Building a Wireless Ice Hockey Personnel Management System

Department of CIS - Senior Design 2009-2010

Samuel J. Lerer
lerersj@wharton.upenn.edu
University of Pennsylvania
Philadelphia, PA

Eric B. Tieniber
tieniber@seas.upenn.edu
University of Pennsylvania
Philadelphia, PA

Jonathan M. Smith
jms@cis.upenn.edu
University of Pennsylvania
Philadelphia, PA

ABSTRACT

Wireless sensor networks (WSNs) are currently one of the most researched topics in the field of Computer Science. One particular area in which WSNs are implemented is human health monitoring systems. In such systems, the focus is on wirelessly observing and tracking physiological data such as heart rate, blood-sugar level, stress level and skin temperature. This paper proposes an extension to such health monitoring systems that is tailored to measure human fatigue in the context of sports, namely ice hockey. While commercial products exist to track basic vital health information for a group of people, the presentation and management of this data is not suited for use in live athletic competitions. Additionally, current commercial products are prohibitively expensive, notably wireless respiratory rate monitors. A novel technique involving audio processing is presented to measure respiratory rates without the need for expensive sensors.

A complete fatigue management system would assist team managers and trainers in making sound personnel decisions based on real-time physiological data. Additionally, such a monitoring system would add a level of safety for the players; when trainers have the ability to detect life threatening health conditions such as irregular heart patterns or severe dehydration, they can act preemptively and avoid catastrophic events from taking place during a game.

1. INTRODUCTION

Networked sensor systems have existed for several decades, dating back to the Cold War when Allied forces developed a scheme to detect the presence of Soviet submarines using acoustic sensors. As technology has improved over time so have the capabilities of sensor systems, allowing for long range wireless networks consisting of hundreds of thousands of nodes. All wireless sensor networks are comprised of a set of nodes, each of which has a certain amount of processing capability, memory, a power source, and a transceiver to send and receive data [4]. The specific characteristics of such networks are largely application specific; Table 1 illustrates several different attributes that vary with each implementation [12].

As an example, consider an intelligent parking service that uses a wireless sensor network. Depending on the size of the lot, there could be several thousand nodes in the network. Each node is responsible for passively monitoring a single parking spot; when a car comes, the node recognizes the spot is now occupied and sends this data back to a centralized source. These nodes aren't mobile and operate within a benign environment [13]. Such a network is structurally much different from an air traffic control system, where the targets are constantly moving and the acquisition of data from the sensors is an active rather than passive process.

One popular application of wireless sensors involves the transmission of human physiological data such as heart rate and blood pressure. "Wearable" technologies have evolved over time, utilizing low cost disposable sensors to monitor a person's vital health information [5]. While such systems are well engineered and are extremely efficient at relaying this information, many of these networks are focused exclusively on improving information flow between physicians and patients in a hospital setting. There is great potential to use health monitoring systems for other purposes such as in the military or in sports, where people are required to perform strenuous physical activity. We seek to create the first ice hockey specific personnel management system.

On October 13, 2008, a 19 year old Russian ice hockey player named Alexei Cherepanov collapsed after taking a shift during a regular season game in the Kontinental Hockey League in Chekhov, Russia. After being attended to by several of the team's medical staff, Cherepanov died. Investigation revealed that he suffered from myocarditis, a heart condition that under normal circumstances would prevent anyone from playing a professional sport [11].

Sensors	Size: small, large Number: small, large Type: passive, active Composition or mix: homogeneous, heterogeneous Spatial coverage: dense, sparse Deployment: fixed and planned, ad hoc Dynamics: stationary, mobile Nature: cooperative, non-cooperative Environment: benign, adverse
Communication	Networking: wired, wireless Bandwidth: high, low
Architecture	Processing: centralized, distributed

Table 1: Sensor Network Classification Criteria

Such an example illustrates one of the primary benefits of developing a product where team managers can observe vital player health information in real-time. It is often the case that reactive treatment to an unexpected injury is not enough to save a life.

An equally important benefit of a complete fatigue management system is the increased team performance that comes with knowing exactly what physical condition players are in at any point in time. Unlike other popular sports such as American football and soccer, where personnel changes occur relatively infrequently, ice hockey is a sport where players are constantly coming in and out of the game. A normal team consists of approximately 18 players, only five of which are on the ice at any given time. Players typically stay on the ice for 30-45 seconds before they are replaced. A direct consequence of this style of play is that it is up to the coaching staff to constantly decide which players should be in the game. It naturally follows that knowing who is tired and who is not is a critical part of making such decisions.

2. RELATED WORK

2.1 Commercial Fitness Monitoring Devices

The technology used to measure human fatigue has significantly advanced over the past decade. Originally, simple pedometers tracked and displayed data without any network or wireless capabilities. Logically, the next step was to use wireless networking technology to help track and analyze this data over time.

Commercially available sensors (listed here in chronological order by release date) include the Nike + iPod fitness system, the BodyMedia sensor, the Polar Electro WearLink heart rate sensor, and the Zephyr Technologies BioHarness. Each of these sensor units measure various vital signs and utilize a varying degree of wireless connectivity.

In May 2006, Apple Computer announced a partnership with Nike to produce an in-shoe personal fitness sensor. The sensor is only a simple accelerometer, but in conjunction with an iPod connected through a wireless link, the system can estimate metrics such as speed, calories burned, pace, and distance. After a workout, users can connect their iPods to a computer and upload collected data to a web interface at nikeplus.com, allowing users to track progress over time. This system is available and widely used today [7].

A company by the name of BodyMedia was formed with the intention of becoming a "pioneer in developing wearable body monitoring systems that are designed to help people lose weight, improve performance, and live a healthier lifestyle [2]." By the end of 2006, BodyMedia had a single-link system with similar wireless properties to the Nike system that could monitor total energy expenditure, active energy expenditure, physical activity duration, and sleep duration. This was all accomplished using a small arm band sensor that could monitor skin temperature, galvanic skin response (to measure electrical conductivity of skin due to sweating), heat flux (to measure the amount of heat released by the body into the environment) and motion. All major processing for this device is centralized to the receiver, as is the case with the Nike + iPod system.

Similarly, in early 2009, Polar Electro received FCC approval for their coded Bluetooth WearLink heart rate belt sensor. Using this device, wearers can wirelessly monitor their heart rate in-real time. The belt is worn on the lower chest and pairs to a wristwatch style monitor that displays the user's current heart rate. Using the WindLink receiver, the WearLink device can also be paired to other devices such as personal computers. While heart rate monitors existed long before this, Polar Electro's WearLink device was among the first to include support for Bluetooth wireless technology.

2.2 Measuring Human Fatigue

At the heart of this project is an effort to quantify human fatigue in real-time. This problem has been addressed and investigated in a variety of industries, leading to several theories as to what the best way to measure fatigue is. Zachrich's "Max VO2 and Ventilatory Threshold in University Ice Hockey" [16] demonstrates a particularly relevant example. His analysis is centered on three main determinants of fitness: maximal oxygen consumption, ventilatory threshold and heart rate.

Maximal oxygen consumption, often referred to as VO2max, is the maximum amount of oxygen that the human body can use to create adenosine triphosphate (ATP), the energy source that the human body needs to perform work. An important distinction to make is that the VO2max level varies from human to human, allowing for relative comparisons between individuals. This metric is one of the primary indicators of cardiovascular fitness.

While VO2max is a strong gauge of overall fitness level, more factors must be considered in order to determine human fatigue at any given point in time. Ventilatory threshold is one such variable that enables observers to determine how close an athlete is to exhaustion (the point where performance significantly declines). After a certain amount of intense exercising, humans experience increases in blood lactate levels. Fatigue occurs precisely when blood lactate begins to accumulate in muscles. This threshold is referred to as the lactate threshold. Both ventilatory threshold and lactate threshold are closely related to one another and are highly correlated with heart rate and respiratory rate. The ventilatory threshold can be computed as a function of breathing rate and VO2max, as seen in Figure 1.

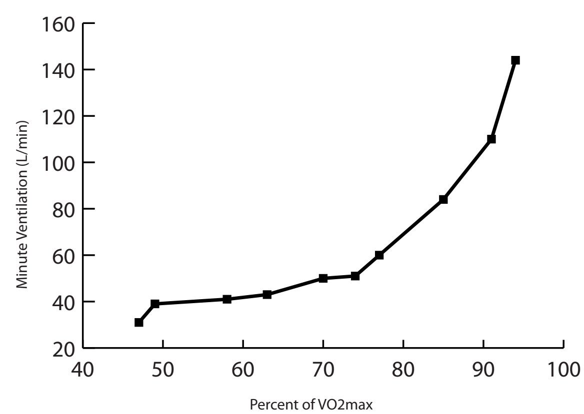

Figure 1: Determining the Ventilatory Threshold

The ventilatory threshold occurs at the point in which the slope of the curve becomes exponential (just below 80%). The results of Zachrich's study of 24 hockey players show that the mean ventilation threshold was 80.9% of VO2max, compared to a similar study of randomly selected individuals that yielded a mean ventilation threshold of roughly 58%. This illustrates the potential of using ventilation threshold and maximum oxygen consumption as valid predictors of fatigue [16].

Human fatigue may also be determined by examining respiratory recovery rates. An important part of ice hockey is that players are exerting large amounts of energy in short bursts followed by periods of rest. For this reason, it is important to look at the changes in recovery rates over the course of an entire game. Intuitively, a player will recover fastest after shifts near the beginning of the game and slowest near the end of the game. However, the variability in recovery times is largely dependent on the fitness level of the individual. A study was conducted in 1997 to determine whether or not overall fitness was an accurate predictor of excess post-exercise oxygen consumption (EPOC) and recovery rate [14]. The results of the study, seen in Figure 2, are interesting in the context of our fatigue management system. The most significant differences in recovery rate between trained and untrained individuals occur within the first four minutes after exercising. The typical hockey player spends between two and half to three minutes on the bench resting in between shifts. The data in Figure 2 supports the hypothesis that higher aerobic fitness levels leads to faster regulation of post-exercise metabolism.

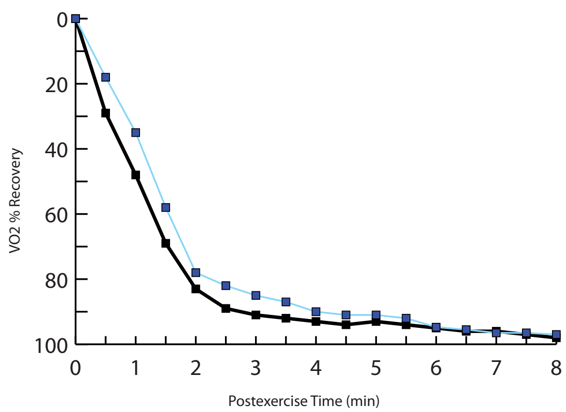

Figure 2: Recovery Rates of Trained(dark) vs. Untrained(light) Individuals

The usefulness of the analysis above is contingent upon the fact that an individual's VO2max is an easily measurable quantity. The only perfect method of calculating an individual's VO2max is using analysis tools only available in a laboratory. In a general sense, the level of oxygen and carbon dioxide in the air inhaled and exhaled by an individual are measured as the person performs an exercise of increasing difficulty, such as running on a treadmill with an increasing incline. When the point is reached where the individual increases his physical exertion or rate of breathing without increasing his oxygen uptake, he has reached his VO2max.

Fortunately, researchers have developed accurate methods of estimating an individual's VO2max using materials available to an ordinary sports team. With access to a track, a stop watch, and a heart rate monitor such as the Polar WearLink described earlier, it is possible to estimate any individual's VO2max. Such a method can determine an accurate estimate of VO2max by measuring an individual's 1-mile jog time and heart rate after that 1-mile jog [10]. This method could be utilized to determine each team member's VO2max.

2.3 Zephyr's Approach

Moving back to the sensor aspect of the project, the general concept of the Nike + iPod system, the BodyMedia system, and the Polar WearLink system could be extended to use many nodes with a common hub receiving and interpreting all collected data. This is exactly what has been developed at Zephyr Technology Ltd [15]. Zephyr's BioHarness sensor can monitor heart rate, breathing rate, body temperature, body posture, and activity levels. Under this system, a general network topology could range from a single node and hub to a large number of nodes all connected wirelessly to a processing hub.

Using its sensor's wireless capabilities, Zephyr has developed a system to monitor and track data received from multiple BioHarnesses in real time. This opens up applications for monitoring the status of squad-based tasks where members maintain high activity levels, such as military crews, fire fighting squads, and athletic teams. In fact, Zephyr has implemented solutions in each of these three areas. The athletic solution is our topic of focus here, as it is perhaps the most closely related system to ours [15].

Figure 3: Zephyr Technology's BioHarness

Zephyr's athletic solution currently has a number of drawbacks that make it an unsuitable solution as an ice hockey specific system. While the Zephyr solution collects many different vital signs including respiration rate, it is unable to correlate this raw data directly to a ventilatory threshold at some percentage of VO2max. Thus, it is unable to estimate a player's fatigue level in real-time. In addition, it does not offer any special functionality or customization towards the unique characteristics of play in ice hockey. Specifically, the rate at which players enter and exit the game demands an interface and metrics that would allow coaches to make split second decisions. Zephyr's team solution is still in its developmental stages and has yet to be released as a finished product [15]. Perhaps the biggest drawback is the cost of the Zephyr product; at $1200 per sensor, implementing this system for a team of 20 individuals would cost nearly $25,000 for the sensors alone.

Measuring an individual's respiration rate is one of the many functions of the BioHarness sensor. There are, however, other methods of measuring an individual's breathing rate. For example, a device known as a thermistor can accurately track a person's breathing rate in real-time [3]. In essence, a thermistor is used to measure small changes in temperature; when placed in the path of an individual's breath, the sensor will heat up and exhibit a change in electrical resistance. Thus, a subject inhaling and exhaling in the path of a thermistor produces a recordable change in resistance that can then be used to determine the subject's respiration rate. We planned to use this technology in our system but were ultimately unable to do so because of the limited commercial availability of wireless thermistors. Commercially available products were either too expensive or had a sampling rate that was too low for our purposes.

This led us to the decision to use a wireless microphone as the source of our respiratory rate data, an alternative that is advantageous because of its simplistic elegance and relatively low cost.

2.4 Audial Breath Detection

The use of audio processing techniques to measure respiratory rate is somewhat novel in the sense that there are no known commercial respiratory rate monitors that use this approach. Kroutil *et al.* present a method for breath detection based on measuring acoustic signals originating in the trachea [8]. Bajowala textitet al. used a similar approach in an attempt to build a monitor to detect sleep apnea occurring among infants [1]. While the overall concept of these works is closely related to our goals, there are several key factors that separate our work from much of the work that has already been done. Perhaps the biggest difference is the very environment in which our system operates; the large amount of ambient and background noise within an ice rink poses a particularly big challenge in terms of extracting breathing rate data. A second difference involves the subjects themselves; our system must be able to function without being so obtrusive that it affects the performance of the players.

3. SYSTEM MODEL

Figure 4 is a block diagram of the information flow throughout our system.

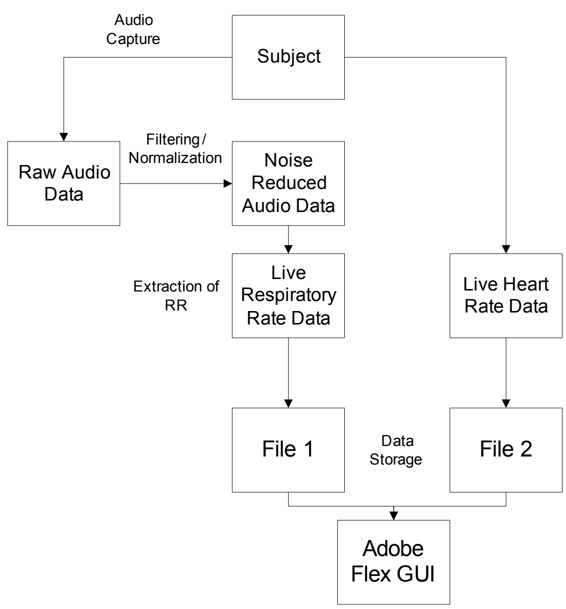

Figure 4: System Block Diagram

For each player, there are two main sources of data as shown in Figure 4: heart rate data and respiratory rate data. The players wear a commercially purchased wireless heart rate monitor as well as a small wireless microphone. The actual transmission of the data occurs via Bluetooth. While the heart rate data is automatically generated from the monitor, there are several processing steps that must occur to get respiratory rate data from the microphone signal. These steps are hereafter referred to as the audio processing subsystem.

3.1 Audio Processing Subsystem

One of the main components in the audio signal processing subsystem is the filtering of ambient or background noise. The reasoning for including this component is intuitive; inside of an ice hockey rink there are several compressors used to maintain the low temperature in the building. Naturally, this constant and loud din will show up in any recorded audio sample. Removing this background noise involves comparing the frequency distribution of the noise as well as the actual sound of human breathing, and then applying high and low bandpass filters to remove as much of the noise as possible while still maintaining the integrity of the breathing. Consider Figure 5, a recording of human breathing in an environment with significant background noise:

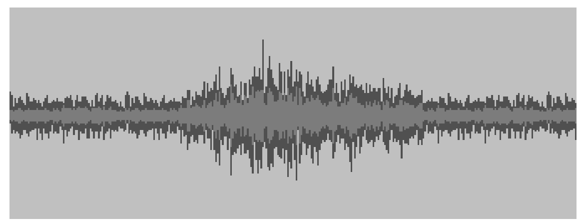

Figure 5: Unfiltered Breathing Signal

The slight differences in amplitude may not be enough to detect the presence of a breath. Therefore, a Fourier transformation was used to determine the distribution of frequencies for this sound. In other words, we can observe which particular frequencies dominate and which are less prevalent. Figure 6 illustrates dominant frequencies based on the shaded areas.

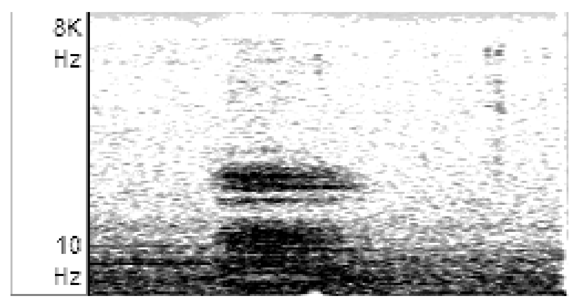

Figure 6: Breathing Spectrogram

The goal is to determine which frequencies are more prevalent in the background noise and less prevalent in the actual breath. For reference, the x-axis of Figure 6 represents time. The spectrogram provides a good start as to what these frequencies might be; note the dark circular band that occurs in the during the breath that does not appear in the ambient noise. Figures 7 and 8 are two frequency distributions that illustrate this discrepancy more clearly.

The goal is to determine which frequencies are more prevalent in the background noise and less prevalent in the actual breath. Figure 7 shows a frequency distribution for a portion of the audio clip in which the subject is in the middle of a breath. Similarly, Figure 8 shows a frequency distribution for a portion of the audio clip in which there is only ambient noise.

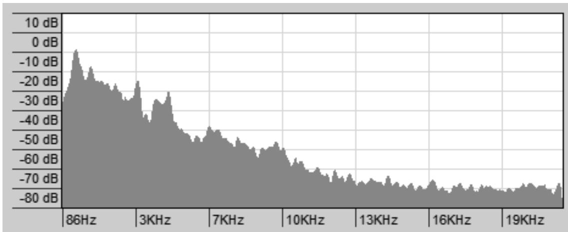

Figure 7: Frequency Distribution of Breath

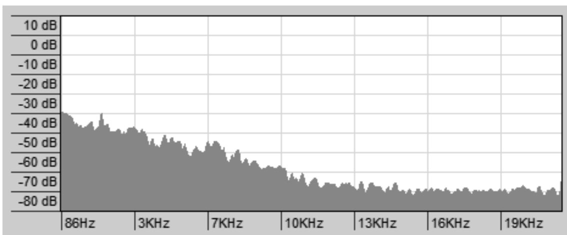

Figure 8: Frequency Distribution of Ambient Noise

For both distributions, the dominating frequencies occur at the far left of the scale. However, in the breath there is a spike in 1500-4000 Hz waves that we do not see in the background noise. This is crucial to the filtering process; if we apply a highpass filter of 1500 Hz and a lowpass filter of 4000 Hz, we are actually filtering out a significant portion of the human breath, in addition to most of the noise. The advantage lies in the fact that what we are left with is band of frequencies that differentiate the two. After applying the two bandpass filters, the resulting waveform looks like Figure 9.

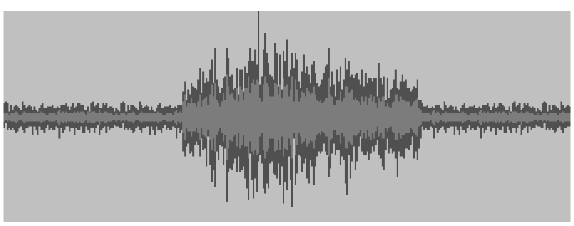

Figure 9: Filtered Breathing Signal

The before and after images (in Figures 5 and 9, respectively) clearly show the effect that bandpass filtering has on accentuating the human breath in the presence of ambient noise. Extracting the respiratory rate from a sequence of waveforms like Figure 9 is the most crucial component of the audio subsystem. Our original approach to finding this breathing rate involved using an amplitude threshold algorithm, which takes a noisy signal and locates the local maxima and minima. One significant problem we ran into using simple amplitude threshold was that within a single "breath" there isn't a single amplitude value such that at any point during the breath, the amplitude is above that threshold. Figure 10 shows a highly magnified version of the wave seen at its maximum amplitude. With this view it is evident that a single amplitude threshold test will not succeed in identifying a single breath.

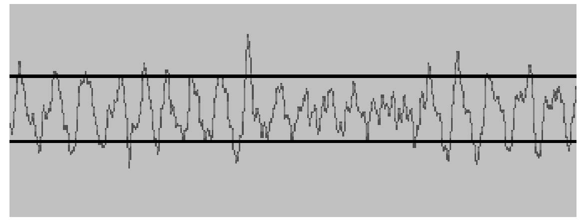

Figure 10: Simple Threshold Detection in a Single Breath

The goal is to find the points at which the absolute value of the amplitude is above a certain threshold T. If we set T too high (as seen in Figure 10), there are peaks within this 10 ms window that fall below the threshold; the result is that within one actual breath the algorithm will detect multiple breaths. Consequently, the method that we eventually developed involves taking the average amplitude over a slice of time and comparing that to a specific threshold. This method is further discussed in Section 4.

3.2 Data Storage

The purpose of this component is to hold two sets of data: data coming from the heart rate monitor and respiratory rate data from the microphone that has been processed and filtered. The data storage component of our system is particularly important because it serves as the intermediary step between the raw data and the interface. Our design includes two files for each player: one file for heart rate data and a second file for respiratory rate data. This limits the amount of data that needs to be loaded each time the program starts: if the user only wishes to monitor one or two players, only these files must be loaded. While a more technical description of our approach can be found within Section 4, one significant advantage to this design is that all of the historical data is stored within the GUI application itself (rather than in the individual files). This is important because each time a new data point is loaded, it will overwrite the previous value instead of simply concatenating it to the end of the file. The result is that each time the GUI updates its view, it must read in only a single data point.

3.3 Interface Display

The purpose of the interface is to enable the user to control the program and view data about the current players as the program is running. Much like the data storage component, there are two primary types of data that the interface must be able to clearly show: the streaming heart rate and respiratory rate data, and the historical data and trends that exist for each player. Much of the display is in the form of graphs and charts, and the user is able not only to view data about a certain player, but also to compare two or more players' data over an arbitrary amount of time. Aside from displaying the data, the interface incorporates ice hockey specific features. These features are particularly important for analyzing player performance and are explained in greater detail in Section 5.2.

4. SYSTEM IMPLEMENTATION

4.1 Audio Subsystem Implementation

The first basic design choice in implementing our audio subsystem was deciding which sound package to use for streaming audio data from a microphone. The Java Sound API provides low-level support for audio operations such as capturing sound and performing basic signal processing. While Java may not be the best choice in terms of speed, our tests revealed that the data processing in Java did not create a bottleneck for the rest of the system. Perhaps the most important reason why the Java Sound API best suited our needs is that it offers a simple way to stream audio data in real time. By connecting with a microphone, data can be read into a fixed-size buffer and then processed by the program without much overhead.

The primary alternative that we considered for handling the audio data was LabView. Despite the excellent analytical tools provided by this software, there were a few major flaws that prevented us from incorporating it into our final implementation. LabView is a large application that requires a significant amount of memory and CPU utilization; simultaneously analyzing multiple audio streams with this software would consume nearly all of the computer's resources and cause unwanted lag in controlling the interface. Additionally, integrating LabView with the GUI proved to be difficult and inefficient. For these reasons, we had to sacrifice the superior analytical capabilities of LabView in favor of the simpler and more manageable Java Sound API.

The algorithm used to extract real time breathing rate from a microphone is similar to the amplitude threshold method described in Section 3.1. Figure 11 is a block diagram of the process, followed by an explanation of the steps involved.

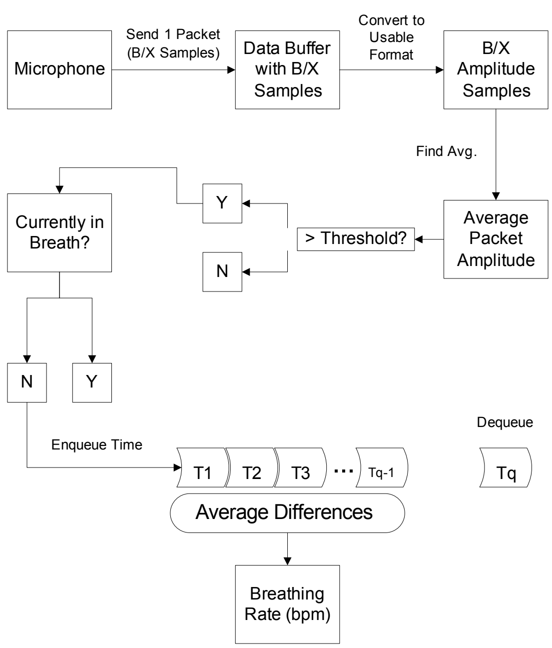

Figure 11: Breathing Rate Extraction Algorithm

There are several inputs to the algorithm that will alter its accuracy:

- S, the sampling rate (in Hz)
- B, the buffer size (in bytes)
- X the sampling size (in bytes)
- Q, the queue size, and
- T, the amplitude threshold.

The amount of time it takes for one packet to be delivered depends on the sampling rate (S), buffer size (B) and sampling size (X). Once the buffer is full the audio data is preprocessed into a usable format. The average amplitude is then found and compared to the predetermined threshold (T). If the average amplitude exceeds T, this means the packet represents data that is part of a breath. If we observe that the last packet that came in was not in a breath, we can conclude that this packet marks the start of a new breath, and we store the time at which the packet was delivered in a queue of size Q. Breathing rate is then determined by calculating the average time between the last Q breaths and converting it into breaths/min (BPM).

4.2 Fatigue Analysis

When the system is initiated, an algorithm is used to determine a player's circulatory capacity, or maximum heart rate, based upon his age and weight. This value is used to determine the current percentage of each player's circulatory capacity at any given time. It has been shown that a person performing at above 90% of his circulatory capacity is unable to sustain performance, and thus becomes quickly fatigued [9]. We have deemed this value the circulatory threshold.

As discussed earlier, the ventilatory threshold can be observed as the change from a linear increase in respiration rate to an exponential one. Observing this change is key to our system, as a player operating at above his circulatory threshold as well as above his ventilatory threshold becomes fatigued as his blood cannot carry oxygen to (and lactic acid away from) his muscles quickly enough, and thus cannot sustain performance.

Our system determines when both thresholds have been crossed in determining fatigue. When a player is fatigued, he is marked as such on the main system GUI, as an indication to the team manager. When a player has dropped back below both of the thresholds, his fatigue marking is removed from the system.

4.3 Interface Implementation and Features

Since the interface is the only component in the system with which the user interacts, our implementation had to satisfy three basic criteria: 1) it had to be able to integrate with Java at the back end so that it would be possible to retrieve data in real-time, 2) the application wouldn't need to constantly wait for new data to appear, and 3) the GUI would have aesthetic and intuitive appeal. For all three of these reasons, we selected the Adobe Flex Software Development Kit as the SDK we used to develop the interface. Flex is an open source SDK built on top of Adobe Flash; the language is a mixture of ActionScript and XML. Java methods and classes can be invoked from a Flex Application using Blaze DS, a Java remoting technology designed specifically for this purpose [6]. In terms of performance, one of Flex's most attractive features is that certain parts of the interface can be addressed individually. Unlike many HTML based applications which require constant reloading of the page as the view changes, Flex uses this individual element addressing to allow the user to make significant changes to the view without have to reload or refresh the page. Considering aesthetics, the Flex SDK provides a full toolkit of GUI items that can be used to present the data in a user-friendly way.

5. PROGRAM FUNCTIONALITY

5.1 GUI Functionality

As described in 3.3, in addition to simply displaying the streaming respiratory and heart rate data, the Flex GUI also stores all of the historical data. Players are represented in memory by "Player" objects which contain (among other things) the following information:

- Historical heart rate data (Integer[])
- Historical respiratory rate data (Integer[])
- Current heart rate (Integer)
- Current respiratory rate (Integer)
- Number of shifts (Integer)
- Length of shifts (Integer[])
- Total time on ice (Integer)
- Whether or not player is currently on the ice (Boolean)

The Flex backend is driven primarily by a timer; on each timer event, the GUI reads the files associated with each player (one file for heart rate data and a second file for respiratory rate data) and stores incremental data into the relevant data structures. This is advantageous from an efficiency standpoint in that each data point is read into its corresponding data structure only one time. Since all of the Player objects are automatically updated on each timer event, the data being displayed through the GUI is always perfectly synchronized with the incoming data.

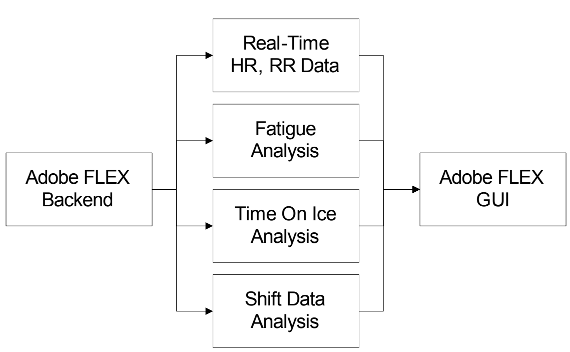

Figure 12: GUI Implementation

5.2 Ice Hockey Specific Features

5.2.1 Managing Players Coming On and Off the Ice

The interface allows coaches to select which players are currently on the ice at any given point in time. This can be done at any point in time by simply clicking on the player's name from any screen within the application. This feature is incredibly important for making personnel decisions; coaches can quickly identify who is on the ice, and analyze how each player's heart rate and respiratory rate moves over the course of a shift.

5.2.2 Managing Stops and Starts in Play

One feature that is unique to our system is the ability to keep track of stops and starts in game play. There is a button on the interface that controls whether or not the game clock is ticking. To illustrate why this is important, consider the following example. A particular player may spend two minutes on the ice during a particular shift, but only 45 seconds of that is spent actually playing. This is a perfectly normal occurrence given the fact that after there is a whistle there is a certain amount of downtime before play resumes. When analyzing player fatigue, it is necessary to be able to identify exactly when players are exerting themselves and not simply standing around on the ice. This built-in feature ensures that all of the data is accurate and not skewed due to pauses in play.

5.3 Application Views

5.3.1 Team-Level View

Figure 13 shows the first screen that the coach encounters, displaying all of the players and their current heart rates and respiratory rates. The color indicates whether or not a particular player is on the ice: a darker rectangle indicates that a player is on the ice, and a lighter rectangle indicates that a player is resting. From this screen, the coach can switch personnel, change the state of the game clock, or choose a single player to view historical heart rate and respiratory rate data.

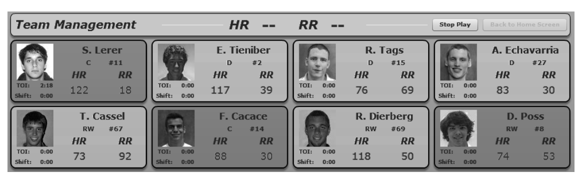

Figure 13: Main Team-Level View

5.3.2 Individual Player View

The view shown in Figure 14 allows coaches to examine all of the data for a single player. The scatter plots are color coded, making it especially simple to look at how a player's heart rate and respiratory rate fluctuate not only when that player is on the ice (darker blocks), but also during the recovery phase (lighter blocks).

5.3.3 Team-Level Shift Analysis

One important hockey-specific feature of our system is shift analysis, seen in Figure 15. A typical shift for a player lasts between 30 and 45 seconds; coaches can see the number of shifts that each player has taken as well as the length

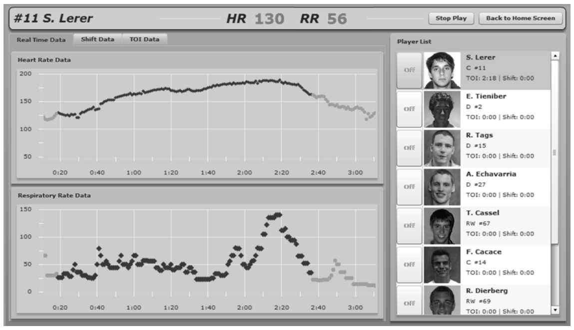

Figure 14: Individual Player View

of each player's average shift. Currently, coaches use clipboards to keep track of the number of shifts that each player takes. However, they do not have the capability of tracking average shift lengths in real-time. This feature makes it easy to identify if specific players are staying out on the ice too long and are likely performing at a lower level.

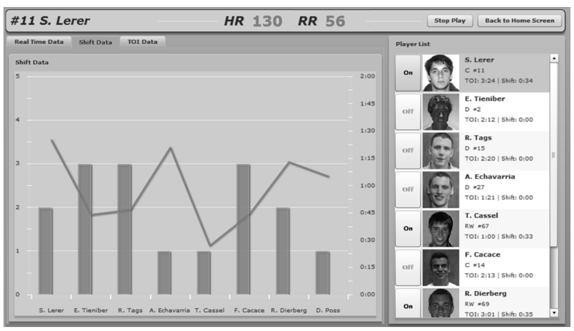

Figure 15: Team-Level Shift Analysis

5.3.4 Time On Ice Analysis

This view (shown in Figure 16) simply shows the total amount of time that each player has spent on the ice. This feature provides another tool that team managers can use to ensure that particular players are not spending a disproportionately large amount of time on the ice. Coaches do not currently have access to this data in the middle of a game.

Figure 16: Time On Ice Analysis

6. SYSTEM PERFORMANCE

Unit tests have been conducted to evaluate both accuracy and CPU efficiency of the system. Different metrics and setups were tested to determine the tradeoff between accuracy, efficiency, and feasibility.

For the first test, we determined the accuracy of 3 different microphone placement models, namely: (1) microphone mounted away from the face, (2) microphone mounted in front of the nose and mouth, and (3) microphone connected through a stethoscope on the neck. Here we determine the accuracy of each setup based on the sound coming from a quiet room, a noisy room, and a noisy room with high and lowpass filtering enabled.

Breaths Counted	Correct Breaths Counted	True Count	Accuracy
8	7	11	72%
13	5	11	45%
15	6	11	55%

Table 2: Placement Test: Microphone Away from Mouth

With the microphone mounted away from the face (for example on the side of the head), test results were extremely inaccurate. External noise also had the most effect in this test. Noise filtering helped to achieve a balance here. Table 2 further represents the results of this test. The leftmost column represents the number of breaths counted by the audio subsystem. The second column represents the number of times the subsystem accurately detected a breath. Accuracy is determined by dividing this value by the true number of breaths in the sample.

In the next test, the microphone was mounted directly in front of the mouth. Test results were generally very accurate. Even with background noise applied without filtering, the program still managed to count 8 out of 9 breaths correctly. When filtering was applied as described in Sec. 4.1, accuracy rose to 100%. Table 3 represents the results of this test.

Breaths Counted	Correct Breaths Counted	True Count	Accuracy
9	9	9	100%
9	8	9	89%
9	9	9	100%

Table 3: Placement Test: Microphone Near Mouth

For the final accuracy test, the microphone was connected to a stethoscope placed on the neck. This allows the microphone to record the sound of air passing through the esophagus. As seen in Table 4, accuracy was perfect in every case. Background noise is also hardly an issue, as the only sound captured by the microphone must first pass through the stethoscope, which was in contact with the neck the during the entire test.

Although the neck-based results were the most accurate (as originally predicted), it represents the least feasible option. The system needs to strike a balance between feasibility and accuracy. A neck-mounted microphone can listen to a player's breathing more accurately at the cost of comfort. Most ice hockey players currently do not wear protective equipment on their necks, so a neck-mounted sensor

Breaths Counted	Correct Breaths Counted	True Count	Accuracy
12	12	12	100%
12	12	12	100%
12	12	12	100%

Table 4: Placement Test: Stethoscope on Neck

would be less than ideal. That being said, some players do in fact wear protective neck guards, which could be used as a mounting point for a microphone. As demonstrated, enhanced filtering can lead to acceptable accuracy rates from a microphone mounted near the mouth. This enhanced filtering comes at the cost of the CPU time needed to develop those filters. Thus, we must ensure that the system will be functional with at least 18 players.

In this next test, CPU utilization is graphed against sampling rate, S. The goal here is to minimize S (and therefore CPU utilization) while maintaining an acceptable level of accuracy. Figure 17 shows a graph of our results.

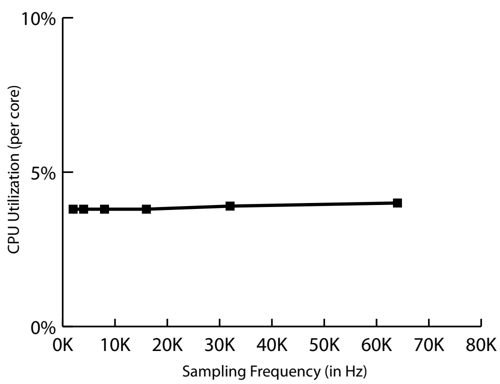

Figure 17: Sampling Rate vs. CPU Utilization

It is clear that microphone sampling rate has little-to-no effect on CPU utilization. Therefore, it is not a factor that needs to be considered from a performance standpoint. It is important to consider, however, that the audio subsystem, as implemented, requires approximately 4% of a modern CPU core per player. With 18 players, that figure rises to 72% CPU utilization per core. This level of multi-threaded processing represents only 36% of the available processing power of a dual-core CPU and less than 20% of a quad-core CPU.

7. FUTURE WORK

Future work in this area focuses heavily on a few major research questions.

7.1 Refining Breathing Rate Algorithm

Although a great deal of work has been completed on the topic of determining a respiration rate given a microphone sample, certain improvements could be made. First, more sophisticated audio processing tools, such as ambient noise reduction filters, could be implemented and refined to improve system accuracy in a variety of conditions. These additional filters, however, come at the cost of increased system processing requirements. Balancing accuracy with filtering capability was an important goal in this system, as it is expected to operate on 18-20 players simultaneously with a single central hub. Conducting complex filtering on 18-20 streams simultaneously would require either a more powerful hub or a more distributed system of computation.

Additionally, the audio subsystem parameters described and tested in Section 4.1 could also be further refined through research and testing in a variety of environments. For example, although the research presented here demonstrates an attempt at developing the most accurate values for the audio subsystem parameters (i.e., sampling rate (S), buffer size (B), sampling size (X), queue size (Q), and amplitude threshold (T)), testing was confined to a limited range of environments specific to our application, namely ice hockey.

7.2 Extension of Fatigue Measurement Capabilities

Aside from the audio processing subsystem, the other major theoretical portion of our work deals with measuring fatigue based on real-time respiration and heart rate data. While much research and has gone into this portion of our work, capabilities could be further refined. Our system determines fatigue based on an anaerobic range of heart rates and an exponential increase in breathing rates. The only personal variables used are age and weight. Thus, the system could be further individualized by measuring other variables such as V02max and lactate threshold, quantizing differences in these values for each subject, and incorporating those factors while calculating fatigue.

Furthermore, an extended system could implement fitness tracking for each player over a period of time, incorporating historical data for training and fatigue measurement purposes. Historical tracking would allow a coach to use the system to oversee players' fitness levels over a period of weeks or months as players gain or lose stamina. Determining changes in a player's overall fitness and progress over time would allow coaches to track the fitness progress of each player as well monitor the fitness level of the entire team. Finally, this type of tracking would allow a coach to rank his players based on fitness ability and progress, something that has never before been quantified.

8. REFERENCES

[1] Adnan Bajowala et al. Baby breathing monitor. Master's thesis, University of Illinois at Urbana-Champaign, May 2001.

[2] D. Andre et al. The development of the sensewear armband, a revolutionary energy assessment device to assess physical activity and lifestyle. *BodyMedia, Inc.*, 2006.

[3] Emil Jovanov et al. Thermistor-based breathing sensor for circadian rhythm evaluation. Master's thesis, University of Alabama in Huntsville, October 2001.

[4] I.F. Akyildiz et al. Wireless sensor networks: A survey. *Computer Networks*, 38(4):393 – 422, March 2002.

[5] Roozbeh Jafari et al. Wireless sensor networks for health monitoring. Master's thesis, University of California Department of Computer Science, 2005.

[6] Adobe Systems Incorporated. Blazeds.

http://opensource.adobe.com/wiki/blazeds/BlazeDS/, 2009.

[7] MD Ken Tegtmeyer. Data, data, data - but how to keep track of it all. *American Journal of Lifestyle Medicine*, 1(2):144 – 145, 2007.

[8] Jiri Kroutil and Miroslav Husak. Detection of breathing. Master's thesis, Department of Microelectronics, Czech Technical University in Prague, October 2008.

[9] Todd Laux. Target heart rate zone training. Master's thesis, Purdue University, 2008.

[10] David E. Nielson. Predicting vo2max in college-aged participants using cycle ergometry and nonexercise measures. Master's thesis, Brigham Young University, October 2009.

[11] Associated Press. Cherepanov probe reopened in russia. http://sports.espn.go.com/nhl/news/story?id=4388247", August 2009.

[12] Proceedings of the IEEE. *Sensor Networks: Evolution, Opportunities and Challenges*, volume 91, August 2003.

[13] Dukhee Yoon Sangwon Lee and Amitabha Ghosh. Intelligent parking lot application using wireless sensor networks. Master's thesis, University of Southern California Ming Hsieh Department of Electrical Engineering, 2008.

[14] Kevin R. Short and Darlene A. Sedlock. Excess postexercise oxygen consumption and recovery rate in trained and untrained subjects. *Journal of Applied Psychology*, 83(1):133 – 159, July 1997.

[15] Zephyr Technology. Zephy bioharness bt. http://www.zephyr-technology.com/, 2009.

[16] Timothy P. Zachrich. Max vo2 and ventilatory threshold in university level hockey players. Master's thesis, Bowling Green State University, May 2008.