Towards Distributed Real-Time Physiological Processing in Mobile Environments

James Meneghello
School of Information Technology
Murdoch University
Perth, Western Australia
Email: J.Meneghello@murdoch.edu.au

Kevin Lee
School of Information Technology
Murdoch University
Perth, Western Australia 6150
Email: Kevin.Lee@murdoch.edu.au

Kiel Gilleade
School of Natural Sciences and Psychology
Liverpool John Moores University
Liverpool, UK
Email: K.M.Gilleade@ljmu.ac.uk

***Abstract*—Physiological monitoring has been used in a wide range of scenarios to assist in disease diagnosis, athlete monitoring and other activities. There are also many opportunities in analysing aggregate data from groups of people rather than individuals such as public event monitoring or athletic team performance optimisation. Numerous difficulties exist pertaining to this, particularly concerning how to process and transform the resulting physiological data in real-time when many devices are producing data. This paper proposes a system that is designed to monitor, analyse and report physiological data in real-time by leveraging mobile devices as distributed processors.**

I. Introduction

Physiological monitoring can provide a plethera of health and fitness data in real-time. Useful measures such as pulse, heart rate and blood oxygen levels can be used successfully to diagnose and aid in the treatment of such diseases as sleep apnoea [1] and other respiratory disorders, as well as assisting in the monitoring of cardiac systems [2]. By utilising readily-available sensor devices with low power requirements, long-term monitoring can enhance the quality of care for patients and providing numerous advances to research associating physiological markers with mental states, such as the measurement and analysis of stress levels [3]. Many physiological monitors can be used to derive useful information from subjects toward a wide range of potential uses.

As well as monitoring individuals, it is possible to examine interconnected physiological monitoring devices operating in a group context. By analysing and aggregating data from groups, conclusions can be drawn regarding the state of the entire group. This technique has a wide range of possible usage scenarios, such as monitoring the life-signs of miners in underground tunnels in order to hasten awareness of emergency situations [4], or monitoring a population for symptoms of influenza to locate outbreaks [5]. Applications for this technique can also be found in monitoring of groups for crowd control purposes, tracking of vehicles for civic planning purposes and enhanced targeting of advertisements.

There are opportunities to be leveraged with real-time processing of physiological data on portable devices. Unfortunately, while the processing capability of mobile CPUs has increased dramatically over recent years, power drain and storage remain bottlenecks. In order to alleviate these issues, real-time processing could be packaged and distributed to other nodes within the local area that are more capable of handling the work. If nodes are unable to handle the workload, data can be passed upstream to the Cloud which can efficiently process the work in the required time frame. Other factors, such as the available bandwidth and/or an inaccessible connection that can make distribution to Cloud resources difficult or impossible to utilise, forcing the devices to rely on each other to process data in as timely a manner as possible.

This paper proposes a system that aims to support the packaging and distribution of real-time physiological processing across a diverse selection of devices by dynamically allocating resources based on contextual/environmental data and specific rulesets. Any transformations required by the devices should be supported and processed in a pipeline that provides the best compromise between efficiency and timeliness. Considerations such as the current battery state in the group of devices and the required computation govern the decisions that need to be made in distributing physiological processing.

The remainder of this paper is as follows. Section II provides a background of physiological monitoring. Section III introduces three scenarios that illustrate the usefulness of the system. Section IV proposes the design of a system to package and distribute the workflow amongst devices, and Section V analyses the suitability of the system to the proposed scenarios. Section VI highlights specific issues and solutions. Finally, Section VII concludes.

II. Background

The monitoring of physiological markers for specific purposes has been utilised extensively in recent years. However, most of these uses have been limited to specialised equipment in hospitals and health care clinics due to expense or because they are simply not portable. Advancements in certain types of monitoring equipment (such as electrocardiogram monitors [6]) have allowed for them to be used in different environments (e.g. mobile), for purposes beyond simple diagnosis and health monitoring procedures. Communications are made trivial by devices that can utilise either a mobile phone network [1] or home wireless internet connection [2] that allows the devices to be monitored remotely. Software updates can also be automatically deployed to suit future purposes without requiring physical interaction by a technician.

As noted, physiological monitoring can be used for a number of other purposes and is not limited to health care. Athletes can be monitored in real-time for performance and referee-assist services [7] by attaching sensors monitoring heart rate and position. Monitoring levels of fatigue in drivers by using a neural network to analyse eye shape and position [8] can also assist in improving road safety. Monitoring equipment is progressing to a point where it is more portable than ever before [9], which allows its use in a wider range of scenarios while being minimally invasive to the wearer.

In order to analyse physiological data, a number of filters and transformations often need to be executed [10]. This can be as simple as converting an inter-beat interval to heart-rate or a sequence of complex algorithms that can require more processing power, such as extracting the inter-beat interval from an electrocardiogram and transforming it into its spectral properties for the purposes of heart rate variability analysis. To process these transformations in a responsive manner, adequate computer resources must be accessible.

Some research has been attempted specifically involving group physiological monitoring [11]. However, a significant amount of groundwork required has been completed; useful research has been completed to allow for the synchronisation of real-time data sources [3], as well as several techniques providing means of routing [7], [12] and management of wireless sensor networks [13]–[15] that can be utilised. By combining these network organisational techniques with our own methods for other components of our theoretical networking stack, a complete solution could be developed.

There remains a number of problems with physiological monitoring, particularly focused on power supply and consumption [16]. Performing transformations upon collected data in real-time is a significant strain on resources, so any improvement in efficiency through intelligent distributed processing is both desirable and achievable [17]. Power consumption is also affected considerably by wireless transmission specifications, so devices with lower power consumption also tend to have low transmission ranges [11]. In order to save additional resources, it is possible to dynamically select the mode of communications [18] in order to achieve significant power consumption reductions.

III. Distributed Physiological Processing

Physiological monitoring is a useful method of deriving data from subjects for a range of purposes, including measuring stress and activity levels amongst others. However, there remains further opportunities in group physiological monitoring - particularly in determining crowd behaviour from physiological markers. In order to effectively transform the data in real-time and aggregate it, significant processing power is required - processing that is not necessarily always available. In order to alleviate the strain on external resources for this method of analysis, the devices collecting and transmitting the physiological data should be able to handle some or all of the processing occurring. For the system to cope with devices with different levels of ability or power resources, processing should be distributed across all devices such that their weighted load provides a more efficient use of group resources.This would allow the system to monitor groups for longer and with greater responsiveness than otherwise possible, which is an important consideration in certain situations.

To ease definition and description of the system being proposed, three scenarios have been designed that would allow the system to demonstrate its benefits in areas that would otherwise encounter problems if group physiological monitoring were attempted. Such scenarios cover a range of different potential applications, from safety monitoring to crowd control and general monitoring for the purposes of enhancing other services (such as advertising or camera control); as follows.

A. Case: Athlete Performance and Service Delivery

There are a number of opportunities in monitoring the physiological data being produced by athletes actively participating in a group sport such as football. By examining localised player groups for levels of exertion, it is possible to focus on areas most likely to be of interest to spectators, such as complex ball plays and brawls. While the ball is not directly involved in these events, such situations tend to be desirable to spectators. Other physiological data can be helpful to team managers, such as general player well-being - the ability to diagnose player fatigue early is desirable, as they can be substituted for more rested players.

As noted in previous studies [7], there are often problems with monitoring players. Communications tend to be sporadic and limited, and attenuation from player's bodies often prevent connections from occurring to some base stations. There is the possibility of device damage occurring due to player contact or other forces applied to the monitoring devices themselves. While physical damage mitigation lies outside the realm of this paper, connection interruption can be worked around.

Because of the nature of the proposed system, processing of physiological transformations can take place at any stage of the device communications pathway. As an example, if a player's monitoring device was unable to contact either; a) a nearby wireless base station; or b) another player that was able to forward the data to the base station, transformations could be completed on the player's device (or distributed among available player devices) with results transmitted when the connection becomes available again. This allows services to remain real-time, as an upstream processor is not required to quickly transform data that was unable to be transmitted - potentially several seconds worth.

Distribution of processing would also allow power management to be handled more effectively. Players that have been on the field for longer periods of time could encounter power supply issues to monitoring devices, so distribution of processing to other devices with more plentiful resources could occur - or, if latency allowed, Cloud resources could be dynamically allocated to compute the transformations.

B. Case: Shared Experiences in Crowds

Aggregation of physiological data taken from crowds could potentially be used to enhance shared experiences in crowds at sporting or cultural events (e.g. musical festivals). Measures of autonomic arousal and movement can be used to provide real-time feedback to the crowd about the state of the people around them which can potentially improve the shared experience of an event (e.g. improved feelings of co-presence [19]).

Distribution of wristbands with built-in monitoring and wireless capability would allow organisers to collect data to provide a visual feedback to attendees of the crowd experience. Wireless base stations would provide connectivity to a wide area of the event, while peer-to-peer connections would provide connections to devices outside specified areas. The devices would monitor physiological signs and transform the raw data as appropriate either individually or in a distributed manner across other devices and available Cloud resources, before sending them to be aggregated.

Distribution of processing becomes essential in situations such as these, where base stations are likely to be positioned sparsely, and there is unlikely to be significant processing resources to maintain real-time transformations. Devices need to assist in routing to base stations for other nodes, and act as supervisors for local node groups to effectively organise processing distribution amongst devices with available resources.

C. Case: Monitoring Miner Safety

Distributed physiological processing can also be extremely useful for safety monitoring in areas with intermittent or weak connections to headquarters - particularly in labyrinthine mining tunnels and similar areas. Physiological monitoring is already proposed in mining scenarios, particular in rescue scenarios [4], although the proposed solutions require a large amount of on-site infrastructure. While monitoring devices and processing equipment would likely represent a significant cost to a resources corporation, the potential for improved safety (and therefore decreased fines/penalties) may justify the cost.

Miners equipped with monitoring devices and small processing units that could be built into existing safety equipment. Large spikes in heart-rate and other indicators within a localised area could indicate a problem not visible to other sensors. If no connection to headquarters is available, units would collaboratively processes the workload and alert localised alarm units to indicate a problem - even without connection to or interaction with superiors. A monitoring system could also include other types of sensors looking for issues with air quality or poisonous gasses, able to be entirely processed by the mobile units. By collating data from units, it would be possible to locate areas with poor habitability and alert workers to avoid certain areas. While workers are likely to hold higher power reserves for processing than other scenarios, efficient distribution of work processing would allow for more accurate monitoring, as more available processing could cater for lower intervals between sensor readings.

IV. Design

A. Overview

The proposed system has been designed as a standard networking stack as seen in Table I, inheriting a number of existing standards and protocols for dealing with tasks such as routing, addressing and physical media access. Existing techniques suffice for our purposes, with the exception of a modified network layer, incorporating dynamic radio switching depending on distance and required bandwidth - a measure taken to save a substantial amount of energy.

TABLE I
Proposed Implementation

Application	Distributed Physiological Processing protocol
Transport	TCP
Network	IPv4, CoolSpots [18]
Link	Dynamic 802.11b/g/n, Bluetooth

The proposed system is a generic architecture for real-time physiological transformations that is designed to manage the packaging and distribution of processing to other devices. Any available device can be utilised for processing, including similar nearby monitoring devices, wireless base stations and Cloud resources. Figure 1 represents the proposed usage architecture of the system, including optional components (such as the super-nodes and available Cloud resources).

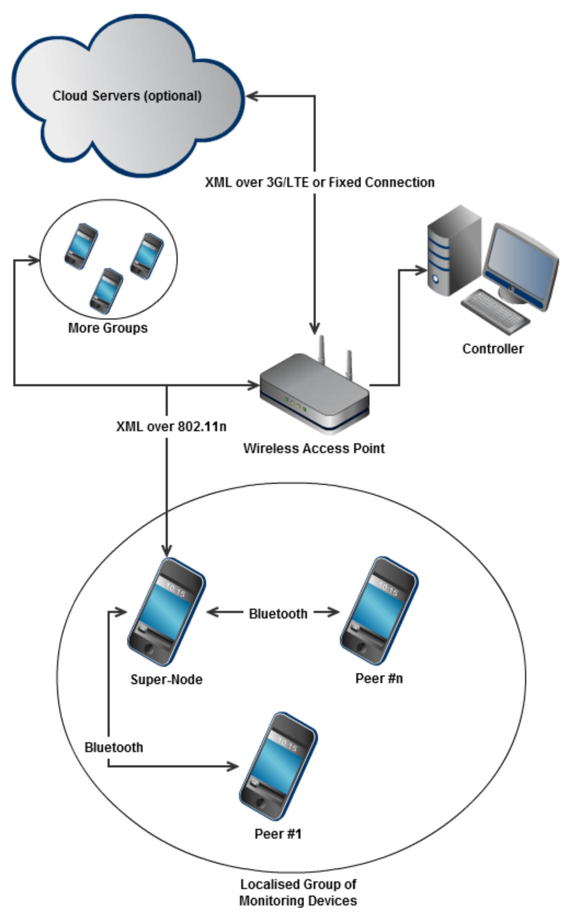

Fig. 1. Proposed Architecture for Distributed Real-Time Physiological Processing

B. Architecture

In order to effectively manage groups of sensor nodes, a hierarchical peer-to-peer network architecture is used to group monitoring devices for data collection and processing, as represented in Figure 1. Within each group of sensor nodes, a super-node is selected. Determined amongst nodes by a number of metrics (such as remaining power resources, processing ability and strength of connection to both other nodes and upstream), the super-node manages load distribution, connections to other super-nodes and data aggregation for upload. In the event that a super-node is disabled or leaves the group, a new super-node is selected. If other supervising devices are available (such as wireless base-stations), these static supervisors effectively replace the role of a super-node in the system.

Individual nodes performing processing as detailed by the super-node, which would often be their own processing. If resources on other nodes are depleted, the super-node would re-allocate work to other nodes with abundant resources, and processing can continue unimpeded. If all nodes have depleted resources, processing is restricted to collection and transmission - the super-node would then send all results upstream to Cloud resources for processing. The benefits and disadvantages of processing at each of these layers is modelled in Figure 2. Importantly, it is noted that latency and difficulty of access increases dramatically as processing is shifted further away from individual nodes, as fast and reliable network connections can be difficult to procure in some usage scenarios.

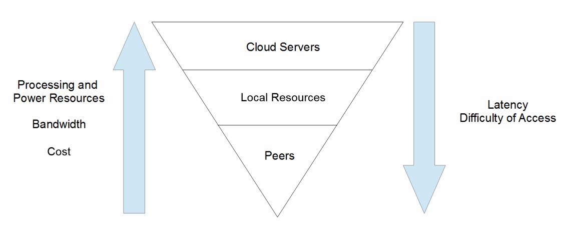

Fig. 2. Benefits and Disadvantages of Processing Layers

In order to effectively load-balance the swarm, nodes are also required to report their current state to the super-node. This includes details such as current load, remaining power resources, current processing details and others. Because the stream registration system already exists for data collection and transmission purposes between client and server, minimal effort is required to extend it to include reporting of device state to the super-node.

The super-node balances load and resources by monitoring the state of clients, allowing it to distribute work to low-load devices with plentiful resources. The super-node also keeps aggregate state details, such as an average power resource level of the group. This provides the super-node with supplementary data, allowing it to decide if the swarm is unable to cope with current work-load for a sustained period of time. In this event, the super-node would trigger offloading of work-load to Cloud resources (if accessible). Logic such as this is decided by pre-determined decision trees, as described in Section VI. The trees are able to be updated over-the-air by controllers, in the case of change situational parameters. The super-nodes examine the trees to make decisions about distribution to individuals or groups.

V. Applications

The proposed processing framework would, in its generic form, handle all three example studies with a minimum of alteration. While minor redevelopment for individual device types may be necessary, a substantial part of core functionality is platform-agnostic. The software itself can also be easily adapted to unusual situations, if required. It is expected that fine-tuned decision tree modifications should enable the majority of situations to be supported by the system as desired.

A. Athlete Performance

Due to the fixed area of operation involved in this scenario, wireless base stations with considerable processing power and virtually limitless power resources are able to be utilised. By positioning these stations as processing nodes with an overarching controller, devices attached to players can be made smaller and lighter. Because they have no need to individually process data, they can be placed in a monitoring-only role - advantageous due to the physical nature of the sport and potential damage to the unit if it were made bulkier with higher processing capabilities.

The application interface can also allow for a range of information to be delivered to different sources as authentication permits. Heart rate and other direct physiological markers can be delivered directly to the teams coaching and medical staff for review, while aggregated data could be made available to camera crews for use in partially-automated filming.

B. Shared Experiences in Crowds

Several challenges are introduced by this scenario; a generally fixed area but with uneven population distribution, varying levels of connectivity to devices and limited power resources. The system should handle the scenario with optional extras implemented and modifications to the primary decision tree to emphasise battery conservation except in times of interrupted connectivity.

Swarm super-node selection and setup may be required for the system to operate correctly, due to the possibility of groups moving out of range of base stations. In the event that a base station ceased operation or a group of subjects were not within communications range, the swarm would select a super-node to manage distribution of processing and report back to the controller whenever possible. Communications could be re-established either through movement of the group back within range of a station, or bunny-hopping through multiple other devices until any device was able to communicate with the controller. Once a connection isestablished, the full backlog of data can be transmitted to the controller for analysis.

In order to prioritise power resources over processing, the decision tree would be modified to emphasise the use of distributed processing amongst peers only in the event of interrupted connectivity, preferring instead to pass data up the chain to more capable processing resources. Combined with swap and/or recharge stations, monitoring devices could more easily last the full length of events described in this scenario.

C. Miner Safety

Adaptation of the system for this scenario follows similar principles to the otherscenarios, as population density is reduced and connectivity becomes sparse. Equipment supporting higher processing ability and more power resources is able to be integrated into safety harnesses and mining equipment.

Due to the importance of maintaining connections in safety-conscious situations such as mining, it is recommended that wireless relays are situated throughout the mines to ensure controllers have connectivity at all times. Unfortunately, bandwidth is not necessarily plentiful with such solutions, so transformations should still be distributed amongst equipment available on mining and safety rigs and only resulting data should be transmitted to controllers.

VI. Issues/Discussion

This section highlights important implementation issues.

A. Implementation Details

For ease of use and compatibility reasons, the processing framework is implementaiton as a standalone Python application utilising the Twisted Matrix Internet development library. The resulting software should execute on any Cloud platform, PC-compatible system or (with minor alterations) Android smart-phone, allowing a broad range of devices to be deployed to support the operation of the system. Programmatic and manual connections are be made to servers through a RESTful web API, allowing connections by any software that implements the correct interface. Monitoring devices also register users, streams and transmit data using this API, as well as hosting their own interface through which users may request connections to specific peers - lessening potential load on the selected super-node or other server resource.

B. Resource Allocation

Determining the quantity of work to allocate to a single node can be a difficult task. There are models able to manage resource allocation, from auction-based approaches [20] to process scheduling in high-performance computing [15]. Techniques used within these models can be adapted to provide resource allocation in the proposed system, which similarly consists of a group of heterogeneous devices with varying resources levels. In order to balance workload across a group, individual nodes report their state to the supervisor in an identical format to normal physiological data. The node registers a stream containing its state to the supervisor, which in turn uses the data to make decisions about work allocation.

To maintain a fair balance of work, current system load and remaining battery resources should be reported by the node to the supervisor. Work should be allocated in such a way that individual node load should be less than 1.0, while remaining battery lifetime should be approximately equal across devices. This should distribute work to more powerful devices in greater amounts, but without compromising battery life.

In the event that aggregate swarm load is greater than 1.0 or remaining power resources are less than a user-specific amount, processing should be distributed upstream to available processing resources. If available, this would be on-site computing hardware to ensure lowest possible cost and latency. If hardware is not accessible on-site but sufficient Internet bandwidth is available, transformations should be distributed to Cloud Computing resources [21].

C. Transformation Distribution

Distribution of transformations involves requests being sent in XML. Likewise, data to be processed by transformations can be sent using XML or (if necessary) using a simple TCP socket to reduce processing overhead. All post-transformation data is available externally by sending specific requests in XML format to the controllers RESTful [22] API.

In normal usage, a node registers with the nearest available super-node or base station. After receiving authentication tokens, the node registers data-streams with the super-node containing physiological data including its state. Any desired transformations is specified to the super-node, which will distribute work as appropriate amongst nodes or other resources. All transformed data is then sent upstream to the controller, though requests for raw data may still be made to the super-nodes. Finally, any aggregated information required externally by the end-user may be requested from the controller.

D. Decision Trees

In order to provide a customisable decision process for use in a range of environments, a decision tree [23] can be utilised by the system to aid in optimisation of system parameters. Ideally, these would be modified by the controllers (either at run-time or during setup) to provide the best possible balance of system responsiveness and resource usage.

Anexample of decision tree usage is for prioritising battery life over processing ability by initialising monitors in monitoring-only mode, which would only pass data upstream to be transformed. If no other processing was available, devices would default to working in a co-operative fashion to transform data for transmission to controllers. A hybrid approach can also be used, with individual resources being used until such a time that either power or processing is depleted, and data is sent upstream. This is desirable if more plentiful processing facilities had a higher cost; for example, using Cloud computing (which incurs costs incrementally as resources are used) as the primary transformation processor.

The decision trees are implemented using a standard XML tree format, which is timestamped and distributed to all peers - while standard monitoring peers would not utilise it, any peer later selected as a super-node would. A simple example of a decision tree in XML format is described in Figure 3.

- Individual
  - $CurrentBattery \leq 20 : MonitoringOnly$
- Group
  - $CurrentBattery \leq 40 : Redistribute$
  - $CurrentLoad \geq 1.0 : DistributeToCloud$

Fig. 3. An Example Decision Tree

E. Super-node Selection

If the system is required to operate standalone without the assistance of a dedicated super-node (such as a wireless base station), the swarm of monitors must decide amongst themselves which peers will act as supervisors. A number of techniques [24] [25] have been developed to cover this eventuality in other peer-to-peer networks, which can be adapted to suit the proposed system. Additional factors will need to be included in decision-making; remaining battery capacity, available bandwidth to controllers and overall processing ability. As the supervisor, the super-node acts as a load-balancer and primary distributor of transformation workloads across the swarm, and must be kept up-to-date as to the swarm state at any given time. For the purposes of this system, normal supervisor software (such as that used on base stations) should be easily adapted to run on any monitoring hardware, allowing for dynamic allocation of super-node status.

VII. Conclusion

This paper describes a number of case studies in which the proposed system could potentially provide data for the purpose of increasing safety or productivity. By leveraging devices within a system for use as distributed processing clients, we re-use resources in an efficient manner to transform physiological data in real-time. This allows for the aggregation and analysis of physiological data covering a whole population, rather than individuals, which opens up a significant range of applications.

The system is designed with the aim of extensibility through the implementation of new physiological transformations, and adaptability through the modification of decision trees. These decision trees would allow the controllers to prioritise device lifetime through minimisation of power expenditure, or real-time processing through distribution of transformation tasks to dedicated processing hardware, such as Cloud resources. This system enables the use of physiological transformations that can be used to improve a range of scenarios.

References

[1] N. Oliver and F. Flores-Mangas, "HealthGear: a real-time wearable system for monitoring and analyzing physiological signals," in *International Workshop on Wearable and Implantable Body Sensor Networks, 2006. BSN 2006*. IEEE, Apr. 2006, pp. 4 pp.–64.

[2] G. G. Mendoza and B. Q. Tran, "In-home wireless monitoring of physiological data for heart failure patients," in *Engineering in Medicine and Biology, 2002. 24th Annual Conference and the Annual Fall Meeting of the Biomedical Engineering Society Conference, 2002. Proceedings of the Second Joint*, vol. 3. IEEE, Oct. 2002, pp. 1849– 1850 vol.3.

[3] E. Jovanov, D. Raskovic, A. O. Lords, P. Cox, R. Adhami, and F. Andrasik, "Synchronized physiological monitoring using a distributed wireless intelligent sensor system," in *Proceedings of the 25th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2003*, vol. 2. IEEE, Sep. 2003, pp. 1368– 1371 Vol.2.

[4] Zephyr Technology, *CASE STUDY: Zephyr Provides Physiological Monitoring of Chilean Miners During San Jose Mine Rescue Operation*, 2010. [Online]. Available: http://www.zephyr-technology.com/resources/case-studies

[5] K. Gilleade and K. Lee, "Issues inherent in controlling the interpretation of the physiological cloud," 2011.

[6] J. Bai, Y. Zhang, D. Shen, L. Wen, C. Ding, Z. Cui, F. Tian, B. Yu, B. Dai, and J. Zhang, "A portable ECG and blood pressure telemonitoring system," *IEEE Engineering in Medicine and Biology Magazine*, vol. 18, no. 4, pp. 63–70, Aug. 1999.

[7] A. Dhamdhere, H. Chen, A. Kurusingal, V. Sivaraman, and A. Burdett, "Experiments with wireless sensor networks for real-time athlete monitoring," in *2010 IEEE 35th Conference on Local Computer Networks (LCN)*. IEEE, Oct. 2010, pp. 938–945.

[8] W. Rong-ben, G. Ke-you, S. Shu-ming, and C. Jiang-wei, "A monitoring method of driver fatigue behavior based on machine vision," in *IEEE Intelligent Vehicles Symposium, 2003. Proceedings*. IEEE, Jun. 2003, pp. 110– 113.

[9] W. Chang, T. Fang, and Y. Lin, "Characterization and fabrication of wireless flexible physiological monitor sensor," *Sensors and Actuators A: Physical*, vol. 143, no. 2, pp. 196–203, May 2008.

[10] J. Healey and R. Picard, *Digital processing of affective signals*. IEEE, 1998, vol. 6, pp. 3749–3752.

[11] E. Jovanov, A. O'Donnell Lords, D. Raskovic, P. G. Cox, R. Adhami, and F. Andrasik, "Stress monitoring using a distributed wireless intelligent sensor system," *IEEE Engineering in Medicine and Biology Magazine*, vol. 22, no. 3, pp. 49– 55, Jun. 2003.

[12] Y. Y. Chen, "The development of wireless sensor network for ECG monitoring," http://ir.lib.nctu.edu.tw/handle/987654321/33957, 2006.

[13] H. Ren, M. Q. Meng, and X. Chen, "Physiological information acquisition through wireless biomedical sensor networks," in *2005 IEEE International Conference on Information Acquisition*. IEEE, Jul. 2005.

[14] P. S. Pandian, K. P. Safeer, P. Gupta, D. T. Shakunthala, B. S. Sundersheshu, and V. C. Padaki, "Wireless sensor network for wearable physiological monitoring," *Journal of Networks*, vol. 3, no. 5, May 2008.

[15] Y. Lee and W. Chung, "Wireless sensor network based wearable smart shirt for ubiquitous health and activity monitoring," *Sensors and Actuators B: Chemical*, vol. 140, no. 2, pp. 390–395, Jul. 2009.

[16] Y. Lin, I. Jan, P. C. Ko, Y. Chen, J. Wong, and G. Jan, "A wireless PDA-based physiological monitoring system for patient transport," *IEEE Transactions on Information Technology in Biomedicine*, vol. 8, no. 4, pp. 439–447, Dec. 2004.

[17] P. Rong and M. Pedram, "Extending the lifetime of a network of battery-powered mobile devices by remote processing," in *Proceedings of the 40th annual Design Automation Conference*. New York, NY, USA: ACM, 2003, pp. 906–911.

[18] T. Pering, Y. Agarwal, R. Gupta, and R. Want, "Coolspots: reducing the power consumption of wireless mobile devices with multiple radio interfaces," in *Proceedings of the 4th international conference on Mobile systems, applications and services*. New York, NY, USA: ACM, 2006, pp. 220–232.

[19] G. Chanel, S. Pelli, N. Ravaja, and K. Kuikkaniemi, "Social interaction using mobile devices and biofeedback: effects on presence, attraction and emotions," *BioSPlay Workshop Fun and Games Conference*, pp. 5–9, 2010.

[20] J. Gomoluch and M. Schroeder, "Market-based resource allocation for grid computing: A model and simulation," in *Middleware Workshops*, 2003, pp. 211–218.

[21] K. Lee, D. Murray, D. Hughes, and W. Joosen, "Extending sensor networks into the cloud using amazon web services," in *Networked Embedded Systems for Enterprise Applications (NESEA), 2010 IEEE International Conference on*, 2010, p. 1?7.

[22] R. Fielding, "A little REST and relaxation," in *The International Conference on Java Technology (JAZOON07), Zurich, Switzerland*, 2007.

[23] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, Mar. 1986.

[24] A. Singla and C. Rohrs, *Ultrapeers: Another step towards gnutella scalability*, 2002.

[25] V. Lo, D. Zhou, Y. Liu, C. GauthierDickey, and J. Li, "Scalable supernode selection in peer-to-peer overlay networks," in *Hot Topics in Peer-to-Peer Systems, 2005. HOT-P2P 2005. Second International Workshop on*, 2005, pp. 18–25.