its community, and its connections to other communities. There is also the possibility to locate nodes with high incoming degrees (they appear larger) and discover influential accounts for every community.

[19] http://hackair-mklab.iti.gr/hackair-network/v2/
[20] http://hackair-mklab.iti.gr/table/

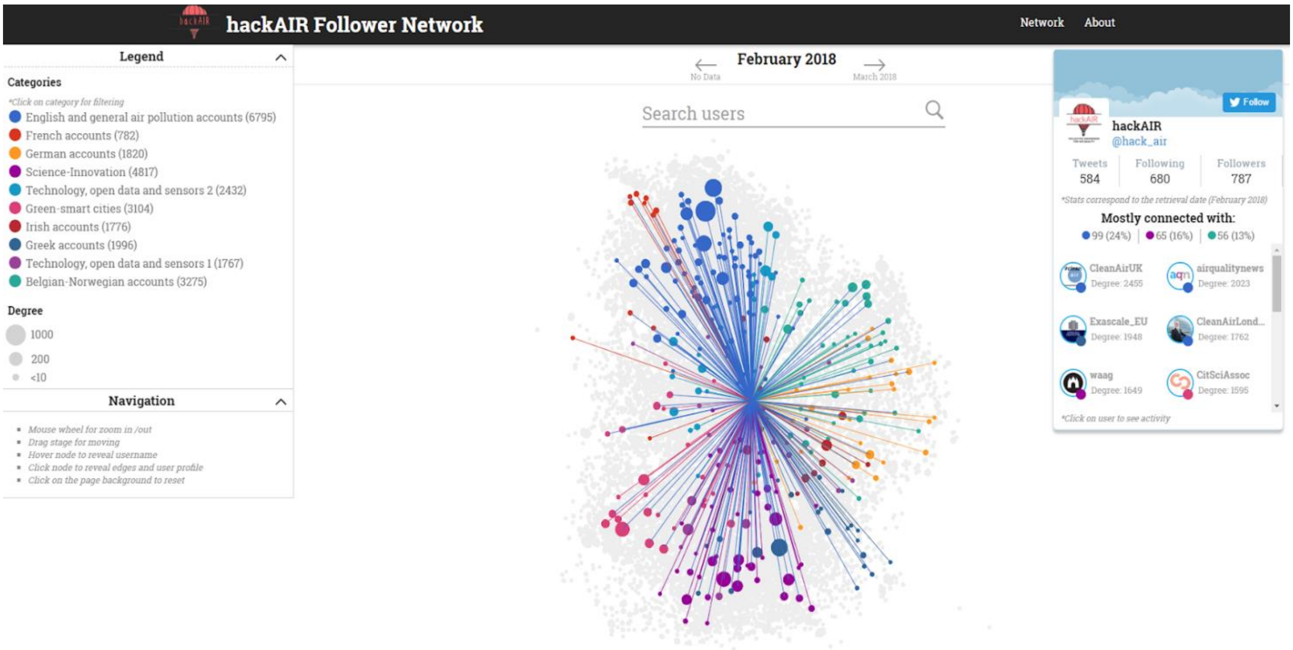

hackAIR Follower Network
Network
About
Legend
Categories
*Click on category for filtering
● English and general air pollution accounts (6795)
● French accounts (782)
● German accounts (1820)
● Science-Innovation (4817)
● Technology, open data and sensors 2 (2432)
● Green-smart cities (3104)
● Irish accounts (1776)
● Greek accounts (1996)
● Technology, open data and sensors 1 (1767)
● Belgian-Norwegian accounts (3275)
Degree
1000
200
<10
Navigation
Mouse wheel for zoom in /out
Drag stage for moving
Hover node to reveal username
Click node to reveal edges and user profile
Click on the page background to reset
February 2018
No Data
March 2018
Search users
Follow
hackAIR
@hack_air
Tweets 584
Following 680
Followers 787
*Stats correspond to the retrieval date (February 2018)
Mostly connected with:
● 99 (24%) | ● 65 (16%) | ● 56 (13%)
CleanAirUK Degree: 2455
airqualitynews Degree: 2023
Exascale_EU Degree: 1948
CleanAirLond... Degree: 1762
waag Degree: 1649
CitSciAssoc Degree: 1595
*Click on user to see activity

Figure 25. Network and community visualization service

Another valuable functionality is that the user can inspect the progress of the graph in time. This can provide strong insights about how communities behave and what are the impacts on them as the network of @hack_air account is evolving. Figure 26 contains two instances of the visualized graph. The first is from February 2018 and the other from November 2018. We can notice that some communities have merged together meaning they are more densely connected to each other. Also the number of accounts in each community has increased significantly. This can be explained by the proactive dissemination strategy and engagement initiatives performed by the communication managers, which played a critical role in graph's evolution.

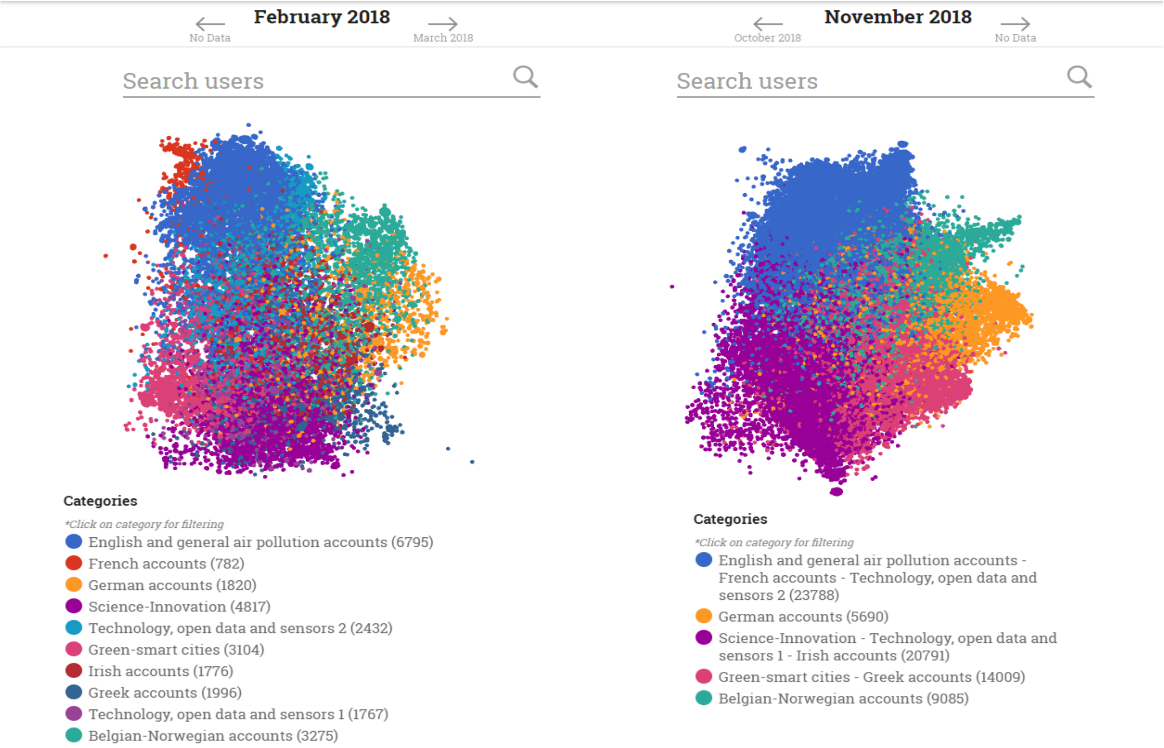

February 2018
No Data
March 2018
Search users
Categories
*Click on category for filtering
● English and general air pollution accounts (6795)
● French accounts (782)
● German accounts (1820)
● Science-Innovation (4817)
● Technology, open data and sensors 2 (2432)
● Green-smart cities (3104)
● Irish accounts (1776)
● Greek accounts (1996)
● Technology, open data and sensors 1 (1767)
● Belgian-Norwegian accounts (3275)
November 2018
October 2018
No Data
Search users
Categories
*Click on category for filtering
● English and general air pollution accounts - French accounts - Technology, open data and sensors 2 (23788)
● German accounts (5690)
● Science-Innovation - Technology, open data and sensors 1 - Irish accounts (20791)
● Green-smart cities - Greek accounts (14009)
● Belgian-Norwegian accounts (9085)

Figure 26. Evolution of network graph from February 2018 to November 2018

2.6.3 Discovery of hackAIR-newsworthy articles from social media

The purpose of this task is to provide hackAIR social media account managers with content that is worth sharing and that could eventually lead to an increased follower base. In addition, this task will help the consortium stay up-to-date with the latest developments and trends in the area of air quality research, technology, activism and policy.

2.6.3.1 Methodology

To provide content that is relevant to hackAIR stakeholders, we devised the following methodology. Initially, we create a list of twitter users that are relevant with air quality. This process is thoroughly explained in D6.3 (hackAIR 2017) section 4. This user list is updated every two weeks. The next step involves the retrieval of every tweet that is posted in the previous 24 hours from every account in the aforementioned user list.

To filter tweets that are irrelevant with air quality, we use an updated version of the classifier that was proposed in D6.3 (hackAIR 2017) section 4.5. To update the classifier (Air_Quality_Tweet), we used a training set of 1800 Tweets, manually labeled with respect to whether they provide information about air quality (relevant) or not (irrelevant). The same type of pre-processing is applied to the Tweets as in the case of accounts (i.e. stemming and stop-word removal) and a tf-idf bag-of-words representation is used. We also use the same classification algorithm, i.e. L2-regularized L2-loss Support Vector Machine with default parameters. Details of the classifier and its performance are shown Table 8.

Table 8. Air_Quality_Tweet classifier details

Classifier	# examples	# relevant	# irrelevant	Precision	Recall
Air_Quality_Tweet	1800	800	1000	90.3%	89.4%

After the classification process we retrieve a list of tweets that are related to air quality. Due to the fact that hackAIR social media managers are interested only in European accounts, we apply the location estimation method from Kordopatis-Zilos et al., 2017 on the tweet text in order to remove non-EU referring tweets.

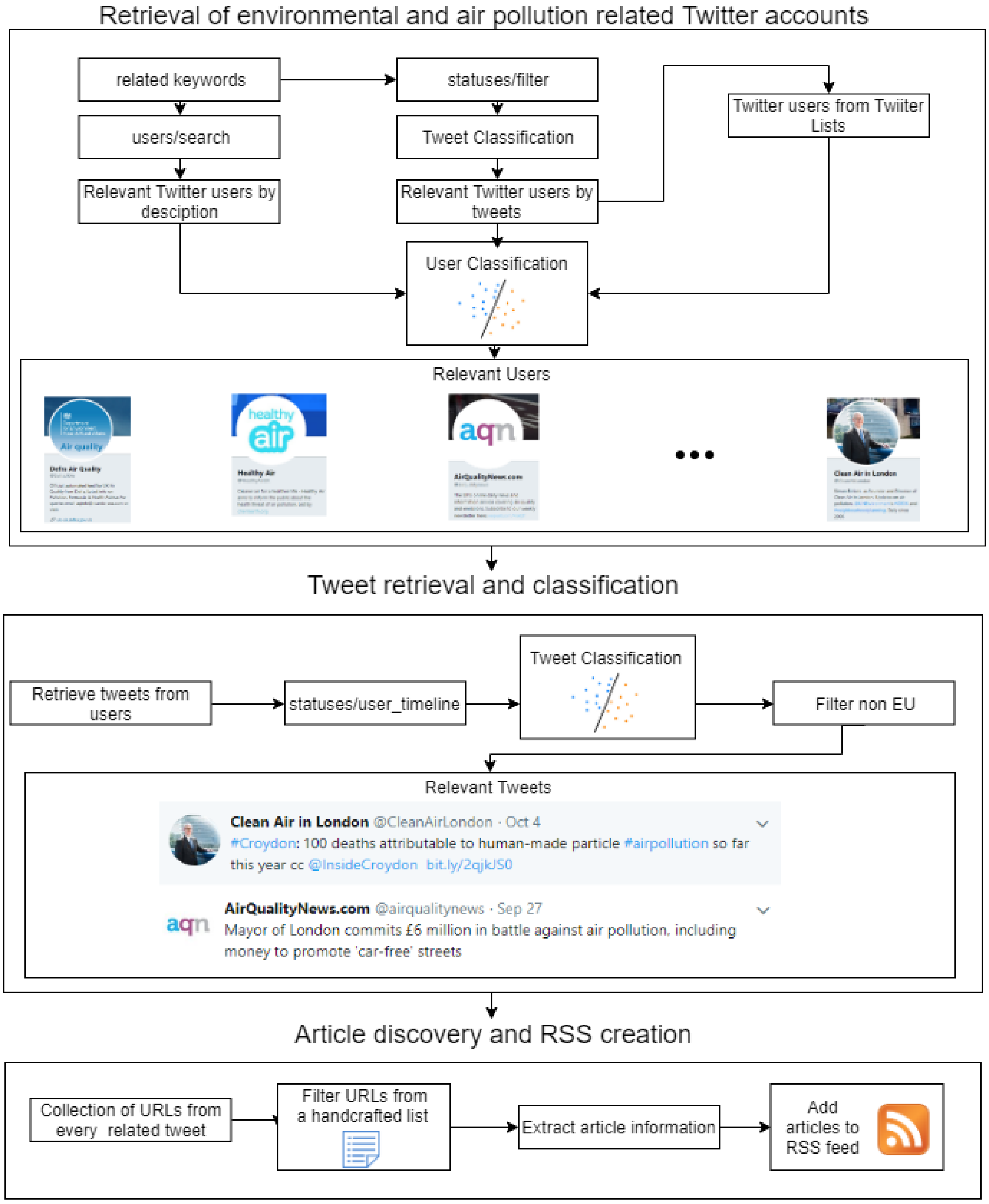
We then utilize the pool of EU-referring air quality tweets to extract URLs from them (if any). In the next step, we filter these URLs using a handmade list[21] of valid URLs and regular expressions from news sites and air quality related resources to retrieve related articles.

The final step includes the title and metadata extraction of the article using the Newspaper3k[22] Python module. Metadata include author, publish date and content summary using natural language processing provided by the module. The resulting RRS feed from the collected articles is updated daily and it is hosted on http://hackair-mklab.iti.gr/feed/rss-aq.xml.

Figure 27 illustrates the updated methodology for hackAIR-newsworthy article discovery.

[21] https://docs.google.com/spreadsheets/d/1j1laT3PNoOxnut1m70JzAuzbWwWIvJ7fqFMjUEBEwDQ/edit?usp=sharing

[22] https://github.com/codelucas/newspaper


Retrieval of environmental and air pollution related Twitter accounts
related keywords
users/search
Relevant Twitter users by desciption
statuses/filter
Tweet Classification
Relevant Twitter users by tweets
Twitter users from Twiiter Lists
User Classification
Relevant Users
Tweet retrieval and classification
Retrieve tweets from users
statuses/user_timeline
Tweet Classification
Filter non EU
Relevant Tweets
Article discovery and RSS creation
Collection of URLs from every related tweet
Filter URLs from a handcrafted list
Extract article information
Add articles to RSS feed

Clean Air in London @CleanAirLondon · Oct 4
#Croydon: 100 deaths attributable to human-made particle #airpollution so far this year cc @InsideCroydon bit.ly/2qjkJS0

AirQualityNews.com @airqualitynews · Sep 27
Mayor of London commits £6 million in battle against air pollution, including money to promote 'car-free' streets

Figure 27. Updated hackAIR methodology for hackAIR-newsworthy article discovery.

2.6.3.2 Updated Results

Table 9 presents a list of relevant examples that demonstrate a sample of useful and worth sharing articles that have been retrieved using the service.

Table 9. Relevant article examples

Title	Summary	URL
Air pollution particles found in mothers' placentas	Scientists have found the first evidence that particles of air pollution travel through pregnant women's lungs and lodge in their placentas	https://www.theguardian.com/environment/2018/sep/16/air-pollution-particles-found-in-mothers-placentas
Illegal pollution is worse than we thought, tests reveal	Client Earth won a High Court ruling in February requiring the government to take more action on air pollution. Pollution is much worse in some	https://www.thetimes.co.uk/article/illegal-pollution-is-worse-than-we-thought-tests-reveal-b50x7th62

	areas than previously believed, the government has admitted	
The school where the windows have to stay shut and the children can taste pollution	So, are pollution levels really that bad in Greater Manchester? At St Ambrose RC Primary on Princess Parkway, pollution levels are so bad that teachers can no longer open windows in the classroom	https://www.thetimes.co.uk/article/illegal-pollution-is-worse-than-we-thought-tests-reveal-b50x7th62
FIVE THINGS WE LEARNT WHEN 20,000 BELGIANS BECAME AIR POLLUTION SCIENTISTS	The biggest ever citizens' investigation into air pollution has produced some interesting results about air quality in Europe	https://metamag.org/2018/10/04/five-things-we-learnt-when-20000-belgians-became-air-pollution-scientists/

In some rare cases the results happen to be irrelevant with air quality. Most of the times the irrelevant article is referring to climate change topics. This can be explained due to the fact that tweet classification cannot fully distinguish between air quality and climate change topics, since similar terms appear in both contexts. Also, in some other cases the location estimation method does not filter out non-EU content. Examples of such cases are listed in Table 10.

Table 10. Irrelevant article examples

Title	Summary	URL	Reason
Scientists say halting deforestation 'just as urgent' as reducing emissions	By protecting and restoring forests, the world would achieve 18% of the emissions mitigation needed by 2030 to avoid runaway climate change, the group of 40 scientists, spanning five countries, said in a statement	https://www.theguardian.com/environment/2018/oct/04/climate-change-deforestation-global-warming-report	Climate change related
Indian startup turns air pollution into ink	There are times when the air pollution in Delhi is so bad, you can barely see the hand in front of your face	https://www.dw.com/en/indian-startup-turns-air-pollution-into-ink/a-45664168?maca=en-rss_en_global_ideas-24361-xml-mrss&utm_source=dlvr.it&utm_medium=twitter	Non EU
The satellite that can clean up space rubbish from Earth's orbit	VideoA net has been successfully fired into space as part of a plan to clean up the millions of pieces of rubbish floating in Earth's orbit	https://www.bbc.co.uk/news/av/science-environment-45569068/the-satellite-that-can-clean-up-space-rubbish-from-earth-s-orbit?ocid=socialflow_twitter	Not air quality related

The service retrieves on average 2520 air pollution related Twitter accounts for sourcing links to articles. The daily average number of all tweets retrieved from relevant users is 4074. From these, the average number of unique relevant tweets number is 65, 12 of which contain URLs leading to 7 relevant articles after filtering. Figure 28 shows the daily distribution of related articles retrieved by the hackAIR-newsworthy article discovery service.

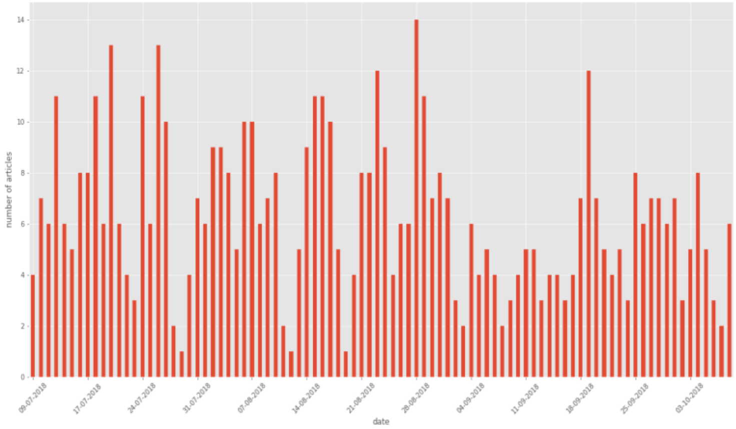


Figure 28. Daily related article distribution

2.7 Updates on the hackAIR platform and support to users and software developers

The most recent developments as regards the delivery of the hackAIR platform were described two months prior the submission of the present document, and specifically in the deliverable D5.3 (M34).

First of all, the development team offered continuous technical support to users through emails, in which DRAXIS offered assistance on the sensor assembly, the use of the hackAIR web/ mobile app, and the interpretation of the produced data. Moreover, all project partners reported issues, mainly bugs and minor improvements, through the JIRA issue tracking system, that have been already resolved by DRAXIS.

The platform codes were released in GitHub (https://github.com/hackair-project) and Zenodo (https://doi.org/10.5281/zenodo.1442608) so that other developers can build on top of it. In addition, an API that enables the exchange of information with the hackAIR web and mobile app was released, accompanied with an API documentation that has been kept up-to-date. The technical team supported also software developers who wanted to use the API. Finally, the air quality data that were created within hackAIR were anonymized and made available in Zenodo (https://doi.org/10.5281/zenodo.2222342) so that external researchers and developers can feed the hackAIR API with them and deliver new services.

More details on the updates of the hackAIR platform development can be found in D5.3 – Final version of integrated and tested hackAIR open platform (https://doi.org/10.5281/zenodo.2276621).

3 Conclusions

The current document describes the modifications performed on the hackAIR services and methodologies based on the user needs emerged from the pilot implementations. The improvement of a service or a product is a work that never ends, as different users may have different needs that you have to satisfy, and technology evolves in such a rapid way that someone should be agile to keep up with it. Within the hackAIR project, the consortium was constantly evaluating the need of updates and improvements on the currently provided services, and proceeded to their update or modification based on the available resources.

As the project comes to an end, the project partners are committed to keep the hackAIR platform live and functional for as long as possible. In case of an opportunity of further funding come up, they are interested to continue improving the provided services for the benefit of the research community and the public.

References

- Bambust, F. (2015). Effectief gedrag veranderen met het 7E-model. Brussel: Politeia
- Belouaer, L., Bouzid, M., and Mouaddib, A.I. (2010). Ontology based spatial planning for human-robot interaction. In Temporal Representation and Reasoning (TIME), 17th International Symposium, pp. 103-110, IEEE.
- Codescu, M., Vale, D.C., Kutz, O., and Mossakowski, T. (2012). Ontology-based route planning for OpenStreetMap. In Terra Cognita 2012 Workshop, p. 62.
- Day, D. E., Malm, W. C., Aerosol light scattering measurements as a function of relativehumidity: a comparison between measurements made at threedifferent sites, Atmospheric Environment 35 (2001) 5169–5176
- hackAIR Consortium (2016). Deliverable 3.1: 1st Environmental node discovery, indexing and data acquisition. Available online: http://www.hackair.eu/wp-content/uploads/2016/12/d3.1_environmental_node_discovery_indexing_and_data_acquisition_1st_.pdf .
- hackAIR Consortium (2017). Deliverable 3.2: 2nd Environmental node discovery, indexing and data acquisition. Available online: http://www.hackair.eu/wp-content/uploads/2016/03/d3.2-2nd_environmental_node_discovery_indexing_and_data_acquisition_v1.2.pdf .
- hackAIR Consortium (2017). Deliverable 4.2: Semantic Integration and Reasoning of Environmental Data. Available online: http://www.hackair.eu/wp-content/uploads/2016/03/d4.2-semantic_integration_and_reasoning_of_environmental_data.pdf.
- hackAIR Consortium (2017). Deliverable 6.3: Social media monitoring tools for assessment and support of engagement. Available online: http://www.hackair.eu/wp-content/uploads/2018/02/d6.3-social_media_monitoring_tools_for_assessment_and_support_of_engagement_v1.2-final%20%281%29.pdf
- hackAIR Consortium (2018). Deliverable 7.4: Intermediate pilot implementation and evaluation report. Available online: http://www.hackair.eu/wp-content/uploads/2018/11/D7.4_Intermediate-pilot-implementation-and-evaluation-report-final.pdf .
- Kosmidis, E., Syropoulou, P., Tekes, S., Schneider, P., Spyromitros-Xioufis, E., Riga, M., Charitidis, P., Moumtzidou, A., Papadopoulos, S., Vrochidis, S., Kompatsiaris, I., Stavrakas, I., Hloupis, G., Loukidis, A., Kourtidis, K., Georgoulias, A., and Alexandri, G. (2018). "hackAIR: Towards raising Awareness about Air Quality in Europe by developing a Collective Online Platform", ISPRS International Journal of Geo-Information 2018, 7.
- Provine, R., Schlenoff, C., Balakirsky, S., Smith, S., and Uschold, M. (2004). Ontology-based methods for enhancing autonomous vehicle path planning. Robotics and Autonomous Systems. Vol. 49, pp.123-133.
- Ramachandran, G., Adgate, J. L., Pratt, G.C., Sexton, K. (2003). Characterizing Indoor and Outdoor 15 Minute Average PM2.5 Concentrations in Urban Neighborhoods; Aerosol Science and Technology 37:1, 33 – 45
- Riga, M., Kontopoulos, E., Karatzas, K., Vrochidis, S., and Kompatsiaris, I. (2018). “An Ontology-based Decision Support Framework for Personalized Quality of Life Recommendations”. In: Dargam, F., Delias, P., Linden, I., Mareschal, B. (eds) Decision Support Systems VIII: Sustainable Data-Driven and Evidence-Based Decision Support. ICDSST 2018. Lecture Notes in Business Information Processing (LNBIP) 313, pp. 38-51. Springer, Cham. Available online: http://dx.doi.org/10.1007/978-3-319-90315-6_4.
- Kordopatis-Zilos, G., Papadopoulos, S., & Kompatsiaris, I. (2017). Geotagging text content with language models and feature mining. Proceedings of the IEEE, 105(10), 1971-1986.

- Kosmidis, E., et al., hackAIR: Towards Raising Awareness about Air Quality in Europe by Developing a Collective Online Platform, ISPRS International Journal of Geo-Information, 7(5), 2018.
- Lahoz, W. A., and P. Schneider, Data assimilation: making sense of Earth Observation, Frontiers in Environmental Science, 2(16), 1–28, doi:10.3389/fenvs.2014.00016, 2014.
- Laulainen, N. S. (1993). Summary of Conclusions and Recommendations from a Visibility Science Workshop; Technical Basis and Issues for a National Assessment for Visibility Impairment, Prepared for US DOE, Pacific Northwest Laboratory, PNL-8606.
- S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering,* vol. 22, no. 10, pp. 1345-1359, 2010.
- Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. Annals of statistics, 1189-1232.
- S. Mei, H. Li, J. Fan, X. Zhu and C. R. Dyer, "Inferring air pollution by sniffing social media," Advances in Social Networks Analysis and Mining (ASONAM), 2014 IEEE/ACM International Conference on. IEEE, p. 534–539, 2014.