Indian Journal of Science and Technology, *Vol 9(19), DOI: 10.17485/ijst/2016/v9i19/53088, May 2016*

ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Mobile Phone Sensing: A New Application Paradigm

Shaukat Ali* and Shah Khusro

Department of Computer Science, University of Peshawar, 25120, Pakistan;
shoonikhan@upesh.edu.pk, khusro@upesh.edu.pk

Abstract

Background/Objectives: The main objectives are to present a comprehensive survey of mobile phones sensing applications by analyzing and classifying them in a unique chronological order, and identifying their features and shortcomings. **Methods/Statistical Analysis**: The literature focusing on using mobile phone sensors in applications is thoroughly tracked and selected for the study. The selected applications are analyzed and classified into different application domains using their features. A comparison framework is defined and comparisons are shown in tabular format. A generalized architecture for people-centric mobile sensing applications is proposed for increasing the potentials of researchers and developers, and saving them from being divided into separate islands. **Findings:** The integration of cheap yet powerful sensors have enabled the development of mobile phone sensing applications for solving problems in different domains which would be impossible otherwise. Analysis revealed that mobile phone sensing paradigm can be beneficent in dynamic capturing of information about different aspects of peoples' lives. However, the available solutions are mainly based on the distributed approach that uses a mobile phone as a gateway and rely heavily on external aids for capturing and storing information. This approach could be problematic in situations where supporting technologies are unavailable. State-of-the-art mobile phones are technologically advanced and are offering the same features and functionalities as personal computers. Mobile phone sensing applications should optimally exploit the potentials of mobile phones for effective solutions. Furthermore, the novel paradigm has a number of open issues and challenges that requires more contributions from the stack holders. The approaches and classification criterions used in the paper are unique and novel and providing more detailed and insight knowledge in an organized fashion as compared to the others. **Applications/Improvements:** This paper is aimed to provide a compact platform for researchers, especially the beginners, in understanding the phenomenon and diagnosing new research problems as well as finding solutions to the existing.

Keywords: Context-Awareness, Mobile Phone Sensing, People-Centric Sensing, Sensors, Smart Applications

1. Introduction

Mobile phones are highly ubiquitous computing devices with millions of mobile phones are in use around the globe to-date. Sales of smartphones (i.e., high-end mobile phones) are increasing each year, reaching to 353 million units sold by different vendors in 2015 with an increase of 15.5% in 2014 (shown in Table 1). Modern mobile phones are powerful communication devices having support of powerful hardware, operating systems, and applications for a variety of domains. The integration of sensors in mobile phones have changed their role from traditional communication devices into life-centric sensors[1]. To date, mobile phone sensing capability includes a rich set of sensors (e.g., GPS, accelerometer, proximity, gyroscope, digital compass, microphone, and ambient light sensors etc.) and a few more are expected to appear in the near future[2]. Using mobile phones as sensors can capture a broad range of dynamic information about people (e.g., physical activities etc.), global environment (e.g., noise level, transportation and road condition, and pollution level etc.), healthcare (e.g., obesity measurements, and cardiovascular diseases etc.), e-commerce (e.g., real-time price comparison and bargaining etc.), peer-to-peer communication (e.g., finding identity of peoples present in a proximity etc.), and indoor positioning and location-based services[3]. The captured information can be used locally for intelligent decision making (e.g., changing mobile phone profile

**Author for correspondence*

according to user context etc.) and disseminated over a network for others to take other real-world advantages (e.g., using a person's social network to find his current location etc.). The combination of mobile phones and sensors can be achieved in two ways: (1) embedded sensors are provided as integral parts of a mobile phone and accessed using APIs, turning mobile phone into a smart sensor, and (2) external sensors can be connected with mobile phones through wireless networking technologies (e.g., Bluetooth etc.) to provide sensory information which could be processed, analyzed, rendered, and forwarded to some remote places. In the future, it is believed that mobile phones will be surrounded by different external smart sensors of different tasks and architectures which would process captured information locally before sending to mobile phones[4]. Mobile phone sensors have several advantages over traditional wireless sensor networks[5,6] including power management, network formation and maintenance, economy of scale, covering hard to deploy geographical areas, capturing accurate information, and improving applications functionalities.

Empowering mobile phones with sensors has given rise to a novel area of research called mobile phone sensing also called participatory sensing[7]. Over the past few years, researchers and developers have leveraged mobile phone sensing capability for introducing a number of exciting people-centric mobile phone sensing applications to solve real world problems in a variety of domains (e.g., healthcare, road and transportation monitoring, security, environmental pollution monitoring, and social networking etc.) which would otherwise be impossible. This growing interest can be contributed to the technological advancements[8]: (1) the availability of cheap embedded sensors in mobile phones have made the creation of disruptive sensing applications possible, (2) mobile phones are open and programmable which eliminates the barriers of entry for third-party programmers, (3) mobile phones can provide converge of hard to deploy geographical areas and capture valuable information about different aspects of peoples' contexts and environments in real-time, (4) each mobile phone vendor has an app store allowing application developers to deliver their applications to large number of user across the globe, and (5) developers can upload mobile services to back-end servers on the cloud to enjoy high valued resources for computation of large scale sensory data and other advanced features. However, mobile phone sensing applications vary numerously such as sensing scale, sensors usage, data analytics, and domains. Furthermore, mobile phone sensing suffers with several issues and challenges which could severely affect the credibility of applications including[5]: (1) users typically put their mobile phones in their pockets or handbags and might not explicitly calibrate their sensors, (2) users spend most of their time indoors or in cars which might induce errors in the data collected while observing a phenomenon, (3) users are not frequent in charging their mobile phone and might at most do it once per day, (4) increased sensors sampling rate would result into increase of precision of the sensing system but at the expense of draining battery power more and more, and (5) privacy and security threats can exists by unattended leakage of users' personal information such as potentially sensitive graphical and voice contents (e.g., videos, audios, and images etc.), location, identity, and health etc.

To date, we have an extended list of mobile phone sensing applications developed by researchers, academia, and organizations which have not only turned mobile phones into people-centric sensing devices but also revolutionaries the field of applications development. These increasing

Table 1. Smartphones sold by different vendors to end users in 3rd Quarter of 2015 (thousands of units)[1]

Company	3Q15Units	3Q15 Market Share (%)	3Q14 Units	3Q14 Market Share (%)
Samsung	83,586.7	23.7	72,929.4	23.9
Apple	46,062.0	13.1	38,186.6	12.5
Huawei	27,262.3	7.7	15,935.0	5.2
Lenovo	17,439.2	4.9	21,314.1	7.0
Xiaomi	17,197.2	4.9	15,772.5	5.2
Others	161,296.6	45.7	141,246.5	46.3
Total	352,844.0	100.0	305,384.0	100.0

[1]http://www.gartner.com/newsroom/id/3169417

efforts have motivated us to combine, analyze, and present the existing important work in some chronological order in a compact document, highlight their important features and functionalities, and indicate the technological and non-technological shortcomings which could affect this new paradigm. Our main contributions of this paper include:

- To the best of our knowledge, the existing mobile phone sensing applications are tracked and analyzed. Their workings and functionalities are sharply highlighted and their main features are presented in tabular format.
- A novel general architecture for people-centric mobile phone sensing applications is proposed. Detail description of the architectural modules and their relationships are clearly articulated.
- A novel classification criterion is formulated for classifying and organizing the available applications into subcategories depending on their functionalities and problem areas.
- A number of technological and non-technological limitations in the paradigm are identified which could help researchers in finding new research dimensions.
- The existing knowledge is organized in a unique chronological order to be helpful for the researcher to find relevant knowledge in a single platform. Furthermore, the knowledge is presented to encourage new researchers by improving their understandings and motivate them to add their contributions in to the area.

2. People-Centric Mobile Phone Sensing

The concept of mobile phone sensing is general and applicable to a wide range of people's real world problems both in rural areas and urban areas. In a people-centric mobile phone sensing application, peoples are the focal point of sensing where information form peoples behaviors, actions, connections, and environments are gathered using mobile phone sensors and results are generated which could be used by other systems or peoples (i.e., family members, and friends etc.) for their respective benefits[9].

2.1 Participatory and Opportunistic Sensing

An urban landscape comprises a number of elements (e.g., people, tree, building, and vehicles etc.). The process of collecting data about urban landscape using sensors and making decision accordingly is called urban sensing. Mobile phone sensing is strongly related with urban sensing because of deployment of most of the mobile phones in the urban areas. However, they could have equal use in the rural areas too. Depending on the users' awareness and involvement in the sensing and decision making processes, urban sensing (similarly mobile phone sensing)can be classified into two types: participatory and opportunistic[8,10,11].

- **Participatory Sensing.** Participatory sensing emphasizes on the explicit active involvement of users in the sensing and decision making processes to determine that what and how to satisfy an application request and how one environmental information should be disseminated with other while abiding privacy constraints[10]. Participatory sensing can range from private personal sensing to sensing data from hundreds or thousands of peoples across a city. Compared to other devices, the promising characteristics of mobile phone make it an ideal tool for engaging participants in sensing their local environments. Using the built-in sensing capability and sensors connected wirelessly, mobile phone can provide an effective platform for building interactive participatory sensing networks enabling public and professionals to capture, analyze, and share local environmental information[12].
- **Opportunistic Sensing.** Opportunistic sensing automatically detects a state of interest and automatically changes device state to satisfy an application request. Opportunistic sensing makes data collection fully automatic and relieves users from the burden of explicitly decisions for sensing, data storing, and satisfying application request. Opportunistic sensing lowers the hurdles for applications to run in people-centric networks by leveraging the sensing capabilities of a device. Again inherent characteristics of mobile phone make it ideal tool for opportunistic sensing such as mobile phones are mostly accompanied by users enabling applications to automatically detect domains of interests and initiate sensing.

Participatory and opportunistic sensing are the two extremes in the design space of mobile phone sensing applications, each with their respective pros and corns[8]. Although opportunistic sensing helps in lowering user burden but keeps people underutilized and give raise to security and privacy problems. Furthermore, mobile phone might not be properly exposed to a context to take accurate samples of a phenomenon and might disseminate or leak personal sensitive information which a user does not want to[10]. Participatory sensing increases user burden and leverages human intelligence to tackle complex operations and solve mobile phone context and privacy problems. However, the quality of data being collected depends on the enthusiasm and reliability of the participants in the sensing process. Comparatively, participatory sensing is suitable for mobile phone sensing due to its support of large scale deployments and applications diversity[10].

2.2 Sensing Scale

Irrespective of participatory or opportunistic sensing, research community assumes people-centric mobile sensing applications to operate at multiple scales ranging from personal sensing to global sensing. At present, three distinct mobile phone sensing scales are identified: personal sensing, group sensing, and community sensing[8,11].

- **Personal Sensing.** A personal sensing application typically collects and analyzes data about a single individual. The collected data is provided to be consumed by the originating user only and do not share with others if no exception exists. For example, applications tracking a user exercises routine (e.g., UbiFit[13]), and a user obesity state (e.g., HealthAware[14]).
- **Group Sensing.** A group sensing application collects data about individuals having the common goals, concerns, and interests etc. and share sensing experiences with others freely or with privacy protections. This kind of applications will gain more popularity due to growing interest of the people in social networks or connected groups (e.g., friends, colleagues, and neighbors etc.). A typical example of group sensing application is CenceMe[15]).
- **Community Sensing.** A Community sensing application takes into account a large number of participants and represents a large-scale of data collection, analysis, and dissemination for the betterment of community. To scale largely requires cooperation from strangers who might not trust each other, requiring community sensing applications to have strong privacy protection and low commitment levels form the users. Typical applications of community sensing can create noise pollution map of a city (e.g., Laermometer[16], and NoiseTube[17] etc.), determining traffic congestion and road conditions (e.g., Nericell[18] etc.), and creating air pollution map of a city (e.g., N-SMARTS[5] etc.) etc.

2.3 Application Architecture

Although, mobile phone sensing applications might be architecturally different from each other but there are some functions and features which are essential for each application. Generally, a mobile phone sensing application is composed of four components: application query submission, device selection, sensor sampling, and data analysis, sharing, and presentation. A query application submits a query to mobile phone specifying required sensor(s) and condition(s) for sampling the sensor(s) data. The device samples sensor(s) data, analyzes samples, returns results to the querying application, and possibly shares the results with others depending on the issues of connectivity and privacy[10]. Architecturally, people-centric mobile phone sensing applications can be classified into two classes: distributed and integrated.

- **Distributed.** Distributed mobile phone sensing applications either use external sensors only or external sensors in conjunction with mobile phone internal sensors to capture a phenomenon. The captured information is provided to mobile phone via communication modules (e.g., Bluetooth or Wi-Fi etc.). Application's functional components are distributed across mobile phone and a remote web server. Low level data processings and storage are performed on mobile phone, whereas, information is transmitted to a remote server using Internet or cellular network protocols for high level data processings and storage. Distributed approach is preferred for mobile phones due to technological constraints (e.g., limited processing power, operating system, memory, and storage etc.), but can suffer from a number of problems which could ultimately degrade performance of an application such as (1) transmission delay can be introduced which is not tolerable by real-time monitoring applications, (2) data uploading can be problematic and can increase delays in areas where networks (e.g., Internet and cellular) availability cannot be ensured, (3) excessive data transmission can deplete battery quickly which is a scarce resource in hand held devices, and (4) data uploading can introduce complex security and privacy issues.
- **Integrated.** Integrated mobile phone sensing applications attempt to extensively exploit the functionalities

and potentials of mobile phones for sensing and processing tasks. Mobile phone internal sensors exclusively or in combination with external sensors are employed for capturing data and all of an application's functional modules (e.g., processing, inferencing, storage, and user-interface etc.) are executed on the mobile phone without requiring any external web-based supplement. Integrated approach provides whole of the features into a single package and overcomes the problems associated with distributed approach. Recent technological developments have proved mobile phone as a complete computing and communication platform, capable of performing complex and time consuming computations and processings. However, mobile phones technologically are still limited to process and store the excessive amount of data generated by sensors and execute complex inferencing and classification algorithms to derive conclusions. Therefore, improvements are needed technologically and non-technologically to overcome the problems.

2.4 General Architecture

Mobile phone sensing applications architecturally have no common consensus and vary greatly from each other. Applications are developed by developers in their own methodologies producing separate islands which would waste the potentials and technology. Therefore, we have proposed a general architecture capable of encompassing sensing from individual level to broader levels and is equally applicable for both distributed and integrated types of applications. This architecture can serve as standard for people centric mobile phones sensing applications to improve the applications' interoperability, components reuse, communications among applications, and minimize their heterogeneity. The architecture broadly divides a people-centric mobile phone sensing application into five primary modules (shown in Figure 1): information domain, sensing module, mobile phone module, web server module, and visualization.

2.4.1 Information Domain

An Information domain represents the area from which a mobile phone sensing application will extract its data. Main sources of information domains for people-centric mobile phone sensing applications could be the users' environments, contexts, activities, social relationships, and their physiological conditions etc. An application's information domain depends on the nature of its functionalities. A user's blood pressure monitoring system will extract its required information from the physiological conditions of the user, whereas, a road condition monitoring system and a weather condition monitoring system will extract their required information from the surrounding environment and contexts of the users.

2.4.2 Sensing Module

Sensing module represents the sensing components of a mobile phone sensing system. Sensors of varied nature which can be either external or internal can be used to deduce required pieces of information from their information domains by the sensing applications. Again, the types and quantity of sensors used by an application depends on the functionality of the application. A real-time ECG processing system would use external wireless ECG sensor, whereas, an old age fall monitoring system would use mobile phone internal accelerometer, microphone, and camera sensors. Sensing module might either start sensing information domain proactively or reactively in response to a command received from the sensing application running in the mobile phone module.

2.4.3 Mobile Phone Module

Mobile phone module represents the actual mobile phone platform where the sensing application will run. A mobile phone sensing application could perform a number of tasks while keeping in view certain limitations such as extracting sensed data from sensing module, performing different types of processing and inferencing to deduce new knowledge and accurate results using sensed data, storing sensed data and conclusion in some data storage mechanisms, and presenting the sensed data and derived conclusion in some meaningful form to the users in such away to not jeopardize or create unnecessary load on the mobile phone. Furthermore, a sensing application can also forward the sensed data and conclusions for further operations to a remote web server using Internet protocols (i.e., GSM, WiFi, GPRS etc.).

2.4.4 Remote Web Server Module

A remote web based server module represents an application running on a remote web server utilizing the sensed data and conclusions forwarded by the mobile phone

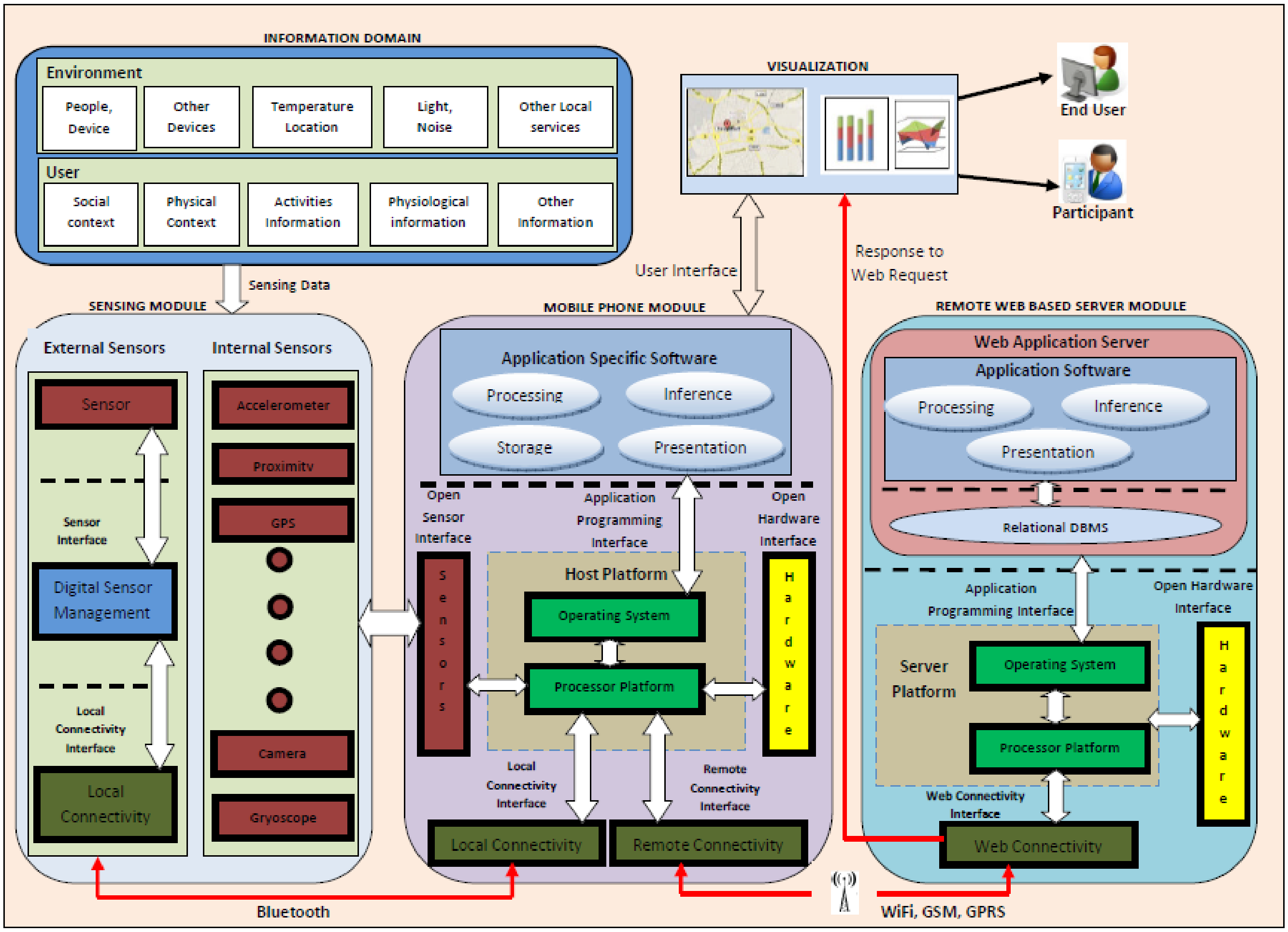

Figure 1. General architecture of people-centric mobile phone sensing applications.

module. Due to the underlying hardware and processing limitations, a sensing application at mobile phone module would not be able to carry out and execute complex operations and algorithms. Thus, to utilize the power of desktop hardware technology, such operations would have to be forwarded to the remote application. A remote application may perform functions such as processing and inferencing on sensed data, storing sensed data and derived conclusions in some RDBMS, and reporting sensed data and conclusion to end users using web-based protocols (i.e., HTTP).

2.4.5 Visualization

Visualization represents the way in which the processed sensory data and derived conclusions should be presented to the users. A visualization component can be either an integral part of sensing application to render information directly on mobile phone or end users can use third party applications (i.e., web browser etc.) to render information from remote web server on their computers using web-based interface.

3. Mobile Phone Sensing Application Areas

The ubiquitous nature and highly penetration of mobile phones in human lives have opened up new disciplines of applications development and facilitated human lives in different spheres. Using the mobility, sensing, processing, and communication capabilities of mobile phones a wide range of applications addressing a verity of domains areas are developed which would otherwise be impossible. In the recent past few years researchers and academia have contributed their efforts in developing high valued mobile phone sensing applications for a number of application areas. After studying the on-hand researches, the avail-

Figure 2. Application areas of mobile phone sensing applications.

able mobile phones sensing applications can be classified hierarchically into groups and sub-groups using their addressing domain areas. Figure 2 depicts the various application areas of mobile phones sensing applications.

3.1 Healthcare Monitoring

In the past few years, health monitoring using mobile phones alone or in conjunction with wearable devices has gained increasing interest of both research and academia[19], and laid foundation of a new research area called "Mobile Health Monitoring (MHM)". MHM has applications areas where peoples/patients health needs to be accessed and monitored on continuous basis such as old age, and handicap population. Most of MHM systems focus on pre-symptomatic testing and alerts peoples before any devastating situation take place. Researchers have shown that Obesity and heart rate disorder can be easily tackled with pre-symptomatic testing which can otherwise cause Cardiovascular diseases (CVDs) such as diabetic, stroke, hypertension, and heart diseases[14,20]. Similarly, cardiac arrest is unpredictable and is not possible for a person under attack to go to a medical institution, so witness has to perform CPR as soon as possible[21]. Experts have predicted that pre-symptomatic testing and timely CPR can save lives of millions of peoples and save millions of dollars in the coming decades[19,21], for example $117 billion dollars are used by US alone for only obesity related healthcare[14]. Using mobile phones as health monitoring devices can bring a number of potential advantages including[19]: (1) enables that early signs of health exteriorization can be easily detected, (2) in critical situations health care providers can be notified, (3) find correlation between lifestyle and health, (4) due to ubiquitous and prime source of communication in most of the population areas, health services can be provided to remote locations and developing countries, and (5) sharing real-time raw or interpreted physiological data, sense of connectedness with loved ones can be enhanced. However, to be advantageous, mobile phone health monitoring applications needs to overcome a number of technological, societal, and legal obstacles. For example, devices needs to be efficient in power, comfortable to wear, preserve privacy, non-intrusive, have user-friendly interface, have very low failure rate, and can trigger alarm accurately if used for diagnostic purposes.

3.1.1 Chronic Illness Monitoring

According to medical professionals, acute illness can be affectively treated in a hospital or office, but chronic illness (i.e., diabetics, chronic pain, cardiovascular disease, hypertension, asthma, and HIV etc.) requires a different methodology emphasizing majorly on patient self-management or environmental support. For example, a person under cardiac arrest attack would need immediate CPR from a witness[21]. A mobile phone assisted chronic illness self-management would be a participatory sensing system which would perform three functions. First, sensors would be used to collect data which could be helpful in understanding the possible causes of chronic illness. Second, data would be analyzed by medical professionals to prescribe an affective healthcare plan. Third, triggering alarm notification to protect a patient as early as possible. Table 2 depicts comparison of mobile phone sensing applications developed for chronic illness monitoring.

3.1.1.1 SPA

SPA[22] continuously monitors a user body, behavior, and environment during his/her daily life activities and notifies emergency responders to take appropriate actions in case of any emergency. SPA architecture is composed of three major components: (1) body area sensor network for collecting environmental and biomedical data, (2) a remote server for storing and analyzing sensors captured data, (3) group of healthcare professionals who could check records and advice healthcare suggestions. Body area sensor network is formed by attaching a number of biosensors and environmental sensors (shown in Table 1) for capturing and sampling environmental and biomedical data. Mobile phone is used as a base station to temporarily store and forward sensory data to remote server tagged with location information obtained from mobile phone's GPS sensor. Remote server stores sensory data in a formatted ware house and uses data mining algorithms to discover time-series patterns, rules, and correlation between environmental, biomedical, and location data. Health care professionals examines health records periodically to advice some follow-up suggestions to improve patients health. SPA mobile phone client prototype is developed in J2ME and remote server is developed in Java SE. Client is implemented on Nokia N95 and Nonin's Purelight 8000AAWO Adult Finger Clip Sensor is used to measure a user's pulse rate and blood oxygen saturation. SPA uses a variety of communication protocols to establish connections among its components. On body sensors communicates with each other and with mobile phone using Bluetooth, mobile phone communicates with remote server using either WLAN or cellular network and healthcare professionals uses Internet to access health records. SPA is succeeded in achieving accepted level of functionality and reliability. However, it can be extended to monitor additional environmental risk factors such as identify locations using GPS to persuade users in avoiding unhealthy locations (e.g., avoiding alcohol, drug use, and brothels etc.).

3.1.1.2 Mobicare Cardio

Mobicare[23] is a mobile phone based real-time ECG processing system to continuously monitor a patient's cardiac status anytime and anywhere by receiving ECG signals from the outer sensor unit, process them locally to identify any abnormality, and transmits information to emergency responders for taking necessary action, if any abnormality is detected. Mobic are architecture consists of a real-time ECG processing algorithm running on mobile phone, a wireless ECG sensor, web based server, a patients' database, and a user interface. Mobic are uses MPS450 multi-parameter simulator from FLUKE biomedical for raw ECG signals generation. The raw ECG signals are transmitted to Alive Technologies ECG sensor via two electrodes which fuses raw ECG signals with 3D accelerometer signal and transfer fused signal data to mobile phone via Bluetooth for processing and evaluation. Mobile ECG acts as a local processor and decodes the accelerometer and ECG data. Mobile ECG perform context-aware ECG processing, and if any abnormality found, ECG data will be send to web server over the cellular network (GPRS/3G). A special protocol called Medical Data Transmission Protocol has been designed for sending data from MobileECG to web server. Web server will route the data to hospitals or care centers which will be analyzed by the physicians for necessary actions such as sending an ambulance immediately or giving instructions to the individual via GPRS or 3G. Patients' database is developed to store all of the patients' personal particulars, clinical history, and all ECG data logs. MobileECG data is displayed graphically on the mobile phone. The graphical display shows ECG data and accelerometer data as graph, heart rate and QT interval as readable measurable values, arterial fibrillation, maximum heart rate, and body movement intensity. The system has been executed on Dopod 595 running with Microsoft Windows Mobile 5.0 and O2 XPhone II running with Microsoft Smartphone 2003 SE. Two separate experiments were conducted to confirm the validity of the framework. Results indicated that ECG signal processing has correctly identified the abnormal conditions, forwarded the detected conditions, and custodians can easily perform non-strenuous activities such as running, and stair climbing etc.

3.1.1.3 HeartToGo

HeartToGo[20] is a mobile phone-based wearable platform, which can continuously monitor and record ECG information in real-time, automatically identify irregular CVD conditions, generate individualized cardio health summary report in plain language, and classify abnormal CVD conditions on spatial and temporal basis. Heart To Go is an integrated mobile phone-based ECG application, which not only uses mobile phone capabilities to capture, store, display, and transmit ECG data in real-time, but also for other functional activities including identifying abnormal heart activities. Heart To Go prototype is implemented using Alive™ Bluetooth ECG and Heart Monitor; Microsoft Windows Mobile™ based mobile phone, and software packages including Microsoft Visual Studio™, MATLAB™, and Lab VIEW™. Alive wireless ECG and Heart Monitor is a light-weight, low-power and wearable 2-lead ECG sensing device, which is used to acquire ECG signals in real-time. ECG sensing device can record information at rate of 300 8-bits samples per second and also contains class 1 Bluetooth transmitter to send information to mobile phone or any other nearby wireless device. ECG processing module on mobile phone would capture transmitted ECG information, dynamically extract various important ECG features, and generate an ECG summary report. The format and information presented in the ECG summary report will be quite similar to the one that is produced by the stationary ECG machine being used by the cardiologists in the hospitals. Rule-based and Hybrid Fuzzy Network based neural classifiers are used to identify ECG features and searches both existing ECG database (e.g., MIT-BIH arrhythmia database etc.) and individual's own personalized ECG database for finding potential matches for any CVD condition. Prototype's user-interface is written in Microsoft Visual C++ and has an optional open source EP Limited arrhythmia detection software package based plug-in. Rules are added to the plug-in for the detection of conditions: sinus bradycardia, ventricular flutter, and left bundle branch block. User-interface shows heart rate updated in real-time, and classifies the beats with "N" as normal beat and "V" as PVC beat. Because of the limited number of features available in EP Limited plug-in, additional detection algorithms are developed, tested and verified in MATLAB. Lab VIEW-based platform is developed for rapid development of algorithms prototypes. The algorithms implemented in MATLAB are tested on the Window Mobile based mobile phone using Lab VIEW Mobile Module for accessing their performance for deploying in real-time mobile environments.

3.1.2 Obesity Monitoring

Obesity has been emerged as an epidemic phenomenon and its ratio is higher in developed countries as compared to under-developed countries. Obesity rate is higher in USA and according to 2009-10 survey report 35.7% of adults, and 16.9% of children and adolescents were suffering from obesity problem[24]. Obesity unawareness can result in severe health consequences including heart diseases, diabetes, and stroke etc. Ignoring precautions in food intake and regular exercises are the major reasons of getting obese problem. Most of the obesity prevention programs concentrate on educating people about healthy eating and physical activities awareness in their daily lives. Leveraging the ubiquitous nature and embedded sensors of mobile phone, researchers have augmented that mobile phone can be an ideal candidate for peoples' physical activities and food intake monitoring. Table 3 depicts comparison of mobile phone sensing applications developed for obesity monitoring and controlling.

3.1.2.1 Health Aware

HealthAware[14] is obesity prevention application, which makes people aware of the relationships between their health, food intake, and daily physical activities. Health Aware measures intensity of users' physical activities in real time and persuade them that how much physical activities are needed to reduce their obesity and remain healthy. Health Aware is integrated obesity control using mobile phone internal sensors and combining all the functionalities within a single mobile phone. Health Aware application is composed of four components: user-interface, on-device database, physical activity analysis, and food-items classification system. User interface displays count of walking steps as well as running steps taken during daily physical activities, number of walking steps or running steps needed to burn the taken food items, time of last activity completed, user current location, and any alarming message set by the application. To enable a mobile phone to serve Health Aware functions, an on-device database is maintained to store information related to food items such as names, pictures, and calories etc. along with other information entered by a user such as data related to physical activities and food items intake. The physical activity analysis works in the background and analyzes accelerometer readings to recognize users' physical activities. Accelerometer counts the number of walking and running steps of user. Information about location (i.e., from GPS), time, and quantified movement information are stored in database to be used by other functional components. The food-items classification system extracts metadata from the incoming pictures (i.e., from camera) and uses them to index the database. The pilot prototype is developed in C++ using Windows Mobile SDK 6. The target code is tested on HTC Touch Diamond Device running with Windows Mobile 6.1 Professional operating system. Results indicated that application successfully differentiated and quantified walking and running activities under various moving intensities in real-time.

3.1.2.2 BALANCE

BALANCE[25] provides an easy way to visualize the balance between the caloric expenditure and caloric intake, minimize users' cognitive overload, and provide real-time feedbacks as well as encouragements. BALANCE makes use of and provides an easy interface enabling users to customize and enter food entries. BALANCE uses Mobile Sensing Platform (MSP) sensors to measure caloric expenditure. Users' physical activities including: sitting, walking, running, and bicycling are recognized using MSP multiple sensors and inference engine. In BALANCE implementation, accelerometer data is used to perform activity inferencing (walking and running only) and compute steps after confirmation. The computed information is transmitted to mobile phone using Bluetooth. BALANCE user interface allows users' to enter consumed foods and activities which could not be detected automatically, and presents a quick summary of users' current caloric intake and caloric expenditures. BALANCE uses two primary food databases: master database and personal database. Master database containing detailed nutrition information and personal database contains food entries which users' have ever enjoyed. To make a food entry, user has to first select it from master database which will be automatically copied into personal database for making future selections easy. Furthermore, to facilitate food entry, the application offers "Favourities" list where a user can enter several food items under a single food item or meal heading. BALANCE is implemented on Nokia N95 mobile phone running with Symbian OS v9.2. The calculated caloric expenditures values are compared with the estimated caloric expenditure values from American College of Sports Medicine's Guidelines for Exercise Testing and Prescription. Results indicated that calculated values are at little lower than the estimated values.

Table 2. Comparison of mobile phone sensing applications developed for chronic illness monitoring

App	Hardware	Software	Protocols	Sensors	Type	Product	Shortcomings
SPA[22]	Nonin's Pure light 8000AAWO Adult Finger Clip Sensor, Nokia N95, and Remote Server.	J2ME, Java SE, Symbian OS v9.2, Nokia S60 UI	Bluetooth, WiFi, Cellular Network (GPRS/3G)	Pulse Oximeter, Blood Pressure Meter, Actigraph, Temperature, Humidity, Light, GPS	**Architecture** Distributed **Operation** Participatory **Scale** Personal	Prototype	Communication is subjected to mobile phone and network availability, and energy constraints which could introduce delays. Delays cannot be tolerated by real-time monitoring system.
Mobicare[23]	MPS450 multi-parameter simulator from FLUKE biomedical, Alive Technologies ECG sensor, Dopod 595 mobile phone, O2 XPhone II mobile phone.	ECG processing algorithm , Microsoft Windows Mobile 5.0, Microsoft Smartphone 2003 SE.	Bluetooth, Cellular Network (GPRS/3G)	Wireless ECG sensor, 3D Accelerometer	**Architecture** Distributed **Operation** Participatory **Scale** Personal	Pilot Prototype	Approach is practical for real-time transmission of abnormalities to remote web server but do not provide solution for situation where cellular network might not be available.
HearToGo[20]	Alive™ Bluetooth ECG and Heart Monitor, Microsoft Windows Mobile™ based mobile phone.	Microsoft Visual Studio™, MATLAB™, and LabVIEW™	Bluetooth.	Alive wireless ECG and Heart Monitor.	**Architecture** Distributed **Operation** Opportunistic **Scale** Personal	Pilot Prototype	Identify only a few conditions. Advanced mechanisms are needed to be developed for identifying and extracting information related to other conditions.

3.1.3 General Health Monitoring and Persuasion

In most of the developed and developing countries peoples often have sedentary lifestyle. According to the World Health Organization (WHO), the proportion of the world's population having lack of physical activities ranges from 60% to 85%. According to medical professionals, physical activities can reduce chances of premature mortality, type II diabetics, coronary heart diseases, osteoporosis, colon and breast cancer, obesity, high blood pressure, lipid disorders, depression, and anxiety[13,26]. Recent advancements in technologies can play an important role in promoting and persuading peoples' to adopt active lifestyle that is a lifestyle which includes physical activities, healthy life choices, and sports etc.[13,26] Researchers have used sensing modules quite effectively to infer activities in fitness domain but the challenge is which of the physical activities should be identified[13]. According to American College of Sports Medicine recommendation, a physical activity should include cardio respiratory trainings, resistance trainings, and flexibility trainings. Therefore, to encourage physical activities for obtaining physically active lifestyle, technologies should support a range of activities instead of concentrating on a single activity such as walking etc.[13]Table 4 depicts a comparison of mobile phone sensing applications developed for general health monitoring and persuasion.

3.1.3.1 TrippleBeat

TrippleBeat[26] argues that motivation can be key element for health monitoring systems in assisting people for having changes in their behavior to improve or maintain their health conditions. The application helps users in a number of ways including: provide interface for users to specify cardiovascular goals as per their desires, guide users during their exercises using real-time musical feedback, establishes a virtual competition to improve users' motivations, and provides glanceable interface to display relevant information and recommendations about their exercises in a easy-to-understandable way. TrippleBeat sensing module consists of sensors, processing board to receive and digitize raw sensor data, and Bluetooth transmitter to transmit digitized data wirelessly to mobile phone running with Tripple Beat. TrippleBeat determines user speed and heart-rate information after processing the raw physiological and acceleration data of the sensors. After understanding the situation, the system guides users for achieving their desired goals though selecting and playing a song form the Digital Music Library (DML). The DML can store 70 songs of different genres and tempo in the phone. A song with fast tempo than the current one indicates users to speed up, slow tempo indicates users to slow down, and with the same temp indicates users to go with the same pace. TrippleBeat uses two persuasive techniques to motivate runners for achieving their exer-

Table 3. Comparison of mobile phones sensing applications developed for obesity monitoring.

App	Hardware	Software	Protocols	Sensors	Type	Product	Shortcomings
HealthAware[14]	HTC Touch Diamond Device.	Windows Mobile 6.1 Professional , Windows Mobile SDK 6, C++	None	Camera, GPS, and Accelerometer.	**Architecture** Integrated **Operation** Participatory **Scale** Personal	Pilot prototype system.	Application can jeopardize resources constraint mobile phone by executing all of the complex activity recognition, picture analysis, and classification algorithms in a single place.
BALANCE[25]	MSP box consisting of multisensor, and storage, Nokia N95.	Symbian OS v9.2 with S60 3rd Edition user interface	Bluetooth	MSP box : 3-D accelerometer, barometric pressure, humidity, light sensors, and GPS	**Architecture** Distributed **Operation** Participatory **Scale** Personal	Pilot prototype system.	Mobile phone sensors and accelerometer in conjunction with other sensors should be used for effective real-time activities inferencing. In addition to walking and running other daily life activities should be considered as well. Cognitive burden over users of entering of food items should be reduced.

cises goals: (1) establishing a virtual competition with other runners, and (2) providing real-time awareness using glanceable interface. There are two ways to select competitors: (1) manual selection, where a user explicitly selects its competitors, and (2) automatic selection, where TrippleBeats uses its database to automatically select the competitors. Virtual competition is used with ultimate goal of encouraging users to achieve their exercise goals in a healthy way. Glanceable interface displays visual information in real-time in an intelligent manner which should fit the small screen of mobile phone and could be easily read by a user with low cognitive effort while he/she is running outdoor. TrippleBeat is implemented on Audiovox SMT5600 (HTC Typhoon) or Cingular 2125 (HT Faraday) mobile phones and tested with real world participants. Results indicated that TrippleBeat is considered more effective by the runners in achieving their desired goals. The effectiveness of TrippleBeat is attributed to its glanceable interface for maximized personal awareness, providing real-time recommendation, and a virtual competition with other runners.

3.1.3.2 UbiFit

UbiFit[13] is designed for people, who wants but cannot or not consistently incorporate regular physical activities into their everyday lives. UbiFit has drawn its features from a number of its predecessor projects including encouraging and inferencing physical activities, correcting inferencing mistakes, engaging users, providing trending information and positive reinforcements, providing opportunities for self-reflections, and integrating use into everyday life. The UbiFit Garden system is composed of three components: (1) fitness device, (2) interactive application, and (3) glanceable display. The fitness device automatically finds physical activities and passes relevant information to glanceable display and interactive application. The interactive application adds detailed information with the inferred physical activity and an interface where to add, delete, or edit activities. The glanceable display remains as a wallpaper in the background screen and uses the metaphor of garden to represent physical activity and goal attainment for motivation. The UbiFit Garden application uses Mobile Sensing Platform for activities sensing and inferencing. The MSP uses 3-D accelerometer, barometer, and classifiers to infer activities (i.e., walking, cycling, running, and using stair machine etc.), and uses Bluetooth to communicate inferred activities to mobile phone running with interactive application and glanceable display. The interactive display is built using My Experience framework, which communicate with MSP, convert sensed data into human-scale interpretation, store activity data, manually journalize and personalize activities, and update the glanceable display. The glanceable display is a wallpaper image uses metaphor of garden (e.g., fresh flowers, butterflies, stormy sky, blue sky, healthy grass, and wilting flowers etc.) to provide information about weekly goal attainment status, physical activity behavior, and a subtle and persistent reminder of commitment to physical activities. UbiFit is tested using real-world participants and results have shown that UbiFit Garden can support a variety of activities recommendations of American College of Sports Medicines (ACSM). Activities are correctly inferred and journalized during testing duration. It has been conceived that by allowing users to manipulate and add to inferred data, the usefulness and credibility of the system is improved.

3.1.3.3 StudentLife

StudentLife[27] is a continuous sensing application to access mental health, academic performance and behavioral trends of a student. Student Life architecture is composed of two parts: mobile phone and cloud. Sudent Life captures data from mobile phone sensors (i.e., accelerometer, proximity, microphone, light, GPS), and application usage. The application automatically infers activity (i.e., stationary, walking, running, driving, and cycling), sleep duration, and sociability by extracting features from the preprocessed accelerometer stream and using physical activity classifier (i.e., decision tree) on the extracted features. Outdoor mobility is identified using GPS samples and indoor mobility is identified using WiFi scan logs. Two classifiers are used for detecting conversation: an audio classifier for to infer human voice and a conversation classifier for to detect conversation. Audio classifier process audio on the fly and extract features and a two-state Hidden Markov Model (HMM) is used to infer speech segment. Output of the audio classifier is input to the conversation classifier to capture the number of independent conversations and their durations. The frequency and duration of conversations is used to measure sociability of a user. Unsocial conversation (i.e., lecture and x-hours etc.) are filtered using location information and removed from the dataset using conversation classifier extended on the cloud. Sleep model is constructed using a sleep classifier. The sleep classifier extracts four types of features: light feature, phone usage feature, activity feature, and sound feature. Sleep model combines theses features for an accurate sleep model. MobileEMA is used to capture additional human behavior which could not provided by surveys and automatic sensing. MobileEMA is integrated into the StudentLife application based on extensions to Google PACO. The inferences and other sensor data are temporarily stored on the phone and are efficiently uploaded to the Student Life cloud when users recharge their phones under WiFi. Student Life is deployed on Google Nexus 4 Android phones and a vast dataset is collected from several students. Results indicate that automatically sensed conversation, activity, mobility, and sleep have significant correlations with mental well-being outcomes. Strong correlations between academic performance and automatic sensing data and mental well-being are also observed.

3.1.3.4 SmartWalker

SmartWalker[28] is an automatic ambulation tool to help elderly and visually impaired people to move around and be supervised at the same time. The system consists of two parts: local system and remotely controlled system. Local system is responsible for detection and notification of obstacles to user and remotely controlled system receives video stream and heart beat value of the user. Four infrared sensors are used for detection obstacles in the vicinity in each direction (i.e., left, right, front right, front left). Pulse sensor is used for providing data about heart rate of a user. Infrared sensors and pulse sensor are connected to the inputs of the Arduino Board. ISD2590 integrated circuit is used for recording sounds which would be used for communication with visually impaired peoples. Playing and stopping of recorded sound is controlled by the infrared sensor output. Ethernet Shield is inserted on top of the Arduino Board for transferring pulse sensor value to router to be read on the Android application on user mobile phone. Android application captures real time data from IP camera installed on the walker in addition to the emergency stop button. A 3G router is also installed on the walker for remotely monitoring of the user and all his vital conditions. Smart Walker is tested and found that all of its components are functioning effectively. However, weight of the walker and power supply are the two significant issues needed to be solved.

3.2 Environmental Pollution Monitoring

Environmental pollution can cause a lot of damage to humans, animals, plants, and trees. In humans, environmental pollution cannot only cause physical disabilities but also physiological and behavioral disorder. Figure 3 depicts the problems caused by the different types (i.e., air pollution, solid pollution, noise pollution, and water pollution etc.) of environmental pollution in humans. Several of the systems have deployed static sensor network nodes in specific local regions for collecting environmental parameters which could be stored in a remote central database. But, the large scale deployment of sensors faces with price, energy, bandwidth, and computational speed constraints problems. Mobile phone provides the potential of environmental monitoring for determining and monitoring the pollution levels and leverages novel services. A mobile phone has certain amazing characteristics such as co-located with a user, being in environment, and sensing and communicating contextual data etc. which can be effectively exploited to determine environment impacts on individuals, communities, cities, and even more globally and vice versa[5]. However, using mobile phones for environmental monitoring will give rise to new challenges and would create a novel research field in pervasive computing. Table 5 depicts a comparison of mobile phone sensing applications developed for different types of environmental pollution monitoring.

3.2.1 Noise Pollution Monitoring

Noise pollution includes noises of aircrafts, vehicles, vehicle horns, loudspeakers, industries, and high-intensity sonar effects which are extremely harmful for the environment. Noise pollution can cause a lot of damages to humans and can be major source of physiological and behavioral disorder. Mobile phone sensing capabilities in coordination with other external sensors can be effectively used for collecting noise pollution information for using in useful applications such as creating noise maps for helping patients having problems with noise pollution, and helping authorities to eliminate noise pollution in a region etc.

3.2.1.1 MobGeoSen

MobGeoSen[29] is developed for determining the noise pollution levels in an environment. MobGeoSen uses mobile phone internal sensors in conjunction with

Table 4. Comparison of mobile phone sensing applications developed for general health monitoring and persuasion

App	Hardware	Software	Protocols	Sensors	Type	Product	Shortcomings
TrippleBeat[26]	Alive TecECG, Audiovox SMT5600 (HTC Typhoon) or Cingular 2125 (HT Faraday)	Windows Mobile Standard 2003 Second Edition, Microsoft Windows Mobile 5.0	Bluetooth	2-lead ECG sensors, 2-adhesive electrodes, 3-D Accelerometer	**Architecture** Distributed **Operation** Opportunistic **Scale** Personal	Pilot Prototype	The system emphasizes on using external sensors instead of mobile phone internal sensors. Runner performance depends on the amount of time spent in the training zone, which is not correct because more calories can be burnt in small time run with a higher speed.
UbiFit Garden[13]	MSP box, Mobile phone	My Experience frame-work	Bluetooth	3-D Accelerometer, Barometer	**Architecture** Distributed **Operation** Participatory **Scale** Personal	Pilot Prototype	Using of MPS put extra burden on the users and makes the system impractical during daily life activities. Some activities such as swimming, kayaking etc. cannot be inferred because of requiring water proof casing for MPS box.
StudentLife[27]	Google Nexus 4	Survey Monkey, Mobile EMA Framework, Google PACO Framework	WiFi	Accelerometer, GPS, Microphone, Proximigy, light	**Architecture** Distributed **Operation** Opportunistic **Scale** Personal	Pilot Prototype	Sensing and exaction of large number of classifiers could jeopardize mobile phone. No security protocol is defined for transferring data from mobile phone to cloud. Data access mechanisms and policies from cloud need to be defined.
Smart Walker[28]	Arduino Board, IP Camera, 3G Router, ISD2590 Sound Recording Circuit	Android	WiFi	Infrared, Pulse Sensor	**Architecture** Distributed **Operation** Opportunistic **Scale** Personal	Prototype	Technical functionalities of the application are not defined. Walker is heavy due to integrating all components in one place. Obstacle detection accuracy depends on the distance between infrared sensors and objects. Efficient power supply is needed to remove heavy wiring.

external wireless sensors for data collection and allows users to annotate data at the time of collection to visualize data spatially and temporally. Mob Geo Sen architecture is composed of three core Bluetooth-enable devices for capturing, storing, displaying, and communicating data (i.e., environmental, contextual, and geo-position). These devices are Nokia series 60 camera phone, sensor data logger, and a GPS receiver. In the prototype Bluetooth-enabled Haicom 406B GPS is used and Science Scope Logbook WL data logger is used, which has LCD display, and can be connected with up to four sensors simultaneously for collecting multitude of data such as temperature, carbon monoxide, electric field strength, and acceleration. Application queries the datalogger, GPS, and the mobile phone built-in microphone periodically and return with data for processing and storing. Users can use mobile phone's camera to take geo-referenced photos to annotate data (in addition to text) during collection which can be rendered automatically along with environmental data using Google Earth. Master mobile phone can initiate multiple Bluetooth connections with slave datalogger and GPS devices to

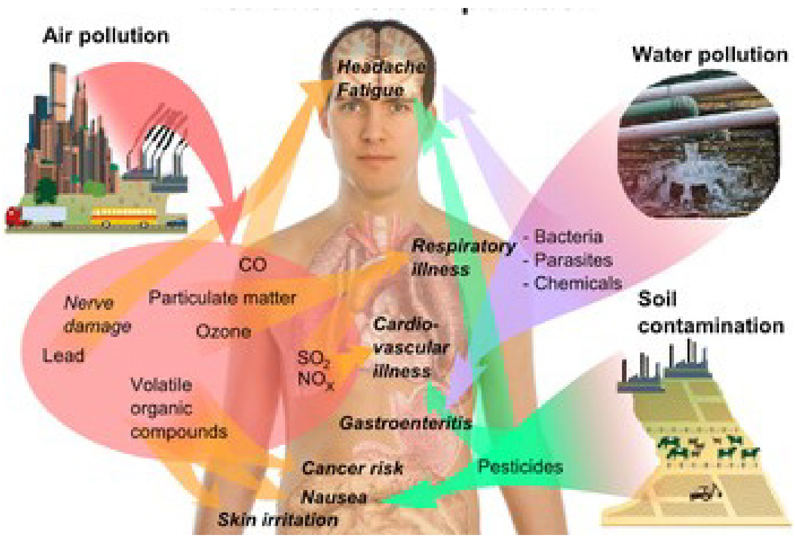

Figure 3. Problems caused by the different types of environmental pollution in humans.

read data. Application can turn on microphone to access the noise information from the surrounding environment and combines the sound data with the GPS location data to generate a map of the sound levels encountered during a journey. The recorded environmental and location data is saved in a time stamped KML file and user generated annotation (photos and text) are linked with the KML file. KML file can be directly loaded into Google Earth visualization tool. MobGeoSen is tested using real-world participants to collect, manipulate, share and visualize noise and carbon monoxide pollution data in their environment. All of the participants found the application very much interesting and fruitful. But, few of the problems were identified: (1) battery life of mobile phone is reduced quickly due to continuous Bluetooth connection and communication with slave devices, (2) some issues related to GPS and Bluetooth connectivity were identified, and (3) storage capacity of mobile phone is found insufficient for storing tones and tones of sensory data.

3.2.1.2 Laermometer

Laermometer[16] is a community-based mobile phone noise mapping application and used to solve the problem of creating noise maps. Laermometer application consists of two parts: client-side and server-side. Client-side consists of mobile phone application responsible for a number of functionalities such as recording sound, different visualizations of sound points, administrating comments, and viewing the noise maps at different points of time via the timeline etc. Server-side is used to store, upload, and retrieve sound points. In addition, a web interface is provided with functionalities such as viewing and editing the data, adding and editing comments, user profiles, noise descriptions, and bookmarks. Once installed, Laermometer uses mobile phone built-in microphone sensor to capture noise information from the environment automatically and forwards captured data to remote server at pre-determined time. The application runs in the background and does not require explicit active users' interaction. GPS module of the mobile phone (either internal or external) is used to capture the device position. To combine GPS coordinates with sound levels, Laermometer utilizes geo-tagging and uploads them to a remote database server. Apache server has been used for controlling the communication between mobile client and server, and MySQL database is used for storing all users' data and sound information. Information between the server and the mobile application are exchanged in XML format. An automatic upload interval can also be determined to define the time after which the collected sound levels should be uploaded to server. The application is automatic requiring users only to upload comments actively. Laermometer has two main advantages: (1) it is based on multi sound levels recordings and is more precise than modeling based systems, (2) it imposes no extra cost due to using existing mobile phone hardware. Yet, it has the lack of identifying the exact placement of mobile phone, which could lead to uploading a wrong noise level. In addition, the different qualities of the microphones can also affect the quality of the recorded sounds. Laermometer is planned to be evaluate for its real-time usage and quantitative information about its quality and any further enhancements would be identified from participants equipped with Laermometer enabled devices. In everyday usage, data about quality of sound levels can also be obtained to determine how far usage can influence sound levels and how to figure out ways to stop them.

3.2.1.3 NoiseTube

NoiseTube[17] is developed with the goal of turning GPS-equipped mobile phones into noise sensors and incorporate users carrying them in the sensing process by enabling them to determine, locate, provide qualitative input, and share their geo-localized measurements as well as personal annotations for the monitoring of urban noise pollution and the production of a collective noise map. NoiseTube prototype consists of two components: client side mobile phone sensing application and server side application. After installing mobile phone sensing application on a GPS-equipped mobile phone, the application collects information from different sensors (e.g., microphone, GPS, time, and user input) and uploads them to NoiseTube server where the data is processed and centralized. A real-time digital signal processing algorithm measures the loudness level of the environmental sound recorded by microphone at a chosen interval and A-weighting filter is used to compute an equivalent sound level in dB (A) from the recorded sounds. To make the noise map more meaningful for both citizens and decisions makers, users can annotate the calculated sound loudness with tagging to specify source and precise location of a noise. The client mobile sensing application is written in Java and intended primarily for running on mobile phones running with Symbian S60 operating system. Actually the application is tested on Nokia N95 8GB mobile phones but is expected to run on any mobile phone providing support for Java J2ME platform with multimedia and localization extensions. A GPS receiver both built-in and external is needed to localize measures. The server side application is implemented using MySQL, Ruby on Rails, Google Earth, and Google Map. A Web API is provided allowing scientists and developers to access individual or collective noise experiences data for creating useful applications (e.g., web mashups etc.) or analyzing for scientific purposes. The system has been experimented to determine the precision of the loudness computed with the mobile phone as compared to sound level meter. After applying the inverse function as a post-processing corrector, the results obtained were almost as expected. Using Nokia N95 8GB built-in GPS receiver, error up to 30 meter in position accuracy is detected, which is considered acceptable in localizing noise sources within a specific street or neighborhood. A slight improvement in position accuracy is observed while using external GPS receiver.

3.2.2 Air Pollution Monitoring

Air pollution (i.e., both indoor and outdoor) has become a serious problem for both developed countries in general, and developing countries in particular due to their significant impacts on health of the inhabitants especially the people[5]. The World Health Organization (WHO) estimates that indoor pollution causes approximately 2 million premature deaths annually in developing countries and outdoor pollution causes approximately 1.3 million deaths worldwide annually. Leveraging mobile phones sensing capabilities enables users to track multiple data points and collect dynamic information about ambient air quality. In developing countries using mobile phones for air pollution monitoring can be of potential benefit for the people in a number of ways (5): (1) help policy makers in impacting environmental policies, (2) raising peoples' awareness will change their willingness for adopting new technologies to benefit their health, (3) leverages economy of scale because of their production in large quantities, and (5) help regions under development due to detecting high level of air pollution with low-cost sensors.

3.2.2.1 N-SMARTS

N-SMARTS[5] is proposed with four major objectives: (1) building of system architecture which should be scalable to millions of users, (2) designing of efficient algorithms which could provide accurate estimations of both the pollution levels and other sensed data, (3) designing of effective algorithms to accurately detect users' behaviors, and (4) building and assembling a suite of useful sensors to integrate. A commercial off-the-shelf sensors platform has been build for developing and testing algorithms which forms the core components of N-SMART application. The integrated platform consists of a number of off-the-shelf sensors (i.e., CO and NOx sensors, a temperature sensor for calibration, an accelerometer for activity inferencing, and a Bluetooth radio for communication with the phone), data loggers (i.e., a NO2, SO2 or O3 data-logger from BW Technologies and a Lascar EL-USB-CO Carbon Monoxide data logger), and a flexible power supply for using with a number of power sources. The integrated platform can be either automotive platform or personal platform. An automotive platform can be mounted externally near car window and a personal platform can be worn on user belt. The data collected by automotive platform is uploaded to database which cannot only be viewed in several formats but can also be overlaid on Google Map. The integrated platform communicates with mobile phone using Bluetooth. A dual board version is also devised to split sensors onto a daughterboard to mechanically separate control circuitry from the sensing unit. The integrated platform is designed in such away to work with any type of mobile phone having integrated GPS. For experimental purpose the application is tested with Nokia N95 mobile phone. A test chamber is designed for testing and calibration of the application. The chamber allows for calibrating sensors as well as performing experiments of different characteristics of the used sensors. The test mechanism consists of pure, dry air source, dry toxic gas source, humidifier, constant rate flow controller, two voltage controlled flow controllers, DA converter to drive the flow controllers, and AD converter to take readings from the flow controllers and any analog sensor used. This test mechanism has been found valuable in systematically testing a range of sensor devices under various conditions.

3.2.3 Visibility Monitoring

Visibility in urban metropolitan areas is greatly degraded by airborne particulate and is a severe threat to health, driving safety, tourism, and environment. In addition, this particulate matter is also cited as a key reason of global warming[30,31]. Haze aerosols are the major atmospheric pollutants which strongly effect visibility. Understanding the importance of air visibility monitoring from health and environment point of views, some of the monitoring stations are thinly deployed based on human observers, optical instruments (i.e., photometers etc.), or chemical sensors (i.e., integrating nephelometers etc)[32]. Although, these systems are quite precise but suffers from subjectivity, expense, and maintenance problems which makes them impractical for developing countries. To deploy air visibility monitoring sensing systems at a large-scale and augment them with the existing infrastructure, off-the-shelf sensors in mobile phones can be used for a number of reasons[32] such as mobile phones proliferation in human society, and human assistance for intelligent data gathering.

3.2.3.1 Visibility Monitoring

A visibility measuring system has been presented that uses optical technique (i.e., camera) and other common sensors found in mobile phones[32]. The application works by requiring a user to point mobile phone to the sky and take an image. The image is tagged with accelerometer, magnetometer, and time information into a tag file automatically. To compute the current solar position, the tag file is also appended with GPS and time information. The tag file along with the image is send to backend server which estimates visibility and returns a value of turbidity to user. Files (i.e., image file and tag file) are transferred using FTP standard protocol from the client program running on mobile phone to FTP server process running on backend server. During this time, the application switches to the background while listening for response from backend server. Upon receiving turbidity estimates, the application displays estimates using notification along with time of image capture. In addition, GPS coordinates can be transferred to backend server as well for displaying visibility information over a map to be shared with other users. The system has been implemented on HTC G1 mobile phone running with Android OS. The significant computational capabilities of the mobile phone make it an ideal platform for instantaneous computation of solar position. In addition, its support MicroSD card for providing potential for storing camera images and tag files. 3 axis magnetic sensor and 3 axis accelerometer data is combined by Android API to identify 3D orientation of mobile phone. Backend consists of FTP and HTTP servers running with MATLAB on desktop computer and communicate with mobile phone over Internet. A Perl script is executed on the backend server as a result of HTTP request. The script initiates image processing modules written in MATLAB for computing image luminance. The resulting visibility values are sent back to the mobile phone as a response to HTTP request. Privacy, communication cost, non-blocking (not jeopardizing mobile phone), taking human assistance, and energy efficiency are the primary factors considered in the design of the application. The application has been tested by

Table 5. Comparison of mobile phone sensing applications developed for environmental pollution monitoring

	App	Hardware	Software	Protocols	Sensors	Type	Product	Shortcomings
Noise Pollution Monitoring	MobGeoSen[29]	Nokia S60	Google Earth	Bluetooth, GPRS	Camera, GPS receiver, Microphone	**Architecture** Distributed **Operation** Participatory **Scale** Community	Prototype	Mobile phone is used for creating KML file from collected sensory information and no specialized rendering application is developed for mobile phone. Visualization is completely dependent on the external PC through loading KML file into it.
	Laermometer[16]	Any GPS equipped mobile phone	XML, MySQL, Apache web server	GSM	Microphone, GPS	**Architecture** Distributed **Operation** Participatory **Scale** Community	Prototype	Mobile phone is used for data capturing and forwarding and all of the other activities are performed on the server. Other problems includes that it cannot accurately determine exact placement of mobile phone which could lead to recording wrong recordings to noise level, and the quality of recordings can vary due to different qualities of microphones.
	NoiseTube[17]	Nokia N95 8GB	Java, Symbian S60 OS, MySQL, Rubyon Rails, Google Earth and Map	Bluetooth, GSM	GPS, Bluetooth, Microphone	**Architecture** Distributed **Operation** Participatory **Scale** Community	Prototype	Mobile phone is used as data capturing and forwarding device, whereas, all of the other data activities such as updating, and editing etc. are performed on the server.
Air Pollution Monitoring	N-SMARTS[5]	Nokia N95	Google Map, Database	Bluetooth	GPS, CO and NOx sensors, Temperature sensor, Accelerometer sensor	**Architecture** Distributed **Operation** Opportunistic **Scale** Community	Prototype	Instead of using mobile phone sensors, an external integrated unit of sensors is used. Data processing operations and storage is also performed using the integrated unit and mobile phone is only used for displaying results received from the integrated unit. The system has several problems regarding identifying users' locations and their behaviors.
Visibility Monitoring	Visibility Monitoring[32]	HTC G1, Backend Server	Android API, MATLAB, Perl, Google Map	HTTP, FTP, GSM, GPRS	Accelerometer, Magnetometer, GPS, Camera	**Architecture** Distributed **Operation** Participatory **Scale** Community	Prototype	Processing modules to extract results from captured images are performed on the backend server. The system could get more fruitful if processing components are deployed on mobile phones because they can support complex operations.

making experiments thoroughly and the results obtained indicate that the application can effectively differentiate between clear and hazy days.

3.3 Road and Transportation Monitoring

Roads and transportation monitoring has been of prime importance for both developed and developing countries since long. Research community has presented numerous propositions to lay the foundation of Intelligent Transportation Systems (ITS)[33,18]. Most of the systems proposed statically deployment of sensors either on vehicles (e.g., GPS based tracking system etc.) or on the roadside (e.g., traffic cameras, and doppler radars etc.) or both of them. These propositions are quite practical but proven to be very expensive and restricted to the busiest areas of roads [18,33]. Using mobile phones sensing capabilities for providing real-time road and traffic information can be effectively used in reducing congestion in a number of ways such as information describing road condition can be forwarded to road drivers to avoid congested or bad roads to save traveling time[34]. Table 6 depicts a comparison of mobile phone sensing applications developed for road and transportation monitoring.

3.3.1 Road and Traffic Monitoring

Orchestrating rich sensing and communication capabilities of mobile phones can ensure powerful road and traffic conditions monitoring applications for helping users such as map annotation which could be used by drivers in deciding driving conditions that would reduce stress by keeping them away from the chaotic roads and intersections[18]. Mobile phone based traffic monitoring approach is classically ideal for developing countries as it eliminates the need of existence of expensive, specialized, and complex traffic monitoring infrastructures. Likewise, it also eliminates the dependency on advanced vehicle features such as Controller Area Network (CAN) which is absent in low-cost vehicles (e.g., 3-wheels autorickshaws etc.) commonly found in the developing countries[18]. However, using mobile phones as road and traffic monitoring devices can pose several significant challenges[33,34] related to energy consumption, inaccurate position sampling, and users privacy etc. which needs immediate attention of research community and industry.

3.3.1.1 Nericell

Nericell[18,33] exploits the rich sensing and communication capabilities of a mobile phone to detect quality of road (i.e., potholes, and bumps etc.), traffic conditions (i.e., breakings etc.), and noisiness of traffic (i.e., honking etc.). Nericell addresses several challenges including virtually reorientation and energy efficiency. Accelerometer is virtually reoriented according to orientation of vehicle and save energy concept called "triggering sensing" is introduced where relatively inexpensive sensors (e.g., accelerometer, and GSM radio etc.) form an energy viewpoint to trigger the operations of energy expensive sensors (e.g., GPS etc.). Moreover, to save energy and get efficiency in communication the sensed data is processed locally before forwarding to backend servers. Simple heuristics is used to identify honking by utilizing audio samples sensed through the microphone. Furthermore, an energy efficient method is constituted for coarse-grain localization using GSM radio. Nericell is mainly implemented with HP iPAQ hw6965 in companionship with HTC Typhoon and Sparkfun WiTilt accelerometer. HP iPAQ is the cornerstone of the system, whereas Sparkfun WiTilt is used as accelerometer and HP Typhoon is used for cellular localization. The virtual reorientation algorithm runs on the HP iPAQ and implemented in C# and python. The algorithm works by the measurements of Sparkfun WiTilt accelerometer via a serial port interface over Bluetooth radio. The audio honk detection algorithm is implemented in C# and runs on HP iPAQ. The algorithm uses FFT library (for Discrete Fourier Transform) and Windows Mobile 5.0 core.dll library for receiving microphone input. The GPS localization algorithm is implemented in Perl and runs on HTC Typhoon to provide localization information using the accessible cell tower information. In case of sensors data being invalid, user interaction identification algorithm is also implemented on Windows Mobile 5.0 which pauses accelerometer and microphone sensing while functioning. The application has been experimented with extensive GPS-tagged cellular tower measurements, GPS-tagged accelerometer data measurements collected from Bangalore (India) and Seattle (USA). A massive amount of accelerometer data about different roads conditions and data about several vehicle honking is recorded during driving. Evaluation tests yielded promising results.

3.3.1.2 Road Bump Monitor

Platform for Remote Sensing Smartphones (PRISM)[35] platform is developed to support developers in developing and deploying community sensing applications with the ultimate goals of generality, security, scalability, and without reinventing the wheel. To justify the potentiality of, three different community sensing applications have been developed using PRISM. Road Bump Monitor[35] is one of them which locate automatically using the GPS and accelerometer sensors of a mobile phone. To reduce data flow, the sensed data is processed locally to extract required features before forwarding to backend server. Client part of the PRISM (having Road Bump Monitor as an integral part) is implemented on mobile phone running with Microsoft Windows Mobile 5.0 and 6.1 (WM 5.0 and 6.1) and the server side components are implemented over a PC running with Windows 7. The system is tested on 15 different mobile phones: 4 HP iPAQ hw6965 running WM 5.0 and 8 Samsung SGH-i780, 2 HTC Advantage 7501, and 1 HTC Advantage 7510, each running with WM 6.1. Each of these mobile phones has variety of sensors including microphone, camera, and GPS etc., whereas 3 HTC mobile phones provide support for external accelerometer sensor as well. Apart from them, each of these mobile phones supports Bluetooth, 802.11b, and EDGE/GPRS/3G radios. However, only EDGE/GPRS/3G radios are used in the application as the network communication protocols for uploading sensed data from the mobile phones to the backend servers. Most of the implementation (both on client side and server side) is performed using C#, whereas, some client side components are implanted in C, and C++.To evaluate the application, a testbed has been designed where 9 road bumps were detected within 2.5 km drive. Afterward the actual locations of road bumps in the same vicinity were recorded manually as a ground truth and 7 bumps were recorded. The manual recordings are compared with the application recordings and matches of 6 bumps recordings are detected.

3.3.2 Traffic Delays and Congestion Monitoring

Traffic congestion is one of the major problems faced by the people in both developed and developing countries[34]. Despite of implementing a number of projects, policies, and programs etc. regions of all sizes are becoming the victims of traffic congestions and delays. Traffic delays and congestion are major sources of money and fuel wastage, peoples potential time wastage, inefficiency, commuter frustration, and overall significant loss of a country. Real time traffic information showing travel time or vehicle flow on the roads can be helpful in lessening traffic congestion in a multiplicity of ways such as informing drivers using real time travel and congestion estimations to find path with smaller travel time and congestion, and combining real time travel information with historic information to calculate travel time in a particular area at a specific time of a day etc[34]. Mobile phones can provide an efficient platform for monitoring traffic delays at a fine spatial and temporal granularity using the position samples collected from the drivers' mobile phones. Using mobile phones will not only provide extended coverage to major roads but will also enable users to receive traffic data back for planning (i.e., trip etc.) extracted from the data which they have collected by themselves.

3.3.2.1 VTrack

VTrack[34] is developed to accurately determine road travel time and provide routes to users with minimum travel time using the data collected from mobile phones. Key contribution of the application is to handle two classes of challenges: (1) use inaccurate position sensors (i.e., WiFi rather than GPS) to reduce energy consumption, and (2) creating precise travel time estimation from inaccurate position samples. The system uses Hidden Markov Model (HMM) to represent a vehicle trajectory over a block level map of an area, identify the most likely road segment driven by a user, and generate travel time estimations for each of the traversed road segment. VTrack uses drivers' mobile phones to periodically report position data from GPS or WiFi sensors to server which estimates travel time on the currently driving road segments using travel time estimation algorithm. At server side, users' positions are identified from WiFi position estimations using a localization algorithm if GPS is not provided or not working. Position information is used by estimation algorithm consisting of two components: map-matcher and travel-time estimator. Map-matcher identifies the currently driving road and travel-time estimator determines the travel-time based on the road segment selected by the map-matcher trajectory. Real-time travel time calculated by the application is used in a couple of applications: detecting and visualizing hotspots, and real-time route planning. The former application shows all of the hotspot within a given geographic area visually in a browser and enables route

Table 6. Comparison of mobile phone applications developed for road and transportation monitoring

	App	Hardware	Software	Protocols	Sensors	Type	Product	Shortcomings
Road and Traffic Monitoring	Nericell[18]	HP iPAQ hw6965, HTC Typhoon, SparkfunWiTilt	Windows Mobile 5.0, Windows Mobile 2003, Python, C#, Perl	GSM, GPRS, EDGE, Bluetooth, WiFi	GPS, Accelerometer, Microphone, GSM radio	Distributed **Operation** Opportunistic **Scale** Community	Prototype	To solve orientation problem, mobile phone accelerometer should be used. Information about server side implementation and visualization should be explored. Communication efficiency is effectively handled but privacy issues are completely ignored.
Road and Traffic Monitoring	Road Bump Monitor[35]	HP iPAQ hw6965, Samsung SGH-i780, HTC Advantage 7501, HTC Advantage 7510	Windows Mobile 5.0 and 6.1, Windows 7, C, C++, C#	EDGE, GPRS, GSM 3G radios	Accelerometer, GPS	**Architecture** Distributed **Operation** Opportunistic **Scale** Community	Prototype	The system is demonstrated effectively to show its accuracy but not detail information about client side and server side implementation is explored. No other issues like users privacy and energy efficiency are considered in the research work
Traffic Delays and Congestion	VTrack[34]	iPhone 3G	Not mentioned	WiFi, GSM	WiFi, GPS	**Architecture** Distributed **Operation** Participatory **Scale** Community	Prototype	Only the sensing components and potential operations of algorithms are explored and no information about the implementation on mobile phone as well as server side implementation is described.

avoidance algorithm to avoid segments which get congested very often at a particular time. The later application will use travel time prediction to empower users to figure out a route with minimum expected time to a particular destination while performing route planning or to be rerouted at mid-drive. An application for the iPhone 3G and a website is built showing users positions, nearby traffic, current traffic delays, receiving route planning updates, offering a navigation service, and sending position estimates to servers. To make both of these applications practical map-matching and time estimation algorithms have been developed which are accurate, efficient enough to run in real-time and energy efficient. The system has been evaluated with extensive datasets obtained from CarTel. Different combinations of WiFi and GPS sampling rates are used for evaluating the accuracy of real time estimates for hotspots detection and shortest path routing. Results have indicated that using WiFi localization or sparsely sampled GPS in order to save energy can produce good accuracy. It is shown that the system can tolerate significant noise and can successfully identify highly delay-prone segments and provide accurate travel time estimates for route planning.

3.4 Commerce Monitoring

Prices of the homogenous products from the different vendors vary even in the same vicinity and are called price dispersion in economics[36]. Visiting competing retailers to collect pricing information is tedious job incurring wastage of time and energy, and increases environmental pollution. Price dispersion is also more prevalent in Internet but a buyer who is well informed and having good idea about Internet searches can get the best value[37]. A study has shown that consumers with sufficient on-line price comparison experience can save up to 16% in buying electronic goods[38]. Leveraging the ubiquity of mobile phones, applications (i.e., Compare Every Where and ShopSavvy etc.) are developed to provide on-the-fly price comparisons to the consumers using online available information. Using participatory mobile phone sensing to collect and disseminate price information can be advantageous in price dispersion. Several researchers have developed ideas for sharing price information collected using mobile phone sensors (e.g., camera sensor etc). However, a number of challenges still needs to be meet[36]: (1) data gathered from mobile phone camera is not in a consistent format which makes aggregating of price information from different retailers difficult, (2) managing and storing price and behavior information of a vast variety of products and consumers in a single database, (3) application of computer vision laws and theorems in price information extraction, (4) optimum positioning of mobile phone camera to capture price image, (5) encouragement of peoples to participate in collection and sharing of pricing information, and (6) ensuring of user's privacy and security, data integrity, and reliability. Table 7 depicts a comparison of mobile phone sensing applications developed for commerce monitoring.

3.4.1 Price Bargaining

Retailers usually fix prices of the products of their own choices to make more money. Homogenous products could be offered by the different retailers in the same market with variation in prices. However, a customer has to visit several stores for making price comparison of a product with the intention of getting the product at the lowest possible price which is a tedious job. Mobile phones participatory sensing can be used in lowering the cost where users can capture price information of a product using the sensing capabilities of mobile phones and share the captured information with others in real time. Availability of price information of a product from different stores in the same vicinity would enable customers to make thorough and easy price comparison. Researchers have developed several mobile phone sensing application which could help users in effective price bargaining.

3.4.1.1 MobiShop

MobiShop[37] is a distributed system enabling users to use their mobile phones for collecting, processing, and delivering of consumer products information to potential buyers from local retailer shops. In addition, it might also work as an effective indirect cheapest medium for advertising retailer shops. MobiShop is implemented as a client-server program, working in two principal operational modes: product price collection, and user query. In contribution mode, a user captures digital image of shop receipt listing the products and corresponding prices using his/her mobile phones. Optical Character Recognition (OCR) program implemented as component of MobiShop extracts pricing information from the image and uploads information to a central server along with GPS and time of purchase information. In the query mode, a user can query the server including his GPS location for the prices information of a specific product in their locality. The server responds with a list of stores offering the requested product along with price within the user vicinity. MobiShop client program is implemented in Java ME and used native Symbian OS 9.2 OCR engine. A captured image is input to the OCR program whose output is a text file containing products names and prices information. Users are provided with an option to fix errors and erase any sensitive information (e.g., credit card number etc.). Network communication module establishes a TCP connection with the server and transfers file along with GPS coordinates and time information to the server using existing cellular or WiFi. A simple GUI interface is provided for user to issues query which is transferred along with GPS location information to the server using a TCP connection. The client interface is also connected with GIS library to render shops location over a street map for ease in navigation. A prototype of client program is implemented on Nokia N95 8GB mobile phones. The server program is written in Java and runs as a daemon on always-on workstation. The central server is the basic repository for storing and maintaining the price information. The server program is implemented using MySQL and Tomcat running on Linux or Windows platforms.

3.4.1.2 PetrolWatch

PetrolWatch[36,38-40] leverages mobile phone sensing capabilities to collect, contribute, and disseminate fuel pricing information. The system is implemented as client-server program and works in two operational modes: fuel price detection and user query. When a contributing vehicle reaches near a fuel satiation (i.e., identified using GIS and GPS), mobile phone camera to be mounted on the dashboard is automatically triggered to take photographs of the roadside fuel price board. The metadata related to the photograph (e.g., location coordinates, brand, and time etc.) is extracted automatically and stored separately. A sophisticated image processing engine is implemented on mobile phone which crops regions within the input image representing fuel brand and fuel price information by using the color thresholding for each of the fuel brand. The cropped color image is converted into binary to extract the individual numerical values. A Feedforward Back Propagation Neural Network Algorithm is utilized for classifying the digits. The extracted prices information are uploaded to a server for storing in a database which is linked with GIS road network database containing fuel stations locations information. Fuel price information of a particular station is updated on the server if price information with new timestamp is provided. Querying the database is returned to the users with up-to-date price information of the nearby fuel stations using GIS and GPS location information. Application prototype is implemented on Nokia N95 mobile phone. The client application is implemented in JavaME which uses open source JJIL image library for implementing the computer vision algorithm. Location coordinates are obtained using either mobile phone built-in GPS receiver or an external GPS receiver connected with the phone via Bluetooth. The system requires GIS software (e.g., Google Maps, Nokia Maps, or TomTom etc.) already installed on the mobile phone. The extracted price information is uploaded to the backend server using cellular or WiFi networks thorough a TCP connection. As an experimental study 60 images are used, out of which the system achieves a hit rate of 75.4% for correctly detecting the fuel price board from the image background and 86% accuracy in reading the price information out of them.

3.4.1.3 LiveCompare

LiveCompare[41] is a participatory sensing system which leverages camera enabled mobile phones for grocery bargaining and improve inter-store grocery prices comparisons. LiveCompare is similar to CompareEveryWhere and ShopSavvy applications but can provide information related to grocery products which are not sold online. The system works by requiring users to take snap pictures of the products price tags which will be used for extracting product and price information located in the tag in the form of UPS barcode. UPC is used by the system as a globally unique key for representing products. Barcode libraries (i.e., ZXling) are used to accelerate the decoding of barcode and the numerical UPC value along with the photograph is forwarded to the LiveCompare central server. The client also sends GPS and GSM information to the server for identifying and localizing current store which can be used by the system to restrict query results to include only nearby stores. By taking picture of a product price tag, a user is actually specifying the product about which he wants to submit a query for price comparison from a grocery shop. The server populates its database by appending the upcoming additional information. Thus, the system enriches its database automatically whenever a user initiates a query. By submitting a query to the server, a user is provided with pricing information for the scanned product at the other nearby grocery stores. The system emphasizes on information integrity and reduces the risks of human error by requiring the product identification and pricing information to be incorporated into the same tag and submitted to the system in the same photograph. Users can flag entries if found incorrect and an entry with sufficient number of flags would be removed automatically. Likewise, a user found owner of several suspicious entries can be banned. The system is implemented on Nokia N95 8GB. A participant location is identified using GPS/GSM and data/photographs are uploaded to central server using HSDPA.

3.4.2 Advisory Assistance

While making a business activity, people, undoubtedly, would wish to buy high quality products at the lowest possible prices. The ratio of success is always subjected to finest buying decision making capabilities. People, women in particular, always feel problems in making buying decision due to a number of reasons including psychological problems, or being puzzled because of large volume of verities etc. To elevate the decision making problem, people usually take assistance of experts, or experiences of the users etc., who are not usually available at ease. The technological advancements have enabled mobile phones to extract contextual and psychological information in a seamless manner. Therefore, several automatic systems are devised by the research community which can help users in the selection and purchase of product by exploiting the users' captured information.

3.4.2.1 ColorMatch

ColorMatch[42] is a mobile phone based cosmetic advisory system to solve foundation color selection problems by enabling women to select suitable colors of cosmetics. Women usually select inaccurate foundation makeup which does not suit their skin tones. To solve the problem, ColorMatch is aimed to provide: (1) expert makeup advice, (2) enabling to choose appropriate foundation makeup in accordance to skin tone, and (3) to get advice anywhere and anytime form the users in the same community. ColorMatch is a is a pipeline consisting of components of image processing, color correction, and an expert system which collectively evaluate a user facial skin coloring and provides an accurate expert opinion. The system requires a user to photograph herself using her mobile phone camera while holding a specially designed color chart in front. The captured image is forwarded by the consumer to a backend server running with advisory service via MMS. At the server, the image quality (i.e., image color, camera specifications, and lighting conditions etc.) is corrected by color science, face in the image is located and extracted by using image processing algorithms, and foundation makeup color is suggested to a user in accordance to her skin color using statistical classifiers in almost the same accuracy as would be suggested by a makeup expert. After server side processings, user would be informed with foundation makeup recommendations best suit their complexion via a SMS.

3.5 Online Social Network Reciprocation

Online Social Networks are computer based networks aimed to bond inhabitants and organizations. In the past few years the viral growth of online social networking applications have facilitated people in forming friends and keeping in contact with old friends and colleagues[43]. These applications provide a platform where people can declare new friends and follow their activities or posts online. A user profile in online social networks provides a great deal of his personal and contextual information including name, picture, contact information, interests, and activities/hobbies etc.[44] With the proliferation of mobile phones and increasing success in developing applications capable of accessing social network information have enabled applications to know about users' preferences, social groups, contacts, and positions[45]. Group sensing applications are getting popularity as a result of growing interests in social networks and representing groups of individuals having the common goals, concerns, and interests[8]. Several mobile phones sensing applications have been developed sharing users' contextual information and experiences with friends using social networks. The Mobile Social Network (MoSoNet) bring the rich contextual information form online social networks to enrich local social interaction such as helping people in identifying identity of a person, whom they are seeking to interact, initiating conversation, and building relationships etc.[44] Sharing this information can be advantageous in numerous ways but the current suggested models have to face with several complex and research demanding problems[15] including operating system problems, API and operational problems, security and privacy problems, and hardware platform problems etc.

3.5.1 Presence Sensing and Sharing

Presence sharing is an accommodative service which could be used by mobile phones custodians to broadcast their identities periodically using short-range wireless technologies (e.g.,WiFi, and Bluetooth etc.)[46]. Incorporation with location based services (e.g., GPS etc.) presence sharing service can provide an efficient platform for emerging mobile phones and ubiquitous applications. Presence sharing can be an attractive alternative as compared to traditional pervasive computing architecture due to low-cost, decentralized architecture, and ease of deployment. Modern mobile phones capabilities (i.e., programmable, and sensing etc.) establishes a new platform for novel people presence sensing applications development to share information through social networking applications for answering questions concerning people such as where are the people and what they are doing?[15]. Table 8 depicts a comparison of mobile phone sensing applications developed for presence sensing and sharing.

3.5.1.1 CenceMe

CenceMe[15] uses sensors enabled mobile phones to infer information for determining presence of a person (e.g., running, in party, or conversion etc.) and share collected

Table 7. Comparison of mobile phone sensing applications developer for commerce monitoring

	App	Hardware	Software	Protocols	Sensors	Type	Product	Shortcomings
Price Bargaining	MobiShop[37]	Nokia N95 8GB	Java ME, Symbian OS 9.2 OCR, MySQL, Tomcat, Linux, Windows	GPRS, GSM, 3G, HSDPA, IEEE 802.11	Camera, GPS	**Architecture** Distributed **Operation** Participatory **Scale** Community	Prototype	OCR technology is used to extract prices information from images captured by mobile phone cameras. But, OCR technology is error-prone and is difficult to parse making results unsatisfactory.
	PetrolWatch[39]	Nokia N95	Java ME, open source JJIL image library	GPRS, GSM, 3G, HSDPA, IEEE 802.11	Camera, GPS	**Architecture** Distributed **Operation** Participatory **Scale** Community	Prototype	Using complex image processing algorithms can be problematic and can jeopardize a mobile phone. The results obtained are also not satisfactory.
	LiveCompare[41]	Nokia N95 8GB	Not mentioned	HSDPA	Camera, GSM, GPS	**Architecture** Distributed **Operation** Participatory **Scale** Community	Prototype	Decoding barcode is accurate than OCR but some situations needs definite use of OCR. Capturing information about a user location and products of interest might reveal sensitive information concerning privacy which is not addressed by the system.
Advisory Assistance	ColorMatch[42]	Not mentioned	Not mentioned	GSM	Camera	**Architecture** Distributed **Operation** Participatory **Scale** Personal	Prototype	Mobile phone is used for image capturing and other processings are performed on the backend server. No information about statistical classifiers and image processing algorithms are given. Privacy concerns such as how images are to be used once loaded on the backend server is not clearly defined.

information using social networking portals. The system enables users to: (1) automatically upload and share enriched forms of presence information to members of their social networks, and (2) historical analysis of their activities. CenceMeruns as a daemon users phones and automatically perform sensing, classification of raw sensory data to produce primitives for visualizing users' presence on their phones, and uploading the primitives to the backend server. CenceMe splits classification process between mobile phone and backend server. Primitives are calculated from the phone classification process, whereas, facts are calculated from the backend server classification process and stored in a database for retrieving and publishing. Primitives are calculated using: (1) the classification of sound samples from a phone's microphone using a discrete Fourier transform (DFT) technique and a machine learning algorithm to classify the nature of the sound, (2) the classification of on board accelerometer data to determine activity (e.g., sitting, standing, walking, and running etc.), (3) scanning Bluetooth MAC addresses in a mobile phone vicinity, (4) GPS readings, and (5) random photos which are taken randomly when a phone keypad key is pressed or a call is received. The backend server runs with classification algorithms used to determine more complex forms of sensing presence. Consuming applications (e.g., Facebook, and the CenceMe portal etc.) and VOIP clients (e.g., Skype etc.) can access the stored data at backend servers (filtered for privacy) using specialized

CenceMe API. After logging in, a CenceMe buddies list (i.e., Facebook friends running CenceMe application on their mobile phones) is downloaded from the CenceMe server and selecting a buddy from the list would retrieve his sensing presence from CenceMe server through GPRS or WiFi. The CenceMe prototype is designed to run on any Symbian-based mobile phone which supports JVM. The system architecture is based on the client-server architecture. Most of the client side data producing and other components are implemented as daemons and written in C++ and Java to support portability while addressing some of the limitations of the JVM system APIs. The client suite is implemented as a thread architecture where each component is implemented as a separate thread to protect components from each other in case of any failure. Primitives are transferred to the backend server using XML-RPC calls using either WiFi or GPRS. Once primitives are received by the backend server, they are stored in the MySQL database. The backend server components are also written in Java and uses Apache 2.2 and Tomcat 5.5 to provide web services. A web service based API is provided by the backend servers to service primitives received from the mobile phones and application requests from the Facebook, or CenceMe portal etc. Server uses RPC technology implement in Apache XML-RPC library to control the communication between mobile phones and backend servers. Applications' requests are handled by the backend servers using Java Servlets in combination with MySQL database for storage. The application has been tested on Nokia N95 and it is found that activity classifier performance is not significantly affected by small variations in locale (e.g., restaurant or office etc.) and people (e.g., weight or body type etc.) but differentiating between sitting and standing states is found relatively difficulty. The audio classifier suffers from a relatively high rate of false positive. Likewise, phone placement on different places on the body has no affect on the accuracy of activities inference. Results have shown that techniques used are very energy efficient and less greedy for resources (e.g., memory and CPU etc.) and are very much promising.

3.5.1.2 Friendlee

Friendlee[43] uses ambient awareness by analyzing users calls and messages history information (i.e., frequency, recency, and duration etc.) to automatically infer and relatively weight contacts in their social networks for depicting a rich picture of their social lives. Without requiring users to wade through an extensive phonebook, the strongest connections are rendered prominently to provide users to have instant access to them. The system share a significant amount of users contextual information depicting his location at different granularities (i.e., street address, city, country etc.), his phone (i.e., on/off/available/ringer/silent/vibrate) and message status, local time, and weather etc. with their intimate in a social network. The system can track down a user preferred services and businesses which he frequently uses (e.g., favorite dentist, restaurant, coffee shop, and health insurance etc.) and use them as recommendations to his social network. These recommendations could be easily obtained by other users in his intimate social network while browsing his connection. To protect privacy, the system uses category-based privacy model for context sharing which allows users to classify their contact into categories (e.g., colleagues, and family etc.) and specify that which categories may view (1) different kinds of their contextual information (i.e., phone status, location etc.), and (2) different categories of their contacts (i.e., friends, family, colleagues etc.). A user-friendly interface is provided which is natural and intuitive to the type of communications commonly take place among close peoples. Friendlee consists of three components: (1) phone-based client, (2) Web-based interface, and (3) backend server. Phone-based client is the Friendlee user-interface which gathers required user and contextual information such as personal status, call and message history, and location etc. and synchronizes with server several times in a minute to upload up-to-date gathered information. The server stores all of the users information in a centralized database, calculates the strength of relationships in a social network using communication history, and propagates contextual information to users social networks while taking into account their defined privacy policies. Users can also access the information stored in backend server using the Web-based interface. Client side is implemented on Android and Windows Mobile operating systems and server side is implemented in Perl and MySQL database. Connection between client and server is accomplished using stateful TCP/IP, stateless HTTP, and SMS.

3.5.2 Social Activities Coverage

The inclusion of high valued camera sensors in wearable devices enables users capturing of high-resolutions views (i.e., photos and videos) of events happening in their daily lives. However, excessive experiences of capturing events

Table 8. Comparison of mobile phone sensing applications developed for presence sensing and sharing

App	Hardware	Software	Protocols	Sensors	Type	Product	Shortcomings
CenceMe[15]	Nokia N95	C++, Java, Apache 2.2, Tomcat 5.5, JVM	GPRS, WiFi	Accelerometer, Microphone, GPS	**Architecture** Distributed **Operation** Participatory **Scale** Group	Prototype	Presence sensing is an aspect of human life people are interested in. For complete presence description, visual experiences are also needed to be captured. Mobile phone would be running with several daemons which could jeopardize mobile phone.
Friendlee[43]	Not available	Android, Windows Mobile, Perl, MySQL	TCP/IP, HTTP, SMS	None	**Architecture** Distributed **Operation** Opportunistic **Scale** Group	Prototype	Privacy model needs improvement to determine that what level of access a person can have when he want to access a friend in an intimate social network.

can produce large amount of information resulting into information overload problem. A biggest challenge is the extraction of relevant contents from the large collection of information and summarizing to the end user. Solving the challenge requires long term research efforts for the development of applications capable of recognizing socially interesting events and stitching video highlights from a number of sources for generating video highlights of an occasion. These types of applications will reduce humans load for creating videos of a social activity highlight. Table 9 depicts a comparison of mobile phone sensing applications developed for social activities coverage.

3.5.2.1 MoVi

Mobile Phone based Video Highlight (MoVi)[47] harnesses mobile phones in social contexts to collaboratively to record videos and create automatic video highlights of social events for answering questions like "what happened at the party?" positively. MoVi assumes client-server architecture where client is a sensors enabled mobile phone and server is a MATLAB-based remote workstation performing most of the architectural functions such as Group Management, Trigger Detector, View Selector, and Event Segment. Group Management module groups mobile phones belonging to the same social context into social groups using both the visual and acoustic ambiance of mobile phones. Trigger Detection module identifies potentially interesting social events by using sensory data from mobile phones belonging to the same group. Once an event is identified, View Selector module recruits a mobile phone with good quality view as the best video view using face count, accelerometer reading ranking, light intensity, and human assistance matrices. Event Segmentation module extracts the appropriate segment of the video by putting logical start and end of an event using sensory data received before and after the trigger. Once all of the video-clips have been identified, they are stored in time and stitched into automatic video highlights of the occasion. MoVi is experimented in controlled settings and natural social occasions' scenarios. Participants used iPod Nanos taped with shirt pockets for video capturing and Nokia N95 phones clipped with belts for ambient capturing of sensory data from accelerometer, compass, and microphone. Predefined activities were defined in controlled setting, whereas, no activities were predefined for natural social occasions. Results obtained in each of the scenarios indicated that video highlights created in offline by the MoVi are almost similar to those created manually. However, MoVi needs improvements in information retrieval accuracy, handling camera views of unsatisfying quality, energy efficient algorithms for video recording and events detection, protecting users' privacy, and algorithms for data mining, signal processing, and image processing.

3.5.3 Protecting users Privacy and Security

Developing mobile phones based applications for presence sharing and accessing social networks information is a potential research area but the available approaches share information insecurely and may result into privacy compromises. Users in proximity can share their presence through broadcasting their identities periodically using low-cost and built-in mobile phones sensor such as Bluetooth, and WiFi etc. but managing presence privacy in such as broadcast environment remains difficult in particular[46]. Information sharing using mobile phones can be abused in a number of ways and may result into a catastrophic situation. Sending information over wireless channels without encryption could enable a nearby person to not only snoop user social network identity and eavesdrop on the data transmitted for launching different attacks including spoofing, replay, and wormhole etc[45]. Furthermore, most of the social network sites (e.g., Facebook, and Myspace etc.) allow users to install third-party applications and widgets on their mobile phones to enhance performance. Third-party applications can access user information as well as the information a user has rights to access (e.g., friends etc.)[45]. Table 9 depicts a comparison of mobile phone sensing applications developed for protecting user privacy and security.

3.5.3.1 Secure SocialAware

Secure SocialAware (SSA)[45] is an improvement of the previous work "SocialAware"[48] and provides location-based personal information without disclosing users' identities or any set of information which could be matched to a user positively. The proposed system is composed to three primary components: Stationary Component (SC), Mobile Component (MC), and Authentication Server (AS). SC is implanted in user environment, MC is deployed on mobile phone, and AS is executed on a web server for providing services to MC and SC. AS is a trusted component and both MC and SC components are needed to have accounts for signing into the AS. AS stores the current locations, and users/administrators names and passwords of the MC and SC during the signing up process. A MC user is assigned a unique Initial Encrypted Identifier (EID) by the AS at account sign up time and provides services to SC for obtaining user information from a social networking site using the EID. To notify its presence, a MC advertises its EID to the nearby active SC using a wireless technology (e.g., Bluetooth etc.). The SC detects the presence of users, obtain information about users from AS, and perform actions accordingly. To obtain information about a user from a social networking web site, SC provides EID to the AS which identify his social network profile using EID. SSA has enough security features to stop security threats like replay and spoofing etc. EID for each mobile phone user is generated using cryptographic hash function with a random salt value and updated continuously by the SC. Using EIDs instead of social network identity can prevent malicious users performing spoofing and replay attacks. To provide protection against eavesdropping, all of the traffic between MSs and SCs as well as between SCs and SA is encrypted. Furthermore, AS does not support to retrieve personal information of a user which is not defined in his preferences. Thus, connecting a user EID with his social network identity becomes impossible. Java Standard Edition (SE) 5.0 platform is used for implementing SC and AS, while MC is implemented using Java Micro Edition (ME) Mobile Information Device Profile (MIDP) platform and all of the interactions between MC and SC are taking place using Bluetooth. The Merge framework is used for managing all of the Bluetooth communication logic in SSA. SA is developed using Reslet framework for Java and all of the resources of SA are accessible by MC and SC as web services confirming to the REST architecture. Each resource is a separate URL accessible supporting HTTP GET, PUT, and POST methods. Open source SimpleJPA tool is used to persistence all data on the AS. AS generates EID for each MC using SHA-1 cryptographic hash function with a 16-bytes random salt value. To obtain contents of a user's Facebook profile AS uses the Facebook REST API web service. Context-aware social networking application called "SocialAwareFlicks" is developed using SSA framework. SocialAwareFlicks displays recommended movie trailers that match the movie preferences of one or more users jointly watching a common display. All performance tests are conducted using a Macbook Pro notebook running with Mac OS X 10.5, 2.4 GHz Core2Duo processor, and 4 GB RAM. The tests results are positive such that AS scales effectively and SC can retrieve information quickly once MC is detected.

3.5.4 Person-to-Person Communication

Social networks would not only enable users to share a rich set of contextual information but also provides mechanisms for searching users with similar interests and establishing and maintaining communication between them[44]. Social networks emergence is the result of natural evolution of Internet, with the next big wave concentrating on facilitating person-to-person communication.

MoSoNet (Mobile Social Network) technology assumes using of mobile phones for accessing online social networks using users' identities to bring relevant social contexts into the local contexts for enriching local human interactions. Leveraging peoples' personal information from online social networks profiles, mobile phones can be used to inform users about identities of the people in vicinity and establish connection between them. Table 9 depicts a comparison of mobile phone sensing applications developed for person-to-person communication.

3.5.4.1 WhoizThat

WhoizThat[44]uses mobile phones to enrich local social interactions by facilitate a user to start conversation with a person of interest in vicinity by leveraging identity and relevant information from online social networks profiles of the person. WhoizThat fulfills the vision of MoSoNet seamless interaction by implementing a simple two phase protocol. In meet-and-grace phase, nearby mobile phones in vicinity periodically advertise and share their owners' social network IDs with each other using short range wireless technologies (e.g., Bluetooth and WiFi etc). In context-creation phase, users identities are used to retrieve social contextual information from social networks into local contexts and assist them in deciding to initiate a conversation or not. WhoizThat protocol is beneficial because of its simplicity, energy/bandwidth efficiency, agnosticism, and extensibility. WhoizThat information sharing protocol is programmed in Java Micro Edition (JME) and implemented on Nokia N80 mobile phone. Mobile phones broadcasts and receives a users social network IDs via Bluetooth and retrieve the corresponding user name from online social network sites using WiFi. WhoizThat framework can be used for developing fruitful applications which could adopt their behavior according to the interests of the nearby located people in vicinity. Using WhoizThat, WZPlaylistGen is developed which is a context-aware music playlist generation application and creates a playlist according to the taste of the people in a bar. The WZPlaylistGen is developed in Java Standard Edition and implemented on a PC. Using users' IDs, WZPlaylistGen retrieves users' musical preferences from Facebook through Facebook web services API and generates a musical playlist by using the Audio scrobbler API and play the music from this list in the local environment. Despite of WhoizThat usefulness, it can give raise to certain privacy and security risks. By broadcasting actual ID can result into spoofing, and cyber stalking attacks. To truly realize the full potential of MoSoNets, full privacy and security protection must be considered.

3.5.4.2 OLS

Opportunistic Localization System (OLS)[49] is a truly ubiquitous mobile phone-centric localization system readily providing location related information to people for finding their peers in a vicinity and make connections with them. OLS uses service-oriented architecture to cope with the increasing demands of reusability across different environments and platforms, and to scale up services to support a vast number of various clients. OLS services including management services, location services, data fusion engine, communication services, database, registration services, and visualization provider resides on the server side making the concept of OLS concrete. The mobile phone client is localized by the OLS server using stream of client sensory data including GSM/UMTS signal strength, embedded accelerometer readings, WiFi signal strength, Bluetooth proximity information, and GPS. A mobile phone's physical environment is considered an important factor for deciding reliability and availability of input information such as using GPS for outdoor environments and using WiFi signal strength as well as GSM/UMTS signal strength for indoor environments. Bayesian Filter is used for fusing all incoming location information at the fusion engine, whereas Particle Filter represents the density of the mobile client's location. To accurately determine localization, psedometery data estimated from accelerometer measurements are used in complement to WiFi signal strength information, GSM/UMTS signal strength information, and GPS data. OLS can also incorporate the availability of Bluetooth signals in his location estimation process and can also detect the proximity of other Bluetooth enabled devices.OLS can be interfaced with external third-party services such as Google Earth enabling users to build an indoor building environment with Google Sketch-Up drawing tool and publish them on Google Earth.

3.5.4.3 Social Serendipity

Serendipity[50] uses mobile phone sensing to facilitate interactions between the people in a physical proximity (i.e., conference, bar, meeting, and bus etc.) through a centralized server. Information about users' profiles and matchmaking preferences is stored in Serendipity central server. Similarity score is calculated by identifying commonalities between two proximate users' profiles and summing them according to the user-defined weights. If score is found more than the defined threshold for both of the users, the application alerts them that someone nearby might be of potential interest. Upon receiving Bluetooth ID (BTID) and threshold values from the phones, Serendipity queries a MySQL database. If a profile associated with the received BTID address is found and similarity score is greater than the threshold, information from the profile (i.e., picture, commonalities, a list of talking points, and contacts etc.) is forwarded to the user mobile phone. Mobile Information Device Profile (MIDP) 2.0 application called BlueAware is developed which runs passively in the background of a Bluetooth-enabled mobile phone for recording with timing information the BTIDs of other devices in an environment. Once a new BTID (which is not recorded) is discovered, the BTID is forwarded automatically to the Serendipity server. To stop draining out the battery, BlueAware scans environment after every five minutes. Another application is BlueDar, a slight variation of BlueAware, which continuously scans environment and forward detected BTIDs to the central server using 802.11b wireless network. BlueAware usage is simple and installation requires just needs sending of installation file (Symbian.sys) to mobile phone through Bluetooth or IR. After installation, the mobile phone is automatically connected to server, mobile phone Bluetooth ID (BTID) profile is created, and users are lined up with each other as friends. Serendipity has been tested and evaluated for almost one year, and has gone through a number of design and implementation changes to gain overwhelming positive results. Serendipity can give raise a number of privacy concerns if not handled carefully. Providing nearby strangers with a user's name and picture can result into a great liability and privacy issues. Serendipity should be improved to ensure that the service never jeopardize a user's privacy expectations. Thus, all of the privacy concerns are required to be reviewed efficiently before releasing the service to the general public use.

3.6 Fall Monitoring

Unintentional falls are a common cause of bringing fatal and nonfatal injuries to people especially older people all around the world. Due to the fact that most of the falls occur at home which could increase severity and danger because of not having any assistance in case of unconsciousness and extreme injuries. To decrease risks of falls and associated outcomes in elderly population both commercial organizations and academic research are motivated for the development of efficient and cost-effective fall detection and reporting products[51]. Most of the modern fall detection systems use three methods[52]: (1) implanting sensing devices in floors for detecting and monitoring signatures of human fall, (2) mounting cameras to a fixed location to track a human inactive state for a long time, and (3) users wearing sensors to track vector forces exerted on the users. Although existing commercial products and academic research systems may perform well but they are suffering with a number of deficiencies[51,52]: (1) designing of special hardware and software which increases cost and restricts applicability to only wealthiest and most impaired users, (2) higher installation costs and higher training time further can effect wide spread adaptation, (3) majorly reporting false positives. Mobile phones are an ideal choice for developing pervasive fall detection systems because they are highly portable, all necessary components (i.e., hardware and software) are already integrated, and have vast communication coverage. A fall detection system requires sensors enabled mobile phones which are very popular and thoroughly accepted in the society. Users will prefer to move with a single mobile phone integrated with all of the fall detection functionalities instead of separate fall detection devices clipped to their bodies. Table 10 depicts a comparison of mobile phone sensing applications developed for fall monitoring.

3.6.1 *iFall*

iFall[52] is a cost-effective and reliable mobile phone based solution for fall detection. iFall is implemented as a low-powered Android service which runs inconspicuously in the background without jeopardizing a mobile phone. Fall monitor service constantly listens to accelerometer to detect a fall. Fall detection algorithm assumes that the forces exerted during a fall are different than forces during normal daily activities. Fall is detected by taking root-sum-of-squares of the accelerometer three axis data. If a fall is suspected, a user is required to restore to the original position within a short period of time. If position is not restored and timer is expired, intent is sanded to the iFall activity which will try to get the user's attention by regularly vibrating, flashing LEDs and screen, and playing an audio message. To prevent false positives, user is prompted with a simple pop-up window to confirm or negate the fall suspicion. By negating alert, iFall activity is closed and interrupted activity is restored. In case of absence of user's response, a simple fall detection SMS containing certain information is sent to every contact in the iFall emergency list asking them to text the fallee mobile phone. In case of social contact confirms to the fall or no social contact text the fallee, emergency services are notified. iFall prototype is implemented on Android and deployed on HTC G1 mobile phones. The prototype is written in Java, run in Dalvik virtual machine, SQL Lite database is used for storing persistent data. The upper threshold value used for fall detection in the fall detection algorithm is resizable due to different carrying methods of a mobile phone. More accelerated body parts will have greater upper threshold and vice versa. Users can edit their emergency contacts lists which have to be used by iFall before confirming a fall. In spite of being simple and efficient, iFall suffers from certain shortcomings. Instead of relying on only one sensor (accelerometer) data, it could be quite beneficial and accurate to confirm a fall by using data from multiple sensors either external or internal. Likewise, Bluetooth can be also be used to acquire data about a fall from multitude of sensor in an environment. The author has augmented that the application can be extended to cover a broad range of peoples' communities instead of limiting to senior citizens only.

3.6.2 PreFallD

PerFallD[51,53] exploits the built-in detection and communication components of mobile phones for developing pervasive fall detection systems. The fall detection algorithm is acceleration-based detection approach requiring a mobile phone to have an accelerometer. The system is considered to be available both indoor and outdoor, user friendly, requiring no extra hardware and service costs, power efficient, and lightweight. The main program runs in the background as daemon requiring a user to view and make changes in his profiles containing user's defined basic fall detection configuration. Power-aware pattern matching algorithm is used to process accelerometer data for determining fall existence. If a fall is detected, alarm is triggered and timer is started requiring a user to turn off the alarm manually within a certain time period. In case of no response, the system would automatically and iteratively initiate calls and texts up to five contacts already defined by a user in his emergency contact list according to his priorities. Power efficiency is ensured numerously in the application design such as the daemon service runs in the background to not jeopardize a mobile phone, users can adjust the sampling frequency, the pattern matching algorithm is launched only if the daemon-collected data exceeds some preset threshold values, and screen is activated only if necessary. PerFallD prototype is developed on Android G1 mobile phone running with ARM-based dual-core processor, 98MB RAM, and 70MB of internal storage. To Evaluate PreFallD, data about 450 falls in different directions (lateral, forward, and backward), different speeds (slow and fast), and in different environment (kitchen, bedroom, outdoor garden, and living room) is collected. Data about daily life activities including walking, sitting, jogging, and sitting is collected as well for several days. PerFallD is compare with two existing fall detection algorithms and one commercial product for performance evaluation. The results obtained have shown that PerFallD outperforms existing algorithms and provide efficient balance between false positive and false negative as compared to commercial product.

3.7 Human-Phone Interaction (HPI)

Human-Phone Interaction (HPI) is a branch of Human-Computer Interaction (HCI) where researchers are continuously searching to deduce new techniques and technologies for users to interact with their mobile phones with less efforts. HPI, to date, has not been much successful like HCI due to addressing unique challenges which are not typically found in HCI environment. Most HCI technology addresses solutions in an ideal environment, whereas, user-mobile phone interaction usually take place under varying conditions and unpredictable environments due to portability. Furthermore, available HCI technologies (e.g., glass mounted cameras, and helmets etc.) due to their size constraint cannot be penetrated directly into mobile phone domain. Therefore, more specialized technologies and enhanced types of techniques to cope with the mobile phone environment constraints. HPI technology relies heavily on a mobile phone internal sensors (i.e., camera, accelerometer, and microphone etc.) to infer and detect gestures made by users for initiating certain actions. HPI technology, however, has to work under certain constraints[54]: (1) mobile phones internal sensors should be used heavily, (2) user interaction should be minimized as much possible to function, (3) lightweight in terms of computations requiring less resources usage, and (4) mobile phone should not

Table 9. Comparison of mobile phones sensing applications developed for protecting users privacy and security, person-to-person communication, and social activity coverage

	App	Hardware	Software	Protocols	Sensors	Type	Product	Shortcomings
Protecting Users Privacy and Security	Secure SocialAware[45]	Not available	Java Micro Edition, SimpleJPA, SHA	HTTP, REST API	WiFi, Bluetooth	**Architecture** Distributed **Operation** Opportunistic **Scale** Personal	Prototype	Transmission delay can be introduced due to processing units at three distinct locations. The entire system will be down if SC is not working. If SC is compromised, the entire data transmission would be compromised. The system will only work at locations where SC is present.
Person-to-Person Communication	WhoizThat[44]	Nokia N80	J2ME, JSE, Facebook web service API, Audioscrobbler API	Bluetooth, WiFi	Bluetooth	**Architecture** Distributed **Operation** Opportunistic **Scale** Group	Prototype	It does not have any provision of using mobile phone sensors which can be used to provide rich contextual information instead of entering manually. Sharing of users social network identities in plain form can result in privacy and security violations.
	OLS [49]	Not available.	Not available.	WiFi, GSM/ UMTS	Bluetooth, GPS	**Architecture** Distributed **Operation** Opportunistic **Scale** Group	Not available	Mobile phone is used as a data provider and all of the location identification, data fusion, and management functionalities are performed on the server side. Information about hardware and software platforms used for implementations are not provided.
	Social Serendipity[50]	Not available.	Not available.	GPRS, WiFi, GSM	Bluetooth	**Architecture** Distributed **Operation** Opportunistic **Scale** Group	Prototype	The application does not provide information about data security and preserving users' privacy. Therefore, the application can raise a number of privacy concerns.
Social Activity Coverage	MoVi[47]	iPod Nano, Nokia N95	MATLAB	GSM, Bluetooth	Camera, Accelerometer, Compass, Microphone	**Architecture** Distributed **Operation** Participatory **Scale** Group	Prototype	The system emphasizes on performing most of the activities on the server instead of mobile client., MoVi needs improvements in different aspects including group management, trigger detection, view selection, and event segmentation etc. to improve the accuracy of creating video highlights.

Table 10. Comparison of mobile phone sensing applications developed for fall monitoring

App	Hardware	Software	Protocols	Sensors	Type	Product	Shortcomings
iFall[52]	HTC G1	Android SDK, Java	GSM	Accelerometer	**Architecture** Integrated **Operation** Opportunistic **Scale** Group	Prototype	Data from other sensors as well as accelerometer should be used accurate fall detection. Fall data should be stored on remote server to keep a brief fall record. Due to different carrying methods and orientations of mobile phone by users iFall is required to adjust the correct parameters for fall detection logic.
PerFallD[51]	Android G1, ARM-based dual-core processor	Android SDK 1.6, Java, Eclipse	GSM	Accelerometer	**Architecture** Integrated **Operation** Opportunistic **Scale** Group	Prototype	Same as iFall.

be jeopardize and battery power should not be depleted quickly. Advances in HPI research can be useful numerously such as improving peoples safety while using their mobile phones (i.e., reducing risks of accidents while driving etc), and facilitating the use of mobile phones by the disabled/handicapped people. Table 11 depicts a comparison of mobile phone sensing applications developed for HPI.

3.7.1 NeuroPhone

NeuroPhone[55] is developed to use neural signals to control mobile phones with hand-free, silent, and effortless human-mobile interaction. NeuroPhone is implemented on iPhone and works by using neural signals from wireless Emotiv EPOC electroencephalography (EEG) headset and capturing wink gestures from users eyes. The encrypted P300 brain signals from the headset are decrypted by Emotiv's closed source SDK on a windows machine which relays raw EEG data to mobile phone through WiFi. Lightweight signal processing and classification algorithms are used on mobile phone to discriminate P300 signals from noise and band pass-filters are used to eliminate any noise that is not in the P300 frequency range. Dial Tim application is developed using NeuroPhone which is a brain-controlled address book dialing application working in two modes: think and wink. Dial Tim uses the same principles of P300-speller brain-computer interfaces: the phone flashes a sequence of photos of contacts from the address book and a P300 brain potential is elicited when the flashed photo matches the person whom user wishes to dial. EEG signals from the headset are transmitted wirelessly to the iPhone where lightweight classifiers are used to discriminate noise from P300 signals. A person number is automatically dialed when a person's picture triggers a P300 signal. The system is evaluated in both modes in different scenarios (e.g., sitting, and walking etc.) consisting of two different Emotiv headsets and three different subjects. In wink mode, results data collected from sitting scenarios are more promising than walking scenarios. In think mode, accuracy is more promising as compared to situation with background music and standing. The application is found very lightweight requiring 3.3% CPU usage on iPhone and total 9.40MB memory space out of which 9.14MB is used by GUI elements and rest is used by lightweight processing and classification components of the application. However, NeuroPhone suffers from a number of challenges: (1) using cheaper headset induces significant amount of noise in the data which require using of more sophisticated signal processing and machine learning techniques to classify neural events, (2) continuously streaming raw neural signals over wireless technologies and running processing as well as classifiers on phone can be energy hungry prohibitive. Likewise transmitting unencrypted neural signal packets wirelessly can also open important privacy challenges.

3.7.2 EyePhone

EyePhone[54] is developed to capture eye movements and actions (instead of voice due to noise) using a mobile phone front-facing camera to drive applications/functions and trigger actions on the phone. EyePhone tracks user eye movement and its position across mobile phone display using machine learning techniques to infer its position on the display and emulates eye blink as mouse click to activate application under view. The eye detection and blinking algorithm is inspired from the algorithms developed for desktop machines and consists of four phases: (1) eye detection phase, (2) open eye template creation phase, (3) eye tracking phase, and (4) blink detection phase. The original algorithm is modified in a way to reduce eye detection error rate and to add two more filter criterions to reduce false eye contours. A user open eye template is created by the eye detection algorithm at the time when EyePhone is used for the first time. The template is saved in persistent memory of the device and is fetched each time EyePhone is invoked. To track eye the template matching function calculates a correlation score between the open eye template and search window. If the normalized correlation coefficient is equal to 4.0, signify the presence of eye in the search window. To detect eye blinks, two threshold values with maximum and minimum normalized correlation coefficients returned by the template matching function are used. The system is implemented on Nokia N810 mobile phone running with Maemo 4.1 operating system installed with C OpenCV (Open Source Computer Vision) library and EyePhone algorithms. To capture video frames from the camera, GStreamer the main multimedia framework on Maemo platforms is used. To determine the accuracy of eye tracking and blink detection algorithms, the performance of EyePhone is experimented under different conditions. It has been found that under different light conditions and user stationary positions, the result obtained are promising. But, the results are found of low accuracy while user is walking or the distance between eye and phone is larger. EyePhone is found more lightweight in terms of CPU and RAM usage. To save battery consumption, EyePhone becomes active only when a user eyes pair is detected while looking at mobile phone display otherwise remains inactive.EyePhone needs improvements in open eye template and filtering algorithms. The one-time template strategy might affect the accuracy under varying light conditions and movements of mobile phone. The template generation strategy should be modified to evolve template according to a user current context. Likewise, the filtering algorithm should be modified to incorporate learning approach instead of completely relying on thresholding technique in order to minimize false positives.

3.7.3 SenSay

SenSay (Sensing and Saying)[56] is a context aware mobile phone that uses a number of sensor mounted on various parts on a user body to extract context information and changes mobile phone behavior to adopt to the dynamically changing environment and physiological states. SenSay architecture consists of five functional modules: sensor box, sensor module, decision module, action module, and phone module. Sensor box consists of sensors and microcontroller and gathers sensors data for extracting contextual information. Software based sensor module polls sensors data from sensor box periodically, stores most current data in an array, and forwards it to the decision module. Decision module uses sensors data and user electronic calendar to conclude user state and command an action. Action module changes the settings and operations of mobile phone using command received form decision module. The basic operations action module performs are: ringer control (off/low/medium/high), vibration control (on/off), sending SMS to caller, making call suggestions, and providing access to electronic calendar. A SMS can be send to the caller in case if user is busy and also instructs caller to call back (after how much time) in case of an important call. Changes in mobile phone behavior are also communicated to user over mobile phone display which user can change manually. A SenSay state corresponds to a user current context which could be Uninterruptible, Idle, Active, and Normal (default state). Uninterruptible state is identified from electronic calendar and voice intensity as well as ambient noise data where a user should not be disturbed. Uninterruptible state has two sub-states: Light-On and Light-Off. A user will be in active state, if he/she is in high-activity physical motion. Accelerometer data and ambient noise sensors can be used to diagnose active state and mobile phone behavior would be set accordingly such as adjusting ringer volume to the loudest level and turning on vibration etc. A user in idle state would be interruptible. When a decision module could not accurately determines mobile phone state, the default state (normal state) would be determined where ringing volume and vibration would be set to default values and no suggestions would be triggered to users. Mobile phone's states are modeled in Moore Finite State Machine, where all states are interconnected and users can make transition from one state into another. Some states have higher priorities based on the information in pre-requisites such as electronic calendar entry has high weight as compared to high physical activity. In the current implementation decision, sensor, and action modules runs on a notebook computer running with windows 2000 and attached with sensor box and mobile phone via the RS232 serial connection. The sensor box contains PIC16F877 microcontroller, a voice microphone sensor (Emkay Innovative Products SP0101NC2-2), an ambient noise microphone sensor (Emkay Innovative Products MD6020ASC-0), accelerometer sensors (Motorola MMA1201P, Motorola MMA2200W), a temperature sensor (National LM35), and a visible light sensor (Panasonic PNA4603H Photo IC). The system has been tested under varying users' physical states and it has been observed that the system can clearly differentiate between users states using the sensory data.

3.7.4 I^2Navi

Indoor Interactive Navigation System (I^2Navi)[57] enables users to navigate within a building easily and conveniently using Near Field Communication (NFC) enabled mobile phones. Mobile phone reads information from a NFC tag embedded in a poster (i.e., called smart poster) to determine current location of a user. Captured information is used to display path directions to the required indoor location on the mobile phone screen. Main components of I^2Navi are: user interface module, NFC reader module, navigation module, database module, and information module. The NFC reader reads location information from NFC tag inside in smart poster whenever an Android based mobile phone running with I^2Navi is placed near it. Location information is transferred to user interface module which retrieves current location from the information module. Navigation module determines route using current location information and destination information retrieved from database module. The suggested rout is marked on the mobile phone display using user interface module. Information module provides additional information related to the destination to administrative staff to ensure the information is always updated. I^2Navi is implement on Android NFC enabled mobile phone. The NFC tags (NXP Semiconductor NTAG 203 (F)) is used in smart posters to store location information. A user has to tap "tap here" spot on his mobile phone screen to obtain his current location while I^2Navi is running. Similarly, user has to specify destination using I^2Navi user interface. I^2Navi is tested by giving task of locating a particular room in a building using the system. Results indicated that the participants were satisfied with performance of the system. All of the participants found I^2Navi simple and robust, and accurately determined their current locations and guide them in locating their desire destinations. Another similar application is NFC Internal[58] which exploits the mobile phone sensing and NFC technologies for indoor navigation. NFC Internal, however, is more innovative and powerful as compared to I^2Navi.

4. Mobile Phones Sensing Limitations/Challenges

Recent technological advancements in small inexpensive sensors, increased processing power, extended internal storage, enhanced operating systems, and improved internet connectivity etc. have attracted the attention of researchers and academia for developing high valued and remarkable people-centric mobile phone sensing applications for a wide variety of domains. Despite of the existence of a number of promising mobile phone sensing application, mobile phone sensing is still in its infancy and suffers from a number of technological and non-technological limitations/challenges. To meet these limitations/challenges, more efforts from the research community and academia are needed. Some of these limitations/challenges are:

4.1 Platform Support and Programmer Freedom

Modern mobile phones have been improved significantly in their computationally capabilities but they provide restrictive environment for application developers due to closed nature. Sophisticated operating systems are developed for mobile phones with advanced features but most of them (i.e., iOS etc.) are still not open and do not allow programmers to access their libraries and core programming modules. Operating systems might deny third party

Table 11. Compression of mobile phones sensing applications developed for human-phone interaction

App	Hardware	Software	Protocols	Sensors	Type	Product	Shortcomings
NeuroPhone[55]	iPhone, Emotiv EPOC EEG headset	Not available	WiFi	Biosensor	**Architecture** Distributed **Operation** Participatory **Scale** Personal	Prototype	Intermediate devices can be removed and decryption function can be directly moved to mobile phones. Several issues related to energy consumption, accuracy, and user's privacy are still needed to be solved.
EyePhone[54]	Nokia N810	Maemo 4.1, C OpenCV. GStreamer	None	Camera	**Architecture** Integrated **Operation** Participatory **Scale** Personal	Prototype	EyePhone has to deal with varying and uncertain conditions such as lighting problems, and mobility. Needs to implements improved function for capturing user eye pair which is difficult in certain situations. Needs to incorporate voice recognition algorithm and ensures its functionality in different user contexts.
SenSay[56]	PIC16F877 microcontroller, voice microphone sensor, ambient noise microphone sensor, accelerometer sensors, temperature sensor, and a visible light sensor, RS232 connectors	Windows 2000	None	Accelerometer, microphone, light, and motion	**Architecture** Distributed **Operation** Opportunistic **Scale** Personal	Prototype	SenSay should be implemented in software which would be more dynamic and provide more chances for enhancements. SenySay usage could be more problematic for users because of clipping external devices with their bodies and which could produce data capturing and interpretation problems. Outer dependencies should be removed by fully exploiting mobile phones technological improvements.
I^{2}Navi[57]	NFC enable Android mobile phone, NXP Semiconductor NTAG203 (F)	Android	Not available	NFC	**Architecture** Integrated **Operation** Participatory **Scale** Personal	Prototype	Accuracy of current location information depends on the distance between NFC tag and mobile phone. In case of a large building, a large number of NFC tags would need to be installed. Difficult to operate by visually impaired people.

mobile phone sensing applications for resources requests (i.e., accessing sensors etc.) and interrupt them at any time to not disrupt the normal functioning of mobile phones. Thus, resulting in to poor performance of the applications and lose of sensitive data. Therefore, programmers have to design high level programming codes to compensate the non-availability of operating system libraries which is a tedious and laborious task. Furthermore, the APIs provided by the mobile phones vendors provides a limited set of operations due to resources constraints and security concerns of mobile phones. APIs provided by the same manufacturer for the different models might not be portable and compatible which may result in missing or malfunctioning of new or existing components. APIs limitations cannot be solved by the programmers and they have to find some alternative ways for solving or eliminating APIs limitations.

4.2 Hardware Support

Most of the mobile sensing applications use advanced machine learning techniques (i.e., supervised and unsupervised) for refining and mapping sensory data into high-level phenomenon to perform accurate decision making. Machine learning techniques are complex and require high computational resources for execution. Therefore, mobile sensing applications majorly deploy machine learning algorithms execution on backend remote severs because they could jeopardize mobile phone and stop them from performing their normal functions. Modern mobile phones have shown tremendous technological advancements (i.e., duel-core and quad-core processor technology, RAM size in GBs, and SDcard in GBs support etc.) in the past few years but but still they are immature for executing machine learning techniques. Executing machine learning algorithms on remote servers and introducing duty cycles could reduce mobile phones load but they can lead to security and privacy issues, and sampling error which would produce inaccurate estimations. Therefore, more technological improvements are needed in mobile phones to fulfill mobile phone sensing applications needs.

4.3 Participatory Sensing or Opportunistic Sensing

What roles people should play in mobile phone sensing and decisions making processs? Using user involvement, mobile phone sensing can be either participatory or opportunistic as discussed in Section 2.1. These approaches are the two extremes having no commonality and having respective pros and cons. However, there exists tradeoff between participatory sensing and opportunistic sensing. Therefore, ways and scenarios should be defined to use both of the approaches in combination to get the best possible sensing and applications experiences.

4.4 Lack of Specialized Sensors

Although modern mobile phones have a number of sensors integrated in a single suit but they still have lack of specialized sensors (i.e., sensors used for special purposes such as sensor for measuring CO_2 in the air etc.), which could be used productively in a number of useful real world applications. Most of sensing applications developed (e.g., applications for health monitoring etc.) relies heavily on external special purpose sensors (e.g., biomedical sensors, wireless ECG sensors, temperature sensors, and humidity sensors etc.) to capture a real-world phenomenon and is forwarded to mobile phones using wireless protocols (e.g., Bluetooth etc) for advanced processings and decision makings. Certainly, these approaches are effective but they can lead to a number of security, privacy, placement, movement, and cost issues. Therefore, the number of sensors in mobile phones should be increased by integrating more and more specialized sensors in onboard features.

4.5 Accurate Sensors Selection

Sensing experiences of all of the sensors are not of the same quality. There are some sensors which might produce accurate estimations but may have other problems such as daring out battery power quickly or introducing time delays etc. Therefore, care must be taken while selecting sensors for capturing a real-world phenomenon. For example, GPS position estimations are error prone not only inside of buildings but also in other situations such as being in a user's pocket, and in "urban canyons" near tall buildings or tunnels. Therefore, low energy sensors (i.e., WiFi and cellular radio triangulation) should be used which could provide more accurate position estimations at low energy cost as compared to GPS.

4.6 Managing Heterogeneous and Multi-model Data Sources

As in mobile phone sensing applications, the data producers can be very in terms of modality (e.g., mobile phone internal sensors, external sensors, and web services etc.). Different types of sensors have different attributes and capabilities (i.e., they might have different accuracy in sensing physical and virtual world) and would produce plethora of heterogeneous data which would intensify the problem of too much data and no knowledge. Integrating information from diverse data sources adds difficulty to mobile phone sensing mining. Raw data from different sensor sources need to be transformed to the some matrices and represented in a standard form to facilitate the learning and inference process. Instead of reading from each sensor in an isolated way, a mobile phone sensing system should consider reading from multiple sensors in the data stream while modeling the behavior of an individual and group. The same sensor may sense the same event in time and space differently under different conditions and user context often leads to different inference results. Therefore, environmental conditions should be considered while deciding inferences from sensors data.

4.7 Mobile Phone Sensors Ontology

Smarting of mobile phone sensors data is essential for advanced analytical processing, integration, inferencing, and interpretation by the real world mobile phone sensing applications. Most of the researchers are calibrating sensors and associated data in their own ways for decision makings in mobile phones sensing applications which is leading to heterogeneity, complexity, and non-reusability problems. Mobile phone sensors ontology is increasingly needed for describing a common agreed dictionary of sensor-related terminology and representing relationships between them in a machine understandable form. Mobile phone sensors ontology should include terms for describing sensors characteristics (i.e., accuracy, resolution, and frequency etc.), observable properties (i.e., radiation, temperature, position, etc.), and terms for unfolding sensor distinctiveness, types, capabilities, platforms, roles, and interfaces. Mobile phone sensors ontology would enable to map low-level sensory data into high-level user contexts and bridge the semantic gap between the low-level individual activities and high-level social events using inference technique. In short, mobile phone sensors ontology would represent sensors data in a compact and semantic format which would be useful in every mobile phones sensing domain.

4.8 Information Security and user Privacy Protection

Most of the mobile phones sensing application uses distributed client-server architecture where mobile phone is used as a client sensing device and most of the processing and storage operations are preformed on remote web server. Mostly external specialized sensors are deployed for capturing rich sensing experiences users which is communicated with mobile phones via wireless technologies (i.e., Bluetooth etc.). Similarly, mobile phones forward the sensory information to a remote server using either cellular technologies (e.g., GSM, Edge, and GPRS etc.) or wireless technologies (i.e., WiFi etc.) for advanced processing and operations. Communicating information wirelessly in plaintext can be catastrophic because of its unsecure nature and a number of security attacks can take place during transit. To protect from malicious attacks new and efficient methods (i.e., encryption techniques) are needed to be explored for ensuring information security during wireless transit.

Respecting and ensuring the privacy of users should be the prime responsibility of a mobile phone sensing application. Sharing and revealing personal digital data could have a number of risks on user privacy. Compare to personal data (i.e., user profile and IDs), data gathered in community can reveal much more information about individual behavior. Most of the sensing applications primarily focus on the collection of large scale sensory data. Peoples are certainly very much curious about how sensors data is captured and used, especially if the sensory data contain enough information to reveal a user identity, location, speech, or sensitive video or picture. The increased sensing capabilities enabled mobile phones to capture information about different aspects of users and might result into compromise of their privacies. Mobile phones may sense information related to users presence, events, or experiences etc. which they may not want to share with others. Likewise, the sensed information may be intercepted by unauthorized people during transit and could be used unfairly. Privacy protection involves many elements, including identity (who is asking for the data?), granularity (how much does the data reveal about people? does it reveal one identity?), and time (how long the data will be retained?). Therefore, privacy of users should be ensured while allowing their devices to reliably contribute high-quality data to large-scale applications. Two possible ways to ensure privacy are: data anonymization and access control.

4.9 Battery Power Management

Despite of developments, battery power lifetime is still limited to meet the advanced needs of modern mobile phones sensing applications. Developers are needed to design applications which might not significantly change the operational life time of a standard mobile phone battery. To save energy and extend mobile phone battery life time, energy efficient duty-cycle strategies are needed to be explored (i.e., not currently provided by the APIs) for sensing and classification algorithms running on mobile phones, usage of power hungry sensors (e.g., Bluetooth, and GPS etc.), and components for uploading of sensory data to global repositories (e.g., GPRS etc.). Using sampling strategy for energy hungry sensors can be helpful but using GPS sampling less frequently or using a noisy sensor data (i.e., WiFi) can result in inaccurate estimations. Therefore, care must be taken while determining sampling rate to reduce power consumption but accuracy should not be harmed.

4.10 Building People Trust

The success of a people-centric mobile phone sensing application depends on the ratio of its adoptability and participation. Peoples are always very cautious about their privacy and often feel hesitated in exposing their personal, environmental, and contextual information to an application without having any prior satisfaction. Likewise people often do not believe on the information they obtain from sensing applications and do not consider them worthy to bring positive changes in their behaviors, movements, actions, and relationships. Therefore, new ways are needed to be founded to attain peoples' confidence and trust levels.

4.11 Standard Models and Toolkits

Mobile phone sensing applications development domain still has lack of standard models and development toolkits. Researchers and developers are creating applications in their own developed methodologies and strategies which are creating heterogeneity, diversity, complexity, and inconsistency in the domain. Therefore, standard models and powerful development toolkits are needed to be investigated to help researchers and developers in understanding the basic structure and features provided in a sensing application belonging to a domain and speed up the development process.

5. Conclusion

The potential power of mobile phone sensing has been realized due to widespread adoption of the sensors enabled mobile phones by the people across many demographics and cultures. Recent technological advances and integration of the high valued sensors in mobile phones have turned them into sensor nodes capable of collecting, processing, analyzing, distributing and presenting dynamic information about people physical states (i.e., health, activities, and social relationships etc.), environmental pollution, road and traffic conditions, commerce, and human-phone interaction etc. Exploiting the sensing capabilities of mobile phones could be helpful for people in a number of ways such as determining uncongested and smooth road path to a destination, purchasing products at the lowest possible prices, passing a person's physiological information to a nearby practitioner for immediate action, finding a location and path to a destination in an unfamiliar place, enriching peoples experiences with their mobile phones, and developing behaviors to be cautious about expanding environmental pollution etc. Understanding the importance of mobile phone sensing, a number of sensing applications are developed by the research community, academia, and organizations for solving real-world problems in different domains of peoples' lives.

In this survey paper, a detailed overview and critical evaluation of the on-hand mobile phone sensing applications is presented. The sensing applications are classified and organized into different real-world domains depending on their functionalities. The working of each of mobile phone sensing application is clearly stated, shortcomings are identified, and improvements are suggested. Furthermore, applications in a domain or sub-domain are compared and results are presented in a tabular format. We have presented a generic model for mobile phone sensing applications representing the primary components that a mobile phone sensing application should have to help in solving the heterogeneity, complexity, and reusability problems. We have also outlined a number of technological and non-technological issues and challenges which could affect mobile phones sensing paradigm and needs immediate solutions for enabling the technology to hold high market place. After analysis, we have found that mobile phones sensing paradigm has the potential to revolutionize peoples' lives by providing very simple, reliable, inexpensive, and timely solutions to plethora of their real-world problems. However, mobile phones sensing paradigm is still in its infancy. Therefore, to bring the vision of mobile phone sensing into reality, more efforts are needed to be invested by the research community, organizations, and academia.

6. Acknowledgement

This research work has been undertaken by the first author as a partial fulfillment of Ph.D degree with support of the Higher Education Commission (HEC) of Pakistan.

7. References

1. Bao X, Choudhury RR. VUPoints: collaborative sensing and video recording through mobile phones. ACM SIGCOMM Computer Communication Review. 2010; 40(1):100–5.
2. Ali S, Khusro S, Rauf A, Mahfooz S. Sensors and Mobile Phones: Evolution and State-of-the-Art. Pakistan Journal of Science. 2014; 66(4):386–400.
3. Yim J, Joo J. Smartphone Sensor Value Pattern Analysis with Neural Network. Indian Journal of Science and Technology. 2015; 8(26):1–7.
4. Grauballe A, Perrucci GP, Fitzek FHP. Introducing Contextual Information to Mobile Phones by External and Embedded Sensors. Proceedings of the International Workshop on Mobile Device and Urban Sensing - MODUS08, St. Louis, Missouri, USA. 2008.
5. Honicky R, Brewer EA, Paulos E, White R. N-SMARTS: Networked Suite of Mobile Atmospheric Real-Time Sensors. Proceedings of the second ACM SIGCOMM workshop on Networked systems for developing regions,Seattle, WA, USA. 2008. p. 25–30.
6. Kansal A, Goraczko M, Zhao F. Building a sensor network of mobile phones. Proceedings of the 6th International Conference on Information Processing in Sensor Networks, Cambridge, Massachusetts, USA. 2007. p. 547–8.
7. Kinkade S, Verclas K. Wireless Technology for Social Change: Trends in Mobile Use by NGOs. Access to Communication Publication Series. United Nations Foundation: USA, 2008.
8. Lane ND, Miluzzo E, Lu H, Peebles D, Choudhury T, Campbell AT. A survey of mobile phone sensing. IEEE Communications Magazine. 2010; 48(9):140–50.
9. Campbell AT, Eisenman SB, Lane ND, Miluzzo E, Peterson RA, Lu H. The Rise of People-Centric Sensing. IEEE Internet Computing. 2008; 12(4):12–21.
10. Lane ND, Eisenman SB, Musolesi M, Miluzzo E, Campbell AT. Urban sensing systems: opportunistic or participatory? Proceedings of the 9th workshop on Mobile computing systems and applications, USA. 2008; 11–6.
11. Khan WZ, Xiang Y, Aalsalem MY, Arshad Q. Mobile phone sensing systems : A survey. IEEE communications surveys and tutorials. 2013; 15(1):402–27.
12. Burke J, Estrin D, Hansen M, Parker A, Ramanathan N, Reddy S.. Participatory sensing. Proceedings of the Workshop on World-Sensor-Web (WSW'06) Mobile Device Centric Sensor Networks and Applications, Boulder, Colorado, USA. 2006; 117–34.
13. Consolvo S, McDonald DW, Toscos T, Chen MY, Froehlich J, Harrison B, et al. Activity sensing in the wild: a field trial of ubifit garden. Proceedings of the twenty-sixth annual SIGCHI Conference on Human Factors in Computing Systems, Florence, Italy. 2008. p. 1797–806.
14. Gao C, Kong F, Tan J. HealthAware: tackling obesity with health aware smart phone systems. Proceedings of the 2009 International Conference on Robotics and Biomimetics, Guilin, China. 2009. p. 1549–54.
15. Miluzzo E, Lane ND, Fodor K, Peterson R, Lu H, Musolesi M. Sensing meets mobile social networks: the design, implementation and evaluation of the CenceMe application. Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems, Raleigh, NC, USA. 2008. p. 337–50.
16. Bilandzic M, Banholzer M, Peev D, Georgiev V, Balagtas-Fernandez F, Luca AD. Laermometer: a mobile noise mapping application. Proceedings of the 5th Nordic Conference on Human-Computer Interaction: Building Bridges; Lund, Sweden. 2008. p. 415–18.
17. Maisonneuve N, Stevens M, Niessen ME, Steels L. NoiseTube: Measuring and mapping noise pollution with mobile phones. In: Athanasiadis IN, Mitkas PA, Rizzoli AE, Gomez JM, editors. Information Technologies in Environmental Engineering, Springer-Verlag: Berlin Heidelberg. 2009. p. 215–28.
18. Mohan P, Padmanabhan VN, Ramjee R. Nericell: rich monitoring of road and traffic conditions using mobile smartphones. Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems, Raleigh, NC, USA. 2008. p. 323–36.
19. Oliver N, Flores-Mangas F. HealthGear: A Real-time Wearable System for Monitoring and Analyzing Physiological Signals. Proceedings of the International Workshop on Wearable and Implantable Body Sensor Networks. BSN'06, Cambridge, MA. 2006; 61–4.
20. Jin Z, Oresko J, Huang S, Cheng AC. HeartToGo: A Personalized Medicine Technology for Cardiovascular Disease Prevention and Detection. Proceedings of the 4th IEEE/NIH Life Science Systems and Application Workshop (LiSSA), Bethesda, MD. 2009 Apr; 80–3.
21. Seek B-H, Sang-Sub P. Effects of Rescuers' using a Smartphone-Band on the Quality of Chest Compression during Cardiopulmonary Resuscitation-Measuredusing a Manikin. Indian Journal of Science and Technology. 2015; 8(18):1–6.
22. Sha K, Zhan G, Shi W, Lumley M, Wiholm C, Arnetz B. SPA: a smart phone assisted chronic illness self-management system with participatory sensing. Proceedings of the 2nd International Workshop on Systems and Networking

Support for Health Care and Assisted Living Environments, Breckenridge, Colorado. HealthNet'08. 2008; 1–3.

23. 23. Chen X, Ho C, Lim E, Kyaw TZ. Cellular Phone Based Online ECG Processing for Ambulatory and Continuous Detection. in Proceedings of Computers in Cardiology, Durham, NC. 2007 Sept; 653–56.
24. Ogden CL, Carroll MD, Kit BK, Flegal KM. Prevalence of Obesity in the United States, 2009–2010. In: Services USDoHaH, editor. Hyattsville: NCHS Data Brief, 2012.
25. Denning T, Andrew A, Chaudhri R, Hartung C, Lester J, Borriello G, et al. BALANCE: Towards a Usable Pervasive Wellness Application with Accurate Activity Inference. Proceedings of the 10th workshop on Mobile Computing Systems and Applications,Santa Cruz, California, USA. 2009; 1–6.
26. De Oliveira R, Oliver N. TripleBeat: enhancing exercise performance with persuasion. Proceedings of the 10th International Conference on Human Computer Interaction with Mobile Devices and Services, Amsterdam, Netherlands. 2008. p. 255–64.
27. Wang R, Chen F, Chen Z, Li T, Harari G, Tignor S. StudentLife: assessing mental health, academic performance and behavioral trends of college students using smartphones. Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing,Seattle, Washington. 2014. p. 3–14.
28. Haidar GA, Achkar R, Maalouf R. Smart Walker Using Android Application. International Journal of Computing and Digital Systems. 2015; 4(3):175–82.
29. Kanjo E, Benford S, Paxton M, Chamberlain A, Fraser DS, Woodgate D. MobGeoSen: facilitating personal geosensor data collection and visualization using mobile phones. Personal Ubiquitous Computing. 2008;12(8):599–607.
30. Dockery DW, Pope CA, Xu X, Spengler JD, Ware JH, Fay ME. An Association between Air Pollution and Mortality in Six U.S. Cities. The New England Journal of Medicine. 1993; 329:1753–59.
31. Prather KA. Our Current Understanding of the Impact of Aerosols on Climate Change. ChemSusChem. 2009; 2(5):377–79.
32. Poduri S, Nimkar A, Sukhatme GS. Visibility Monitoring using Mobile Phones, Annual Report: Center for Embedded Networked Sensing. 2010; 125–27.
33. Mohan P, Padmanabhan VN, Ramjee R. TrafficSense: Rich Monitoring of Road and Traffic Conditions using Mobile Smartphones, Technical Report, 2008; 29.
34. Thiagarajan A, Ravindranath L, LaCurts K, Madden S, Balakrishnan H, Toledo S, et al. VTrack: accurate, energy-aware road traffic delay estimation using mobile phones. Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems,Berkeley, California, USA. 2009. p. 85–98.
35. Das T, Mohan P, Padmanabhan VN, Ramjee R, Sharma A. PRISM: platform for remote sensing using smartphones. Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services, San Francisco, California, USA. 2010. p. 63–76.
36. Bulusu N, Chou CT, Kanhere S, Dong Y, Sehgal S, Sullivan D, et al., Participatory sensing in commerce: using mobile camera phones to track market price dispersion. UrbanSense'08. 2008; 6–10.
37. Sehgal S, Kanhere SS, Chou CT. MobiShop: Using Mobile Phones for Sharing Consumer Pricing Information. Demo Session of the Intl Conference on Distributed Computing in Sensor Systems,Santorini, Greece. 2008. p. 1–2.
38. Dong YF, Kanhere S, Chou CT, Bulusu N. Automatic Collection of Fuel Prices from a Network of Mobile Cameras. Proceedings of the 4th IEEE International Conference on Distributed Computing in Sensor Systems, Santorini Island, Greece. 2008. p. 140–56.
39. Dong YF, Blazeski L, Sullivan D, Kanhere SS, Chou CT, Bulusu N, editors. PetrolWatch: Using Mobile Phones for Sharing Petrol Prices. Proceedings of the 7th Annual International Conference on Mobile Systems, Applications and Services,Krakw, Poland. 2009 Jun.
40. Dong YF, Kanhere S, Chou CT, Liu RP. Automatic image capturing and processing for PetrolWatch. Proceedings of the 2011 17th IEEE International Conference on Networks. Singapore. 2011. p. 236–40.
41. Deng L, Cox LP. LiveCompare: grocery bargain hunting through participatory sensing. Proceedings of the 10th workshop on Mobile Computing Systems and Applications; Santa Cruz, California. 2009; 1–6.
42. Jain J, Bhatti N, Baker H, Chao H, Dekhil M, Harville M, et al. Color match: an imaging based mobile cosmetics advisory service. Proceedings of the 10th International Conference on Human Computer Interaction with Mobile Devices and Services, Amsterdam, Netherlands. 2008. p. 331–34.
43. Ankolekar A, Szabo G, Luon Y, Huberman BA, Wilkinson D, Wu F. Friendlee: a mobile application for your social life. Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services, Bonn, Germany. 2009. p. 1–4.
44. Beach A, Gartrell M, Akkala S, Elston J, Kelley J, Nishimoto K, et al. WhozThat? evolving an ecosystem for context-aware mobile social networks. IEEE Network. 2008; 22(4):50–5.
45. Beach A, Gartrell M, Ray B, Han R. Secure SocialAware: A Security Framework for Mobile Social Networking Applications,Technical Report, University of Colorado, Boulder, USA. 2009 Jun.
46. Cox LP, Dalton A, Marupadi V. SmokeScreen: flexible privacy controls for presence-sharing. Proceedings of the 5th

International Conference on Mobile Systems, Applications and Services, San Juan, Puerto Rico. 2007. p. 233–45.
47. Bao X, Choudhury RR. MoVi: mobile phone based video highlights via collaborative sensing. Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services,San Francisco, California, USA. 2010. p. 357–70.
48. Gartrell CM. SocialAware: Context-Aware Multimedia Presentation via Mobile Social Networks, University of Colorado: Boulder, USA, 2008.
49. Klepal M, Weyn M, Najib W, Bylemans I, Wibowo S, Widyawan W, et al. OLS: opportunistic localization system for smart phones devices. Proceedings of the 1st ACM workshop on Networking, systems, and applications for mobile handhelds, Barcelona, Spain. 2009; 79–80.
50. Eagle N, Pentland A. Social Serendipity: Mobilizing Social Software. IEEE Pervasive Computing. 2005; 4(2):28–34.
51. Dai J, Bai X, Yang Z, Shen Z, Xuan D. PerFallD: A Pervasive Fall Detection System Using Mobile Phones. Proceedings of the 8th IEEE International Conference on Pervasive Computing and Communications Workshops,PERCOM Workshops,Mannheim. 2010. p. 292–97.
52. Sposaro F, Tyson G. iFall: An Android Application for Fall Monitoring and Response. Proceddings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society,Minnesota, USA. 2009 Sep. p. 6119–22.
53. Dai J, Bai X, Yang Z, Shen Z, Xuan D. Mobile phone-based pervasive fall detection. Personal Ubiquitous Computing. 2010; 14(7):633–43.
54. Miluzzo E, Wang T, Campbell AT. EyePhone: activating mobile phones with your eyes. Proceedings of the second ACM SIGCOMM workshop on Networking, systems, and applications on mobile handhelds,New Delhi, India. 2010; 15–20.
55. Campbell A, Choudhury T, Hu S, Lu H, Mukerjee MK, Rabbi M, et al. NeuroPhone: brain-mobile phone interface using a wireless EEG headset. Proceedings of the second ACM SIGCOMM workshop on Networking, systems, and applications on mobile handhelds,New Delhi, India. 2010; 3–8.
56. Siewiorek D, Smailagic A, Furukawa J, Krause A, Moraveji N, Reiger K, et al. SenSay: A Context-Aware Mobile Phone. Proceedings of the 7th IEEE International Symposium on Wearable Computers. 2003; 248–57.
57. Choo JH, Cheong SN, Lee YL. Design and Development of NFC Smartphone Indoor Interactive Navigation System. World Applied Sciences Journal. 2014; 29(6):738–42.
58. Ozdenizci B, Coskun V, Ok K. NFC Internal: An Indoor Navigation System. Sensors. 2015; 15(4):7571–95.