OULUN YLIOPISTO
UNIVERSITY of OULU

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Tommi Käsmä

Putt swing and hit measurement using a wrist device

Master's Thesis
Degree Programme in Computer Science and Engineering
September 2015

Käsmä T. (2014) Putt swing and hit measurement using a wrist device. Department of Computer Science and Engineering, University of Oulu, Oulu, Finland. Master's thesis, 60 p.

ABSTRACT

Developing automated electronic training aids for sports has become more and more viable in recent years. Modern smartphones provide ubiquitously available computing resources. Combined with ever smaller, increasingly accurate and energy efficient motion measurement technologies, this has enabled the creation of automated systems able to offer feedback to people engaged in various sports, thus aiding them in their training.

In many sports, including golf, it is of interest to track the movement of the hands and arms to gain insight into the player's technique. Wrist mounted devices, such as wrist watches, are widely used and accepted to be relatively unintrusive to wear in many sports. The wrist is also directly connected to the hands that control the used equipment, such as golf clubs.

Following a review of currently existing golf training aids, wrist device was chosen as a suitable compromise between user comfort and performance for measuring movement in golf. In the context of this thesis, a prototype device for measuring golf putts was designed and implemented. In addition, a simulation set-up was created for measuring acceleration data during golf putts performed by a human subject and a purpose-built human analogue. The resulting signals were analysed with a view towards use in algorithm development.

The implemented device proved able to produce data of sufficient quality and quantity to enable algorithm design and implementation. A preliminary test was performed to verify that detection of putts using this device was viable, thus laying the base for creating an automated system able to give instant feedback to the user.

Keywords: Bluetooth, accelerometer, data logging, golf

Käsmä T. (2014) Putin suorituksen mittaus rannelaitteella. Oulun yliopisto, Tietotekniikan osasto. Diplomityö, 60 s.

TIIVISTELMÄ

Viime vuosina urheiluharjoittelua tukemaan on kehitetty sähköisiä apulaitteita. Älypuhelimien laskentaresurssit yhdistettynä yhä pienempiin, tarkempiin ja energiatehokkaampiin liikkeenmittausteknologioihin ovat mahdollistaneet automaattiset järjestelmät, jotka kykenevät palautteella tehostamaan eri urheilulajien harjoittelua.

Käsien ja käsivarsien tarkka seuraaminen on hyödyllistä monessa urheilulajissa, mukaan lukien golfissa. Tieto käsien liikkeestä voi auttaa ymmärtämään pelaajien tekniikkaa paremmin. Rannekellon tyylisten laitteiden koetaan olevan huomaamattomia käytössä. Ranteet ovat myöskin suorassa yhteydessä käsiin, joilla urheiluvälineitä usein hallitaan.

Tässä työssä tarkasteltiin tällä hetkellä olemassa olevia golfharjoittelun apuvälineitä. Rannelaitteen katsottiin olevan sopiva kompromissi käyttäjämukavuuden ja mittaussuorituskyvyn väliltä liikkeen mittausta varten golfissa.

Toteutetulla prototyyppilaitteella mitattiin puttien kiihtyvyysdataa. Putteja suoritti sekä testihenkilö, että mekaaninen aitoa puttiliikettä matkimaan rakennettu laite. Mitattuja signaaleja tarkasteltiin algoritmikehityksen kannalta. Laitteen todettiin käytännössä kykenevän tuottamaan määrällisesti ja laadullisesti riittävää dataa algoritmikehitystä varten. Alustavassa testauksessa varmistettiin, että puttien automaattinen tunnistaminen laitetta käyttäen on mahdollista. Tulokset ovat perusta automaattisesti palautetta antavan järjestelmän luomiselle.

Avainsanat: Bluetooth, kiihtyvyysanturi, datan keräys, golf

TABLE OF CONTENTS

FOREWORD

This master's thesis was done at CaddieON Inc. in Oulu. In this work the use of small mobile devices for motion tracking in sports was investigated and a prototype of a wrist device for detecting golf putts was created.

I wish to thank the supervisor of my thesis, Professor Olli Silvén. The guidance and feedback he provided was invaluable during the writing of this thesis. Thanks also to the second reviewer of my thesis, Dr. Esa Rahtu. I am also very grateful to CaddieON Inc. in general, and to Tuomo Lalli and Juha Pikkarainen in specific, for giving me the opportunity to do this work, and to Osmo Voutilainen and Hannu Kallio-Kokko and many others at CaddieON Inc. for their various contributions that collectively made this work possible.

Oulu, Finland September 22, 2015

Tommi Käsmä

ABBREVIATIONS

Bluetooth LE	Bluetooth Low Energy
CPU	Central Processing Unit
DMA	Direct Memory Access
DOF	Degrees Of Freedom
FPU	Floating Point Unit
GPIO	General Purpose Input/Output
GPS	Global Positioning System
I2C	Inter Integrated Circuit
IDE	Integrated Development Environment
IMU	Inertial Measurement Unit
IO	Input/Output
LCD	Liquid Crystal Display
LED	Light Emitting Diode
Li-Ion	Lithium-Ion
MEMS	Micro-Electro-Mechanical System
MPU	Micro Processor Unit
MTU	Maximum Transmission Unit
RAM	Random Access Memory
RTOS	Real Time Operating System
UART	Universal Asynchronous Receiver/Transmitter
UI	User Interface

1. INTRODUCTION

In many sports the players wield a variety of clubs, bats and rackets to play the game. The items wielded by the player include such things as the projectile itself, for example the disc in disc golf, and tools used to interact with other objects such as an ice hockey stick that is used to control the puck. The player's skill in manipulating the item is of vital importance to that player's performance in the sport. This skill is generally honed over time through numerous repetitions of basic gestures in various conditions typical to the sport at hand.

Objective feedback can be invaluable when consciously striving to improve any given skill, but the players themselves are often unable to accurately judge the exact motions taken during any particular gesture. For players who wish to improve their skills in a systematic manner, instant feedback is invaluable. An outside observer with suitable expertise, such as a coach, may be able to provide useful feedback to the player, but this is not available to everyone. A mechanical appliance capable of objectively measuring the player's movements and providing instant feedback can be of great value, and thus it is of interest to develop such tools.

In practice, such a device would need to be able to measure the gestures typical for the sport. In many sports most gestures usually involve the entire body of the player to some degree, and several different metrics can provide useful information of the player's movements. Many methods to track such movements exist, with a variety of possible sensor types and different ways to attach the sensors to the player, their equipment, or the immediate surroundings being possible.

1.1. Metrics in sports

Collecting sports related metrics is a long established practice. Before the advent of digital automation, even fairly simple metrics such as duration, location, speed and distance had to be measured and tracked manually. Tracking physiological signals was difficult, if not impossible.

As technology has developed it has become trivial to automatically track many metrics with the help of devices such as the heart rate monitor shown in Figure 1. For example even many recreational runners use smartphones equipped with GPS (Global Positioning System) to track their route, distance, speed and elevation changes in real time. Besides simple heart rate monitors, there are also ones that transfer the heart rate data directly to a smartphone in real time, enabling combining location and time information to the heart rate information.

Tracking sports performance metrics can offer several possible benefits. Systematically tracking over longer periods provides objective statistics on long term change in performance. These metrics can be used to simply quantify long term progress, or they can be used to get feedback to consciously guide long term training. For example a long distance runner could adjust the intensity and duration of future training events to reach peak performance at a desired time, or a tennis player could adjust their training to achieve a faster serve.

Conscious practice is important in the short term as well. Repetition is important in first establishing a new motor skill, but repeating a gesture with a systematic mistake

Figure 1. Polar heart rate monitor system with a sensor belt and a wrist receiver.

could be detrimental. An external observer such as a coach can help identify problems and suggest fixes, but not everyone always has access to a personal coach.

Individual training events could be measured to provide objective feedback to guide practice. For example, a tennis serve could be measured, tracking the acceleration and trajectory of the racket and the hand holding the racket. These measurements could then be analysed to for example identify areas where changes are needed to achieve better form, in the manner presented in Figure 2.

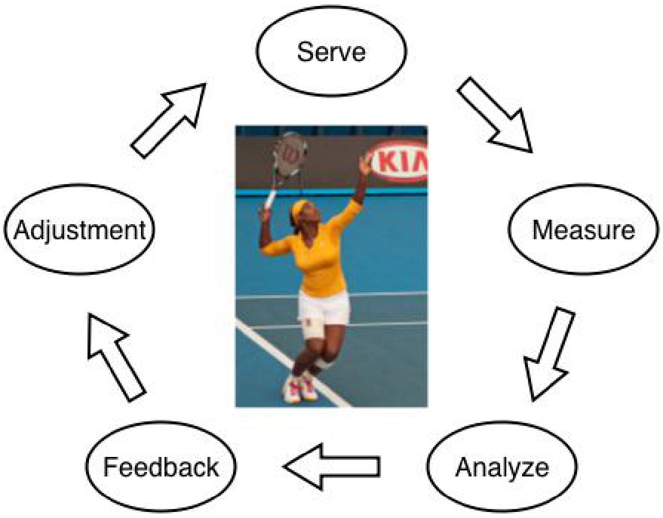

Figure 2. Machine assisted iterative feedback cycle.

Accurately tracking metrics such as the characteristics of individual gestures, e.g. a tennis serve, requires a sensor system to get useful measurements, and computing to analyse them. With the increasing ubiquity and computing power of smartphones, the idea of a personal coach that travels in your pocket is becoming ever closer to reality.

The smartphone is only one part of the system, though. A phone's sensor array typically does not provide sufficiently detailed data to track quick movements, and the phone's placement in a pocket or a phone case can be quite poor for measuring data. At the same time, progress in wireless communications has made it possible to develop light-weight and small external sensor platforms.

1.2. Goal of the thesis

The goal of this thesis is to create a prototype device for use in development of a personal coaching system to provide automated feedback specifically for the sport of golf. It is not intended to result in a finalized product ready for consumer use. Instead, the purpose is to allow for evaluation of the components, development of algorithms and evaluation of their viability to facilitate development of a complete implementation in the future.

A complete system would consist of the sensor platform hardware, the embedded software controlling the hardware, and the software that makes use of the measured and transmitted data in the end device. The main focus of the work is analysing the problem, defining system specifications that the hardware of a sensor platform should satisfy, and implementing its embedded software. While the main component choices for the sensor platform are covered, the specifics of the design and implementation of the sensor platform hardware are outside the scope of this work. This also applies to the end device software, and the detailed analysis and modelling of the measured data.

1.3. Outline of the thesis

Chapter 2 is an introduction into the game of golf and the current state of the commercially available training aids. Various products are covered and evaluated focusing particularly on intrusiveness, ease of use and effectiveness. In Chapter 3 the most relevant embedded motion measurement technologies are looked at in more detail, and the current state of putt detection using such technologies is reviewed. Chapter 4 presents the most important hardware components and the rationale for selecting them for use in the device with the focus on performance, simplicity of implementation and energy efficiency. Chapter 5 covers the software design, including the main timing requirements, the structure of the software and the high level design of the implemented drivers. Chapter 6 describes the putt simulation set-up used to generate repeatable data for algorithm development, and the resulting measured signals are described. The created algorithm is also evaluated to gauge the system's viability for automatic putt recognition. The final outcome of the thesis and some possibilities for future development are discussed in Chapter 7.

2. GOLF TRAINING AIDS

Golf is a good example of a sport that combines a requirement for a high degree of precision and the use of full body power in the same sport. The goal in golf is to deliver a ball no smaller than 42.67mm in diameter from the designated teeing ground to a hole 108mm in diameter, located up to several hundred meters away, with the smallest number of strokes possible by hitting the ball with a golf club. [1]

Typically courses consist of 18 holes or 9 holes that are played twice for a full 18 holes. Each hole is designated a so called par number, which is defined as the number of strokes a good player is expected to need to get the ball from the tee into the hole. The par numbers typically range from 3 to 5, and they are given with consideration to the length of the hole and features of the terrain. For example two different par 4 holes would both be expected to be completed with 4 strokes, but one could be a long hole downhill and the other a shorter one but uphill. When assigning par numbers 2 strokes are allocated to putting and the other strokes for the approach to the green [2]. A typical combined par number of a round of 18 holes in tournament play is between 69 and 73.

Each stroke intended to hit the ball is counted whether successful or not. As a result, a consistently good outcome on each individual stroke is vital for a good total score. Strokes taken each round of golf vary in form and length from hundreds of meters long drives, using the full strength of the body, all the way to centimetres long putts using one hand. Putts contribute on average about 41% of the total score of a tournament round for a PGA Tour player, with the rest coming from penalty points and all the other strokes [3]. It is clear that being able to both putt and hit longer strokes well is important.

It is well understood among those who play golf that it is a sport of refined technique and practised skill. As a result a plethora of different training aids exist on the market today, and many are willing to spend money on devices that promise to help them improve faster than they otherwise could when practising alone.

The available devices differ in many ways although they could roughly be separated into two main categories: mechanical aids and electronic aids.

2.1. Mechanical systems

Mechanical aids are the most common with hundreds of different products available on the market today. Most devices are fairly simple and are meant to help the player with a very specific part of proper golf swing form. This is usually accomplished either by providing tactile or visual feedback, a visual reference point, or by limiting the player's range of movement such that improved form is achieved. In most cases the training aids do, however, change the look or feel of the swing from the normal situation either by giving visual feedback that is not available during a normal swing, or by changing the feel of the motion. If the player is used to swinging correctly when certain visual or tactile feedback is present, there is no guarantee the skill is retained when this feedback is absent.

One inexpensive example of a simple mechanical training aid is the PureShot Sync Ball. It is a ball hanging on a lanyard from the player's neck and squeezed between the forearms. The diagram in Figure 3 shows the structure of the device. If the player tends to let their arms get too far from each other or does not have proper extension of their arms during the back swing and down swing portions of the swing, they could employ the Sync Ball to correct this issue. Simply holding the ball in place requires the player to keep their arms sufficiently close together through the motion. If the player also correctly maintains a slight tension in the cord through the swing, it creates extension on the player's backswing and downswing leading to a larger swing arc and more power and speed.

In this case the player both receives tactile feedback during the swing, and their range of movement is restricted. Their forearms must stay close enough to each other to hold the ball. The ball must also be held far enough from the neck to maintain the tension in the cord. At the same time they may get used to having the Sync Ball between their forearms while swinging and this may lead to an exaggerated motion when the ball and the feel it gives are absent.

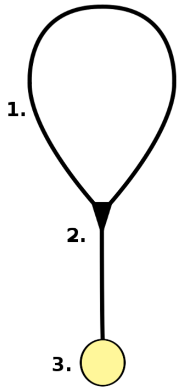

Figure 3. Structure of the PureShot Sync Ball training aid: 1) the loop of the lanyard that goes around the neck, 2) the adjustable length cord, 3) the ball held between the arms.

In another example a product called the Swing Jacket [4] restricts the player's arm movements to those of a proper swing. The jacket has two armbands worn around the upper arms. The armbands are attached to rails on the chest of the jacket. As the player swings the club the armbands are forced to follow the path dictated by the rails. This prevents the player's arms from getting too far from the body thus forcing correct form.

Similarly to the Sync Ball the Swing Jacket also both gives tactile feedback and restricts the player's range of movement. If the player is used to the Swing Jacket forcing them to a specific path of motion, they may not perform the same exact movements with the restrictions to their movement gone.

Both of the above training aids can be used while practising at the range, as well as through a normal round of golf while practising or playing outside of a competition setting. They do however alter the situation and the feel of the swing.

At the other end of the scale from these portable, inexpensive and relatively simple training aids, there are also devices designed to guide the entire swing motion. For example the Plane Perfect Golf Machine [5], shown in Figure 4, guides the club's path through the entire motion, and does not allow the user to deviate from the correct plane of the swing. The device offers various adjustments to fit players of different sizes and swinging angles.

The previously mentioned potential problems also apply to this device. Since the correct plane of the swing is forced by the device, the learned swing may not completely transfer to a normal game of golf. As seen in Figure 4, the device is also too large for it to be used through a normal round of golf, limiting its usability purely to a dedicated practice situation. This may hinder transferring the trained skills to practice.

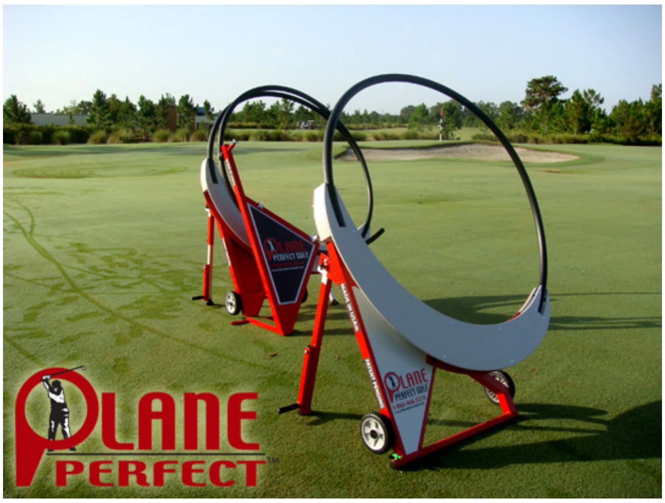

Figure 4. The Plane Perfect Golf Machine. (Courtesy of Perfect Golf Labs)

2.2. Electronic systems

The other main category of training aids are the various electronic aids. These usually function by measuring the motion of the club during the swing, or the motion of the ball in flight. These measurements are then analysed and the player is given for example visual or numerical feedback. Electronic training aids often do not appreciably change the feel of the stroke so there is less risk of acquiring bad habits purely because of using a training aid. At the same time the feedback they provide is not quite as immediate or constant, and its accuracy does depend on the accuracy and interpretation of the measured data. The usefulness of the feedback also relies on the user being able to understand it correctly and then being able to adjust as needed. For example if the feedback gives a club face angle figure, the user needs to know what kind of number they are looking for and what change to make to achieve the desired result.

2.2.1. Doppler radar based systems

Ernest Sports ES14 launch monitor [6] uses doppler radar sensors to track the club head and the ball during a swing. It is a stand-alone device slightly bigger than a smartphone. It is placed on the ground close to the ball, where it can use its two doppler radar sensors. One sensor is used to measure the club head speed before impact and the other measures the speed of the ball after impact. Based on these directly measured data other statistics such as spin rate and distance are also calculated. The device is able to give audible feedback as well as communicate with a smartphone to show the feedback on an accompanying mobile application. To make use of the device it needs to be set up prior to the stroke at each new location.

Another doppler radar based system, the Zelocity PureLaunch Tracker [7] uses 3 separate doppler radar sensors positioned in a triangle formation around the position of the ball, in the way shown in Figure 5. As the system uses multiple sensors it is potentially able to give a more complete picture of the swing compared to the ES14, but this comes at the cost of even more set-up required. The system weighs just over 3kg and includes two separate sensor pods that need to be set a specific distance behind and in front of the ball [8]. The bulk and the necessary set-up exclude the use of the system outside of dedicated practice situations.

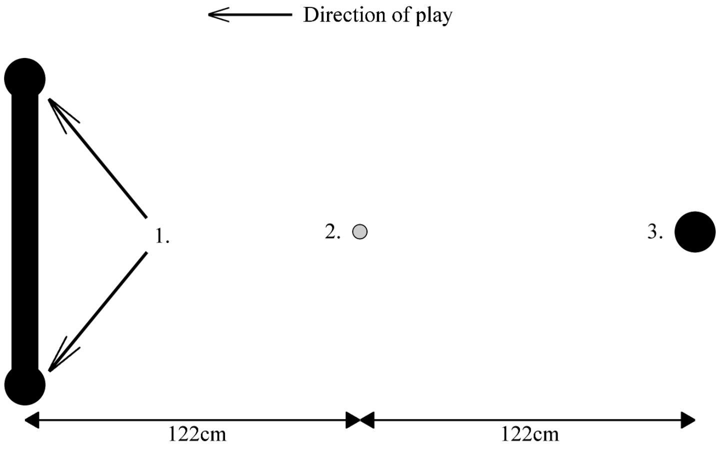

Figure 5. Set-up of the Zelocity Purelaunch Tracker system: 1) the front sensor pod with two sensor units connected by a mat, 2) the ball in the middle of the two pods, 3) the rear sensor pod.

2.2.2. Inertial measurement unit based systems

These systems typically consist of a sensor platform that is attached directly to the club and communicate with a smartphone over a wireless connection such as Bluetooth.

The 3Bays GSA Pro Golf Swing Analyzer [9] and the 3Bays GSA Putting Analyzer [10] are systems that consist of a small rechargeable battery powered sensor unit that is attached to the end of the grip of a club, and an accompanying mobile application that presents the measured data. The sensor devices have three axis gyroscope and accelerometer sensors which are used to record approximately 10000 data points per a swing event. The system uses the data to present the user with a variety of information about their swing, including an animation of it as well as numerical statistics on metrics such as club head speed, face angle, swing tempo and so on. The device works for up to 5 hours, which is long enough for a typical full round of golf, but its usability while playing is limited by the need to move the sensor device to each club as they are used if one wishes to record all of their swings.

Another similar system is the Arccos Golf GPS & Stat Tracking System [11]. In contrast to the 3Bays GSA systems the Arccos system shown in Figure 6 includes 14 individual sensor devices attached to the end of each club's grip and an accompanying mobile application the sensor devices communicate with. The sensor devices communicate with the iPhone over Bluetooth LE (Bluetooth Low Energy). The system uses the sensor devices and the iPhone's GPS location data to record locations of strokes automatically. The sensor devices are small, battery powered and need to be able to last significantly longer than a single round of golf between battery changes. The system uses Bluetooth LE for transferring the data which saves significant amounts of battery power, but also limits the amount of data that can be recorded and transferred to the phone. This limits the kinds of feedback the system is able to generate. Only the time and location of the strokes taken is recorded. Because there are 14 separate sensor devices changing the batteries as they run out may also be bothersome for the user, but otherwise the system is relatively unintrusive.

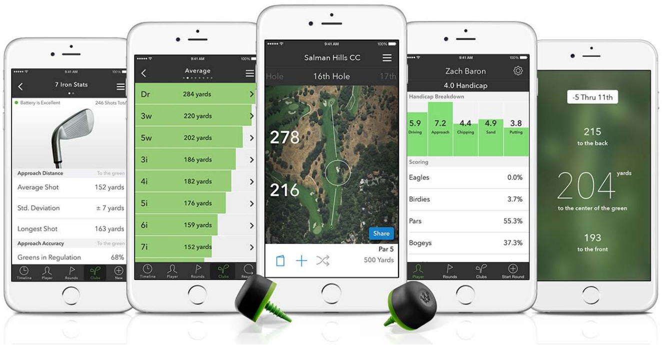

Figure 6. The Arccos Golf GPS & Stat Tracking System: the small sensor modules are screwed into the end of each club's grip, and they communicate with the smartphone over Bluetooth LE. (Courtesy of Arccos Golf)

The Zepp Golf Swing Analyzer [12] also avoids the need for the user to move the sensor devices when changing clubs. Instead of attaching the device directly to the club it is attached to the back of the player's glove using an accompanying clip. The device contains a three-axis gyroscope and dual accelerometers and communicates over Bluetooth with the user's smartphone, which is used to show the user feedback on their swing. The device is so small and light that the user is unlikely to be distracted by it, and the expected battery life of up to 5.5h [13] should suffice for most golf rounds. However, it does need to be recharged before each round and even then it is possible it will run out of power. The system is also unable to provide any feedback on putts.

2.2.3. Vision based systems

Perhaps the simplest vision based approach is a video camera. The Swingshot Cyclops^{TM} range of cameras [14] for example are made purely for use on the course. The system includes a high-definition digital video camera attached to the end of a pole with a sharpened end. The player can simply stick the pole into the ground and press a button to start recording. The device also contains an LCD screen and a GPS module so the location of each clip is also recorded.

While the player can use the embedded LCD screen to immediately check their swing on the field, due the small size of the screen the recordings are best analysed on a separate device after the practice or game is over. Thus the feedback is not immediate. The quality of the feedback is also limited by the player's own ability to analyse the recording, since there is no external assistance to help them.

Another way vision is used is to track objects and people through different machine vision systems. One relatively simple way to do this is to attach beacons or other markers to the tracked objects. By finding these markers in the recorded picture or video, their coordinates can be calculated and thus the movement of the object they are attached to can be tracked. A significant drawback in using this method is that since both the camera and the markers are needed, a certain level of infrastructure is always needed.

One example of such a system is TOMI [15]. It is focused on measuring and giving feedback on putts. It consists of a battery powered clip featuring four LEDs (Light emitting diode), a camera that is used to track the locations of the LEDs through the putt stroke and a computer application that handles the raw data. The system was tested in [16] by using a putting robot to produce predictable and repeatable putts and a laser grid system and a high-speed camera to independently and accurately track the club. While the raw LED coordinate data was handled using standard 3D data-processing techniques rather than the software application of the TOMI system, it showed potential to reliably give valid measurements on various metrics of the putting stroke.

The TOMI system is limited to only measuring putts, and it requires a stationary camera which is connected to a computer. It cannot easily be moved so it is only practical to use in dedicated practice situations. As the clip is attached to the front side of the putter's shaft its weight and size may also slightly hinder the user while training.

2.2.4. Golf simulators

Golf simulators are perhaps the most complex electronic golf training systems. They are located indoors and they attempt to simulate a real world location on a real or imaginary golf course. Indoor golf simulators typically consist of a swinging pad with fake grass and a tee in front of a view of the simulated golf course projected by an overhead projector. The user uses a real club and ball and hits the ball against a net that captures it. The club, ball and player are tracked using a variety of technologies such as high-speed cameras, doppler radars, pressure sensors and infrared sensors. The measured data is processed by a computer.

As the user hits the ball the system calculates the ball's trajectory and displays it on the screen. The location where the ball lands on a simulated golf course is calculated.

The next stroke takes place there. In this manner the user is able to play entire rounds of golf without ever going outside. Golf simulator systems are usually installed permanently into a dedicated space and as such they cannot be moved. They also only simulate the conditions of the game so the experience is never quite the same as out on a real golf course.

The Full Swing Golf simulator [17] depicted in Figure 7 is a fairly typical simulator. It uses a high-speed camera located above the tee spot to measure data such as club head speed, club path, club face angle, and the ball's spin characteristics. Additionally two infrared gates measure the ball's speed, launch angle and direction. Based on this data a computer calculates the ball's trajectory which is then displayed on the screen as previously described. The system, as most golf simulators, is quite large in size. Even the smallest model is 3.5m wide, 2.6m tall and 5.3m deep.

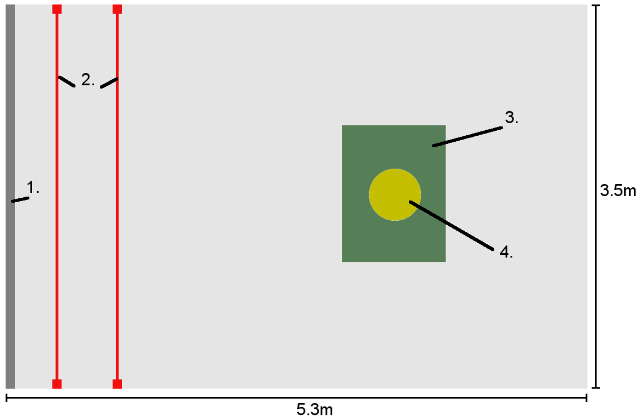

Figure 7. A diagram of the Full Swing Golf simulator as seen from above: 1) screen, 2) infrared gates, 3) swinging pad, 4) the area seen by the ceiling mounted high-speed camera.

2.2.5. Game systems

Golf simulators can be considered to be the sport of golf made into a highly advanced game. Many other examples of less realistic golf games exist, ranging from computer games that are played using a mouse and a keyboard to games on modern console gaming systems which leverage some form of motion tracking to incorporate golf-like movements into the gameplay.

The Nintendo Wii U Sports Club is a collection of games that use the Nintendo Wii U controllers to simulate several sports. One of the simulated games is golf. [18] The Nintendo Wii U Golf is in principle quite similar to the previously mentioned more serious golf simulators. The Wii U game system controllers consist of a GamePad which has a 6.2" LCD (Liquid Crystal Display) screen and a Wii Remote, which contains a three-axis accelerometer and an optical sensor that is used to track an accompanying Sensor Bar which has 10 infrared LEDs. With these sensors the Wii Remote's motions can be tracked. The GamePad serves the purpose of the swinging pad. To play the game the player sets the GamePad on the ground and the club head and the ball are displayed on the screen of the GamePad. The player controls the angle and motion of the displayed club head with the Wii Remote held by the player like the grip of a golf club. The Wii Remote is swung as if it was a real club and the system measures the swing and simulates the ball on the GamePad's screen being hit. The game console is also attached to a display device, such as a television or a projector, and the simulated ball and its trajectory is displayed on the screen.

The biggest differences between the Wii U Golf and the previously mentioned golf simulators are mostly in the accuracy of the simulation and the motion measurement. As the main motivation behind the game is simply providing entertainment with a low learning curve it forgives many mistakes in the swing to easily give players the feeling of achieving a successful swing. The golf simulators on the other hand try to simulate each swing to the highest possible accuracy, including mistakes and failed swings. As such the game systems are generally not useful as training aids for golf, although the sensor systems themselves could potentially be used for a more realistic simulation as well.

2.3. Conclusions on training aids

An ideal training aid would be inexpensive, would help with all kinds of strokes and in all situations, including practice rounds, and would be unintrusive, meaning it requires little maintenance or set-up, and is unnoticeable during use.

All mechanical training aids rely on being felt or seen during the performance of the swing or putt, so they are by definition intrusive. Most electronic aids on the other hand do not affect the look and feel of the stroke, the user merely checks the feedback after the stroke, but they fail in other respects. Many require set-up before each stroke. Many others are locked to a single club meaning the user either receives feedback only on that specific club or has to move the device to a new club when changing clubs.

The Arccos Golf GPS & Stat Tracking System, which includes separate sensor devices for each club, does not require significant set-up before each stroke. After the initial set-up, the main maintenance task needed is changing the batteries of the sensor devices as they are expended. The system is not without flaw though, as the system only detects when a stroke has happened without providing any feedback on the quality of the stroke.

Perhaps the closest fit for these criteria of an ideal training aid is the Zepp Golf Swing Analyzer system, which requires little set-up and is mostly unnoticeable during use. However it is only restricted to providing feedback on full swings. Additionally, while it would be possible to implement putting feedback using the same or a similar device, it does require the user to wear the glove the device is attached to through the whole exercise. This may not be to every user's liking.

Overall, any system that requires attaching battery powered sensor devices to the club faces the same issues. The number of clubs used during each round of golf requires multiple devices or moving the device. The need for the devices to be unintrusive limits their size and weight, which results in restrictions on processing power and the amount of data transferred. This limits what the devices are able to do. As such, wearable devices seem like an attractive option when considering the amount of set-up and maintenance required, as well as from the perspective of battery life versus processing power and data bandwidth.

3. EMBEDDED MOTION MEASUREMENT TECHNOLOGIES

All the electronic training solutions described in Chapter 2 have characteristics that affect their practical day-to-day usability. These shortcomings include intrusiveness, inability to cover all training scenarios, and short battery life. This thesis aims at a solution that tackles these compromises. Taking a look at the technology base of the currently available training solutions is thus needed.

The previously described training solutions comprise of measurement instrumentation and analytics algorithms. Analytics are most straightforwardly implemented with "complete" measurement data. However, compromises need to be made due to cost and practicality reasons. Furthermore, less complete measurements may suffice with advanced analytics. In the following measurement methods potentially suitable for measuring motion in golf are considered.

3.1. Legacy systems

3.1.1. Magnetic tracking

Magnetism can be used to track objects using the ambient magnetic field present at the location, or a field deliberately created with permanent or electric magnets. This is one of the oldest working wireless motion tracking solutions.

In 1969 Polhemus introduced their first electromagnetic tracking system for head tracking for the military. In the decades since they have continued to develop tracking solutions for military and commercial applications. Their current products range from Micro Sensor 1.8, which is a 6DOF (Degrees Of Freedom) tracking element 1.8mm in diameter, to the LIBERTY LATUS^{TM} system, shown in Figure 8, able to cover areas of tens of square meters. [19] The systems provide 6DOF tracking natively and without drift, which means that both position and orientation are measured automatically and independently of each other and each measurement is independent of the previous one so measurement errors are not accumulated over time. The system includes a source that uses alternating current to create an electromagnetic dipole field, one or more battery powered markers that are attached to the tracked objects, and one or more receptors that are connected to a computer.

The markers can be made small enough to be embedded into a golf club or to be worn by the player, but the amount of external infrastructure this technology requires limits its usability in golf. Additionally the markers are affected by metal in the near vicinity, which might limit the selection of suitable golf clubs. The system also requires calibration. Even in an indoor practice space increasing the distances between the components of the system would also quickly cause the measuring accuracy to deteriorate. The number of simultaneously tracked objects is also limited. [20]

3.1.2. Vision based systems

Similarly to the Polhemus electromagnetic tracking system, all vision based systems require at least some external infrastructure. Instead of being worn by the player or

Figure 8. Polhemus LIBERTY LATUS$^{\text{TM}}$ system's components. (Courtesy of Polhemus)

attached to the equipment they require at least one camera to observe the player. To be able to completely track the motion, several cameras may be required to capture the movements from several directions as the player's motion obscures parts of the body and the equipment from any single angle.

The task of tracking objects accurately from visual data can be simplified by attaching markers or beacons to the objects to use as reference points. This comes at the cost of added inconvenience to the person whose motions are being tracked. Even with markers problems may still persist [21].

3.2. Inertial measurement units

With the development and improvement of MEMS (Micro-Electro-Mechanical System) technologies in the past few decades [22] it has became possible to embed inexpensive, small, accurate and low power sensors of various kinds into new devices.

By combining different MEMS sensors it is possible to achieve full 6DOF tracking of objects. For example, the data from an accelerometer that measures linear acceleration across the X, Y and Z axes combined with a gyroscope measuring rotation around the same axes can be used to track the object's path in all three dimensions, as well as its rotation and orientation. As opposed to the previously described magnetic and visual tracking systems, IMU (Inertial Measurement Unit) based systems typically do not require any external infrastructure. This combined with their small size and relatively low power requirements enables creation of truly unintrusive and mobile applications for motion measurement.

MEMS sensors are made using modified semiconductor fabrication technologies, the same processes that are used to manufacture for example modern microprocessors. The same integrated systems contain both analog and digital circuitry and even moving mechanical components at the micrometer scale [23]. Figure 9 demonstrates the scale and detail of modern MEMS components.

The level of integration of MEMS devices varies, but quite complex systems are possible today. For example the Freescale FXTH87 product family of Tire Pressure Monitoring Sensors integrate an air pressure sensor, a single or dual axis accelerometer, an 8-bit MCU, 512 bytes of RAM (Random Access Memory) and RF (Radio Frequency) receiver and transmitter units all in a single package 7 x 7 x 2.2 mm in size [24]. For example, the accelerometer integrated into the units with single axis accelerometers offers dynamic ranges of up to -270g to +350g [25].

It is apparent that MEMS sensors are capable of offering sufficient dynamic range to measure golf swings, particularly at the wrist where the accelerations are significantly smaller than for example at the head of the club. There are also numerous sensors on the market with smaller dynamic range and higher accuracy than the accelerometers on the above sensors.

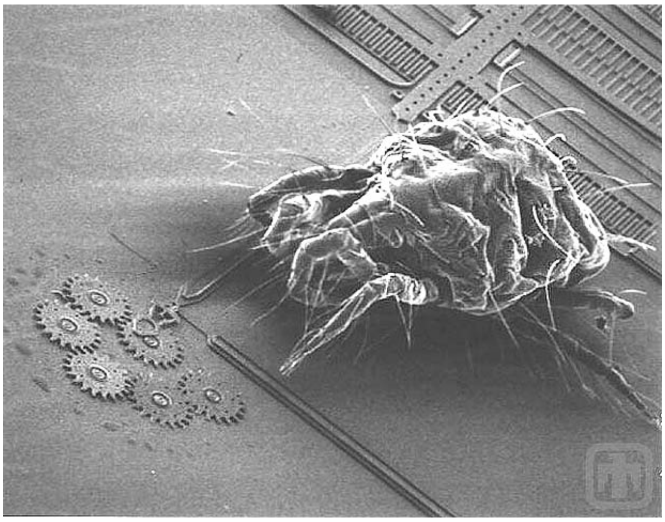

Figure 9. A spider mite next to some MEMS gears with typical lattice structures used in for example MEMS accelerometers and gyroscopes in the back. (Courtesy of Sandia National Laboratories, SUMMiT^{TM} Technologies, www.sandia.gov/mstc)

3.2.1. Gyroscopes

Gyroscopes are used to measure the change in orientation of an object, but they cannot detect movement along the axes. Figure 10 shows the elemental rotations α, β and γ around axes X, Y, and Z respectively. To obtain an orientation based on these individual elemental rotations, they need to always be considered in the same order starting from a known initial orientation. These sequences of rotations are called Euler angles and there are several different ways to define them, any of which can be used. Additionally, there are a number of other ways to handle rotations mathematically.

The first gyroscopes were mechanical in nature, based on the principle of conservation of angular momentum. Figure 11 shows the basic structure of a mechanical

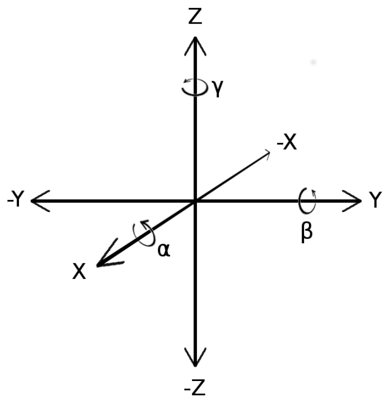

Figure 10. 6 degrees of freedom: axis X, Y and Z, and the rotations around them, α, β, and γ.

gyroscope. These devices were bulky and heavy and as such not suitable for mobile applications modern MEMS gyroscopes enable.

There are several types of MEMS gyroscopes, such as for example tuning fork gyroscopes [26], vibrating-wheel gyroscopes [27] and wine glass resonator gyroscopes [28]. All these variations are based on the Coriolis effect on a vibrating mass in a rotating reference frame [23]. For example the tuning fork gyroscope [26] has a pair of masses, like the prongs of a tuning fork, that are made to vibrate with same amplitude but in opposite directions. When the system is rotated the Coriolis force creates a vibration that is orthogonal to the plane of vibration of the vibrating masses. This induced vibration can be measured using several different methods and the measurements can be used to deduce the amount of rotation.

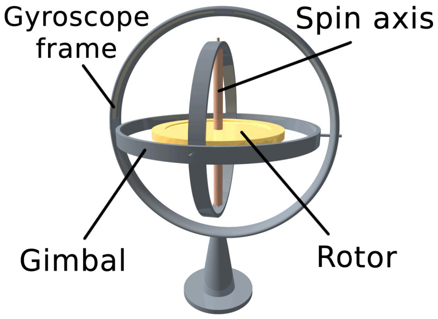

Figure 11. The basic structure of a mechanical gyroscope: as the frame rotates, the gimbals rotate to allow the spinning rotor to constantly maintain the same orientation.

MEMS gyroscopes suffer from several different types of error: bias, sensitivity scale, sensitivity non-linearity and cross-axis sensitivity. Bias means a constant offset from zero while in reality there is no rotation happening. Sensitivity scale refers to change in sensitivity over time and sensitivity non-linearity to the differences in sensitivity at different parts of the sensitivity scale. Cross-axis sensitivity happens in the case of multi-axis gyroscopes where the axes are not perfectly perpendicular and rotation around one axis causes an erroneous reading on another axis. [2]

All these small errors contribute to drift, which is the accumulated error in the gyroscope orientation output. As the current orientation is always the sum of all previous changes in orientation, even any small rotation measurement errors slowly add up over time. Thus, without external calibration data to allow for correcting the error, the measured orientation slowly drifts away from the actual orientation. For the purposes of tracking single golf strokes this does not necessarily pose a problem, as due to the short duration of the event the accumulated error is insignificant. Additionally, relatively safe assumptions about orientation and position in the starting position of a golf stroke can be made. This reduces the impact of any measurement error accumulated before the start.

Similarly to how assumptions can be made about the starting position of the stroke, the actual stroke always follows largely the same pattern that is repeated each stroke. This makes it possible to form a model of a stroke based on the data a gyroscope provides, and to recognize strokes and deviations from optimal technique in the measured data. As such gyroscopes alone could suffice for measuring the motion of a golf stroke.

One downside of MEMS gyroscopes is that compared to some other MEMS sensor types such as accelerometers, they consume more power. This is because they function by measuring the Coriolis effect on a vibrating mass which has to be actively excited to vibrate with a specific amplitude for any measurement to take place. Additionally, the measurements must be continuous, as the orientation measurement is a sum of previous rotations and always relative to the orientation at the time of starting the measurements. In the context of a mobile application even small differences matter.

3.2.2. Accelerometers

An accelerometer measures acceleration along the axes X, Y, and Z, but it cannot detect rotation around the axes. In practice, an accelerometer's output is a sum of static acceleration caused by gravity and dynamic acceleration caused by the motion of the sensor.

If the sensor is not rotated and it is moved in a plane that is perpendicular to the ground, its velocity can be calculated as an integral of the measured acceleration values and the position as an integral of the velocities. If the sensor is rotated in addition to moving, or the motion is not in a plane that is perpendicular to the ground, the effect of gravity must be removed before integration to obtain these values. An accelerometer cannot measure rotation. Therefore, accounting for the effect of gravity in the case of rotating the sensor requires the use of additional sensors such as gyroscopes to measure the current orientation. [2]

An accelerometer consist of a proof mass that is suspended from spring structures that are anchored to a fixed frame. Figure 12 shows the basic structure of a MEMS accelerometer. Knowing the mass of the proof mass and the spring factor of the spring structures allows acceleration to be measured by determining the displacement of the mass relative to the fixed frame. [29]

Accelerometers suffer from all the same sources of error as gyroscopes [2]. If velocity or position are calculated based on the acceleration measurements, they are also

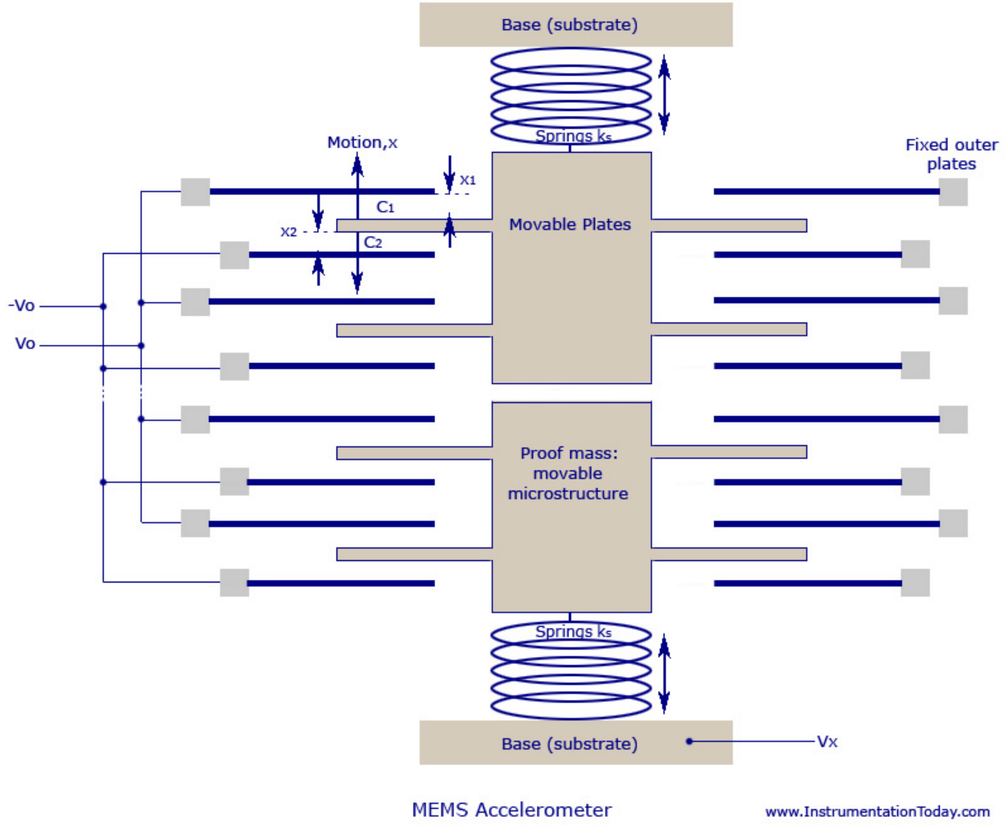

Figure 12. MEMS accelerometer basic structure: as the mass moves the plates attached to it move in relation to the fixed outer plates, affecting capacitances C1 and C2, which can be used to calculate the acceleration the mass is experiencing. (Courtesy of Instrumentation Today)

vulnerable to drift, as they are calculated as integrals of the individual acceleration measurements. As with gyroscopes, this is not considered to be an issue when measuring golf strokes due to their short duration.

In the context of golf strokes, the sensor's velocity and position during a putt can be tracked more accurately than during a full swing, when using only an accelerometer. This is because the plane of the motion is close to perpendicular to the ground and the rotation of the sensor is minimal compared to rotation during a full swing. As previously stated, a golf stroke also follows a predictable pattern, which can be used to detect strokes by means of an accelerometer without additional sensors.

3.3. Embedded motion measurement in golf

When considering the use of motion measurement in golf in this thesis, the focus was primarily on putting rather than the full swing. Both movements are restricted movements that follow a specific expected pattern. Putting seems both easier to simulate and harder to detect, as the range of movement, velocities, and the forces applied, are significantly smaller.

Several methods for automatic putt recognition and technique feedback have been proposed in the literature. Most systems use sensor platforms attached directly to the club which is obviously a good choice purely for measuring the stroke itself, as it allows for direct measurement of the position and rotation of the club. Additionally, most such sensor systems use both accelerometers and gyroscopes so they are capable of full 6DOF motion tracking.

In [30] and [31] a SHIMMER^{TM} sensor platform with the gyroscope daughterboard [32] was used attached to the putter head combined with an Android smartphone to create a system for automatically detecting and giving feedback on putts. The SHIMMER^{TM} platform transmitted three-axis acceleration data as well as three-axis gyroscope data to the smartphone over Bluetooth radio. On the phone a custom made Android application received and processed the data in real time.

A model of a putt was created by measuring putts of 15 completely inexperienced golfers after they had received a basic introduction to the putting movement, but no coaching. The system was then used to detect putts performed by 11 completely inexperienced subjects during repetitive training without coaching or feedback other than whether the putt was holed or not.

They achieved an overall putt detection result of 68.2%. Each test subject's detection rates ranged from 3.1% up to 98.9%. If the three subjects with the worst detection results were excluded the overall putt detection rate increased to 83%. It was found that the detection rate depended strongly on whether the test subject's technique followed the technique used in making the model for putts. The false-positive rate was 2.4%.

In another study [33] a system was developed that tracked the putting motion and automatically classified the experience level of the subject that performed the putt. The classification was based on putting motion data recorded using the same SHIMMER^{TM} measurement system mentioned previously. It was found that the system could differentiate between experienced and inexperienced players with a classification rate of 86.1% based on a single putt.

Based on these studies [30] [31] [33], some conclusions can be drawn. It seems that the putting motion is consistent enough between different subjects that a generic model can be used to detect putts performed by several people. However, at least for inexperienced golfers, there are differences that may reduce detection accuracy.

Additionally, there are sufficiently distinct differences between experienced and inexperienced golfers that it is possible to distinguish between the two automatically on a putt by putt basis. This indicates not only that it is possible to offer feedback aimed at improving the golfer's performance, but also that the motion of putting is consistent enough to be detected from golfer to golfer.

Another study brings up the issue of user comfort. In [2] a framework for the use of MEMS inertial sensors as a low-cost putting coach for golf is developed. A detachable sensor platform able to transmit three-axis acceleration and gyroscope data over Bluetooth is custom built and attached to the backside of the shaft of the club close to the grip. In the design of the sensor platform, not only the purely technical concerns are considered, but also the comfort of the player.

To minimize the intrusiveness of the device it is positioned close to the grip on the shaft, so that its weight does not affect the balance of the club noticeably. The device is also placed on the back side of the shaft so that it does not block the golfer's line of sight down the shaft of the club towards the ball. It is also mentioned that attachments, such as this sensor platform, must be easily removable from the club to allow the player's practice club to conform to the rules of the game.

In [34], another non-intrusive wireless 6DOF sensor platform is designed. The device is small enough to fit inside the shaft of the club and so light that it weighs less than half the weight of the grip itself. This means that once the device has been installed into a club, it is practically unnoticeable.

The device allowed highly accurate tracking of the club. On the other hand, this implementation only allowed for about two hours of continuous run-time which is not sufficient for an entire round of golf. The device also cannot easily be removed.

In another study [35] a system with a graphical user interface for providing feedback to improve the motor skills of a subject performing a golf putt was created. The system presented feedback in the form of video taken of the putt, combined with tempo information derived from data measured by a three-axis gyroscope attached to the base of the club. The majority of the participants in this study had improved their putting tempo after using the system, and their tempo became more consistent.

While the main purpose of this study was not to automatically detect individual putts, but rather measure a specific statistic of the putting motion, it is of interest to note that the tempo measurement was successfully performed using a single sensor. This is possible because the motion can be expected to follow a certain fairly strongly restricted pattern, which greatly simplifies the problem of recognizing the gesture in question.

This pattern has been identified for example in [36]. Putting movements performed by both inexperienced and expert level subjects was tracked using a visual motion capture system. It was found that particularly for inexperienced golfers, the pattern took the shape of a pendulum that reaches the same amplitude at both sides of contact with the ball. For the expert level subjects the follow through phase of the putt was extended a little further, while the motion was performed more precisely in a plane perpendicular to the ground. The motion of the club could be described as a pendulum motion of the club-subject system in both cases. It should be possible to track the putting motion as well from the wrist as the club. They both are parts of the same pendulum system.

It is also worth noting that using a three-axis accelerometer it is possible to determine the direction of earth's gravity during the address phase of a golf swing, as mentioned in [2] and [34]. This is not possible using only a gyroscope.

3.4. Conclusions on embedded motion tracking in golf

Despite advanced features, none of the systems presented here seem to fit the description of an ideal training device presented in Chapter 2.3. All of the above methods involve attaching a sensor platform directly to the club. The official rules of golf state that "All parts of the club must be fixed so that the club is one unit, and it must have no external attachments. Exceptions may be made for attachments that do not affect the performance of the club." [1].

This seems to exclude the possibility of detachable sensor platforms and even ones integrated directly into the club would not necessarily be acceptable. This is significant, because it limits the usability of all of the above methods. Using such a system would likely be limited purely to practice conditions, where a more casual approach to the rules may be more permissible.

Even when considering playing entire rounds of golf outside of tournament settings the presented systems have limitations. The most significant one is that normal play practically always involves using several different clubs during the round. This would mean that either the user needs to manually transfer the sensor device from one club to another, or the system needs to include separate sensor devices for each used club. The first solution would be intrusive during normal play and the second would increase the complexity and the combined cost of the system.

Using a sensor worn on the body, for example a wrist device, would remove these limitations. Little research seems to have gone into sensing the motions of golf putts using such platforms. It is undeniably challenging to detect the delicate motions and angles of a golf putt without isolating the motion of the club. Nevertheless, the qualities of the putting motion indicate that it should be trackable at the wrist. Considering the simplicity of use offered by using a wrist device this is worth investigating more closely.

Furthermore, it seems that due to the restricted nature of the putting motion it could possibly be detected with sufficient specificity using only a single sensor. Both gyroscopes and accelerometers could potentially be used to such an effect, but using an accelerometer offers the benefit of being able to detect the starting condition of a golf swing. That is, the address position taken before starting the back-swing.

4. WRIST DEVICE HARDWARE DESIGN

After considering currently existing training aids, it was concluded that a wrist device could be an unintrusive way to track the movements of the player's hands during a golf stroke. Most existing devices based on MEMS sensors use both gyroscopes and accelerometers, which results in a reduced battery life. Since battery life is an important factor in the user experience, it was of interest to attempt to create a system using only a single sensor.

It was decided to create a prototype of a wireless wrist worn battery powered sensor platform. The goal was to use this device to collect data from real situations for use in algorithm development and verification, as well as to test the viability of individual components and the concept as a whole.

A detailed description of the design and implementation of the hardware of the device is outside the scope of this thesis. It is however of interest to look at component selection from the point of view of both performance characteristics and impact on software implementation.

It was important for the first prototype to be ready quickly, so challenges of implementation from both the software and hardware point of view affected component selection and other choices. Another major consideration was that the device had to be battery powered. The obvious solution was to use a rechargeable 3.7V Li-Ion (Lithium-Ion) battery, which come in a range of sizes and capacities and offer many positives such as high energy density, no memory effect and affordable prices.

4.1. Motion sensing

4.1.1. Sensor set-up options

Golf swings and hits can be considered as gestures that start from a stationary position and proceed through sequences of translations and rotations. The combination of both a three-axis accelerometer and three-axis gyroscope is the only one of the considered sensor set-ups that is able to track such a gesture accurately and reliably. A golf swing should however be a repeatable and predictable gesture, so less than perfect tracking should suffice for reasonable recognition accuracy.

Based on this assumption, three primary sensor set-ups were considered. In addition to the more commonly used combination of both accelerometers and gyroscopes, set-ups with a single three-axis gyroscope or a single three-axis accelerometer were also considered. Table 1 shows some features of each set-up.

Considering a set-up with only a three-axis gyroscope, it seems that it should be possible to recognize the set of rotations that happen during a golf swing gesture. However, it is not possible to deduce anything about the plane of the motion. Thus, should the same gesture be performed in some other plane than the intended, there would be no way to distinguish between the two. For example, waving a hand as a greeting could potentially be mistaken for a swing. Additionally, it may be difficult to detect the moment of impact between the ball and the club based on the slight change in rate of rotation at the wrist.

Table 1. IMU sensor set-up options

	Accelerometer	Gyroscope	Both
Can detect movement	Yes	No	Yes
Can detect rotation	Indirectly by using the direction of gravity if stationary	Yes	Yes
Can detect relation to ground	If stationary	No	Yes
Can accurately track entire gesture	No	No	Yes

Using a set-up with only a three-axis accelerometer, it should also be possible to recognize an acceleration signal resulting from the movement and rotations that take place during a golf swing. It is also possible to detect the direction of gravity when the sensor is stationary, which should make it possible to detect the address position at the start of a swing. The moment of impact is assumed to be easily detectable from the acceleration signal even at the wrist.

The combination of accelerometers and gyroscopes would naturally be the best option for accurately tracking golf swings and hits. However, using both sensors would obviously also result in higher power consumption than either sensor on its own.

It was decided that only a single sensor would be used. As shown in Table 2, gyroscopes have significantly higher power consumption than accelerometers. Additionally accelerometers offer the advantage over gyroscopes of being able to detect the address position at the start of the swing. Therefore it was decided that a single accelerometer would be used.

Using separate single axis accelerometers was also considered, but it was concluded that using a single three-axis accelerometer chip would simplify both hardware and software implementation and reduce the chances of cross-axis sensitivity issues.

4.1.2. Sensor selection

Table 2 shows the sensors that were considered for use. Besides gyroscopes, analog accelerometers such as the Analog Devices ADXL326 were also rejected, due to the added complexities in handling them when compared to the digital alternatives.

Out of the three digital accelerometers considered the ST LIS3DSH was chosen due to its internal programmable state machines. All of the digital accelerometers under consideration are able to independently output an interrupt based on certain acceleration events, such as a tap, a double tap, any movement and so on. However, the ST LIS3DSH is unique in that it has two 16 state fully user programmable state machines, making it the most flexible of the three in this regard. This was considered valuable as at a later point it could be used to detect the moment of address during a golf swing. The accelerometer could run independently until a potential golf swing is happening, thus allowing the system to save energy. Active measurements would start only when an actual golf swing is likely to occur.

Table 2. Inertial measurement units

Item	Range		Output datarate		Power consumption
	Min	Max	Min	Max	
Analog accelerometers					
Analog Devices ADXL326		±19G	0.5Hz	X & Y: 1600Hz, Z: 550Hz	350μA
Digital accelerometers					
Analog Devices ADXL345	±2G	±16G	100Hz	3200Hz	150μA @>= 100Hz
Bosch Sensortec BMA222	±2G	±16G	8Hz	1000Hz	139μA @1000Hz
ST LIS3DSH	±2G	±16G	3.125Hz	1600Hz	225μA @1600Hz
Digital gyroscopes					
Bosch Sensortec BMG160	±125dps	±2000dps	100Hz	2000Hz	5.0mA
Freescale FXAS21002C	±250dps	±2000dps	12.5Hz	800Hz	2.7mA
ST LG3200DH	±250dps	±2000dps	95Hz	760Hz	6.1mA

4.2. Wireless communication

4.2.1. Communication system selection considerations

Certain features were considered when selecting the wireless communication system. These are listed in Table 3.

Table 3. Wireless communication system selection considerations

Consideration	Explanation
Minimum data rate	Must support data rates of at least 80 kbit/s.
Minimum range	Must have at least 1 meter range.
Device support	Must be widely supported by end devices, particularly smartphones and tablets.
Power consumption	Lower is better. Must support low-power mode when there is no data to transmit.
Application profiles	Application profile for transmitting acceleration data from the wrist device to the application in the end device should exist or be easily implementable.

The absolute minimum required data transmission rate is largely determined by how much data the sensor can output. For the ST LIS3DSH sensor, at maximum sampling rate the acceleration data itself with no overhead requires $1600Hz \times 3axes \times 16bits/sample/axis = 76800bit/s$. Some bandwidth must also be reserved for the overhead resulting from packaging of the data and control messages in both directions. It was estimated that about 80kbit/s average transmission rate would be sufficient to be able to constantly transmit the measured data to the end device.

It was also noted that the intended application would present the system with somewhat challenging conditions. The device is worn on the body, and the human body can significantly hinder the propagation of radio waves [37]. In a real life situation not only is the wrist device worn, but the smartphone could be located close to the body. In the case of a right-handed golfer with the phone in the front right pocket and the wrist device on the left wrist, at the end of a full swing their entire torso would be on the direct line-of-sight between the receiver and the transmitter.

4.2.2. Wireless communication system selection

The radio link solutions shown in Table 4 were chosen for closer inspection.

Table 4. Wireless data transmission systems

	Bluetooth Classic	Bluetooth LE	ANT	ZigBee
Theoretical data rate	1-3Mbit/s	1Mbit/s	1Mbit/s	250kbit/s
Range	1-10m	10m	10-30m	10-100m
Peak power consumption [38]	∼45mA @3.3V	∼28mA @3.3V	∼22mA @3.3V	∼16.5mA @1.8V
Device support	Most widely adopted	Some	Some	Unclear

ANT was immediately rejected. Despite the maximum theoretical data rate of 1Mbit/s, the maximum data rate available at the application level is limited to 20kbit/s [39]. This does not satisfy the minimum requirements.

Looking at the remaining three, Bluetooth LE and ZigBee both offer lower peak power consumption than Bluetooth Classic. All of the technologies offer ways to reduce power consumption, when no data is being sent. It is also important to note that while Bluetooth Classic does have higher peak power consumption, this is at least partially offset by the higher data rate.

The quoted theoretical data rates are also significantly higher than what can actually be achieved at the application level. For example, in the case of Bluetooth LE, an experiment was performed to test the data rate between two devices located 0.5 meters apart [40]. The maximum was 58.48kbit/s, which would not be sufficient to meet the requirements in this case. The same authors had also previously modelled the maximum application layer throughput of a Bluetooth LE connection in an error-prone environment [41]. They concluded the theoretical maximum was 236.7kbit/s.

Consequently, Bluetooth Classic was chosen due to its higher maximum data rate and its ubiquitous device support.

4.2.3. Bluetooth Classic module selection

Due to hardware implementation factors and other considerations outside the scope of this thesis, it was decided that the Panasonic PAN1322 Bluetooth module would be used in this device.

This module was suitable from software implementation point of view as it contains the entire Bluetooth protocol stack, leaving only the high-level control for the designer to handle. Additionally, the module supports the Bluetooth SPP (Serial Port Protocol) profile which offers a very simple to use way for transmitting data to the end device. Communication between the microcontroller and the module happens using a simple proprietary protocol over UART (Universal Asynchronous Receiver/Transmitter). [42]

4.3. Microcontroller hardware requirements

The CPU (Central Processing Unit) is perhaps the single most important component of the system. In the case of an embedded system the CPU is often integrated with a number of other systems such as RAM and various peripherals. In this case, the chosen microcontroller would have to be able to support at least a single full-duplex UART bus to connect to the chosen Bluetooth module, and a single I2C or SPI bus to connect to the chosen sensor module. Based on these requirements and the other features of the selected sensor and wireless communication modules as well as on the predicted software features, a number of requirements for the microcontroller could be derived.

4.3.1. Minimum RAM requirements

It was decided that to protect against possible intermittent interruptions in communication the device should preferably be able to buffer at least a single stroke event's worth of data. Taking into consideration the different phases of a golf stroke, the length of a single golf swing event is about two seconds. However, needing to be able to buffer the entire stroke was considered to be the worst case scenario, so one second of data was set as the absolute minimum buffer length. This would allow for the connection to be severely affected before data would be lost.

The chosen accelerometer module provides 16-bit samples of three axes for each sample at a maximum sampling rate of 1600Hz. To achieve the minimum one second long buffer at this sampling rate, $1600Hz \times 3axes \times 16bits/sample/axis \times 1s = 76800bits$, or 9600 bytes of memory would be required. Two seconds, or 19200 bytes of memory, would however be preferred.

Additional memory would also be required for controlling the buffering, as well as for other parts of the wrist device's program. Therefore, the absolute lower limit for the amount of RAM available on the microcontroller was set at 12kB. Twice that was preferred for the added flexibility for the software design and implementation as well as increased tolerance to poor Bluetooth connection conditions.

Adding an external Flash or RAM memory unit was considered, but rejected in favour of choosing a more powerful microcontroller with larger memory capacity. This was primarily due to the simplicity of both hardware and software implementation, resulting in a shorter implementation time, compared to handling an external memory unit.

4.3.2. Word length

While many 8-bit microcontrollers are capable of addressing as much as 64kB of RAM, they typically come equipped with significantly less. Few 8-bit microcontrollers have more than 8kB of RAM available and as such the majority of them are immediately excluded from consideration. One exception is the Atmel XMEGA range of microcontrollers with up to 32kB RAM available [43].

The first prototype was required to only be able to record a continuous stream of data and send it to the end device. For possible future development the device was to be able to perform additional tasks, for example, filtering or compressing the data prior to sending. This processing should preferably be done on the full accuracy data. Reducing the accelerometer data symbol length to 8-bits per axis per sample would make it possible for an 8-bit microcontroller to handle the arithmetics natively without any extra overhead. However, this would result in a loss of accuracy due to the quantization going from 16-bits to 8-bits. Should compression be needed for transmitting the data, multiplication of 16-bit data would be unavoidable, which would cause significant processing overhead on an 8-bit processor lacking 16-bit hardware multiplication support.

While this was not considered an absolute requirement, 16-bit and 32-bit microcontrollers were considered more desirable than 8-bit units.

4.3.3. Power saving states and features

It should be possible to enter a suitable low-power mode, when no processing is to be done, and wake up on an external interrupt from either the sensor or Bluetooth unit. In practice, all modern microcontrollers have low power states available.

4.4. Development support tools

While the previously mentioned hardware requirements are arguably the most important considerations when selecting a microcontroller, other considerations exist as well. The hardware is only one half of the entire embedded system and, for example, poor toolchain support and a lack of suitable early software development platforms can significantly lengthen development time. Therefore, it is important to consider these factors before choosing the microcontroller, in particular, when development time is at a premium.

4.4.1. Software toolchain

A suitable software toolchain is required to develop the software for an embedded device. The minimum requirements are a source code editor, a compiler that translates the source code into computer language objects, a linker that combines these fragments into an executable program and a programmer that allows for installing the generated firmware into the target device. Additionally a debugger that makes it possible to follow the code being run line by line is usually considered mandatory.

There are many commercial IDEs (Integrated Development Environment) that incorporate these parts into a single tool, with support for many microcontroller manufacturer's products. Most polished IDEs were, however, considered too expensive.

Many free options exist as well, and for budgetary reasons it was decided one of these would be selected. Some examples of free toolchain options are listed in table 5.

Table 5. Toolchains

Toolchain	RTOS support	MPU support	Limitations
Keil MDK-ARM Lite Edition	KEIL RTX	Several	32kB binary code size limitation
Coocox CoIDE	Coocox CoOS	Most Cortex-M MPUs	N/A
Atollic TrueSTUDIO	Several	ARM based MPUs	Free 30-day trial
Silicon Labs Simplicity Studio	Several	Silicon Labs MPUs	N/A
Open source tools	Several with plugins	Several. Varying level of support.	Requires set-up

4.4.2. RTOS support

A number of proprietary as well as free, closed- and open-source RTOS (Real Time Operating System) exist in the market today. While creating a port for almost any sufficiently powerful processor is possible, an existing one for the chosen microcontroller greatly reduces the time required for software development.

4.4.3. Development kits

Development kits where the target microcontroller is wired into a demonstration board with different peripherals such as Flash card slots, LCD displays, sensors, buttons and LEDs are commonly available for most microcontrollers. These kits also typically come with a number of demonstration applications, which can work as examples while developing custom software. Due to the time constraints on the development of the prototype device it was decided that only microcontrollers with such development kits available would be considered.

The development board should contain at least the desired microcontroller, or one that is code and pin compatible with the chosen one, and a prototyping area or other easy access to the IO (Input/Output) pins of the microcontroller to allow for connecting the chosen Bluetooth and sensor modules to the microcontroller. This would allow for the software development to begin during the development of the prototype hardware. Table 6 lists a number of inexpensive development kits that meet these requirements.

Table 6. Development kits

Item	MPU on board
STM32L152C-Discovery	STM32L152RCT6
Atmel® XMEGA-C3 Xplained	ATxmega384C3
Freescale FRDM-KL26Z	MKL26Z128VLH4
MSP430F5438 Experimenter Board	MSP430F5438
The LPC1549 LPCXpresso™	LPC1549
EFM32™ Leopard Gecko Starter Kit	EFM32LG990F256

4.5. Microcontroller selection

The previously mentioned requirements and considerations have been collected together in Table 7. With these requirements in mind, the microcontrollers mentioned in Table 8 were chosen for closer inspection.

Table 7. MPU selection considerations

Requirement		Priority
Available RAM	At least 12kB	Must have
Available RAM	At least 24kB	Nice to have
Core type	8-bit, 16-bit or 32-bit	32-bit preferred but not required
Available comm. peripherals	1*I2C or SPI, 1*UART	Must have
Toolchain support	Basic support	Must have a low cost toolchain available
RTOS support	Existing RTOS support	Should be supported by at least one suitable low cost RTOS
Development kits	Low-cost, able to facilitate connecting the chosen external modules	Must be available
Power consumption	<1200μW/MHz	Must have. Lower is better
Up- and down-grade options	Pin/code compatible options available should more or less computing power be required in the future	Nice to have

Out of these microcontrollers the Atmel ATxmega384C3 and TI MSP430F5438A were immediately excluded due to their relatively high run mode power consumption, memory limitation and lack of upgrade path. The lack of upgrade path combined with the limitations of these specific devices was considered a significant disadvantage. Should more resources be needed in the future, changing to an entirely different processor range would be necessary. At that point substantial hardware and software rework would be required.

The devices left are all based on the ARM Cortex-M0+ and Cortex-M3 cores. They all have very similar features and level of development support, and as such it is somewhat hard to find clear differences between them.

As Table 9 shows, in most respects the devices are quite similar. Some differences were however found. Out of the considered microcontrollers the Freescale

Table 8. Some microcontroller alternatives

Item	Clock frequency		Word length	RAM	μW/MHz
	Min	Max			
Atmel ATxmega384C3	32kHz	32MHz	8 bits	16kB	852
Freescale MKL16Z256	32kHz	48MHz	32 bits	32kB	417
NXP LPC1519	32kHz	72MHz	32 bits	36kB	750
Silicon Labs EFM32LG	32kHz	48MHz	32 bits	32kB	633
ST STM32L151RC	32kHz	32MHz	32 bits	32kB	768
TI MSP430F5438A	32kHz	25MHz	16 bits	16kB	1068

MKL16Z256 microcontroller uses the least power in run mode. This is because it is based on the Cortex-M0+ core, which is by design slightly more power efficient than the Cortex-M3 core the other devices are based on. There is a trade-off though, as it also supports fewer interrupt vectors and does not have hardware divide capability, which could have a significant performance impact, depending on the code being run. The NXP LPC1519 microcontroller on the other hand offers the most processing power of these devices, along with slightly more RAM and decent energy efficiency. The ST STM32L151RC offers a good upgrade path with up to 80kB RAM available within the STM32L1 product family. If necessary, swapping to STM32L4 Cortex-M4F range of microcontrollers is also possible. That would provide 128kB RAM, an FPU (Floating Point Unit) and 80MHz clock frequency.

Table 9. Cortex-M MPU comparison

	MKL16Z256	LPC1519	STM32L151RC	EFM32LG
Processing power	+	++		+
Memory		+		
Upgrade options	+		++	
Downgrade options				
Toolchain support				
RTOS support			+	
Development kits				
Power consumption	++			
Total	++++	+++	+++	+

Considering the current needs versus future requirements, the ST STM32L151RC was chosen as the microcontroller for the wrist device. The difference in power consumption between the Freescale MKL16Z256 and ST STM32L151RC, while noticeable, was not considered as significant as the upgrade path, and the better RTOS support for the ST microcontroller.

4.6. Hardware design summary

Several options were reviewed for each major hardware component from the point of view of the software implementation and putt recognition. Based on the findings of the review, one option was selected for the actual implementation in each case.

Three-axis accelerometer ST LIS3DSH was chosen for motion sensing purposes due to its internal state machine system. Using only an accelerometer does not allow for full 6DOF motion tracking, but it was assumed that recognizing the putt motion would be possible. Not using a gyroscope results in significant power consumption savings.

Panasonic PAN1322 module was chosen for wireless communications. Bluetooth Classic was considered the best option to ensure sufficient datarates.

Microcontrollers based on the ARM Cortex-M0+ and Cortex-M3 cores were considered the most suitable. After comparing the microcontrollers in more detail, ST STM32L151 was chosen as a compromise of its performance specifications and available upgrade options.

With these components it was possible to quickly develop a functional prototype device, while allowing for future changes with minimal impact if necessary.

5. WRIST DEVICE SOFTWARE DESIGN

The software implementation for the device was greatly facilitated by the considerations made during component selection. Design decisions were made with a short implementation time and long term maintainability in mind. The intention was to create a modular system with as few interdependencies as possible. Maximizing battery life was not a high priority in the creation of the first prototype device. Nevertheless, there are significant constraints concerning the amount of processing, memory use and timings.

5.1. System software timing requirements

The timing requirements are primarily derived from possible external events, such as button presses, sensor readings and interrupts generated by integrated peripherals. Table 10 shows the main timing concerns in the case of this wrist device prototype.

Table 10. Timing requirements

Item	Priority	Scale	Explanation
Retrieving an acceleration sample	High	>625μs	Each generated sample must be retrieved before the next one is generated.
Retrieving a byte from the UART peripheral	High	>10.9μs	Each received byte must be read before the next one is received as the peripheral can only buffer a single byte at a time.
Button debouncing	Low	>200ms	Button responsiveness was not a priority in this first prototype device.
LED update	Low	>100ms	Small delays are unnoticeable and should the delay be noticeable, there are likely to be larger problems that need to be solved first.

The primary purpose of the device is to retrieve each sample from the accelerometer as it is generated. This is the highest priority event. Each sample must be processed before the next one is ready, while simultaneously sending the recorded data to the end device over Bluetooth.

If the system fails to do this consistently, it cannot fulfil its intended purpose. The maximum output datarate of the LIS3DSH accelerometer is 1600 samples per second. Therefore, fetching and processing each sample must take place under 1/1600Hz$\approx$625μs to avoid loss of data.

Similarly, for the Bluetooth connection to function reliably, the system must be able to send data faster than it is generated, and to receive each byte as it arrives from the PAN1322 over the UART bus. The UART peripheral of the STM32L151 microcontroller only buffers one byte at a time so the bit rate of the bus determines the minimum response and handling time for each byte.

In Chapter 4.2 it was estimated that an average transmission rate of 80kbit/s would be required. To achieve sufficient throughput with the PAN1322 Bluetooth module a minimum UART baud rate of 921600 is required [44]. The bus was configured in 8N1 (8 data bits and 1 stop bit) mode. In addition, each byte is preceded by a start bit.

Thus, for each byte of data to be transmitted 10 bits need to be transmitted. In the case of the UART bus, baud rate is the same as bit rate, so the number of bytes per second the system needs to be able to process is $\frac{921600b/s}{10b/B} = 92160B/s$. Fetching and processing each byte must therefore happen in under $\frac{1}{92160B/s} \approx 10{,}9\mu s$.

The UI related requirements were not a high priority. Any small variations in LED update timings are not likely to be visible to the human eye. Should the variations be noticeable, there would probably be bigger issues to fix as well. Button debouncing, however, is necessary. Debouncing always results in at least a short delay. In this case the solution was a long debouncing delay. Thus, a long press of the button is required.

5.2. RTOS

Approaches to designing and implementing an embedded real time system range from a simple infinite loop, known as superloop, to versatile operating systems. The choices in between include lightweight Real Time Operating Systems such as for example FreeRTOS, CoOS and Keil RTX. Each approach has some advantages and disadvantages, some of which are shown in Table 11.

Table 11. Superloop vs RTOS comparison

	Superloop	RTOS
Dependency on 3rd party software	None	Dependent on the RTOS software so the licensing terms may matter
Processing overhead	None	Task switching takes a little processing time
Memory overhead	None	RTOS structures require both some ROM and RAM
Timing flexibility	Inflexible. Adding flexibility quickly increases complexity	Flexible. Adjusted dynamically based on priorities
Feature scaling	Poor. Adding features quickly increases complexity	Good. Allows for more modular design and better control over interdependencies
Efficiency	Requires constant polling even if nothing to process	Can wait until there is something to process
Power saving positives	If there is known idle time, can stay in power saving mode for the entire time	Can use an idle task that is automatically entered when nothing else to do
Power saving negatives	Requires careful consideration from the programmer to ensure timing requirements are met	May be woken prematurely due to a tick interrupt even when there is nothing to process

In a superloop each function is performed in succession as long as the program is running. In the case where the timing of each process is fixed and known in advance, and there are few if any asynchronous events, this approach is a quick way to implement the code. It also requires the least computing resources.

Anything that can be done with an RTOS can also be done using a superloop, but as more features are added, and more external events are introduced, the complexity of the code required to handle everything increases. As a result, it becomes difficult to add new features or change the existing ones, while guaranteeing that the timing requirements are met.

An RTOS enables a modular approach for handling both new features and timing requirements. The main downside is the increased memory and processing overhead. Processing overhead comes from the context switches, memory overhead originates from the RTOS code, which adds to the application code, and the RAM required by the RTOS control structures.

On the other hand, timing control is more flexible as tasks can be assigned different priorities, and the scheduler ensures that the highest priority task always gets the processing time it requires. The same priority based processing also makes it easier to add new features requiring processing time, while ensuring that the previous ones do not get starved as a result. This flexibility helps reduce development time and makes maintenance easier.

In this project using an RTOS was considered the best option. It was expected to result in a shorter development time as well as provide a suitable platform for future development.

5.2.1. Features of the chosen RTOS

FreeRTOS, which is a market leading [45] free embedded operating system, was chosen. Table 12 covers some features of FreeRTOS [46]. It was chosen largely due to its attractive license terms. Other factors were its substantial user community and the feature set it provides. For example FreeRTOS' tickless idle support helps with "Power saving negatives" mentioned in Table 11.

The FreeRTOS Open Source License was appealing as it allows use even for commercial applications as long as a web link to the FreeRTOS website is included. Additionally, there is a compatible commercial version called OpenRTOS, if a more conventional commercial license would be needed at a later date. Vendor support is not offered for the product, so unofficial community support was considered valuable.

5.3. Tasks and communication paths

The wrist device software was divided into four main tasks: communications, sensor, UI (User Interface) and control. Figure 13 shows the overall structure of the software. The timer service task and the idle task, which are provided by the FreeRTOS, are not shown. All tasks are event driven, processing each event in turn before the next one, and blocking, when there is nothing to process.

Table 12. FreeRTOS features

Feature	Explanation
Scheduling	Preemptive, cooperative and hybrid configuration options. Round-robin with time slicing.
RAM usage	236 bytes for the scheduler. 76 bytes plus queue storage area for each queue. 64 bytes plus task stack size for each task.
ROM usage	<10kB for the RTOS kernel itself
Context switch overhead	Down to <100 CPU cycles when optimized for speed.
Information passing tools	Task notifications, queues, binary semaphores, counting semaphores, recursive semaphores and mutexes. Mutexes with priority inheritance.
Software timers	Software timers that do not consume processing time unless a timer needs to be serviced.
Stack overflow detection	Two options for detecting stack overflow.
Power saving support	Configurable idle task offers a convenient place to enter sleep when all tasks blocked. Tickless idle support (with demos for select MCUs including the STM32L range).
License & restrictions	Modified GNU GPL. Changes to RTOS code need to be made open source. FreeRTOS may be used freely even for commercial applications as long as use of FreeRTOS is mentioned and a link to the FreeRTOS website is provided.

5.3.1. Sensor task

The sensor task, shown in Figure 14, handles all communications with the accelerometer, as well as the acceleration data buffering. This includes retrieving new samples when they are generated, storing them in a buffer, queueing data to the communication task to send, and processing any control events sent by the control task. Notably, in the case of a buffer overflow sampling is stopped, and any remaining stored samples are transmitted if possible. Regular sampling is only resumed, when the buffer has been emptied again. This is because recording data that cannot be sent serves no purpose.

Communication with the accelerometer occurs over the I2C bus, which is serviced by a simple polling driver. This driver is not designed to be thread safe. If it would be used from more than one task at the same time, data loss or corruption, deadlocks and livelocks, or even a crash might result. A simple solution to this problem is to centralize all communication over the I2C bus into a single gatekeeper task.

Centralizing all actions involving the accelerometer and buffering the acceleration data also makes semantic sense. As the task encapsulates all sensor related processes, it would for example be possible to change to a completely different sensor unit with minimal changes to the code outside the sensor task.

In the future, a separate data processing task may be implemented. This could result in a structure, where the sensor task and the proposed processing task would function in a producer-consumer relationship. Then, the sensor task "produces" and stores data in a buffer, while the processing task "consumes", processes and queues data for transmission. This would allow for retrieving the acceleration samples to remain a lightweight and quick process that is not impacted by the amount of processing on the data. There is little to no actual processing required at the current point of development.

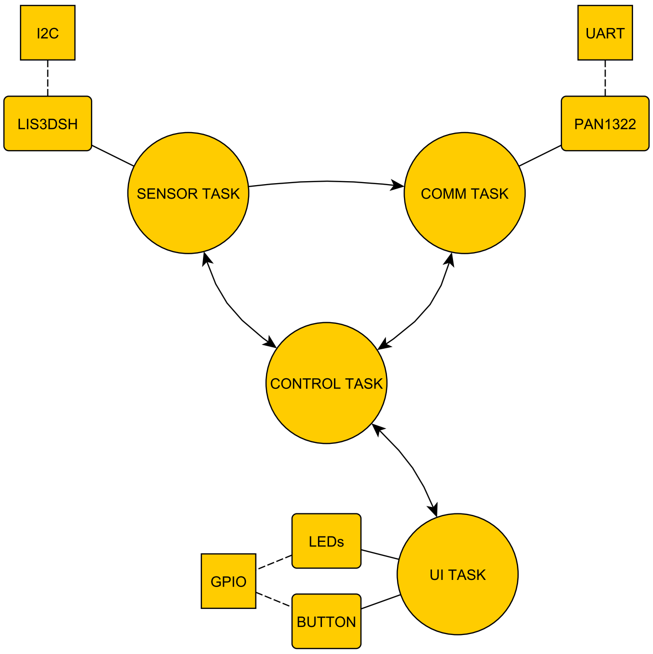

Figure 13. Tasks and communication paths.

5.3.2. Communication task

The communication task, shown in Figure 15, handles all communication with the Bluetooth module. It controls the Bluetooth connection at a low level, notifying the control task of any relevant events. It also forwards any messages coming from the end device to the control task, and handles transmitting the data messages sent by the sensor task, and any other messages sent to the end device.

Similarly to the sensor task, the communication task functions as the only access point to the Bluetooth module, and encapsulates all Bluetooth related actions. This makes it possible to change other parts of the system without affecting the communication task and vice versa. For example should the used Bluetooth module be changed to another one, no software changes would be needed to the sensor task, and at most only minor changes would be needed to the control task.

The communication task could have been split into two separate tasks, i.e. a sender task and a receiver task, for sending and receiving messages to the Bluetooth module. However, the PAN1322 Bluetooth module uses a request-response communication scheme, where most requests can only be sent once a response to the previous one has been received. This means that should sending and receiving be done by separate tasks, careful synchronization between them would be required. Thus, it was decided that a single task would be used to handle both directions of communication.

5.3.3. Control task

The control task, shown in Figure 16, controls the overall state of the device. It receives and processes event notifications from the other tasks, and based on them sends appro-

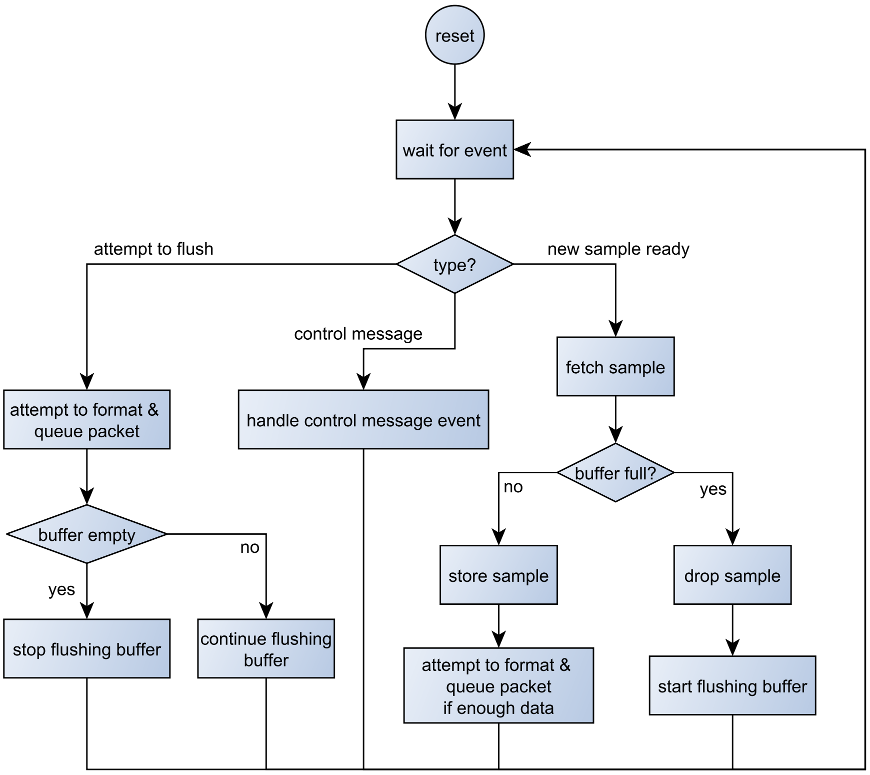

Figure 14. sensor task basic structure.

priate control events to each task. For example, in the event of a Bluetooth connection being disconnected, while the sensor task is sampling data, the control task notifies the sensor task to stop sampling.

It would be possible to implement the system without using such a central control task. This would, however, require each part of the software to be tightly coupled to each other. As a result, any changes to one part could result in any number of changes to the others, leading to maintainability difficulties.

5.3.4. UI task

The UI task is the simplest one. Its only functions are controlling the LEDs to notify the user of the state of the system, and notifying the control task of button presses.

The user interface includes three LEDs, which are used to communicate the device's current state by blinking the LEDs in predetermined patterns. Controlling the blinking itself is handled by means of a software timer offered by FreeRTOS. This makes it possible to turn on and off each LED at a millisecond resolution. The role of the UI task is configuring the timer correctly according to the current state of the system, as notified by the control task. Thus, the task only needs processing time for controlling the LEDs when changing from one blinking pattern to another.

The user interface also has a single push-button, which is used to tell the device to enter Bluetooth pairing mode to allow forming the Bluetooth connection. As men-

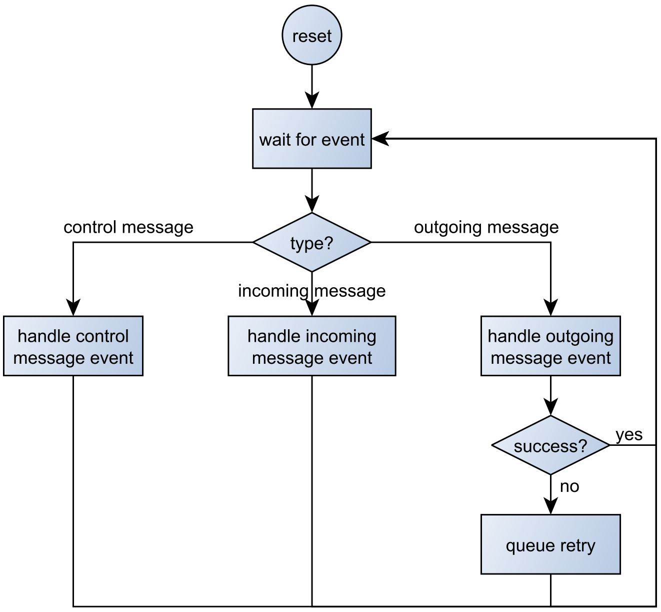

Figure 15. Communication task basic structure.

tioned in Chapter 5.1, the responsiveness of the button is not a priority, which makes it possible to handle debouncing with a very simple algorithm.

5.3.5. Task priorities

The priorities define how the scheduler decides to allocate time for each task. Higher priority tasks are processed first, with lower priority ones running when higher priority tasks are idle. The priority assigned to each task is decided with consideration to meeting the timing requirements presented in Chapter 5.1, and the overall software structure. The priorities are represented by numbers, where a higher value means a higher priority.

The control task is given priority 4, which is the highest one. Since it controls the overall state of the device it must be able to react to each event immediately. For example, if a Bluetooth connection is lost, it must immediately notify the sensor task to stop sampling and the UI task to display the connection loss to the user. Generally, the control task must never prevent the lower priority sensor task from retrieving a generated sample, or the communication task from processing the events it receives from the Bluetooth module. No time consuming processing or polling delays are allowed in the control task.

The sensor task is given priority 3, which is the second highest. Retrieving each acceleration sample before the next one is generated is vital.

The communication task is given priority 2. There is some leniency to the task's timing as the most time critical action, receiving data over the UART bus, is handled by an interrupt driven driver. The driver buffers the received messages independently until the communication task has time to process them.

Finally, the lowest priority value, 1, is given to the UI task.

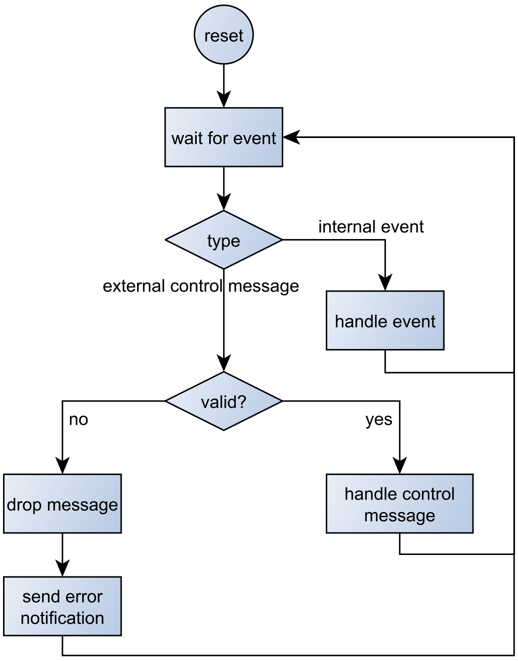

Figure 16. Control task basic structure.

5.4. Device drivers

The software communicates with the hardware using the I2C and UART buses, as well as the GPIO pins. Drivers were implemented as needed for controlling each external hardware module.

5.4.1. LIS3DSH

The I2C bus is used to communicate with the LIS3DSH accelerometer module. A simple polling driver was implemented, facilitating writing and reading one or more bytes at a time. A polling driver was chosen as a concession to a shorter implementation time, as it proved significantly easier to implement than an interrupt or DMA based scheme. Figure 17 shows the basic structure of the driver.

A polling driver is technically the worst option. Processing time and thus energy is wasted instead of staying in sleep mode or processing something else while waiting. Implementing a better driver is a future improvement.

5.4.2. PAN1322

The UART bus is used to communicate with the PAN1322 Bluetooth module. It was implemented using DMA transfer to send requests to the module, and using an interrupt driven driver to receive responses from the module. Figure 17 shows the basic structure of the driver.

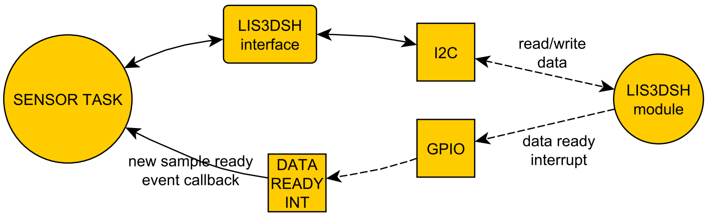

Figure 17. Basic structure of the LIS3DSH driver.

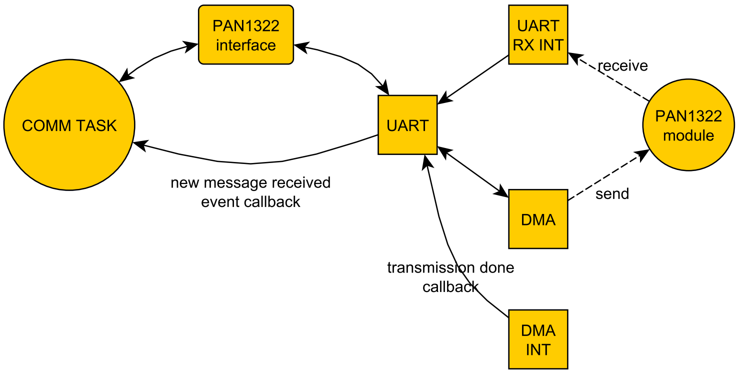

Figure 18. Basic structure of the PAN1322 driver.

DMA transfer is often optimal if any significant amount of data needs to be processed. It only requires processing time when setting up the transfer and when the transfer has concluded. The largest MTU (Maximum Transmission Unit) supported by the PAN1322 module is 500 bytes [44]. If an interrupt driven driver was used, each byte would be handled through interrupts resulting in up to hundreds of individual calls for each message sent. Similarly, sending a single packet of acceleration data over the UART bus to the Bluetooth module takes several milliseconds, which makes using a polling driver impractical.

However, using the DMA for receiving data from the Bluetooth module was not considered practical. The STM32L microcontroller's DMA peripheral requires the number of bytes to be transferred to always be configured before initiating the transfer [47]. This is difficult to do, since the messages can vary in length from four bytes to several hundred bytes. Therefore, receiving messages transmitted by the PAN1322 was implemented as an interrupt driven driver. Each received byte is buffered until the communication task can handle the entire received message.

5.4.3. Miscellaneous

Other hardware connections such as the LEDs and buttons were handled with very basic GPIO functions. Controlling the GPIO pins was very straightforward and no dedicated drivers were needed. Each pin was handled in a manner appropriate for its purpose from where it was needed using the low level driver libraries provided by ST for the STM32L151 series of microcontrollers.

For example, when the accelerometer module finishes generating a new sample, it generates an interrupt and asserts an output pin, which is connected to an input pin of the microcontroller. This input pin is configured to trigger an interrupt when asserted, and from the interrupt handler a callback function alerts the sensor task. This connection is shown in in Figure 17.

Some other pins were simply polled, such as for example the input pin connected to the button, or set to the desired value, as with the output pins controlling the LEDs.

5.5. Software design summary

The aim of the software design was to quickly implement software that could facilitate the initial data logging with the prototype device for algorithm development, as well as function as a starting point for future development. This goal was achieved. However, as time was of the essence, some compromises were made.

After analysing the timing requirements of the system, the software was implemented using the FreeRTOS real time operating system. Using an RTOS facilitated rapid development of the system while meeting all the timing requirements. The task based approach enabled by the RTOS made it possible to implement the software in a modular way, which makes further development relatively easy.

Four tasks were created to handle the different functions in the system. Drivers were also implemented for controlling the I2C and UART buses and for interfacing the accelerometer and Bluetooth modules to their respective tasks. Among the compromises made, the I2C driver which was implemented as a polling driver. Redesigning it as an interrupt driven or DMA controlled driver in the future would result in significant processing time savings and lower power consumption.

6. DEVICE EVALUATION

Reliable data is needed to develop and verify algorithms for swing and hit detection. Data can be recorded from actions performed by human actors. There are, however, certain issues with this approach. For example, if the recorded samples are performed by inexperienced golfers, there may be significant flaws and variability in their technique. As a result it may be difficult to formulate a model that correctly represents the form of a good golf stroke. Someone who can be assumed to have good technique, such as a golf pro, may produce more repeatable results.

Another way to produce repeatable results is to use a mechanical human analogue. In principle, a golf putt is a fairly simple movement that can be imitated by means of a very basic robot that swings the putter. Small variations in the set-up, such as for example the rotation of a wrist device around the wrist of the user, can also be simulated in a controlled fashion.

6.1. Putt simulation set-up

6.1.1. Structure of the human analogue

A simple human analogue shown in Figure 19 capable of swinging a putter and hitting a ball was built. The figure also demonstrates some challenges in creating a general recognition system for different people. The person in the figure is significantly taller than the person used as a model for the simulation system, contributing to the disparity in size between the two.

The system is at its core a rigid pendulum, where the pendulum arm is a triangular frame that approximates human arms and shoulders. The triangle frame swivels around the middle of the horizontal beam, which represents the line between the simulated golfer's shoulders. The diagonal beams the golf club is attached to simulate the golfer's arms. As the frame swivels, the diagonal arms follow approximately the same path as the golfer's arms in a natural putt do, offering a good position for the wrist device to be mounted as the sensor orientation respective to the direction of movement through the motion is the same. As can be seen in Figure 19, the prototype device is located to the side of the simulated pendulum arm, and it is at a diagonal angle relative to the direction of the pendulum motion. The device can also be rotated to different angles around the "wrist" of the human analogue.

6.1.2. Limitations of the set-up

The same set-up is not directly suited to simulating a full swing. Accurately simulating a full swing requires a system with at least two degrees of freedom as for example in the golf swing robot presented in [48]. Besides the rotation of the shoulders present in a putt, the club also rotates with respect to the arms, giving the second degree of freedom that this system does not provide.

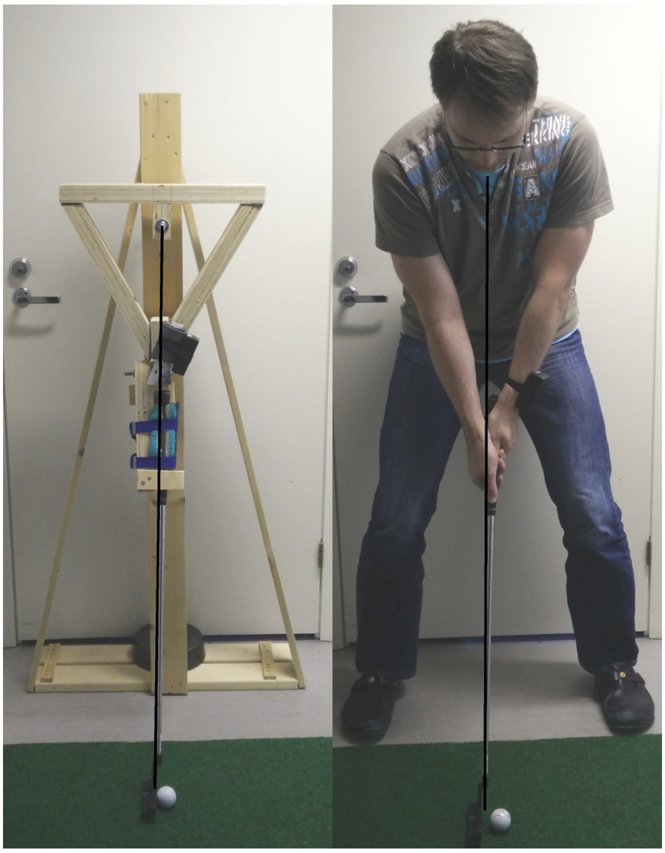

Figure 19. The putting simulation system and a human player in putting address position, with the simulated pendulum arm depicted for both.

It also proved challenging to simulate the human body's capacity to absorb the energy of the club's impact with the ball, which is visible in the unfiltered data as higher-frequency vibrations immediately following the impact.

Despite these shortcomings the simulation set-up was not developed further as the focus of the work is on the wrist device rather than the putting simulation. The simulation system did produce usable data.

6.2. Description of measured signals

For algorithm development purposes putts of three meters were measured on an artificial putting green, both simulated and from an experienced amateur player. Both normal putts and practice swings were recorded, to see whether discerning between the two was feasible. Following recording the acceleration signals were analysed for algorithm development. Some of the features that were found are described here.

6.2.1. Phases of the stroke

Golf swings including putts can be split into five distinct phases called the address, back-swing, down-swing, impact, and follow-through. Figure 20 shows the phases on an acceleration signal of a putt performed by a human measured at the wrist of the leading arm. Figure 21 shows the same for a simulated putt.

The swing starts at the address position, between (1) and (2) in Figure 20, with the player standing next to the ball with the head of the club held at rest behind the ball. Following the address, during the back-swing between (2) and (3), the club is swung away from the ball. After the back-swing the club is swung down, shown between (3) and (4) in the figure. After the impact point at (4) comes the follow-through. At the end of the follow-through at (5) the stroke has been completed.

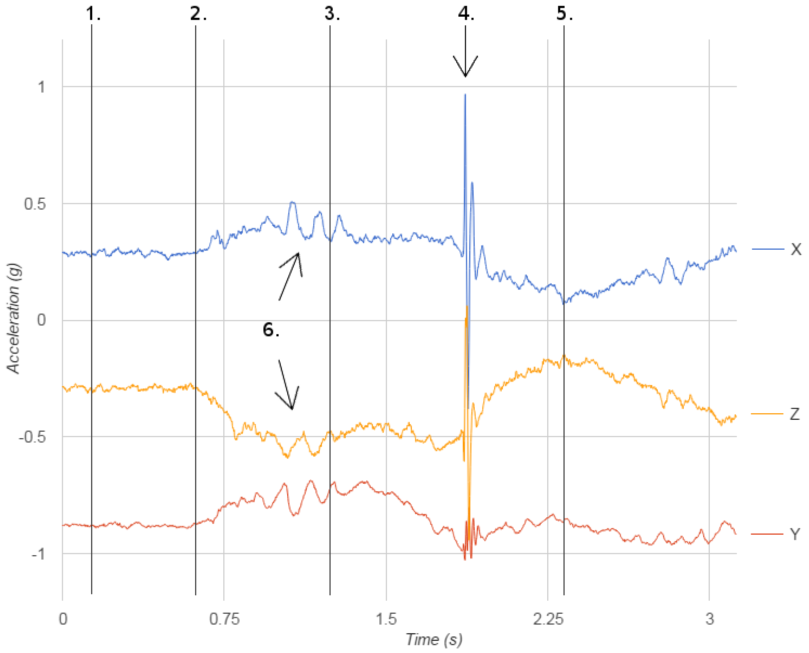

Figure 20. Acceleration at the wrist during a typical natural putt of a moderately experienced player: 1) address, 2) start of back-swing, 3) start of down-swing, 4) ball impact and start of the follow-through, 5) end of the follow-through, 6) shaking that would be absent in an optimal signal.

6.2.2. Visual comparison of natural and simulated putt signals

Comparing the natural putt to the simulated one, some differences become apparent. Firstly, as previously mentioned it is clear that the simulation set-up fails to absorb the energy of the impact as quickly as the human body does. Looking at the natural putt, the vibration caused by the impact vanishes very quickly, while in the simulated signal the peak is not as large and the vibration lasts significantly longer.

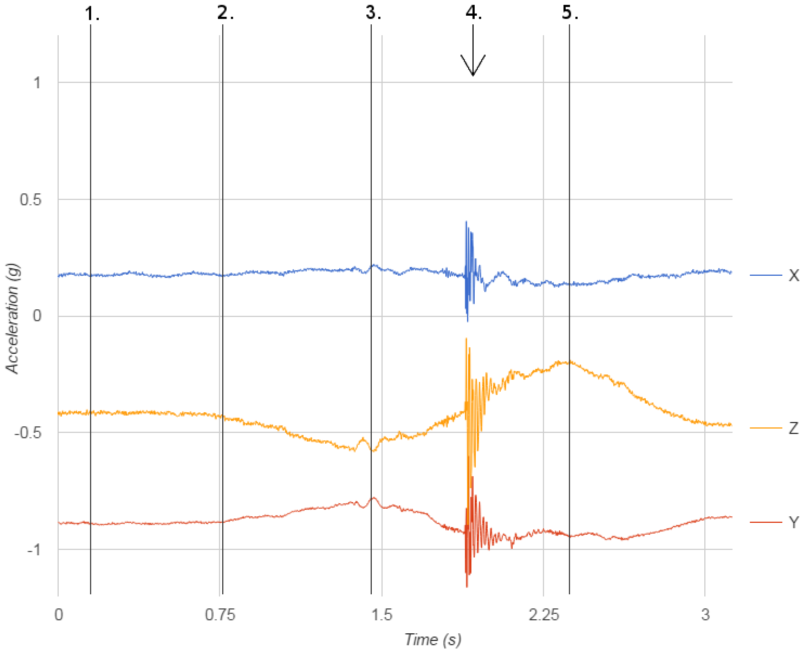

Figure 21. Acceleration at the "wrist" during a typical simulated putt: 1) address, 2) start of back-swing, 3) start of down-swing, 4) ball impact and start of the follow-through, 5) end of the follow-through.

However, at the same time the simulated signal displays fewer low frequency movements, such as the shaking pointed to by (6) in Figure 20, that are sometimes caused by inaccuracies of the human motor control. Similar low frequency vibration is visible on the simulated putt signal at the transition from back-swing to down-swing (3) and during the follow through (between 4 and 5). This is speculated to be caused by the inertia of the club and the effects of it not being mitigated sufficiently quickly for the same reason the previously mentioned high frequency vibration is not immediately absorbed.

6.2.3. Timing comparison

The tempo of the simulated and the natural putt are similar. There are however some differences. The back-swing of the simulated stroke is slightly slower and the down-swing portion slightly faster than those of the natural putt. The down-swing of the simulated putt happens under the power of gravity, with minimal other forces affecting the motion. During the measured natural putt the player appears to actively control the movement of the club, slowing it down. This tempo variation is not considered to be an issue as similar variations in tempo are expected to happen between any two different players as well.

6.2.4. Stable features

Assuming that the sensor is held stationary during address, the average acceleration values for each axis can be used to calculate the direction of earth's gravity and thus to determine the orientation of the wrist device. We can see that the orientation in the simulated situation is not exactly the same as in the natural putt, which indicates that the device was not in the exact same position as when recording the natural putt. This is not considered to be a problem with the set-up, since this will be the case between any two different players as well. As such, any practical swing and hit detection method must be able to function under this limitation regardless.

Another useful thing to note is that even a relatively weak putt of just a few meters has a distinct acceleration spike at the moment of impact. This could perhaps be used to recognize potential putt candidates in the data.

6.2.5. Applicability of the simulated data to algorithm development

The impact of the higher frequency vibration following the moment of impact in the simulated putt can be reduced by filtering it out if necessary. Additionally, the low-frequency shaking apparent in the natural putt could potentially obscure more reliable features that could be used to detect a putt. The simulated putt is largely devoid of these artefacts and thus provides a sufficient approximation of a "perfect putt" for algorithm development.

Visual examination of the signals suggests that the whole stroke from address to the end of the follow-through can be used to detect a putt. However, it may be beneficial to only consider a part of the stroke to minimize the effects of any shaking in the natural putt during the back- and down-swing portions of the putt as well as to minimize the effects of tempo variations.

6.3. Algorithm evaluation

Based on the recorded data, a proprietary algorithm for swing and hit detection was developed. The algorithm development process itself is outside the scope of this thesis, but following the algorithm development a preliminary evaluation was carried out. A separate set of putts by another player was recorded for this evaluation. The results for the recorded test data are presented here.

6.3.1. Data collection

Three sets of 10 3.0m long putts were recorded using the simulation set-up. Between each set, the device was rotated by a few degrees around the "wrist", to simulate potential variations in real humans' way of wearing a device similar to a wrist watch. This was expected to not significantly lower detection rates as it was also considered during algorithm development.

In addition, three sets of 20 putts were recorded as performed by an amateur golfer who had received little formal coaching for putting. Each set consisted of putts from a different distance, with the first set consisting of 3.0m long putts, the second of 1.0m long putts and the last one of 0.5m long putts. For each set the subject was given 20 balls, and instructed to attempt to hole them at their own pace. They were also instructed to perform some practice swings without ball or ground contact in each set, but otherwise their movements were not restricted.

Finally, the developed algorithms included a simple method to personalise certain parameters based on a single sample putt. For this purpose, the subject was instructed to perform another normal 3.0m long putt, which was recorded. The subject was also notified that if he felt the putt had failed, he was allowed to repeat it a few times until he was satisfied with it.

6.3.2. Putt detection

For evaluating putt detection, algorithm outputs were categorized into true positive and false positive results. For a detection to be considered a true positive, it had to be verified to be a true detection of a putt swing with a ball contact. Any other detections were considered false positives, including practice swings if any happened to be detected as putts.

Detection accuracy was evaluated by calculating the detection rate and false positive rate. Detection rate is calculated as $DR = \frac{N_{TP}}{N_P}$, and false positive rate as $FPR = \frac{N_{FP}}{N_P + N_{FP}}$. N$_{TP}$ denotes the number of true positive results, N$_{FP}$ denotes the number of false positive results and N$_P$ the number of actually performed putts.

The detection results for the recorded datasets are presented in Table 13. All three sets of experiments are included.

Table 13. Putt detection results

Distance	N$_P$	N$_{TP}$	N$_{FP}$	DR(%)	FPR(%)
Simulated putts, general parameters					
3.0m	30	30	1	100.0	3.2
Amateur golfer, general parameters					
0.5m	20	13	0	65.0	0
1.0m	20	20	1	100.0	4.8
3.0m	20	16	2	80.0	9.1
Total	60	49	3	81.7	4.8
Amateur golfer, personalised parameters					
0.5m	20	17	0	85.0	0
1.0m	20	20	1	100.0	4.8
3.0m	20	18	2	90.0	9.1
Total	60	55	3	91.7	4.8

All of the putts done with the simulation set-up were detected. This was to be expected as the general parameters used were based on a simulated putt. However, interestingly a single false positive was also found. On closer inspection it became apparent that this happened during resetting of the set-up between putts, from a random knock on the device combined with the pendulum arm happening to swing with the correct timing. It remains to be seen whether events similar to this will happen with any frequency during normal play.

For the putts performed by the test subject the general parameters were used first, resulting in a total detection rate of 81.7% for 60 putts and 4.8% false positive rate. The shortest putts seemed to be the hardest to detect, and it was assumed that they were falsely categorized as practice swings. Unexpectedly all of the 1.0m long putts were recognized however, while the detection rate for the 3.0m long putts was only 80.0%.

Using the personalized parameters for the same data improved the results. The total detection rate increased to 91.7%, with particularly the shortest putts' detection rate increasing from 65.0% to 85.0%. The false positive rate did not change compared to the general parameters, and in fact for both data sets the same events triggered the false positive recognitions.

This system faces the same challenges discovered in [31]. The variations in the techniques of different players can reduce detection rates significantly. Besides the simulation system, only a single subject was tested here, so the results are most comparable to the detection rate of 83% found in [31], when the subjects with the worst detection rates were excluded. Considering measurements in [31] were done at the head of the club and using both a gyroscopes and an accelerometer, the detection rates achieved here are promising. The false positive rate of 2.4% in [31] is smaller than the 4.8% found here. However, more testing is required to determine whether these results will hold in general.

6.4. Device evaluation conclusions

The developed device was used to record data for algorithm development and evaluation purposes. As a first step, the goal was to be able to automatically detect the putts rather than to generate technique feedback.

Simulated and natural putts were visually compared for potentially useful features for putt detection. These observations were leveraged during algorithm development. The algorithm and the details of its development are not covered here. However, the developed algorithm was evaluated for the purposes of this thesis based on a small set of putts performed by a mostly self-taught amateur player.

As the data set was not very large and it was recorded at an early stage of development, and in an artificial environment rather than on a golf course during normal play, only preliminary conclusions can be drawn: the detection rate was sufficient, and the false positive rate was low enough, to suggest that putt detection with a wrist device using only a three-axis accelerometer is indeed feasible. The achieved detection rates were comparable to those found in literature.

More testing is required to reliably judge the accuracy of the system. In particular, a larger number of test subjects and a wider range of different kinds of putts is required to continue refining the algorithms. Additionally, these results are only valid for putts. The method's applicability to detecting full swings remains to be shown.

7. SUMMARY

The goal of creating a prototype device for use in the development of an ideal, unintrusive, automated feedback system for golf was reached. The hardware and software were implemented successfully and the device was used to record data for algorithm development. Preliminary evaluations of the developed algorithms indicate that using an accelerometer to track the movement suffices for automatic recognition of putts. However, the device does not allow for accurate 6DOF tracking. In addition, with a wrist device no direct measurements of the club are possible.

The device enabled the evaluation of the chosen components and enables future refinements. The designed software is modular, allowing for easy implementation of additional features. The modularity also makes it possible to change individual hardware components, such as the Bluetooth module and sensor unit, without widespread changes to the software. The chosen hardware components have features that allow for significant energy consumption reductions.

Lines of further research and development towards the ideal device were found. The recorded signals could be analysed in more detail to explore the kinds of feedback that can be generated from the recorded data. The communication between the microcontroller and the Bluetooth and accelerometer modules could be made more efficient to improve battery life. Through preprocessing the system could also be further optimized to minimize the amount of transmitted data, one obvious example of this being data compression. If the minimum required data rate could be made low enough, more energy efficient wireless communication technologies, such as Bluetooth LE or ZigBee, could be employed.

From the application aspect, a logical step would be widening the scope of measurements and detections to all golf strokes. It is assumed that measuring and detecting swings other than putts is possible using the developed method.

8. REFERENCES

[1] Anonymous, USGA Rules and Decisions. Last accessed 23.4.2015. URL: http://www.usga.org/rules/rules-and-decisions.html.

[2] Burchfield R. & Venkatesan S. (2010) A framework for golf training using low-cost inertial sensors. In: Body Sensor Networks (BSN), 2010 International Conference on, IEEE, pp. 267–272.

[3] Anonymous, PGA TOUR Statistics. Last accessed 23.4.2015. URL: http://www.pgatour.com/stats.html.

[4] Anonymous, Swing Jacket. Last accessed 11.5.2015. URL: http://www.swingjacket.com.

[5] Anonymous, Plane Perfect Golf Machine. Last accessed 11.5.2015. URL: http://perfectgolflabs.com/plane-perfect-golf-machines.

[6] Anonymous, Ernest Sports ES14. Last accessed 15.5.2015. URL: http://ernestsports.com/es14.html.

[7] Anonymous, Zelocity PureLaunch Tracker. Last accessed 15.5.2015. URL: http://www.xtremeradar.com/PureLaunch_Tracker.html.

[8] Anonymous, Zelocity PureLaunch Tracker Launch Monitor Operating Instruction Guide Version 1.1.2-1. URL: http://www.xtremeradar.com/PureLaunch_Tracker.html.

[9] Anonymous, 3Bays GSA PRO Golf Swing Analyzer. Last accessed 15.5.2015. URL: http://www.3bayslife.com/gsa/product.php?pid=102i.

[10] Anonymous, 3Bays GSA PUTT Golf Putting Analyzer. Last accessed 15.5.2015. URL: http://www.3bayslife.com/gsa/product.php?pid=103i.

[11] Anonymous, Arccos Golf GPS & Stat Tracking System. Last accessed 15.5.2015. URL: http://arccosgolf.com/.

[12] Anonymous, Zepp Golf Swing Analyzer. Last accessed 25.5.2015. URL: http://www.zepp.com/golf/.

[13] Anonymous, Zepp Golf User Guide. Last updated 30.4.2015. URL: http://www.zepp.com/assets/docs/user_guide_zepp_golf.pdf.

[14] Anonymous, Swingshot Golf Video Cameras. Last accessed 20.5.2015. URL: http://www.swingshot.com/.

[15] Anonymous, T.O.M.I | The Optimal Putting Instructor. Last accessed 15.5.2015. URL: http://tomi.com/.

[16] Mackenzie S.J. & Evans D.B. (2010) Validity and reliability of a new method for measuring putting stroke kinematics using the TOMI®system. Journal of sports sciences 28, pp. 891–899.

[17] Anonymous, Full Swing Golf. Last accessed 20.5.2015. URL: http://www.fullswinggolf.com.

[18] Anonymous, Nintendo Wii U Golf. Last accessed 20.5.2015. URL: http://wiisportsclub.nintendo.com/golf/.

[19] Anonymous, Polhemus company website. Last accessed 20.5.2015. URL: http://polhemus.com/applications/.

[20] Anonymous, Polhemus LIBERTY LATUS brochure, MS055 Rev. URL: http://polhemus.com/_assets/img/LIBERTY_LATUS_brochure.pdf.

[21] Blake A. & Grundy C. (2008) Evaluation of motion capture systems for golf swings: Optical vs. gyroscopic. In: Information Technology Interfaces, 2008. ITI 2008. 30th International Conference on, IEEE, pp. 409–414.

[22] Fujita H. (1997) A decade of mems and its future. In: Micro Electro Mechanical Systems, 1997. MEMS'97, Proceedings, IEEE., Tenth Annual International Workshop on, IEEE, pp. 1–7.

[23] Gardner J.W., Varadan V.K. & Awadelkarim O.O. (2001) Microsensors, MEMS, and smart devices, vol. 1. Wiley Online Library.

[24] Anonymous, Freescale FXTH87 Tire Pressure Monitor Sensor Family website. Last accessed 21.5.2015. URL: http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=FXTH87.

[25] Anonymous, FXTH870x TPMS Family Fact Sheet. URL: http://cache.freescale.com/files/sensors/doc/fact_sheet/FXTH87FS.pdf.

[26] Bernstein J., Cho S., King A., Kourepenis A., Maciel P. & Weinberg M. (1993) A micromachined comb-drive tuning fork rate gyroscope. In: Micro Electro Mechanical Systems, 1993, MEMS'93, Proceedings An Investigation of Micro Structures, Sensors, Actuators, Machines and Systems. IEEE., IEEE, pp. 143–148.

[27] Juneau T., Pisano A. & Smith J.H. (1997) Dual axis operation of a micromachined rate gyroscope. In: Solid State Sensors and Actuators, 1997. TRANSDUCERS'97 Chicago., 1997 International Conference on, vol. 2, IEEE, vol. 2, pp. 883–886.

[28] Zotov S.A., Trusov A.A. & Shkel A.M. (2012) Three-dimensional spherical shell resonator gyroscope fabricated using wafer-scale glassblowing. Microelectromechanical Systems, Journal of 21, pp. 509–510.

[29] Yazdi N., Ayazi F. & Najafi K. (1998) Micromachined inertial sensors. Proceedings of the IEEE 86, pp. 1640–1659.

[30] Jensen U., Kugler P., Dassler F. & Eskofier B. (2011) Sensor-based instant golf putt feedback. Proc. of the IACSS , pp. 49–53.

[31] Jensen U., Schmidt M., Hennig M., Dassler F.A., Jaitner T. & Eskofier B.M. An imu-based mobile system for golf putt analysis. Sports Engineering , pp. 1–11.

[32] Burns A., Greene B.R., McGrath M.J., O'Shea T.J., Kuris B., Ayer S.M., Stroiescu F. & Cionca V. (2010) Shimmer$^{\text{TM}}$–a wireless sensor platform for noninvasive biomedical research. Sensors Journal, IEEE 10, pp. 1527–1534.

[33] Jensen U., Eskofier B. & Dassler F. (2012) Classification of kinematic golf putt data with emphasis on feature selection. In: Pattern Recognition (ICPR), 2012 21st International Conference on, IEEE, pp. 1735–1738.

[34] King K., Yoon S., Perkins N. & Najafi K. (2008) Wireless mems inertial sensor system for golf swing dynamics. Sensors and Actuators A: Physical 141, pp. 619–630.

[35] Kooyman D.J., James D.A. & Rowlands D.D. (2013) A feedback system for the motor learning of skills in golf. Procedia Engineering 60, pp. 226–231.

[36] Delay D., Nougier V., Orliaguet J.P. & Coello Y. (1997) Movement control in golf putting. Human Movement Science 16, pp. 597–619.

[37] Hall P.S. & Hao Y. (2006) Antennas and propagation for body centric communications. In: Antennas and Propagation, 2006. EuCAP 2006. First European Conference on, IEEE, pp. 1–7.

[38] Patel M. & Wang J. (2010) Applications, challenges, and prospective in emerging body area networking technologies. Wireless Communications, IEEE 17, pp. 80–88.

[39] Anonymous, ANT Burst Transfers Application Note, ANT AN04 Rev 2.1. URL: `http://www.thisisant.com/resources/an04-burst-transfers`.

[40] Gomez C., Oller J. & Paradells J. (2012) Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology. Sensors 12, pp. 11734–11753.

[41] Gomez C., Demirkol I. & Paradells J. (2011) Modeling the maximum throughput of bluetooth low energy in an error-prone link. Communications Letters, IEEE 15, pp. 1187–1189.

[42] Anonymous, PAN1322-SPP User's Manual Revision 1.3.

[43] Anonymous, 8/16-bit Atmel XMEGA C3 Microcontoller datasheet (11/2014). URL: `http://www.atmel.com/Images/Atmel-8492-8-and-16-bit-AVR-microcontroller-ATxmega32C3_64C3_128C3_192C3_256C3_datasheet.pdf`.

[44] Anonymous (2013), PAN1322-SPP-AT SW, Version 3.3 Release Note, Revision 1.0 .

[45] Clarke P. (2013), EE Times, News & Analysis, "Android, FreeRTOS top EE Times' 2013 Embedded survey". (Last accessed 23.7.2015). URL: http://www.eetimes.com/document.asp?doc_id=1263083.

[46] Anonymous, Freertos.org website. (last accessed 24.7.2015). URL: http://www.freertos.org/RTOS.html.

[47] Anonymous (2015), STM32L100xx, STM32L151xx, STM32L152xx and STM32L162xx advanced ARM$^{®}$-based 32-bit MCUs, RM0038, Rev 13. URL: http://www.st.com/web/en/resource/technical/document/reference_manual/CD00240193.pdf.

[48] Ming A. & Kajitani M. (2003) A new golf swing robot to simulate human skill–accuracy improvement of swing motion by learning control. Mechatronics 13, pp. 809 – 823. URL: http://www.sciencedirect.com/science/article/pii/S0957415803000047, computational Intelligence in Mechatronic Systems.