Eywa: Crowdsourced and cloudsourced omniscience

Murat Demirbas Yavuz Selim Yilmaz Muhammed Fatih Bulut

Ubiquitous Computing Lab
Department of Computer Science and Engineering
University at Buffalo, SUNY, Buffalo, NY, 14260
{ demirbas | yavuzsel | mbulut }@buffalo.edu

***Abstract*—Here we present our ubiquitous computing vision, Eywa. Eywa is an open publish-subscribe system that employs crowdsourcing for tasking and social networks & machine learning for identifying relevance. We argue that crowdsourcing (and the social networks and machine learning that enable it) should be a first class citizen in ubiquitous computing. We also observe that cloud computing is a natural platform to host such future ubiquitous computing systems. We discuss about some applications enabled by Eywa, and focus on our *CuratedLiving* application (which emphasizes "less choice more relevance" approach) as a case study.**

***Index Terms*—Crowdsourced collaboration; Smartphones; Internet of things; Social networks; Cloud computing**

I. Introduction

Mark Weiser depicted a clear picture of his ubiquitous computing vision back in 1991 [1]. This vision, although looked vivid, did not get realized in the next couple decades. Instead the technology trends went the opposite way. While Weiser called for computers to come outside and be embedded in the fabric of our physical world, what happened was that a lot of physical services became virtualized and moved into the computer, or more accurately to the Internet/cloud. For example, email has seen widespread adoption and almost eradicated physical (snail) mail. Ecommerce services found great adoption, and grew into big businesses which now constitute a significant portion of our economy. Even friendships got virtualized and we started *friending* in social networks, such as Facebook and Twitter. Most recently, computing ultimately become a virtualized commodity being served at the cloud.

Yet, this virtualization trend does not mean Weiser's vision was lacking or flawed in any aspect and was rather a reflection of the natural flow of things. A ubiquitous computing system has both cyber/virtual and physical components, and the cyber side received most of the attention first as it was easier to attain. Behind the scenes, we have also seen a growth in the physical component as well. With the advances in MEMS technology in the previous decade, it has become feasible to produce various types of sensors (such as magnetometers, accelerometers, passive-infrared based proximity, acoustics, light, heat) inexpensively, in very small-form factor, and in low-power usage. What started with wireless sensor networks research for instrumenting the world with sensors [2], found market adoption lately in the form of Arduino devices [3], Maker movement [4], and Internet of Things [5]. Smartphones have found widespread adoption[1], and brought computers with instant access to the clouds to our fingertips. Finally, maps are replaced by GPS navigation and WiFi-assisted localization, which provide ubiquitous localization. This paved the way to realizing the physical component of the ubiquitous computing vision. Indeed we have observed a quick proliferation of location-based services and location-aware services, which are tell signs of the pendulum swinging back to physical; this time empowered with a very strong cyber component. As a result, we are seeing innovative location-based services such as cab hailing [6], line wait-time monitoring [7] [2], and suggest-nearby services by search engines [8].

Although there is good progress, we are still not there. Despite the availability of the devices to fulfill the ubiquitous computing vision, the-state-of-the-art falls short of this vision. We argue that the reason for this gap is the lack of an infrastructure to task/utilize these devices for collaboration and coordination. In the absence of such an infrastructure, the state-of-the-art today is for each device to connect to Internet to download/upload data and accomplish an individual task that does not require collaboration and coordination. Providing an infrastructure for publish/subscribe and tasking of these devices enables any node to search the data published by several nodes in one region to aggregate and decide on a question, as well as task several nodes in one region to acquire the needed data (if the data is not already being published to the infrastructure).

Consider DARPA's 2009 network grand challenge on the occasion of the 40th anniversary of the Internet. The challenge was to accurately find 10 weather balloons deployed in arbitrary locations of the U.S. within a day [9]. There was an award of $40,000 for the team that would first report the locations of the 10 balloons accurately. This challenge was solved within 9 hours. The winning team, from MIT, campaigned aggressively for a month before the challenge day and employed social networks and a multi-level incentive structure (distributing prize money among the people who contributed towards finding the balloons). Although this DARPA challenge

[1] There are more than 5 billion cellphones with 20% of these smartphones

[2] In our earlier work, LineKing [7], we have used the sensors on the smartphones to detect the wait times at coffee shops, and leveraged on the computational power provided by cloud to provide future wait time estimations with less than 3 mins mean absolute error.

was solved by leveraging on existing technologies, it required a month of campaigning and preparation to solve. Why can we not have an app that is able to solve similar collaboration and coordination problems automatically? Why are the existing apps so limited and person-centric?

II. Eywa vision

In order to task/utilize ubiquitous devices for collaboration, we propose a middleware which provides an "open" publish-subscribe infrastructure for sensors and smartphones, and paves the way for crowdsourced sensing and collaboration applications. We call this middleware Eywa, and describe our Eywa vision in the rest of the paper.

(Remark:) Eywa is the name of the guiding force and deity of the planet Pandora and the Na'vi people in the fictional universe of Avatar, the movie directed by James Cameron. The Na'vi believe that Eywa acts to keep the ecosystem of Pandora in perfect equilibrium. All living things on Pandora connect to Eywa through a system of neuro-conductive antennae; this is why Na'vi can mount their direhorse or mountain banshee steeds and ride them immediately without going through the necessary steps required to domesticate such wild animals. The Na'vi can upload or download memories from Eywa using their queues and it can even be used for mind transfers in certain cases. http://james-camerons-avatar.wikia.com/wiki/Eywa *(End of Remark.)*

Next we discuss about the components of Eywa. Eywa aggregates information from many devices over many channels, provides an open publish-subscribe to allow extensible ways of rehashing this information, employ social networks for tasking, and deploy machine-learning over a cloud computing platform to achieve relevance.

A. Sensing and aggregating component

Eywa gets input from many types of devices/people, and through many different channels. Parking lots (rather sensors deployed at parking lots [10]) stream to Eywa the available parking spots information. Traffic reports through sensors, public transportation schedules [11], GPS and sensor readings from cars [12], and cloud services such as Google Maps/traffic also input to Eywa. GPS and indoor WAP signals, Bluetooth connection information, and many different modality sensory streams (accelerometer, magnetometer, microphone, ambient audio, ambient lighting, gyroscope, temperature, proximity) from smartphones of mobile users flow to Eywa. Eywa's reach also extend to Internet of Things devices, street webcams, satellite photos, social network status updates (Twitter, Facebook, Foursquare), organization event calendars [13], and in some cases on-demand requests to humans for taking a picture or providing an answer.

Eywa aims to provide a flexible/extensible architecture to allow mashing and combining these information in a variety of ways. To this end, Eywa exposes these streams through carefully designed publish/subscribe APIs (to protect privacy, prevent abuse, etc.). The open publish-subscribe architecture of Eywa implies that different actors may integrate user data differently. Moreover, third parties can use the gathered data in unanticipated ways to offer new services with them. For example consider the example of streamed GPS locations from trucks to Eywa. The truck company may access/use this information through Eywa to check on the progress made by the drivers and send them warning messages if there are concerns. The highway authority may use this information to monitor the utilization congestion levels on the roads, and may also use this to send weather/road related warnings. The highway service plazas may also want to use this information to plan their food/accommodation services more accurately. Finally unanticipated applications may also spring. This information may be used for estimating the fresh vegetable/food deliveries to the city and to schedule shopping of health-conscious/dieting users.

The information flow in the Eywa ecosystem is not unidirectional. Eywa and the applications built on top of Eywa have symbiotic mutualism. In the Eywa ecosystem, the apps receive data from Eywa, evaluate and use data for their purposes. But that is not the end of the information flow cycle. The apps also publish their output data to Eywa in a relevant way. This symbiosis relation provides embedded collaboration and helps the mutual growth of the Eywa ecosystem.

B. Computing and Machine learning component

Eywa is hosted on the cloud to provide scalability and high-availability. Cloud computing employs virtualization technologies over general-purpose servers (with an aim to hit the price/performance sweet spot) to enable elastic scalability and provides the illusion of infinite computing resources available on demand. Since Eywa is maintained on the cloud with high bandwidth, it is easy to upload high traffic information to Eywa and many hosts can query and get service from Eywa without scalability, availability, and latency problems.

Highly scalable cloud computing architectures and software for machine learning are now mature and widely used. Eywa employs BigData analytics (using MapReduce [14], Pregel [15]) and parallel batch processing (over AWS EC2 [16], mainly for handling the computationally intensive computer vision and image processing jobs) to process, interpret, and learn from the huge input data streams it aggregates. The machine learning methods Eywa employs include interpolation (for missing data), finding appropriate people for a task (e.g. finding some friends, who has knowledge of the topic, for suggestions), activity recognition [17] (to detect events or for personal assistance), seasonality detection, and social network change detection. These methods enable Eywa to build on the raw input data (published by sources to Eywa), and add more meaning/context and provide processed/refined information as output.

C. Social networking component

Eywa employs social networks to enable crowdsourced tasking and collaboration applications. Social networks are important for tasking and collaboration applications, because a person can task (ask for favors from) her friends and social

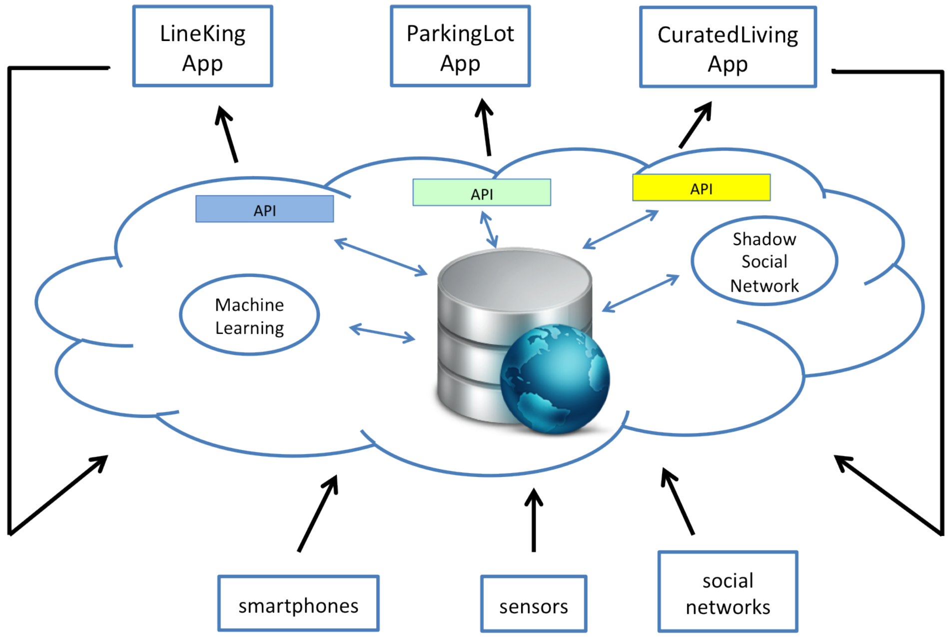

Fig. 1. Eywa ecosystem

network with more ease than complete strangers. Alternatively, from the other perspective, one may enjoy volunteering his time for friends/family than complete strangers. Moreover, suggestions sourced from friends have more relevance as friends know more about the targeted person than complete strangers do.

Eywa maintains a shadow social network mirrored from Facebook and Twitter to keep track of the social network of the users. This is a shadow social network in the sense that Eywa provides a middleware to the public social network API's (e.g. Facebook Open Graph API), and via this middleware apps can get Eywa empowered query results from 3rd party API's. Furthermore, Eywa also extends and maintains a superset of existing networks as it needs to employ the user's device (primarily smartphone) for automated tasking/publishing to this network. Eywa may task appropriate smartphones from the users social network to start sensing and publishing information to Eywa, so Eywa keeps this device-user association as part of the social network. Also since this is a shadow social network, the Eywa related sensing/tasking information do not litter the main social network timeline of the user.

In addition to social networks of devices, Eywa also considers location-induced networking of devices. Although the device users may not be acquainted and in each others social circles, it may happen that these users share the same spaces regularly. This concept is also known as *familiar strangers*. For example, two students taking the same class in a big auditorium, or people working in the same building but for different companies, or people living in the same neighborhood and use the same shopping malls may fit into this category. These people have relevant information to share with each other, but not the appropriate social connections/channel to share this information. Eywa fills in this gap, by also taking into account the location-induced networking of devices (we call this as spatiotemporal device networks) as a relevant component. Eywa captures the spatiotemporal device networks using WiFi-direct [18] technology.

D. Crowdsourcing component

Crowdsourcing is a component that enables and feeds into Eywa. We argue that crowdsourcing should be a first class citizen in ubiquitous computing. Crowdsourcing refers to distributing a query to several users in order to exploit the wisdom-of-crowds effect. Examples of crowdsourcing may be a participant-powered weather/rain-radar (with better precision and ground-truth than meteorological weather radars), friend-sourced suggestions (restaurants, music, places, experience, topics, other friends), expertsourced advice, crowdsourced location-based queries, crowdsourced database [19] and distributed crowdsourced map-reduce [20].

In Eywa, we rely primarily on smartphones as the devices for crowdsourced tasking. Smartphones cover large areas due to their mobility. Smartphones are personal and administrated by their users, and provide the potential of interacting with the phone user for tasks requiring human intelligence and intervention, such as taking a picture of a requested location, answering a question.

Eywa maintains an extensive virtual currency model for its incentive mechanism. An example of such model might be the bitcoin virtual currency [21]. In Eywa incentive model, every client needs to pay in order to get something from Eywa, and every client gets rewarded when they publish to Eywa. By this way, Eywa maintains the quality of queriers and eliminates spam queries (by making them harder/costly to initiate). Consider DARPA network challenge example [9]; in Eywa ecosystem, such an experiment is easy if the client possesses and agrees to spend her virtual money on it. This is mostly not a big deal for the apps that create meaningful results (which they will in return submit to Eywa and earn virtual money). This Eywa incentive mechanism forces the apps/users to be profitable to each other, and maintains a healthy ecosystem based on symbiotic mutualism.

Some of our work on crowdsourcing include:

- Location-based querying app over Twitter [22]
  This application categorizes Twitter users based on their familiarity to the location types taken from Foursquare, and then queries the users with respect to their categories. Experimental results showed that, for non-factual question, crowdsourcing significantly improves the success rate of finding an answer compared to traditional search engines. As a case study, we developed an app that queries Twitter users based on their locations to obtain fine grained location based weather conditions [23].
- Monitoring changes in location related tweets in cities [24]
  The heart of a city beats at its places, and CityPulse reveals this secret by observing location type based tweets. CityPulse uses passive crowdsourcing to identify the events happening in the city. Our experiments showed that CityPulse is able to identify 75% of the events correctly and accurately.
- Identifying breakpoints in public opinion for a topic [25]
  In order to see the trends in public opinion, we used a passive trend sensing algorithm which analyzes the tweets and categorizes them based on a multi-emotion category corpus. Our system can detect and represent opinion changes continuously. Our experiments show that the system is able to identify the breakpoints with 75% accuracy and are able to represent the events causing the breakpoint successfully.
- Building an expert-sourced system to play "Who wants to be a millionaire"
  This work aims to utilize the crowd to answer "Who wants to be a millionaire" questions faster and more accurately than the contestants.

Recently we started the PhoneLab [26] project to enable more of these experiments. PhoneLab is a large-scale smartphone testbed that provides an order of magnitude more participants than typical smartphone experiments. As the first phase of Phonelab project, we already distributed 250 phones to participants at UB. Phonelab provides access to both kernel layer and app layer and overcomes the limitations of App Store or Google Play. We plan to leverage on PhoneLab, as an information outlet and smartphone testbed, to build a first prototype of Eywa for enabling campus-wide ubiquitous collaboration and coordination applications.

III. Curated Living Service as a Case Study

The *CuratedLiving* service builds on top of Eywa, and aims to deliver less complex more relevant experiences to its users. The user tells CuratedLiving what type of day/week she wants to have and CuratedLiving provides a select number of paths (carefully curated choices) to arrange these goals. For example, the user may say "Over the weekend, I need to buy some dresses, do groceries (the grocery list is automated/learned), and also want to see a movie, meet with 2-3 friends to chat". The CuratedLiving then suggests a couple plans, which may also be refined along the way opportunistically, for realizing these goals. This is a sophisticated endeavor and builds on Eywa as the underlying service. CuratedLiving also requires cloud computing hosted machine learning to learn the preferences and the mood of the user and provide customized plans based on these. For example, initially, CuratedLiving chooses a direct and low-congested road (based on the traffic reports posted to Eywa), however, taking into account the mood of the user (sensed via her smartphone), the radio show or music the user is enjoying at the car (again sensed via the smartphone), CuratedLiving may instead opt for a longer but more comfortable or scenic drive. Based on the best available parking spots (again learned from Eywa), CuratedLiving may determine where to enter the mall, and this in turn may affect the plans as to which activity to take on first. At some point in the day, based on the availability of friends nearby (whose mobile positions are learned from Eywa), CuratedLiving may arrange a joint lunch. Alternatively based on the interests of the friends, this is converted to an art gallery visit followed by snack at a cafe (whose line-wait times are input to Eywa via LineKing [7]). Finally, based on customized/bargained deals, CuratedLiving may guide the user to buy the dresses and groceries.

CuratedLiving application uses Eywa via the Eywa APIs. These APIs enable CuratedLiving to record subscriptions to Eywa events/notification based on <keyword, location, time> as well as semantic clues. In real-world there is always some uncertainty so the Eywa notifications also include associated certainty levels. The APIs also include ways to modify the granularity and the fidelity level of information to be obtained from the publishing sources. For example, if the user trusts a friend more on some topic, the CuratedLiving service may instruct Eywa to prioritize data collection or tasking regarding that friend.

We envision that CuratedLiving will be able to improve itself and customize itself to serve better to the habits/peculiarities of its users by processing the data collected by the service, including the feedback/ratings of users about the experiences. The service can process these data and cluster users into groups and suggest a user specific experiences that users similar to her rated as most satisfying/pleasing. Enabling technologies for the CuratedLiving application are already in place. Siri provides a nice natural language/audio interface to the application. Google self-driving cars enable extra level of automation in the mobility of the user to provide a seamless curated living experience. Social networks and smartphones are other enabling technologies that have established themselves deeply and found widespread adoption.

A. Discussion about CuratedLiving

Curated living has always been around in some form. The culture/society we are raised in shapes/curates our lives. We also curate our lives indirectly by reading books (self-help books, cooking books, travel books) or talking to friends. CuratedLiving provides a more direct/efficient way of curating our lives. But, of course, making something fast and efficient is not always necessarily a worthy goal. An inefficient life curation process may have its advantages; it makes us more conscious and aware of the choices we need to make, and this strengthens our sense of self. As we make mistakes and regret them, we develop our tastes, likes and dislikes, and our ambition in life. *An unexamined life is not worth living* as Socrates declared.

We welcome self-driving cars because driving is a rot, and our greatest aspirations are not to be *chauffeurs*. But could it be the case that by giving away the rot of curating our lives, we are giving away too much? How much is too much? To answer these questions it is important to involve social sciences into this discussion. In any case, the CuratedLiving service will be tunable to provide a spectrum that gives only nominal guidance up to giving full service. Even in the full service mode, CuratedLiving will sprinkle some (hard-computed) perfectly reasonable deviations from the beaten track to make our day and life richer. Only by providing these opportunities to explore, the choices to make mistakes, it would be possible for the users to develop their tastes, whims, and sense of self.

IV. THE NEXT 100 UBIQUITOUS COMPUTING APPLICATIONS

In addition to the CuratedLiving application that we presented as a case study, Eywa enables several novel applications:

- Extremely customized/personalized and specific news
  These set of apps personalize the user's news timeline based on her needs, interests, and context (location, time, mood). For example, when a user drinks coffee, she may like to read longer articles, whereas on the go, she may want to glance at news about the locale.
- Charity deeds (community-organization events, running errands for elderly/disabled)
  These apps know what is needed by elderly people, what services volunteers might provide based on their abilities, interests, and context; then distribute workload based on priority and the availability of volunteers.
- CuratedLearning
  These apps empower users to learn outside the class in a manner customized for their learning style. The apps use opportunities (movies, examples, music, events) to enable hands-on and lifelong learning.
- CuratedTravel This app personalizes the travel experience based on the interests of people who share the same tastes, and provides opportunistic travel activities.
- Audubon society national bird counting survey
  This app empowers the bird watching hobbyists to collaborate on accurate and precise surveying of the bird species [27].
- Crowdsourced surveillance (Amber –missing child– alert, homeland security)
- Ad-hoc self-improvement clubs where everyone teaches something to the group
- Socializing (finding/meeting interesting people in nearby crowds)
- Forming opportunistic flashcrowds to make a difference (art, cleaning campaigns, etc.)
- Physical world gaming with role playing
- Social collaboration applications (pick-up soccer games, arranged ride-sharing, support groups for addicts, and support groups for exercizing and weight-watching)

Combined with microfinancing/microtransactions, Eywa enabled applications such as CuratedLiving, CuratedTravel, etc. may give people the chance to provide service related to their hobbies/passions and make a living. This grassroots participation economy can make a leeway into a more productive and creative society.

REFERENCES

[1] M. Weiser, "The computer for the 21st century," *Scientific American*, vol. 265, no. 3, pp. 94–104, 1991.
[2] http://research.microsoft.com/en-us/projects/senseweb.
[3] http://www.arduino.cc.
[4] http://p2pfoundation.net/Maker\textunderscoreMovement.
[5] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787 – 2805, 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128610001568
[6] https://www.uber.com.
[7] M. Bulut, Y. Yilmaz, M. Demirbas, H. Ferhatosmanoglu, and N. Ferhatosmanoglu, "Lineking: Crowdsourced line wait-time estimation using smartphones," in *MobiCASE: Fourth International Conference on Mobile Computing, Applications and Services*, 2012.
[8] http://googleblog.blogspot.com/2010/02/refine-your-searches-by-location.html.
[9] http://archive.darpa.mil/networkchallenge/.
[10] R. Sudhaakar, A. Sanzgiri, M. Demirbas, and C. Qiao, "A plant-and-play wireless sensor network system for gate monitoring," *The 10th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WOWMOM)*, 2009.
[11] P. Zhou, Y. Zheng, and M. Li, "How long to wait?: predicting bus arrival time with mobile phone based participatory sensing," in *Proceedings of the 10th international conference on Mobile systems, applications, and services*, ser. MobiSys '12. New York, NY, USA: ACM, 2012, pp. 379–392. [Online]. Available: http://doi.acm.org/10.1145/2307636.2307671
[12] http://cartel.csail.mit.edu.
[13] http://ubicomp.cse.buffalo.edu/ubupdates.
[14] J. Dean and S. Ghemawat, "Mapreduce: simplified data processing on large clusters," *Commun. ACM*, vol. 51, no. 1, pp. 107–113, Jan. 2008. [Online]. Available: http://doi.acm.org/10.1145/1327452.1327492
[15] G. Malewicz, M. H. Austern, A. J. Bik, J. C. Dehnert, I. Horn, N. Leiser, and G. Czajkowski, "Pregel: a system for large-scale graph processing," in *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, ser. SIGMOD '10. New York, NY, USA: ACM, 2010, pp. 135–146. [Online]. Available: http://doi.acm.org/10.1145/1807167.1807184
[16] http://aws.amazon.com.
[17] S. Abdullah, N. D. Lane, and T. Choudhury, "Towards population scale activity recognition: A framework for handling data diversity," in *AAAI*, 2012.
[18] http://www.wi-fi.org/discover-and-learn/wi-fi-direct.

[19] M. J. Franklin, D. Kossmann, T. Kraska, S. Ramesh, and R. Xin, "Crowddb: answering queries with crowdsourcing," in *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, ser. SIGMOD '11. New York, NY, USA: ACM, 2011, pp. 61–72. [Online]. Available: http://doi.acm.org/10.1145/1989323.1989331

[20] A. Kittur, B. Smus, S. Khamkar, and R. E. Kraut, "Crowdforge: crowdsourcing complex work," in *Proceedings of the 24th annual ACM symposium on User interface software and technology*, ser. UIST '11. New York, NY, USA: ACM, 2011, pp. 43–52. [Online]. Available: http://doi.acm.org/10.1145/2047196.2047202

[21] http://bitcoin.org.

[22] M. D. M. F. Bulut, Y. S. Yilmaz, "Crowdsourcing location-based queries," *IEEE Workshop on Pervasive Collaboration and Social Networking (PerCol)*, 2011.

[23] M. Demirbas, M. A. Bayir, C. G. Akcora, Y. Yilmaz, and H. Ferhatosmanoglu, "Crowd-sourced sensing and collaboration using twitter," *11th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2010.

[24] http://ubicomp.cse.buffalo.edu/citypulse.

[25] C. G. Akcora, M. A. Bayir, M. Demirbas, and H. Ferhatosmanoglu, "Identifying breakpoints in public opinion," *Workshop on Social Media Analytics*, 2010.

[26] R. Baldawa, M. Benedict, M. F. Bulut, G. Challen, M. Demirbas, J. Inamdar, T. Ki, S. Y. Ko, T. Kosar, L. Mandvekar, A. Sathyaraja, C. Qiao, and S. Zawicki, "Phonelab: A large-scale participatory smartphone testbed," in *NSDI 9th USENIX Symposium on Networked Systems Design and Implementation*, ser. NSDI '12, 2012.

[27] http://birds.audubon.org/christmas-bird-count.