CHALMERS

RyttarJalp

Säkerhetsapp för ryttare

Kandidatarbete inom Signaler och system

Lotta Bergbom

Calle Engelbrektsson

Sofia Granberg

Linnéa Streling

Projektnummer SSYX02-15-11
Institutionen för Signaler och system
Chalmers tekniska högskola
Göteborg, Sverige, 2015

Sammanfattning

Ridning är idag en populär sport i Sverige men tyvärr också en av de mest olycksdrabbade. Skador på grund av fall från hästrygg är en av de vanligaste olycksorsakerna inom sporten och är oftast såpass allvarliga att ryttaren behöver uppsöka vård. Nästan 12 900 människor uppsökte år 2012 en akutmottagning på grund av skador inom sporten.

Detta arbete syftar till att öka säkerheten vid ridning genom att med hjälp av en smartmobil-app detektera eventuell olycka och meddela anhörig om så inträffar. Problemställningen yttrar sig i svårigheten att särskilja rörelsemönstret från en olycka med rörelsemönstret från normal ridning.

En stor mängd data samlades in via datainsamlings-appen LogYard. Datan samlades in från normal ridning vid lektioner på ridskola, privata lektioner i hoppning samt från uteritter via rekryterade testryttare. Totalt samlades ungefär 42 timmar data in, varav 15 timmar var från lektioner vid ridskola och hoppning samt resterande 27 timmar från rekryterade testryttare. Data samlades även in vid 17 simulerade olyckor, vilken sedan kunde jämföras mot datan från normal ridning. Den insamlade datan analyserades och användes sedan för att konstruera en olycksdetektions-algoritm. Denna algoritm programmerades för att identifiera och detektera eventuella olyckor och implementerades sedan i appen.

Resultatet blev en fungerande app som författarna anser vara redo för lansering på Google Play. Appen kan med hjälp av smartmobilens sensorer identifiera händelser som påminner om fall från hästrygg. Om dessa detekteras har ryttaren först möjlighet att avbryta ett eventuellt falsklarm, annars larmas efter en minut de angivna nödkontakterna via sms. Sms:et informerar om att ryttaren kan ha råkat ut för en olycka samt vilken ryttarens senast registrerade position var. Appen kan utöver grundfunktionen identifiera gångarterna skritt, trav och galopp samt vid avslutad ridtur informera hur stor andel av varje gångart ridturen innehållit. Total tid som ridturen pågått, karta över turen samt max- och medelhastighet för turen kan också ses.

Algoritmen som söker efter eventuell olycka har testats mot insamlad data ($\approx 32\,\mathrm{h}$) och utfallet visar att algoritmen fungerar väl, vid främst uteritt, med en tillförlitlighet på totalt 87.3 %. Risken för falsklarm förväntas vara i storleksordningen ett falsklarm var tionde timma. I ridhus och vid hoppning är algoritmen mer känslig för falsklarm, men här är förhoppningsvis behovet mindre för denna app.

Appen kan alltså ses som ett hjälpmedel, som ger ryttaren en ökad känsla av säkerhet vid ridning. Vid en eventuell olycka larmas anhörig till ryttaren direkt via smartmobilen, vilket medför snabbare och mer effektiv hjälp. Denna app kommer inte bara ge ökad säkerhet för häst och ryttare, utan även för nära anhöriga till ryttaren som känner en trygghet i att veta att om ryttaren råkar ut för en olycka så finns hjälpmedel som kan ta hand om detta.

Abstract

In Sweden, horseback riding is one of the most common sports according to the number of active equestrians. Unfortunately the sport also has a high number of accidents. Accidents caused by fall from horseback are the most common and are often so difficult that they lead to hospital visits. Almost 12 900 equestrians sought medical attention during the year of 2012 because of accidents related to the sport.

The aim of this study is to improve safety for equestrians, by developing a smartphone app that detects possible accidents and if so, alerts an emergency contact. The main difficulty is to identify movement patterns that distinguish accidents from normal riding.

A large amount of normal riding data was collected using the app LogYard. The data was acquired from riding school lessons, private lessons of show jumping and outdoor riding by volunteers. Overall, 42 hours of data were collected. 17 accidents were simulated and compared to this normal data to be able to create an algorithm that tries to separate these cases.

The result is a fully developed app that is regarded by the authors to be ready for release on Google Play. The app identifies signals that resemble those of accidents, and if these occur the user is first notified and has time to cancel an eventual false alarm. Else, a message is sent to the user's predefined emergency contacts. The message informs that the equestrian may be exposed to an accident and also contains information about the riders current location. The app is also able to identify the most common horse gaits; walk, trot and canter. These are displayed to the user upon completion of the ride. Information regarding total riding time, maximum and average velocity as well as a map with GPS coordinates for the ride is also displayed to the user.

The algorithm that seeks to detect possible accidents was tested towards the majority of data collected during this study (≈ 32 h) and is considered to be functioning well, primarily during outdoor riding, with a detection rate of 87.3 %. The expected outcome of false alarms is in the magnitude of one false alarm occurring every tenth hour. During inside riding or show jumping, the algorithm is more susceptible to false alarms. However, it is presumed that these are cases where the app is less needed.

The app is considered to be of great aid for equestrians by signaling a greater sense of safety during horseback riding. The app provides a quick way to call an alert in a critical situation, which in turn makes the eventual assistance more efficient. The app is also of great value to friends and relatives, by knowing that the rider is better protected and aided by new technology if an accident would occur.

Förord

Denna rapport redogör för ett kandidatarbete utfört under våren 2015 av Lotta Bergbom, Calle Engelbrektsson, Sofia Granberg och Linnéa Streling, vid Institutionen för Signaler och system på Chalmers tekniska högskola. Gruppen bestod av studenter från civilingenjörsprogrammen Teknisk Fysik, Maskinteknik samt Automation och Mekatronik. Arbetet har genomförts i samarbete med Jalp Systems AB. Leif Sandsjö och Stefan Candefjord fungerade som handledare och Bengt Arne Sjöqvist var examinator.

Vi vill rikta ett stort tack till följande personer och organisationer: Stefan Candefjord och Leif Sandsjö för deras engagemang och samarbete, Jalp Systems AB för konsultation och samarbete, Billdals ridskola som besöktes för att studera och samla in data från ridlektioner, Laserdome Göteborg för lån av deras mekaniska tjur, Lena som lät oss studera och samla in data från hopplektioner, samt alla testryttare som samlat in värdefull riddata till arbetet.

Specifika värden samt exakt uppbyggnad av algoritm har valts att ej offentliggöras i denna rapport på grund av sekretess. Detta ska dock inte påverka förståelsen och helhetsintrycket av arbetet.

Ordlista

Android	Ett operativsystem som utvecklas av företaget Android Inc som är ett dotterbolag till Google.
app, mobilapp	Ett enskilt program för en smartmobil.
Appstore	Apples appbutik för iPhone/iPad.
arbetstrav	En variant på trav som används när ryttaren intensivarbetar med hästen, speciellt vid dressyrträning. I vanliga fall rör sig ryttaren upp och ned i takt med hästens tvåtaktiga trav. I arbetstrav däremot sitter ryttaren ned i samtliga takter/steg, vilket gör att ryttaren kan påverka hästens rörelser mer.
bockning	Rörelse där hästen plötsligt sparkar uppåt och bakåt med bakbenen.
fall, olycka	I rapporten avses dessa ord innebära händelser som påminner om fall från hästrygg i form av sensordata.
galopp	Den snabbaste av hästens tre gångarter. Denna gångart är tretaktig och framförs genom att ena hästen först flyttar ett av frambenen, sedan diagonala benparet fram- och bak och till sist det andra bakbenet.
Google Play	Googles appbutik för Android-mobiler.
gångart	Hästens rörelsemönster. De vanligaste är skritt, trav och galopp.
iOS	Operativsystem som utvecklats av Apple för deras iPhone/iPad.
iPhone	Smartmobil som utvecklats av företaget Apple. Använder operativsystemet iOS.
logga	Registrera sensordata från smartmobil.
markarbete	En träningsform för häst och ryttare, där dressyrarbete och lydnad kombineras med lägre hinder.
ponny	Mindre häst med mankhöjd upp till och med 148 cm. Ponny delas upp i fyra kategorier efter sin mankhöjd: A-ponny: Upp till 107.0 cm, B-ponny: 107.1–130.0 cm, C-ponny: 130.1–140.0 cm och D-ponny: 140.1–148.0 cm.
skritt	Hästens långsammaste gångart. I den flyttar hästen ett ben i taget vilket skapar en fyrtakt.
stegring	Rörelse där hästen plötsligt hoppar upp med frambenen i luften.
stor häst	Häst med mankhöjd över 148 cm.
trav	Detta är hästens näst snabbaste gångart och i den flyttar hästen dess diagonala ben samtidigt—höger fram och vänster bak och vice versa, vilket skapar en tvåtakt.
tråd	Programsnutt som schemaläggs i tiden av en processor.

Innehåll

1 Inledning **1**
1.1 Bakgrund . . . 1
1.2 Syfte . . . 2
1.3 Problembeskrivning . . . 2
1.3.1 Datainsamling och konstruktion av algoritm . . . 2
1.3.2 Programmera för smartmobil . . . 3
1.4 Avgränsningar . . . 3
1.5 Ridning och olyckor . . . 3

2 Teori **4**
2.1 Sensorer och sensordata . . . 4
2.2 Frekvensanalys . . . 5
2.3 Apputveckling . . . 6
2.3.1 Appstruktur . . . 6
2.3.2 Verifiering av programkod . . . 7
2.3.3 Versionsstöd . . . 7
2.4 Confusion matrix . . . 7
2.5 Liknande utvecklingsarbete . . . 8
2.5.1 ICEdot . . . 8
2.5.2 *jalp!* . . . 8
2.5.3 RunKeeper . . . 9
2.5.4 *eCall* . . . 9

3 Metod **10**
3.1 Datainsamling . . . 10
3.1.1 Normaldatainsamling vid ridskola . . . 10
3.1.2 Datainsamling från frivilliga testryttare . . . 10
3.1.3 Informationsinsamling via enkät . . . 11
3.1.4 Simuleringar . . . 11
3.1.5 Promenader . . . 12
3.2 Konstruktion av algoritm . . . 12
3.3 Apputveckling . . . 13
3.3.1 Användargränssnitt . . . 13
3.3.2 Appstruktur . . . 13
3.4 Testning . . . 14
3.4.1 Verifiering av programkod . . . 14
3.4.2 Verifiering av algoritm . . . 16
3.4.3 Slutverifiering . . . 16

4 Resultat **16**
4.1 Datainsamling . . . 16
4.1.1 Resultat av enkät . . . 16
4.1.2 Total mängd insamlad data . . . 17
4.1.3 Insamling av verifierad normaldata . . . 18
4.1.4 Simulering av olycka . . . 26
4.2 Algoritm . . . 29
4.2.1 Utformning . . . 29
4.2.2 Utfall . . . 30

4.3 Apputveckling . . . 33
4.3.1 Första utkast av app . . . 33
4.3.2 Realiserad app . . . 36

5 Diskussion **41**
5.1 Datainsamling . . . 41
5.1.1 Enkät . . . 41
5.1.2 Verifierad normaldata . . . 41
5.1.3 Simulerade olyckor . . . 42
5.2 Algoritmen . . . 43
5.3 App . . . 44
5.4 Generell diskussion . . . 44
5.5 Vidareutveckling . . . 45
5.5.1 Ytterligare funktioner . . . 46
5.5.2 Marknadsföring . . . 46

6 Slutsats **47**

Referenser **48**

Bilaga A Enkätfrågor **50**

Bilaga B Enkätsvar **52**

1 Inledning

1.1 Bakgrund

Ridsport är populärt med en halv miljon utövare som är engagerade i någon form i Sverige [1]. Tyvärr är det också en av de mest olycksdrabbade sporterna. År 2012 uppsökte nästan 12 900 människor en akutmottagning på grund av olycksskada i samband med ridning och/eller hästhantering [1]. Olyckorna är ofta svåra att förutse och förebygga då hästen är ett flyktdjur och som vid oanade situationer reagerar utefter det. Under framförallt uteritter kan oförutsedda händelser ske, som kan leda till flykt hos hästen och därmed ge upphov till skador [2], [3].

Den vanligaste typen av olycka är fall från hästrygg [1], [4], där även de mest allvarliga skadorna uppkommer [5]. Sker detta under uteritt finns risk att ryttaren blir liggande skadad och hästen skenar iväg. Om ekipaget dessutom rider ut ensamt kan ryttaren bli liggandes utan förmåga att tillkalla hjälp på egen hand. Ryttare rekommenderas att ha med mobiltelefon vid ridning för att vara kontaktbara/kunna kontakta andra [6], men det kan ändå bli problematiskt om ryttaren exempelvis bryter en arm eller skadar sig svårt. Det är vanligt med olika former av huvudskador som följd [7], [8] av fall från hästrygg, vilket medför att ryttaren löper stor risk att svimma eller bli medvetslös [1]. Genom att i förtid meddela anhöriga om planerad ridtur kan de få en uppfattning om hur lång turen förväntas vara samt var ryttaren kan befinna sig. Om ekipaget inte återvänder i tid och inte heller hör av sig kan en olycka förmodas ha skett. Dock finns ingen information om exakt position vilket försvårar räddningsarbete i form av till exempel ambulans. Även i ridhus kan ryttare befinna sig ensamma och utan någons vetskap. Är olyckan väl framme kan information om var ryttaren befinner sig vara relevant även här.

Via smartmobilers GPS-funktion finns det nuvarande lösningar som gör det möjligt att följa vänners eller anhörigas position, genom till exempel ”Google+” [9] som löpande uppdaterar positionen på en karta. På så sätt kan en olycka misstänkas ha skett om personen i fråga exempelvis är stilla en längre tid. Denna metod är mindre användbar då den kräver ständig övervakning. Risken att anhöriga glömmer att titta på kartan är stor och kan leda till att det tar lång tid innan en olycka upptäcks. Det finns också en lösning som riktar sig direkt till ryttare, ”Horse Rider SOS”, en app som bland annat använder smartmobilens inbyggda GPS för att detektera ryttarens hastighet och position [10]. Om ryttaren inte har rört sig under en viss tid, antar appen att ryttaren kan ha ramlat av och skickar då ett larm. Här kan istället risken för falsklarm förmodas vara stor då appen enbart mäter inaktivitet. Dessa två hjälpmedel kan fungera bra i vissa lägen, men under vissa förutsättningar kan de fungera sämre, till exempel då GPS-mottagningen är dålig. Efter genomsökningar av de stora plattformarna för appar, App Store och Google Play, har det inte påträffas någon lösning för det specifika problemet; att på ett tillförlitligt sätt kunna larma om hjälp vid olycka i samband med ridning.

Tidigare har ett kandidatarbete genomförts vid Chalmers tekniska högskola, där en app togs fram som registrerar rörelse vid cykling [11]. Denna app heter *jalp!* och finns tillgänglig för Android. Appen använder smartmobilens inbyggda sensorer och detekterar en avvikande händelse som kan påvisa en olycka under cykling. Rörelsemönstren kan dock skilja sig mycket mellan cykling respektive ridning vilket gör att denna app inte med säkerhet kan vara tillförlitlig vid ridning.

1.2 Syfte

Syftet med detta arbete är att utveckla en smartmobil-app som kan bidra till ökad säkerhet för ryttare, genom att via smartmobilens sensorer identifiera eventuell olycka och larma anhörig när den inträffar.

Gruppen har också som mål att inom tidsramen för projektet uppnå tillräckliga kunskaper inom programmering och signalbehandling, för att på egen hand kunna konstruera/utveckla appen till en nivå där den ska kunna introduceras på marknad för appar.

1.3 Problembeskrivning

För att uppfylla syftet delas arbetet upp i två problemområden som behöver lösas under arbetets gång. Centrala frågor som utgör grunden för delproblemen är:

- På vilket sätt avviker rörelsemönster vid olyckor/fall från hästrygg från det som normalt förekommer vid ridning?
- Hur kan vi utifrån information om hur rörelsemönstren ser ut, konstruera en algoritm som med hög tillförlitlighet kan detektera händelser som påminner om olyckor?
- Hur utformas appen så att den blir intressant och lockar användare till att använda ett hjälpmedel som detta?

Delproblemen presenteras närmare i nästkommande stycken.

1.3.1 Datainsamling och konstruktion av algoritm

För att kunna identifiera rörelsemönster hos ryttare behöver sensordata från smartmobiler i samband med ridning samlas in och undersökas. En svårighet vid denna process är att få in tillförlitlig och användbar data som går att basera och starta upp arbetet på. Datan som samlas in behöver analyseras för att identifiera vad som skiljer en olycka från normal ridning. Frågor som behöver utredas är:

- Är skillnaden såpass utmärkande att en algoritm går att applicera vid alla typer av ridning?
- Har det någon betydelse var mobilen är placerad på kroppen?
- Har hästens och/eller ryttarens storlek någon betydelse?

En av utmaningarna är att algoritmen inte ska utlösa larm vid rörelser och situationer som liknar ett fall från hästrygg, till exempel vid hoppning över hinder eller vid avsittning. Algoritmen behöver därför vara så pass specifik att den kan skilja på dessa fall. Insamling av data från verkliga olyckor är mindre lämpligt, då ryttare och hästar inte får komma till skada. Därför kommer ett substitut för ekipaget att behöva användas. Problematiken ligger här i att kunna efterlikna verkligheten med avseende på storlek, tyngd samt kroppsliga rörelser vid ett fall, för att kunna erhålla data som påminner om ett riktigt fall.

1.3.2 Programmera för smartmobil

När appen detekterar en olycka ska ett larm utlösas. En frågeställning är vem som ska larmas samt hur många som blir larmade. Även vilken information som ska skickas med i larmet och på vilket sätt larmet når mottagaren är relevant.

En svårighet med appen är att göra den lätt att använda, vilket medför att användargränssnittet blir mycket viktigt. Enkelhet bör eftersträvas, då det under ritt är olämpligt att navigera i telefonen. Appen måste därför kunna startas upp innan ritt och sedan köras i bakgrunden utan att kräva ytterligare hantering under denna period. När ritten är slut ska ryttaren enbart behöva låsa upp smartmobilen för att sedan kunna stänga av appen.

För att ryttare ska använda och ha glädje av appen anses det värdefullt att införa fler funktioner som gör den mer intressant. Här krävs avvägningar mellan vilka funktioner som är relevanta för ryttare och hur många som kan implementeras under arbetet.

1.4 Avgränsningar

För att kunna utveckla en väl genomarbetad app under den relativt korta tid som arbetet pågår, behöver avgränsningar göras.

Appen kommer enbart att utvecklas för operativsystemet Android. Android tillåter en hög grad av frihet för programmeraren som gör att vissa funktioner, såsom att skicka sms per automatik, blir lättare att implementera jämfört med i andra operativsystem. Appen ska i första hand utvecklas för att fungera med avseende på säkerhet. De extra funktioner som gruppen önskat, men inte hunnit implementera innan arbetets slut, presenteras istället under punkten *Vidareutveckling* i Avsnitt 5.5.

En annan avgränsning inom arbetet beror de fall från hästrygg som ska undersökas. Olyckor är svåra att fånga i det ögonblick de inträffar, samt att försöka framkalla fall med verkliga ryttare och hästar är farligt och kommer inte att tillämpas i arbetet. Istället kommer simuleringar som motsvarar verkliga fall från hästrygg i största möjliga utsträckning att utföras.

1.5 Ridning och olyckor

Ridning är som tidigare nämnt en populär sport, men tyvärr också skadedrabbad. I artikeln *Statistik och analys, Fysisk aktivitet och skador* analyseras skadorna från de vanligaste sporterna i Sverige idag [12]. Som ses i Tabell 2 är skador vanligast hos fotboll, utförsåkning, ishockey och ridning i fallande ordning. Trots att ridsporten bara hamnar på en delad tredjeplats i antal skadade utövare per år, bidrar dessa skador till avsevärt störst andel uppsökande av vård och behandling på sjukhus. Antal vårddygn per skada sett på ett år är störst vid ridning utav de fem sporterna ovan. Denna koefficienten räknas ut genom att dividera sportens antal vårddygn med antal skadade utövare.

Huvudskador står för majoriteten av de allvarligare skadorna inom ridsporten [8] och nästan hälften av erfarna ryttare har någon gång drabbats av hjärnskakning [13]. Det är stor spridning mellan ridhjälmars skyddsförmåga [14] och i Sverige är hjärnskakning den tredje vanligaste typen av skada vid fall från hästrygg (12 %) [1]. Vi anser därför att

Rang	Sport	Skadade	Vårddygn	Vårddygn per skada
1	Fotboll	34 000	4 200	0.12
2	Utförsåkning	9 400	3 500	0.37
3	Ishockey	9 200	1 500	0.16
4	Ridning	9 200	8 000	0.87
5	Innebandy	7 400	900	0.12

Tabell 2: *De fem mest skadedrabbade sporterna i Sverige. Skadade och vårddygn anges i antal per år. Koefficienten i sista kolumnen anger antal vårddygn per skada. Källa: [12]*

en säkerhetsapp som kan ge ökad trygghet och hjälpa ryttare att få snabbare hjälp om olyckan är framme, är ett värdefullt komplement till befintlig utrustning såsom hjälm och väst, och borde i framtiden vara en självklar del av den.

2 Teori

I detta avsnitt förklaras de tekniska hjälpmedlen och teorier som användes under utvecklingsarbetet av appen.

2.1 Sensorer och sensordata

I dagens smartmobiler finns som standard flertalet inbyggda sensorer som i takt med den tekniska utvecklingen blir allt mer precisa. För arbetet används en mängd olika sensorer, men en av de mest använda i arbetet är accelerometern. Denna ger data som visar enhetens acceleration a i de tre axlarna i x-, y- och z-led samt belopp. Främst används absolutbeloppet av accelerationen för att göra avläsningen oberoende av smartmobilens orientering och beräknas genom Ekvation (1).

$$|a| = \sqrt{a_x^2 + a_y^2 + a_z^2} \tag{1}$$

Absolutbeloppet är avståndet till origo och är en sammanslagning av variabelns olika delvarabler eller axlar och används för att få en mer enhetlig och sammanställd data som är lättare att analysera. Eftersom smartmobilen inte alltid placeras i samma läge för alla användare under loggning, varierar värdena för x-,y- och z-variablerna. Därför är beloppet mer korrekt; det tar ingen hänsyn till hur smartmobilen är placerad hos användaren under loggning.

En annan sensor är gyroskopet som läser av enhetens vinkelhastighet ω (rad/s) i tre axlar. Beloppet används även här av samma anledning som accelerometern och beräknas genom Ekvation (2). Maximal samplingsfrekvens för både gyroskop och accelerometer varierar mellan enheter.

$$|\omega| = \sqrt{\omega_x^2 + \omega_y^2 + \omega_z^2} \tag{2}$$

Tredje sensorn som används är GPS-sensorn som bestämmer enhetens position och hastighet. Sensorn använder sig av satelliter och noggrannheten bestäms mycket av hur bra täckningen är.

2.2 Frekvensanalys

Vid dataanalys kan en intressant aspekt vara att studera om datan på något vis är periodisk, det vill säga att det finns ett mönster som återkommer med en viss frekvens. För att urskilja ett periodiskt mönster är det vanligt att använda sig av Fouriertransform. Fouriertransformen är ett verktyg som konverterar en signal eller ett antal mätpunkter i tidsrummet till frekvensrummet [15]. Detta görs enligt Ekvation (3) där ω är vinkelfrekvensen $\omega = 2\pi f$, f är frekvensen och $\hat{g}(\omega)$ är fouriertransformen av funktionen $g(t)$.

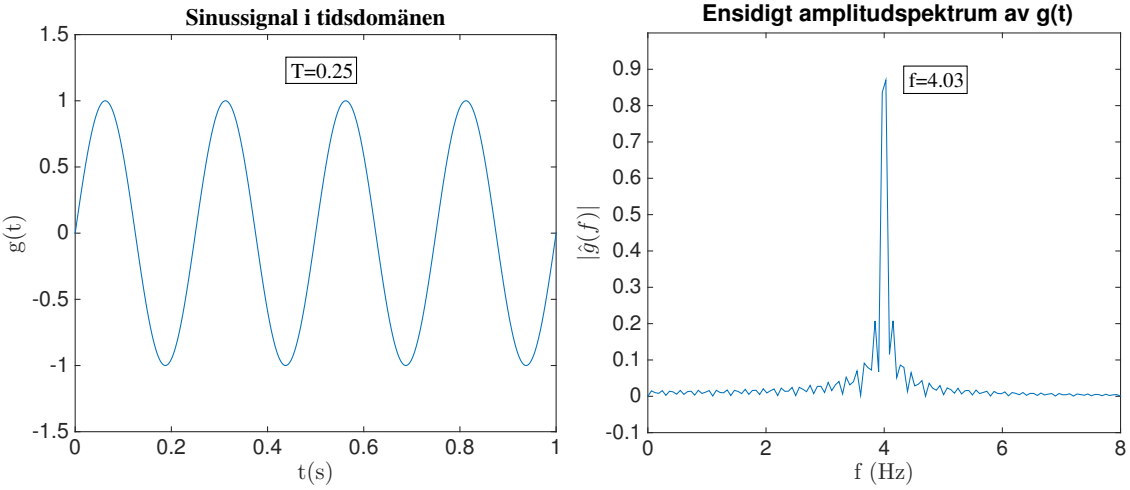
$$\hat{g}(\omega) = \int_{-\infty}^{\infty} g(t)e^{-i\omega t}dt \tag{3}$$

Fouriertransformen är en komplexvärd funktion av frekvens (alternativt vinkelfrekvens). Absolutvärdet av Fouriertransformen talar om hur återkommande en viss frekvens är i signalen, därmed kan ett spektrum plottas över hur förekommande en viss frekvens är. Som ett enkelt exempel kan en sinussignal väljas i tidsdomänen med periodtiden $T = 0.25\,\mathrm{s}$, denna kan ses i Figur 1a. Efter att ha använt fouriertransform på signalen och sedan tagit absolutvärdet på transformen så fås istället Figur 1b som då ger en tydlig topp på frekvensen $f = 4.03\,\mathrm{Hz}$ vilket svarar bra mot periodtiden ($f = \frac{1}{T}$). Att frekvensen inte blir exakt $4.00\,\mathrm{Hz}$ beror på hur många mätpunkter som används vid transformen.

Vid diskret data såsom insamlade mätpunkter (eller som i exemplet med sinussignalen ovan) används den diskreta fouriertransformen, även förkortad DFT. Den diskreta fouriertransformen påminner mycket om den analytiska fouriertransformen och dess definition kan ses i Ekvation (4). Transformen utförs då på en sekvens av N stycken komplexa tal, $x_o, x_1, ..., x_{N-1}$, tagna från tidsrummet och resulterar i en N-periodisk sekvens av komplexa tal, X_k, k är heltal.

$$X_k \equiv \sum_{n=0}^{N-1} x_n e^{-i2\pi kn/N} \tag{4}$$

Vid beräkningar av X_k krävs det N^2 operationer om man utgår från definitionen, vilket kan vara tids- och kapacitetskrävande för datorn. För att kunna göra detta snabbare har algoritmer som ger samma resultat men enbart kräver så lite som $Nlog_2N$ beräkningar skapats. Ett samlingsnamn för dessa är snabb fouriertransform, förkortat FFT från engelskans *Fast Fourier Transform*. I arbetet kommer beräkningar göras med FFT för att urskilja periodicitet i hästens olika gångarter .



(a) *En sinussignal med periodtiden $T = 0.25\,\mathrm{s}$ uppritad i tidsrummet, $g(t)$.*

(b) *Samma sinussignal men nu omvandlad med FFT till en signal i frekvensrummet, $\hat{g}(f)$. På y-axeln ses beloppet av $\hat{g}(f)$ vilket svarar mot hur frekvent en viss frekvens förekommer i tidsdomänen. Den distinkta toppen vid frekvensen $f = 4.03\,\mathrm{Hz}$ stämmer väl överens med den väntade frekvensen $f = \frac{1}{T} = 4.00\,\mathrm{Hz}$.*

Figur 1: *En fouriertransform av en signal från tidsdomänen till frekvensdomänen. I den vänstra figuren syns signalen i tidsdomänen och i den högra figuren syns samma signal i frekvensdomänen efter transform med FFT.*

2.3 Apputveckling

2.3.1 Appstruktur

Android-appar är i utgångsläget singeltrådade. Med det menas att bara en sak utförs åt gången. Standardtråden kallas för ”Main” och det är också på den som användargränssnittet körs [16]. Denna tråd är i regel tillräcklig för de flesta appar, men om intensiva beräkningar görs påverkas prestandan i gränssnittet och det kan sluta med att appen ”hänger sig”. Lösningen är att implementera fler trådar för dessa beräkningar, men ger i sin tur problem med att kommunicera till gränssnittet som alltid körs via ”Main”-tråden.

Android erbjuder flera sätt att hantera kommunikationen mellan trådar, exempelvis via ”Handlers”. Handlers rekommenderas av Android Developers [17] på grund av dess prestanda men kan ge svårigheter beroende på appens struktur då den bland annat måste skickas med och vara känd i alla led. En mer optimal lösning för denna app anses vara att kommunicera via ”Local Broadcasts”. Genom att hos mottagartråden sätta upp en ”Broadcast Receiver” som lyssnar efter en viss fördefinerad sträng, tar den sedan emot de meddelanden som sänds via samma sträng från en sändare i en annan tråd. Sändningen sker lokalt inom appens process och denna metod anses även den vara effektiv med avseende på prestanda [18].

En app pausas som standard av Android-systemet när användaren antingen byter till en annan app men också när skärmen släcks. För att kringgå detta beteende och kunna köra

appen även om den inte är aktiv på skärmen krävs att en ”Service” startas. Den fungerar i praktiken som en app fast utan användargränssnitt [19]. När servicen väl startats körs den i bakgrunden tills det att den stoppas från appen eller tills Android-systemet behöver frigöra resurser för andra appar. För att öka prioriteten ytterligare i Android-systemet kan servicen defineras som ”Foreground”. Kravet är då att en notifikation måste visas för användaren som meddelar att servicen körs.

2.3.2 Verifiering av programkod

”Logcat” är ett inbyggt verktyg för felsökning i Android. Genom att definera små text-meddelanden i programkoden vidarebefodras dessa till Logcat när programsnutten som innehåller meddelandet körs [20]. Programmeraren kan då enkelt se i loggen vid vilka tidpunkter saker inträffar.

2.3.3 Versionsstöd

Android utvecklas löpande och för varje ny version tillkommer ny funktionalitet som blir tillgänglig att använda för appar. Den är dock inte bakåtkompatibel i större utsträckning med tidigare versioner av Android. Enklast är därför att programmera för den senaste versionen. Ett problem med det kan dock vara att det tar tid för nya versioner att nå slutanvändares mobiler och det är inte säkert att alla har uppdaterat till den senaste versionen. Vissa äldre telefonmodeller klarar inte heller att uppdateras till de nyare versionerna.

2.4 Confusion matrix

En ”Confusion matrix” kan visualisera utfallet av en klassificeringsalgoritm och gör det enkelt att jämföra hur parameterförändringar påverkar resultatet. En förklarande bild ses i Figur 2. Algoritmen klassificerar här om data är positiv eller negativ. Utfallet kan ses i kolumnerna jämfört med om datan faktiskt är positiv eller negativ (raderna). Önskvärt är att så många testfall som möjligt hamnar i den gröna diagonalen och så få som möjligt i de röda fälten. Andelen inuti fälten visar andelen av det totala antalet data som hamnar i respektive fält (summan av alla fält = 100 %). $tot_{F,x}$ visar totala antalet faktisk data av varje typ och andelen som algoritmen klassificerar korrekt ses nedanför. $tot_{U,x}$ visar totala antalet data som algoritmen klassificerar till respektive typ medan andelen visar hur stor del som faktiskt är rätt klassificerad av dessa. Slutligen visas summan av alla positiva och negativa testfall och hur stor andel som klassificeras korrekt.

		Utfall		
		positiv	**negativ**	
Faktisk	**positiv**	sann positiv %	falsk negativ %	$tot_{F,p}$ sann %
	negativ	falsk positiv %	sann negativ %	$tot_{F,n}$ sann %
		$tot_{U,p}$ sann %	$tot_{U,n}$ sann %	$tot_{F,p} + tot_{F,n}$ sann %

Figur 2: *Exempel på Confusion matrix som visualiserar utfallet av en klassificeringsalgoritm. Här eftersträvas att så många testfall som möjligt hamnar i den gröna diagonalen*

2.5 Liknande utvecklingsarbete

Idag finns många appar på marknaden baserade på liknande teknik som den arbetet ämnar till att utveckla. Användningen av sensordata i smartmobilerna är ännu en ny teknik som är under ständig utveckling och går snabbt framåt. Det finns lösningar som utnyttjar dessa sensorer på varierande sätt för att redovisa intressanta data för användaren, se exempel på sådana nedan.

2.5.1 ICEdot

ICE står för In Case of Emergency och med denna produkt får användaren hjälp vid olycka eller skada. Den fungerar vid både ridning och andra sportaktiviteter. Det är en liten produkt [21] som fästes på ridhjälmen eller annan hjälm och som känner av rörelse, det vill säga acceleration och rotation i x-,y- och z-riktning. Vid detekterad olycka varnar produkten om detta via en installerad app i din smartmobil. Då har ryttaren/användaren möjlighet att stänga av larmet, annars skickas ett larm till förvalda nödkontakter med information om position och att en olycka har skett.

Produkten är relativt dyr, cirka 1500 kr, samt att den kräver både en app och en produkt som ständigt behöver bäras.

2.5.2 *jalp!*

jalp! är en app som används av cyklister för att detektera fall från cykelsadel. Den utvecklades genom ett kandidatarbete vid Chalmers Tekniska Högskola under våren 2013 [11]. Appen *jalp!* känner av cyklistens rörelse med hjälp av smartmobilens inbyggda sensorer. Genom att en kombination av olika mätparametrar överskrider bestämda tröskelvärden detekterar appen en olycka. Efter detekteringen larmar appen till på förhand valda anhöriga med information om att en olycka troligen har skett. Vid falsklarm har användaren möjlighet att stänga av appen innan ett larm går till anhöriga.

Appen är anpassad för cyklister och därmed fall som kan tänkas ske från en cykel. Vid cykling får användaren andra värden på parametrar (acceleration, gyro, hastighet) än vad en ryttare kan tänkas få vid ridning och fall från hästrygg. Även fast rörelserna kan påminna om varandra till viss del, är de inte helt identiska. Det medför att appen inte är tillförlitlig vid ridning.

Under en konferens som hölls i Göteborg under 2014, där bland annat *jalp!* var representerat, skrevs en artikel om huruvida funktionaliteten likt *eCall*:s kunde implementeras för cyklister och inte enbart för förare av motorfordon [22]. Där diskuterades vilka möjligheter som fanns för att enbart utnyttja dagens moderna smartmobiler för att uppnå samma resultat. Eftersom smartmobiler idag oftast innehåller både accelerometer och gyroskop samt eftersom de används i så stor utsträckning, borde detta vara möjligt.

2.5.3 RunKeeper

En välkänd app med över 26 miljoner användare är ”RunKeeper”. Appen loggar data under träningsaktiviteter såsom löpning, promenad eller cykling [23]. Dock finns ridning inte med som ett träningsalternativ appen fungerar fullständigt för, men kan ändå användas till att samla data om hur långt man ridit och vilken runda som utförts. Det bör dock poängteras att det inte finns någon inbyggd säkerhetsfunktion eller algoritm som detekterar olycka, utan appen registrerar enbart insamlad data genom att bland annat visa erhållen sträcka, tid och position.

2.5.4 *eCall*

Årligen skadas och omkommer tusentals förare av motordrivna fordon i trafiken. Detta sker inte enbart på trafikerade motorvägar där andra bilar ständigt passerar utan även på mindre vägar, där det kan ta lång tid för en skadad eller omkommen förare att få hjälp [24].

eCall är en säkerhetstjänst som skickar ut ett automatiskt meddelande till larmcentral vid detekterad olycka i motorfordon. Tjänsten känner av vissa parametrar i fordonet och antar utifrån dem om en olycka skett. Om exempelvis krockkudden utlöses vid körning antas en olycka ha skett och ett meddelande skickas då ut till larmcentral. Meddelandet innehåller bland annat information om när olyckan skett, fordonets geografiska placering och information om fordonet. Tjänsten är alltså kopplad till fordonet och inte till föraren.

eCall kommer bli obligatoriskt i nya typgodkända bilar från och med 2018 [25]. Att automatiskt kunna skicka larm till en larmcentral blir vanligare och kan i framtiden vara intressant även för appen som arbetet ämnar att utveckla.

3 Metod

För att lyckas med målet för arbetet, att utveckla en väl fungerande smartmobil-app för ökad säkerhet och trygghet för ryttare, delades arbetet upp i mindre moment. Det resulterade i fyra delmoment, som kunde utföras parallellt med varandra. Dessa var datainsamling, algoritmer, apputveckling samt testning och momenten presenteras närmare nedan.

3.1 Datainsamling

För att samla in data, som skulle användas till analys och utformning av algoritmen, användes appen ”LogYard”. Appen registrerade position, hastighet, acceleration och rotation hos smartmobilen. Appen är framtagen av Jalp Systems AB och finns tillgänglig på Google Play och på Appstore. För att kunna använda appen krävdes ett användar-ID som gavs ut av handledarna, som också var ansvariga forskare. Gruppmedlemmarna fick användar-ID för sina privata smartmobiler, dels för att själva kunna testa hur datan representeras vid olika rörelser, men också för att enkelt kunna låna ut telefoner till ryttare utan att det skulle krävas ett nytt unikt användar-ID. Utöver dessa delades även unika användar-ID ut till rekryterade testryttare.

3.1.1 Normaldatainsamling vid ridskola

Ett besök på en stor ridskola i Västsverige genomfördes för att samla in normaldata. Med normaldata menas insamlad data som normalt förekommer vid ridning, till skillnad från olycksdata som samlats in vid olyckssimuleringar. Gruppens smartmobiler lånades ut till ridskoleelever, som sedan samlade in data under lektionerna. Nivån på lektionerna varierade, allt från nybörjare (”ridlekis”) till mer avancerade klasser där galopp och hoppning över lägre hinder utfördes.

Fem telefoner fanns tillgängliga för datainsamling via appen LogYard. Varje gruppmedlem rekryterade innan lektionsstart en elev ur varje ridgrupp, genom att fråga om denna ville delta i en undersökning och förklarade förutsättningarna. Sedan placerade sig gruppen på läktaren och studerade lektionen. Anteckningar fördes angående vilken gångart den rekryterade ryttaren höll vid olika tidpunkter. Även avvikelser noterades, som när ryttaren klev av hästen under ritt eller om hästen var orolig. Denna typ av data benämns som *verifierad normaldata*, som syftar till att gruppen har kunskap kring vilken typ av ridning och vilka gångarter som varit aktuella under ridpasset.

3.1.2 Datainsamling från frivilliga testryttare

Ett antal testryttare i gruppens närhet rekryterades parallellt med arbetet och dessa använde LogYard i privat ridning. Dessa fick unika användar-ID som delades ut av handledare, under förutsättning att testryttarna godkände användarvillkoren för studierna. Det var okänt vilka ryttare som hörde ihop med vilka användar-ID av sekretesskäl och för att ryttarna skulle vara anonyma i studien. Testryttarna rekommenderades att kommentera vilken typ av ridning som de genomförde samt eventuella avvikande beteenden som förekommit hos häst eller ryttare.

3.1.3 Informationsinsamling via enkät

En enkätundersökning utarbetades, som kan ses i Bilaga A, med frågor rörande ryttarnas erfarenheter och beteenden vid ridning, var de brukar placera sin eventuella mobiltelefon vid ritt samt vilka extra funktioner som önskades i ett hjälpmedel för ökad säkerhet. Enkäten genomfördes vid besök på ridskola samt i större utsträckning genom insamling via ett Google-formulär som skickades ut till ryttare som gruppen kände.

Resultatet låg bland annat till grund för vilka extra funktioner som ansågs intressanta att implementera i appen. Gruppen beslutade att de tre alternativ av extra funktioner som fick högst procent i enkäten var de som skulle prioriteras högst att inkludera i appen. I mån av tid och kunskap skulle även de tre nästkommande alternativen i storleksordning, alltså nummer fyra till sex, försöka arbetas in. De valbara funktioner som fanns angivna i enkäten hade gruppen på förhand valt då dessa ansågs relevanta och genomförbara för appen.

3.1.4 Simuleringar

För att jämföra normaldatan med olycksdata användes en mekanisk tjur för att simulera fall från hästrygg. Gruppen ansåg att relativt verklighetsbaserade fall kunde uppnås på detta vis, samtidigt som en mjuk matta och skyddande rink omgav tjuren och förhindrade skador. Gruppen besökte ett stort Laserdome-center i Göteborg för att använda deras mekaniska tjur. Det genomfördes medvetet många olika typer av fall för att få tillräckligt mycket data att analysera. Ett fall definieras i rapporten som intervallet mellan då personen på tjuren förlorar balansen fram till personen landar i mattan och datan från dessa simuleringar benämns vidare som falldata.

De fall som simulerades för att efterlikna bockning eller liknande kast från häst skedde oplanerat, genom att låta gruppmedlemmen falla av tjuren när den inte längre kunde hålla sig kvar. Övriga falltyper som simulerades var planerade i den mån att gruppmedlemmarna på förhand bestämt hur de skulle falla av och tjurens rörelse var begränsad. En fullständig tabell över de 17 fallen kan ses i Avsnitt 4.1.4, Tabell 7. För att i efterhand kunna avgöra när fallen ägde rum filmades försöken, vilket förenklade arbetet när datan skulle analyseras. Filmen kunde då i efterhand synkroniseras med datan. Gruppen bedömde att de olika falltyper som testades skulle representera majoriteten av de olika typer av fall från hästrygg som förekommer. Det gjordes även simuleringar som skulle efterlikna uppsittning respektive avsittning, då dessa ansågs utgöra en potentiell risk för falsklarm. I Figur 3 ses en av gruppmedlemmarna sittandes på tjuren medan de andra entusiastiskt styrde kontrollerna.

Figur 3: *Gruppmedlemmarna vid den mekaniska tjuren. Denna användes för att simulera fall och samla in olycksdata som var så verklighetbaserad som möjligt.*

3.1.5 Promenader

Två försökspersoner fick i uppdrag att logga planerade promenader på jämnt underlag. Detta för att i ett senare skede analysera typiskt rörelsemönster vid denna aktivitet. Sju loggade promenader genomfördes på asfalt samt grusväg.

3.2 Konstruktion av algoritm

All data som samlades in analyserades med hjälp av MATLAB. Program för data-analys som utvecklats i samband med LogYard fanns tillgängliga att använda för gruppen. Dessa sammanställde datan och presenterade den på ett sådant sätt att det gick att få en visuell uppfattning om vad som hänt under loggningarna.

Falldatan studerades för att få idéer om vad som utmärkte ett fall från hästrygg, medan normaldatan från ridskolebesöken främst användes för att kunna undersöka olika gångarter för sig. Värdena jämfördes med varandra genom absolutvärden samt frekvensanalys och visualiserades sedan i histogram. Detta gjordes för att sedan sätta gräns- och tröskel-värden som skiljde normala värden mot värden från simulerade olyckor.

Nästa steg var att i MATLAB upprätta en algoritm som kunde skilja ett fall från normaldata och sedan integrera de tröskelvärden som hittats. För att vara tillförlitlig behövde algoritmen baseras på flera villkor, alltså villkor som behövde uppfyllas för att algoritmen skulle detektera en olycka. Dessa villkor utformades kring en del av variablerna som sensordatan gav; acceleration, vinkelhastighet, hastighet samt frekvensanalys och byggde då upp en algoritm som kunde testas.

När en tillförlitlig algoritm utvecklats (se mer om tillförlitligheten i Avsnitt 4.2.2) översattes algoritmen från MATLAB till Java, som är Androids programmeringsspråk. Därefter överfördes algoritmen till appen för vidare testning i en Beta-version.

3.3 Apputveckling

Programmering av appen skedde parallellt med dataanalysen och framtagandet av algoritmen. Användargränssnitt och grundläggande funktioner påbörjades tidigt eftersom de inte var beroende av algoritmen i detta skede. Av högsta prioritet var att få appen fungerande med avseende på olycksdetektering, men ytterligare funktionalitet planerades också in efter preliminära resultat från enkätundersökningen i Avsnitt 4.1.1.

Enligt Googles mätningar [26] använde de flesta mobiler version 4.1 eller senare av Android (fler än 85 %) när arbetet påbörjades. Med anledningar nämnda i Teoriavsnitt 2.3.3 var det därför denna version som beslutades att stödja fullt ut.

3.3.1 Användargränssnitt

Efter att resultaten från enkäten sammanställts började gruppen fundera på hur appen kunde tänkas se ut och fungera. Brainstorming ledde fram till en önskvärd layout på gränssnittet. De extrafunktioner som flest önskat i enkäten inkluderades i förslaget. Eftersom resultaten var ganska spridda på vissa områden, så utnyttjades gruppens egna erfarenheter och idéer för att komma fram till bra förslag i vissa avseenden.

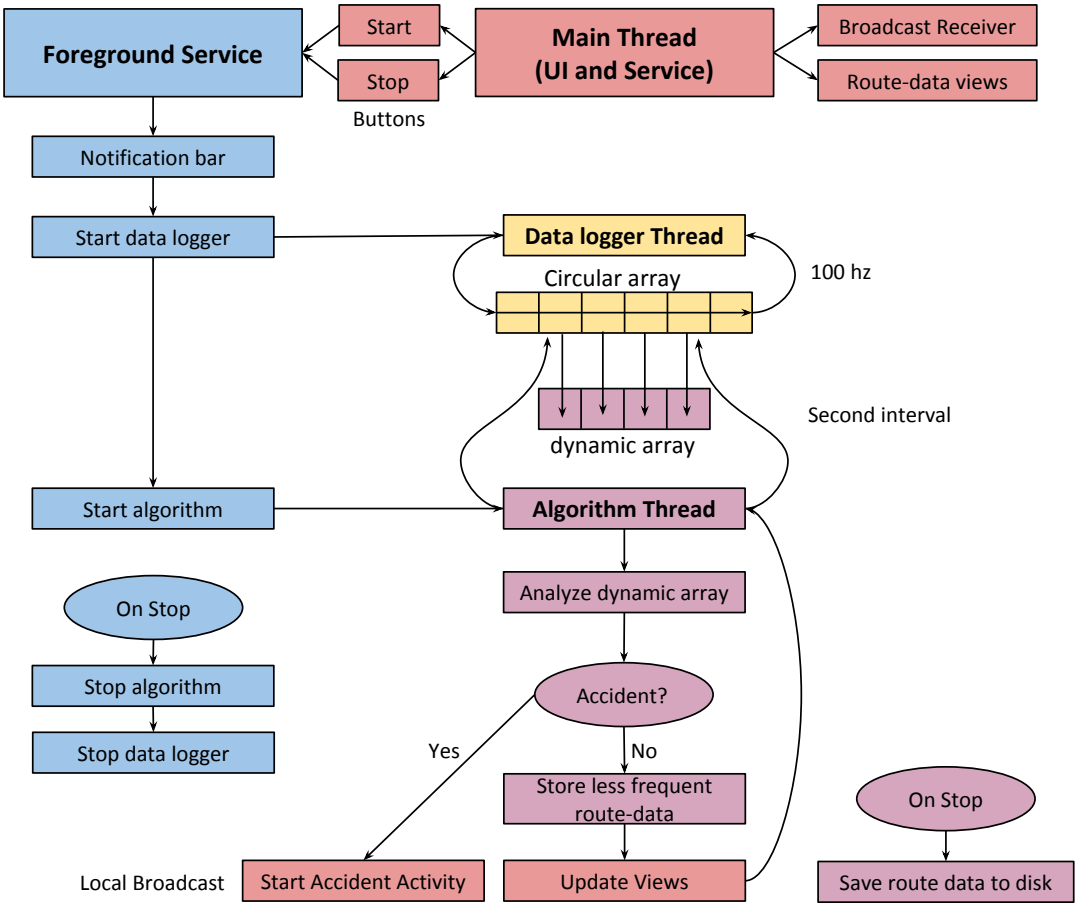
Skisser ritades som efterliknade hur det kunde tänkas se ut i telefonen, hur användaren bör navigera mellan olika fönster samt på vilket sätt start och stopp av loggningen görs. Eftersom appen skulle anpassas till Android-telefoner studerades hur appar för denna typ av smartmobil brukar se ut, för att inte avvika för mycket från ett visst standardutseende som användare är vana vid. Enkelhet var ett ledord för appen och dess användning, därför bör användare lätt kunna navigera i appens layout och gränssnitt.

Hur loggan på appen ser ut har ingen teknisk betydelse men kan motivera användaren till att faktiskt använda appen vid ridning. Därför togs idéer fram under brainstorming för hur den kan tänkas se ut.

3.3.2 Appstruktur

Appen skulle komma att bestå av flera olika programmeringstekniska delar. Ett gränssnitt som ger interaktion till och visar information för användaren, en rutin som läser av smartmobilens sensorer samt en algoritm som tolkar denna sensordata och larmar vid olycka. Ett viktigt moment för fortsatt arbete var att få strukturen på appen bestämd tidigt. Det skulle göra det lättare att se omfattningen av utvecklingen och var prioritet borde läggas.

Utifrån vetskapen om trådar, services och problem som kan uppstå (Teoriavsnitt 2.3.1) gjordes en schematisk bild över appens uppbyggnad enligt Figur 4. Via Main-tråden startas en Foreground Service som i sin tur ansvarar för att driva Data logger-tråden och Algorithm-tråden. Dessa tre trådar körs därmed parallellt. En ”Circular array”, som är en vektor av bestämd storlek som börjar om från start när den är fylld, underhålls av Data-loggern som ser till att senaste sensordata finns tillgänglig. Algorithm-tråden hämtar samtidigt härifrån senaste tillgänglig sensordata som ännu inte har analyserats, till en mindre men dynamisk vektor för analys. Att den är dynamisk (varierande storlek) är för att motverka synkroniseringsproblem mellan trådarna då störningar från andra delar av operativsystemet kan påverka när och i vilken frekvens trådarna körs.



Figur 4: *Schematisk överblick av appstrukturen. Här visas hur alla de trådar, alltså "Main"-tråden, "Data logger"-tråden och "Algorithm"-tråden, interagerar med varandra för att skapa strukturen i appen.*

Algorithm-tråden kommunicerar till gränssnittet via "Local Broadcasts" antingen när en olycka detekteras eller när skärmbilden behöver uppdateras med ny information om ridturen ("Route-data"). Route-data avser här mindre viktig information men som kan vara intressant för användaren såsom andelen av de olika gångarterna, max- och medelhastighet med mera.

3.4 Testning

3.4.1 Verifiering av programkod

Testning av appen och programkoden genomfördes kontinuerligt allt eftersom nya funktioner introducerades. Främst användes Androids inbyggda felsökningsverktyg "Logcat" (Avsnitt 2.3.2) för att bland annat verifiera att de olika trådarna startade, stoppade och synkroniserade med varann. I Figur 5 ges ett exempel på när en loggning startas och cirka fyra sekunder senare stoppas via en knapptryckning i appen. Logcat rapporterade under denna sekvens de förprogrammerade meddelanden som definerats i de områden som exekveras.

I senare skede kunde programkoden testas genom att köra appen och testa funktionerna, såsom att starta/stoppa ridtur, lägga till kontaktpersoner samt att se till att menyer och sidor interagerade som de skulle med varandra. En diagnostikdel implementerades i appen för att lätt kunna se att sensorer och algoritmdelen fungerade som förväntat, ett exempel på detta kan ses i Figur 6. Möjlighet fanns också att köra insamlade filer genom appen och på så sätt ytterligare verifiera att appens delar samverkade på korrekt sätt.

```
03-10 16:08:00.503  31210-31210/com.example.zw4mp.ryttarjalp I/LogService: Received Start Foreground Intent
03-10 16:08:00.508  31210-31235/com.example.zw4mp.ryttarjalp I/Algorithm: started
03-10 16:08:00.511  31210-31210/com.example.zw4mp.ryttarjalp I/MainActivity: Got Start Message
03-10 16:08:00.512  31210-31234/com.example.zw4mp.ryttarjalp I/DataLogger: started
03-10 16:08:01.508  31210-31235/com.example.zw4mp.ryttarjalp I/Algorithm: looping
03-10 16:08:01.511  31210-31235/com.example.zw4mp.ryttarjalp I/Algorithm: Array size: 95
03-10 16:08:02.511  31210-31235/com.example.zw4mp.ryttarjalp I/Algorithm: looping
03-10 16:08:02.512  31210-31235/com.example.zw4mp.ryttarjalp I/Algorithm: Array size: 98
03-10 16:08:04.456  31210-31210/com.example.zw4mp.ryttarjalp I/LogService: Received Stop Foreground Intent
03-10 16:08:04.458  31210-31235/com.example.zw4mp.ryttarjalp I/Algorithm: stopped
03-10 16:08:04.459  31210-31210/com.example.zw4mp.ryttarjalp I/MainActivity: Got Stop Message
03-10 16:08:04.461  31210-31234/com.example.zw4mp.ryttarjalp I/DataLogger: stopped
03-10 16:08:04.610  31210-31237/com.example.zw4mp.ryttarjalp I/AlarmActivity: Did not cancel alarm
```

Figur 5: *Exempel från en verifiering i Logcat under uppbyggnaden av strukturen i appen. Här verifieras att koden fungerar som avsett, bland annat att trådarna startade, stoppade och även interagerade med varandra enligt programkod.*

Figur 6: *Diagnostikvyn i appen. Här kan värden från bland annat accelerometer och gyroskop plottas i realtid. Värden relaterade till algoritmen visas också för att verifiera att data kommer in vid förväntade tidpunkter.*

3.4.2 Verifiering av algoritm

Efter att algoritmen färdigställts testades den i MATLAB mot både normaldata, där algoritmen inte skulle identifiera någon olycka, samt mot olycksdata, där olycka skulle identifieras. Den normaldata som algoritmen testades mot var ridskoledata samt insamlad data från testryttare. Olycksdatan den testades mot var falldata från den mekaniska tjuren. Efter att ha kört igenom filerna utvärderades om resultatet var tillräckligt bra eller om det krävdes förbättringar.

Funktionen i algoritmen för igenkänning av gångarter verifierades också, dock endast mot ridskoledatan. Eftersom den datan loggades under granskning av gruppen, var det den enda data där gångarterna definitivt kunde identifieras. Sedan jämfördes algoritmens resultat för antal sekunder av respektive gångart mot antal sekunder gångarterna faktiskt skett under lektionen.

3.4.3 Slutverifiering

För noggrannare testning av slutresultatet på produkten var tid avsatt i slutet av arbetet för att säkerställa kvaliteten och funktionaliteten. Testningen genomfördes av både gruppmedlemmar och rekryterade testryttare under ridning. Testryttare kunde ladda ned en beta-version av appen via internet till sin telefon och sedan använda den under ridning. Det gjordes för att upptäcka eventuella buggar samt om falsklarm genererades.

4 Resultat

Resultatet delas upp i tre delar: *Datainsamling*, *Algoritm* och *Apputveckling*. Den första delen *Datainsamling* beskriver resultaten från praktiska moment under arbetets gång, medan *Algoritm* beskriver resultatet för algoritmen och *Apputveckling* den realiserade appen. I början av delen *Algoritm* finns en kort sammanfattning av de resultat som ligger till grund för resultatet av algoritmen och appen.

4.1 Datainsamling

4.1.1 Resultat av enkät

Totalt samlades 53 svar in på enkäten, både från insamling på ridskola och via internet. Enkätfrågorna samt det sammanställda resultatet kan ses i Bilaga A respektive Bilaga B.

Nästan tre fjärdedelar av de tillfrågade har ibland känt sig oroliga för att ramla av och skada sig i samband med ridning. Majoriteten av de tillfrågade ramlar av mellan 1–2 gånger per år och upp till 1–2 gånger per halvår.

De tillfrågade fick även svara på hur lång tid de skulle kunna lägga på att starta/stoppa ett hjälpmedel för ökad säkerhet vid ridning. Svaret blev att ungefär hälften tycker det kan ta upp till en minut. Genom att begränsa tiden det får ta att starta/stoppa hjälpmedlet till under 20 sekunder anser 94 % av de tillfrågade att de kan lägga ner den tiden.

Funktion	Röster	Procent
Vilken tur som ridits och hur lång den var	34	64
Tid för olika gångarter under ridturen	27	51
Max- och medelhastighet	25	47
Om/när det är jakttider	17	32
När solen går upp/ned	17	32
Kalender	14	26
Antal kalorier man förbrukat under passet	13	25

Tabell 3: *Resultat från enkäten angående vilka ytterligare funktioner som skulle vara värdefulla eller önskvärda för användaren. Tabellen visar endast de mest populära funktionerna. För fullständig lista över svarsalternativen, se Bilaga B. Första kolumnen siffror visar antal röster alternativet fått och andra kolumnen visar hur många procent rösterna på varje alternativ utgör av totalt antal röster.*

I Tabell 3 visas en sammanställning över de funktioner som de tillfrågade ansåg var mest intressanta. ”Vilken tur man ridit och hur lång den var”, ”Total tid man skrittat, travat och galopperat under ridturen” samt ”Max- och medelhastighet under ridturen” var de tre som hade högst procentandel av svaren.

Enligt de tillfrågade har ungefär 90 % med sig smartmobilen under en ridtur och den vanligaste placeringen av telefonen var i jack- eller byxfickan. Fördelningen mellan Androidanvändare och iPhone-användare är ungefär 30 % Android och 70 % iOS.

4.1.2 Total mängd insamlad data

Total insamlad mängd normaldata kan ses i Tabell 4. Denna avser data insamlad från frivilliga testryttare samt verifierad data där gruppen antecknat hur ritten genomförts. Hos vissa smartmobiler finns problem med att LogYard registrerat data med varierande samplingsfrekvens och ibland även haft längre avbrott mitt i loggningen. Denna data är bortplockad då den inte förväntas ge ett tillförlitligt resultat. Den data som återstår benämns som ”bra kvalitet” och används till att testa algoritmen mot och behandlas i Avsnitt 4.2.

Smartmobil	All normaldata (min)		Bra kvalitet (procent)		Max a (g)
LG Nexus 4	266		100		4
HTC One X	158		100		2
Samsung S2	763		100		2
Samsung S5	295		53		2
iPhone 4	142		100		2
iPhone 5	435		63		8
iPhone 5s	447		21		8
iPhone 6	40		100		8
Summa	2545	(≈ 42 h)	74	(≈ 32 h)	

Tabell 4: *Visar hur mycket av den insamlade normaldatan som varit av bra kvalitet för respektive smartmobil. Hos iPhone 5/5s samt Samsung S5 har data behövts sållas bort på grund av problem med LogYard. Kolumnen längst till höger visar vilken max-acceleration enheten klarar att mäta i varje axel.*

4.1.3 Insamling av verifierad normaldata

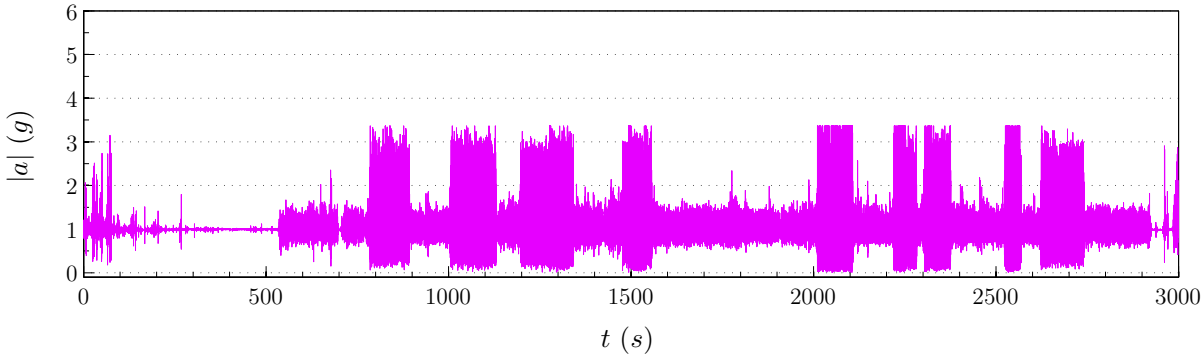
Verifierad normaldata samlades främst in för att kunna studera de olika gångarterna för sig och få en uppfattning av hur de vanligtvis ser ut samt urskilja eventuella skillnader eller likheter mellan dem.

Totalt samlades det in verifierad data från 20 ridpass. Av dessa ridpass är samplingsfrekvensen för opålitlig vid nio av passen. Totalt sett ligger fem av de resterande elva ridpass till grund för analysen av gångarterna. För skritten analyserades även ytterligare ett ridpass. Att inte alla elva användes till analysen beror på att de loggades sent i arbetet. En fullständig lista över hur mycket tid som legat till grund för gångartsanalysen kan ses i Tabell 5.

Hästarna på de ridpass vars gångarter har studerats utfördes på två stycken B-ponnyer samt fyra stycken C-ponnyer. Som en jämförelse mot gångarterna har också data från promenader studerats. Totalt har sju promenadturer samlats in varav en sorterades bort på grund av samma anledning som vid normalridningen; ojämn frekvens med flera långa avbrott under loggningen.

Datasort	Antal ridpass	Antal hästar/ personer	Tid (min)
Alla verifierade ridpass	20	20	894
Verifierade ridpass av bra kvalitet	11	11	503
Verifierade ridpass använda i gångartsanalys	6	6	88
Alla promenadpass	7	3	88
Promenadpass av bra kvalitet	6	2	74

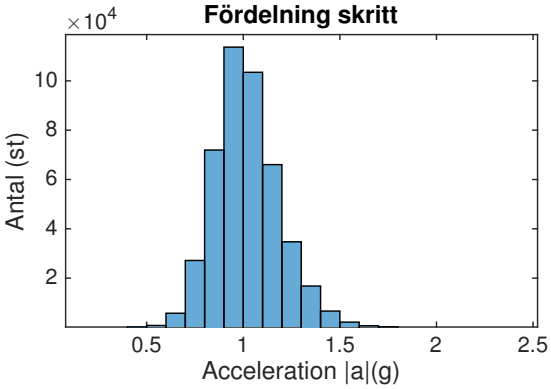
Tabell 5: *En jämförelse mellan total mängd verifierad data som samlats in samt hur stor del av datan som kunnat användas vid gångartsanalysen. Andra kolumnen visar antal utav sort data och tredje kolumnen visar hur många olika hästar och-/eller personer dessa kommit från. Fjärde kolumnen visar total tid insamlad data i minuter.*



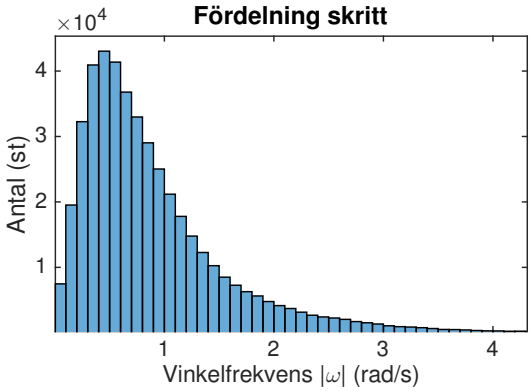
Figur 7: *Beloppet av accelerationen som funktion av tiden under en hel ridlektion. Delarna med högre accelerationstoppar svarar mot trav och galopp medan styckena med lägre accelerationstoppar svarar mot skritt. Skritt kan ses som fält där accelerationen når upp till ca* 1.5 g*, medan fälten med acceleration når upp till ca* 3 g *motsvarar trav eller galopp. Accelerationstopparna i början och slutet av filen hänger ihop med handhavande av telefonen. Mellan cirka* 2000–2600 s *syns det att accelerationen når max av vad sensorn klarar av att mäta.*

Ett klipp från en loggning av verifierad normaldata ses i Figur 7. Här kan det ses att accelerationen varierar i och med olika gångarter. Accelerationen är ungefär lika stor för trav som för galopp medan den för skritt är avsevärt lägre, detta är ett mönster som återkommer i de ritter som studerats. Det syns tydliga accelerationstoppar i samband med start och stopp av loggning, detta uppkommer ofta i samband med hantering av telefonen och ses även i Delavsnitt 4.1.4. De avsnitt med hög acceleration och helt raka toppar visar hur det ser ut då accelerometern har gått i taket. Vissa telefonmodeller loggade inte högre än upp till ett visst värde på acceleration.

För att få en tydligare bild över hur de olika gångarternas acceleration och gyroskoputslag skiljer sig åt gjordes histogram över dessa. En förhoppning var att på så sätt kunna skilja dem åt samt att få en uppfattning om hur dessa värden kan variera under normal ridning. I Figur 8 visas fördelningen av normaldata för absolutaccelerationen respektive absolututslaget på gyroskopet för 4506 s skritt. För absolutaccelerationen syns en tydlig fördelning kring $\approx 1.0 \pm 0.5$ g. I Figur 9 visas motsvarande parametrar tagna från 725 s travdata. Både fördelningen av accelerationen och gyroskoputslaget skiljer sig åt från skritten. Accelerationen sträcker sig över ett längre spektrum och har en tydlig topp vid ≈ 0.4 g men även en mindre höjning vid ≈ 1.75 g. Värden förekommer också upp till 4 g. Gyroskoputslagen är betydligt högre i genomsnitt med en fördelning centrerad kring ≈ 2 rad/s. I Figur 10 visas även motsvarande fördelningar för 92 s galopp. Detta är galopp från flera olika ridpass. Det finns ej lika stor tidsmängd galopp jämfört med skritt och trav då detta är en gångart som inte genomfördes lika ofta under de loggade ridpassen. Både fördelningen över accelerationen samt fördelningen av gyroskoputslaget påminner om motsvarande för trav men är båda förskjutna. Accelerationen har även här en tydlig topp vid ≈ 0.4 g men en mindre höjning vid ≈ 1.25 g. Här förekommer också värden ända upp till 5 g. Slutligen presenteras i Figur 11 datafördelningarna för 4440 sekunder promenad. Utslaget på accelerometern påminner mycket om detsamma för skritt, här med de flesta värden inom intervallet $\approx$ 0.5–1.7 g. För gyroskoputslaget däremot påminner det främst om motsvarande fördelning för galopp.

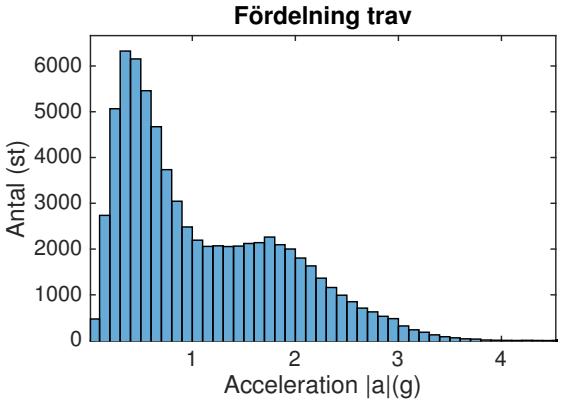


(a) *Fördelningen för absolutaccelerationen vid skritt är tydligt centrerad kring* $\approx 1.0 \pm 0.5\,\mathrm{g}$.

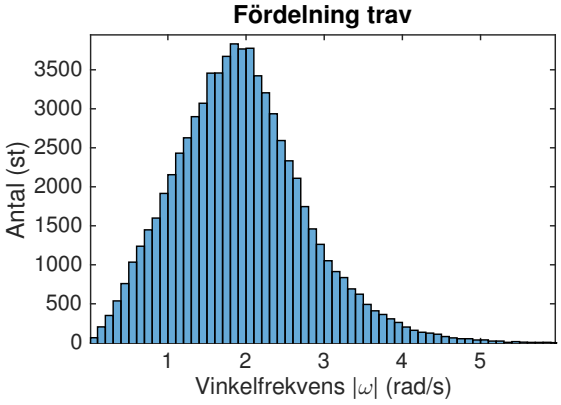


(b) *Fördelningen för absolutbeloppet av gyroskoputslaget är generellt lågt och* $\approx 68\,\%$ *ligger under* $1.0\,\mathrm{rad/s}$.

Figur 8: *Fördelning av acceleration och vinkelhastighet vid skritt.*

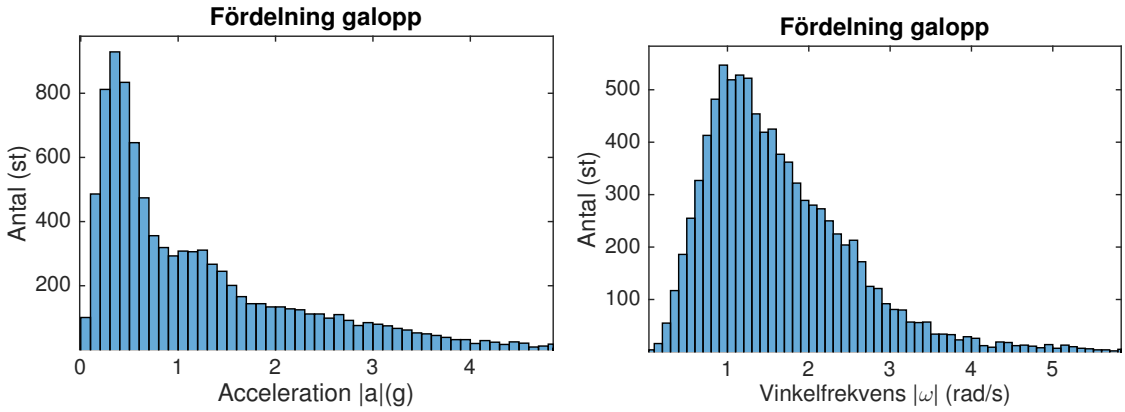


(a) *För accelerationsvärdena insamlade under trav visas en bredare utsträckning än för skritten och med en tydlig topp vid* $\approx 0.4\,\mathrm{g}$ *men även en mindre höjning vid* $\approx 1.75\,\mathrm{g}$.



(b) *Absolutbeloppet av gyroskoputslaget för travdata ligger relativt symmetriskt centrerat kring* $\approx 1.9\,\mathrm{rad/s}$

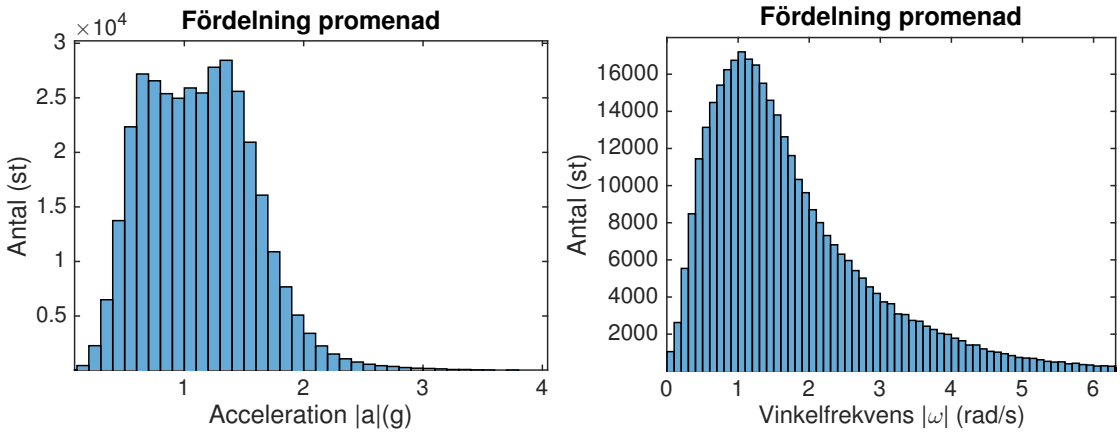
Figur 9: *Fördelning av acceleration och vinkelhastighet vid trav.*



(a) *Fördelningen av accelerationsvärden under galopp påminner till stor del om motsvarande värden för trav. En tydlig topp ses även här vid* $\approx 0.4\,\mathrm{g}$ *men den mindre höjningen vid* $\approx 1.25\,\mathrm{g}$.

(b) *Utslaget på gyroskopet är ett mellanting mellan motsvarande data för skritt respektive trav. Toppen vid* $\approx 1.0\,\mathrm{rad/s}$ *är inte symmetrisk utan förskjuten lite till vänster om fördelningens centrum.*

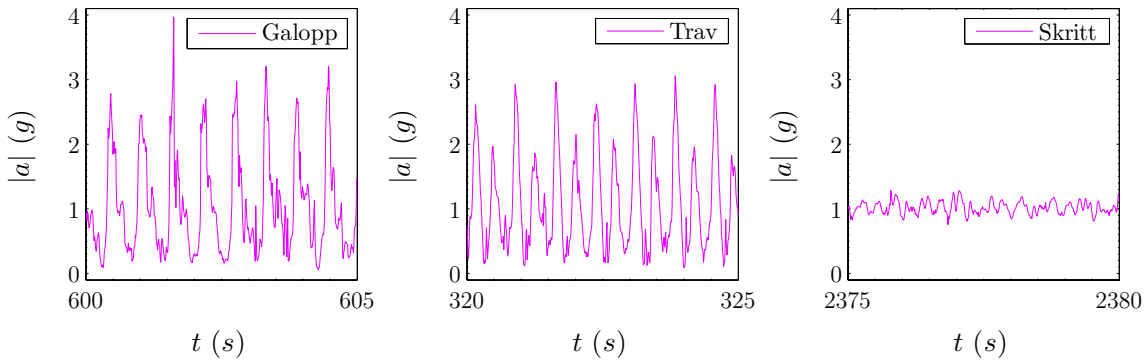
Figur 10: *Fördelning av acceleration och vinkelhastighet vid galopp.*



(a) *Fördelningen hos accelerationen för promenaden påminner om den för skritten men med något större utsträckning. Största andelen ligger inom intervallet* 0.5–$1.7\,\mathrm{g}$.

(b) *Utslaget på gyroskopet påminner till stor del om motsvarande data för galopp. Toppen ligger även här vid* $\approx 1.0\,\mathrm{rad/s}$ *och är förskjuten till vänster jämfört med fördelningens centrum.*

Figur 11: *Fördelning av acceleration och vinkelhastighet vid promenad.*



(a) *För galoppen syns en tydlig periodicitet hos accelerationen. Att galoppen är tretaktig syns i att en stor accelerationstopp snabbt följs av en liten mindre topp innan den sjunker ner nästan till* 0 g.

(b) *Även för traven syns en tydlig periodicitet i accelerationsvärdena. Traven är till skillnad från galoppen tvåtaktig vilket kan ses i att de flesta toppar är relativt jämnstora.*

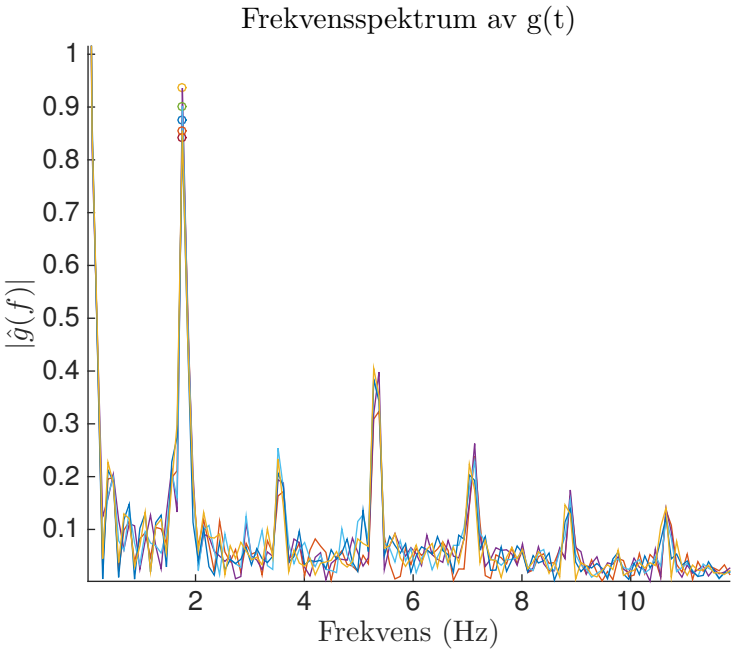
(c) *För skritten noteras ett generellt lägre utslag på accelerometern men ingen tydlig periodicitet som för traven och galoppen.*

Figur 12: *Den absoluta accelerationen som funktion av tiden för hästens tre gångarter galopp, trav och skritt.*

Då kortare snuttar av de olika gångarterna studerades som funktion av tiden gavs en indikation om att gångarterna skulle kunna skiljas åt även på annat sätt än enbart baserat på fördelningar av acceleration och gyroskoputslag. I Figur 12 visas absolutaccelerationen som funktion av tiden för de olika gångarterna skritt, trav respektive galopp. För både traven och galoppen kan en tydlig periodicitet urskiljas, vid en titt med ögat anas det att galoppen har lägre frekvens men för att övertygas om detta analyserades gångarterna med hjälp av FFT i MATLAB.

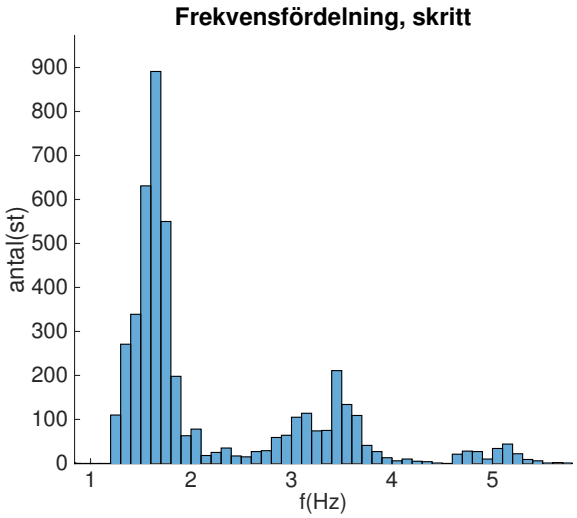
Efter att ha använt FFT på delintervall med längden sju sekunder (dessa sju sekunder uppdaterades till de sju senaste varje sekund) på alla verifierade gångartsfiler kunde frekvenser samt dess förekomst (amplitud) bestämmas. En hög amplitud innebär att frekvensen är vanligt förekommande i det intervall som studerats. FFT kan ibland hitta flera frekvenser men arbetet har fokuserat på den vanligast förekommande frekvensen, den så kallade grundfrekvensen, då det är denna som stämmer överens med den frekvens som kan urskiljas med ögat i Figur 12. Ett exempel på hur värden tagits fram med hjälp av FFT kan ses i Figur 13, cirklarna i figuren motsvarar de värden som sparats och som visualiseras i Figur 14–17. I alla amplitudfördelningar som följer visas histogram över amplituden på den högsta toppen (amplituden hos grundfrekvensen) motsvarande värdet på y-axeln hos cirklarna i Figur 13. I alla frekvensfördelningar som följer visas histogram över de frekvenser som gett upphov till den högsta amplituden (grundfrekvensen) i det ensidiga frekvensspektrumet, motsvarande x-värdet hos cirklarna i Figur 13. För mer information se Avsnitt 2.2.

För skritt var grundfrekvensen mindre framträdande, något som kan ses i Figur 14b där amplituden aldrig nådde högre än till 0.3. Vid analys av skritt med FFT hittades oftast grundfrekvensen vilken ligger inom intervallet $\approx 1.7 \pm 0.3$ Hz men ibland framträdde skrittens övertoner mer änd grundfrekvensen. Detta medförde att övertonerna av FFT misstogs för grundfrekvensen, vilket Figur 14a visar. För traven och galoppen däremot

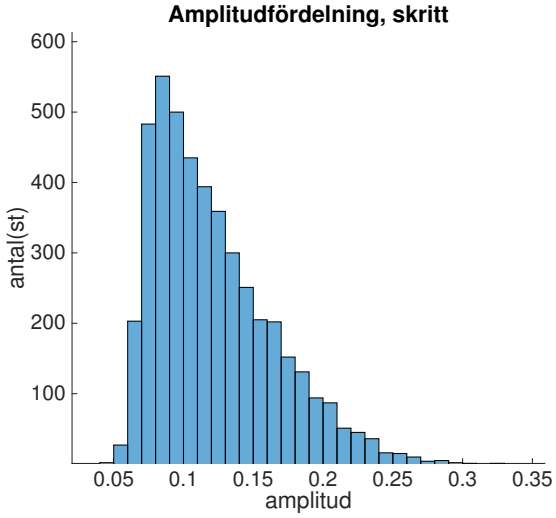


Figur 13: *Fem stycken sjusekundersintervall för galopp som har transformerats med FFT. Cirklarna markerar de värden som sparats och visualiseras i Figur 14–17. Detta exemplet kommer alltså resultera i fem frekvensvärden som ligger på* $\approx 1.75\,\mathrm{Hz}$ *och fem amplituder i intervallet* 0.84–0.94.

är frekvensen tydligt framträdande med de flesta amplituderna större än 0.4 i amplitudspektrumet, se Figur 15b och Figur 16b. För både trav och galopp är frekvensen till skillnad från skritten koncentrerad kring små områden. För traven var den koncentrerad kring $\approx 2.75 \pm 0.25\,\mathrm{Hz}$ med få undantag och för galoppen kring $\approx 1.7 \pm 0.2\,\mathrm{Hz}$. Som kunde förväntas från Figur 12 så hade galoppen en lägre frekvens jämfört med traven utifrån den data som analyserats. För promenaden är frekvensen också tydligt framträdande med majoriteten av de hittade grundrekvenserna inom frekvensintervallet 1.8–2.2 Hz, se Figur 17a, vilket till viss del överlappar med frekvensen för galopp. Liksom för trav och galopp är dessa frekvenser tydligt framträdande, men med amplituder mellan 0.05–0.75.

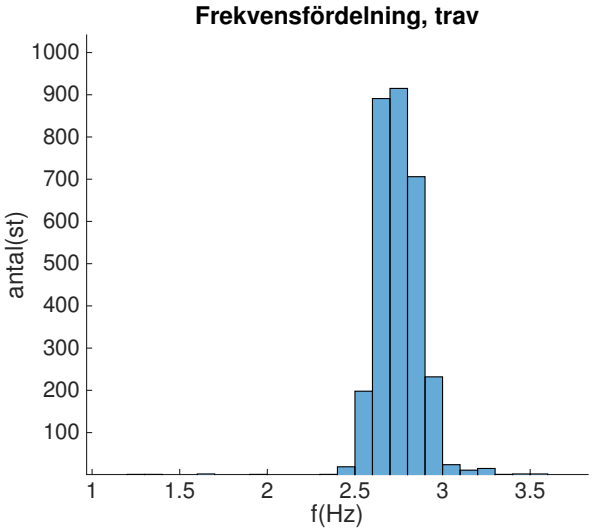


(a) *Frekvensintervallet för skritten. Mest förekommande frekvenser finns kring* 1.7 Hz *men övertoner som misstolkats som grundfrekvensen gör att frekvenser upp till* 5.5 Hz *också förekommer.*

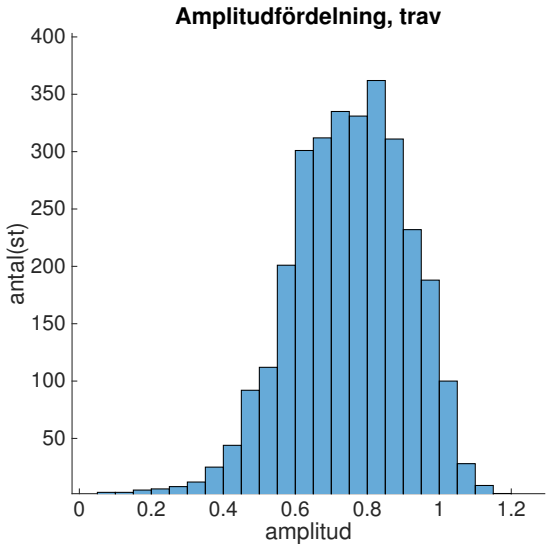


(b) *För skritten visades lägre amplituder än vid trav och galopp av det FFT genererade frekvensspektrumet. Amplituden varierade mellan* ≈ 0.05 *och* ≈ 0.3 *men mest förekommande var amplituder på* ≈ 0.085.

Figur 14: *Fördelning av frekvenser med den högsta amplituden vid varje mättillfälle under skritt.*

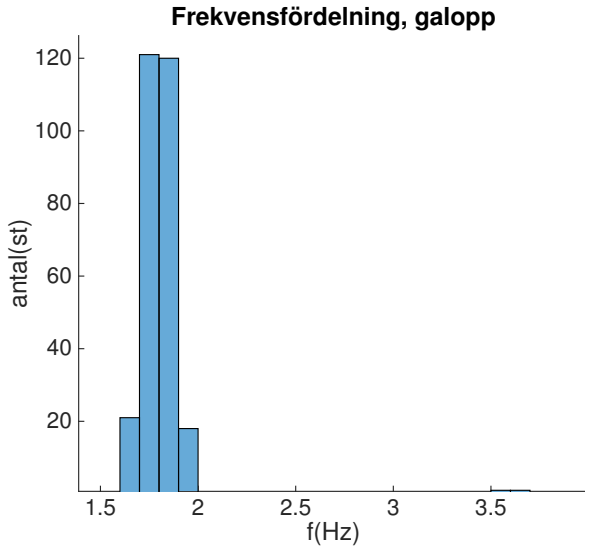


(a) *Frekvensintervallet för traven. Här infaller den vanligaste frekvensen i princip alltid inom intervallet* 2.5–3.0 Hz.

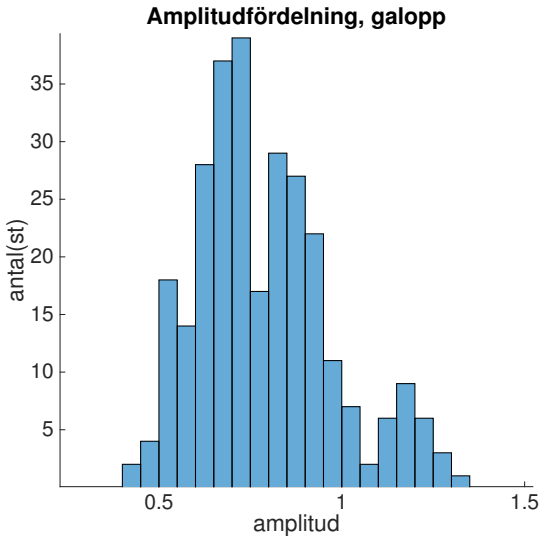


(b) *För traven urskiljdes högre amplituder i det av FFT genererade frekvenspektrumet än för skritten. Amplituden för vanligaste frekvensen är sällan lägre än* 0.4 *och når vanligast upp till* ≈ 0.6–0.9.

Figur 15: *Fördelning av frekvenser med den högsta amplituden vid varje mättillfälle under trav.*

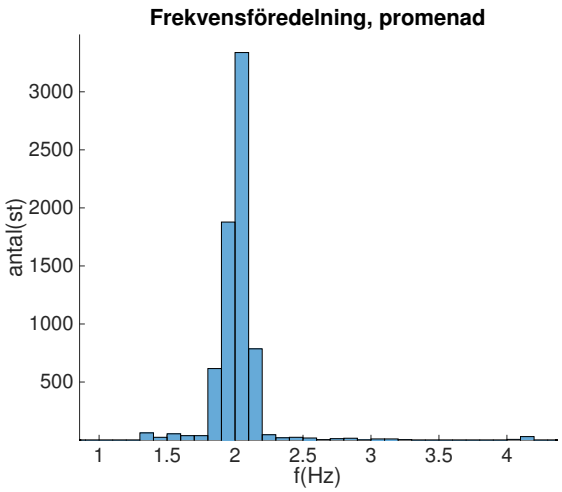


(a) *Frekvensintervallet för galoppen. Galoppen har liksom traven ett tydligt frekvensområde men här med frekvenser mellan* 1.6–2.0 Hz.

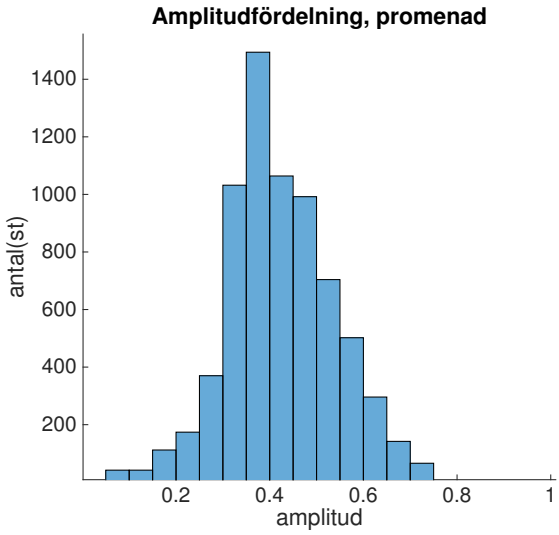


(b) *Liksom för traven (Figur 15b) så urskiljdes högre amplituder i det av FFT genererade frekvensspektrumet än för skritten (Figur 14b). Amplituden för den vanligaste frekvensen är sällan lägre än* 0.4.

Figur 16: *Fördelning av frekvenser med den högsta amplituden vid varje mättillfälle under galopp.*

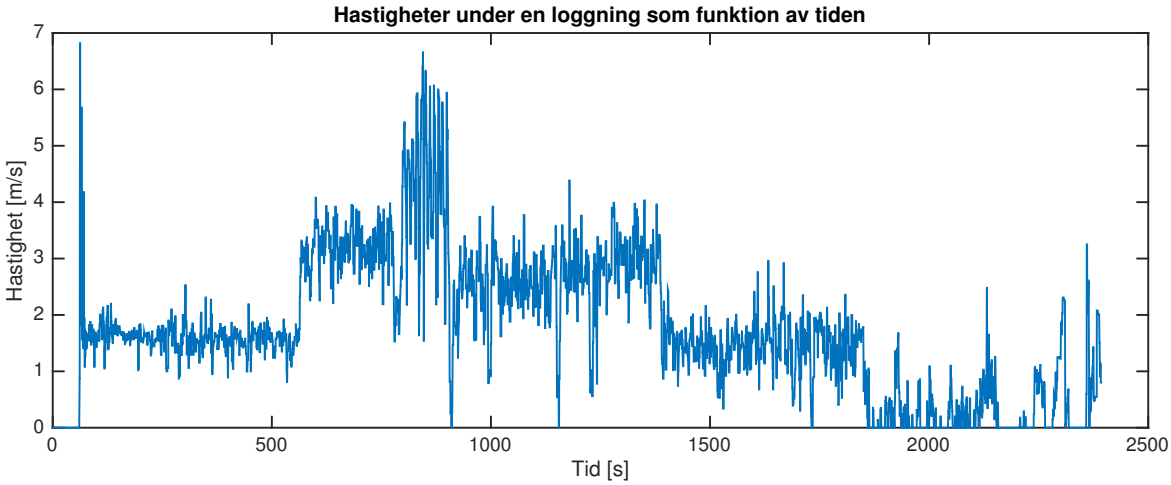


(a) *För promenaden fanns ett tydligt frekvensområde för grundfrekvensen med frekvenser mellan* 1.8–2.2 Hz, *vilka tydligt överlappar med frekvensen för galopp.*



(b) *Även för promenaden urskiljdes relativt höga amplituder i det av FFT genererade frekvensspektrumet. Amplituderna vid de vanligaste promenadfrekvenserna är tydligt utmärkta med de flesta amplituder högre än* ≈ 0.3, *med vissa undantag.*

Figur 17: *Fördelning av frekvenser med den högsta amplituden vid varje mättillfälle under promenad.*



Figur 18: *Plot av hastighet mot tid för loggad ridning av testryttare. I denna kunde fält med verifierad skritt, trav, arbetstrav, galopp och promenad plockas ut, för att sedan räkna ut medelvärde för hastighet. Jämför hastigheterna med Tabell 6. Hastigheten efter* 2200 s *analyseras ej då loggningen stundtals har avbrutits här.*

ID	Gångart	Medelhastighet (m/s)	Tid (s)
1	Promenad—med häst	0.50 ± 0.48	60
2	Skritt	1.60 ± 0.22	350
3	Trav—lättridning	3.20 ± 0.38	150
4	Trav—arbetstrav	2.60 ± 0.43	100
5	Galopp	4.50 ± 1.25	80

Tabell 6: *Medelvärden på hastighet från normalridning utomhus på en D-ponny, där en av gruppmedlemmarna närvarade för att granska loggningen.*

En av ritterna som genererade verifierad normaldata skedde utomhus och kunde därför användas för att göra en uppskattning av hastigheter för gångarterna. Hastigheterna som funktion av tiden för denna ridtur kan ses i Figur 18. För att hitta de specifika hastigheterna för varje gångart granskades varje parti av hastigheter för sig, där en viss gångart identifierats. För dessa partier räknades sedan ett medelvärde ut för varje gångarts hastighet. Dessa hastigheter kan ses i Tabell 6.

4.1.4 Simulering av olycka

Resultatet av de 17 simulerade olyckorna (Metodavsnitt 3.1.4) kan ses i Tabell 7. Av intresse är att se vilka utslag i acceleration och vinkelhastighet som inträffar vid fall. Även här finns problem med att vissa smartmobiler inte kunnat logga med konstant samplingsfrekvens, men datan är ändå relevant eftersom det är minimi-värden av acceleration och vinkelhastighet vid fallet som eftersöks. Maxvärden av acceleration och vinkelhastighet uppnås när testpersonen landar i mattan. I tabellen ses att det är stor spridning av resultaten gällande accelerationer. Den minsta uppmätta accelerationen är 3.1 g och den största 8.6 g. Vinkelhastigheterna är till stor del samlade mellan 7–12 rad/s med några undantag. Minsta vinkelhastigheten är 4.8 rad/s och största är 21.8 rad/s. I Figur 19 plottas sex av fallen efter varandra för att ge en inblick i hur de ser ut i tiden.

Fall	Typ av fall	Max $\lvert a \rvert$ (g)	Max $\lvert \omega \rvert$ (rad/s)	Δt (ms)
1	Bockning eller liknande kast	7.2	10.6	0*
2	Stegring eller liknande fall bakåt	3.7	12.8	103*
3	Bockning eller liknande kast	7.3	4.8	310*
4	Bockning eller liknande kast	5.6	11.3	290*
5	Bockning eller liknande kast	4.8	5.6	130*
6	Bockning eller liknande kast	3.4	7.4	70
7	Bockning eller liknande kast	3.4	8.6	20
8	Bockning eller liknande kast	3.4	10.7	52
9	Bockning eller liknande kast	8.6	7.8	0*
10	Bockning eller liknande kast	6.0	8.5	21
11	Bockning eller liknande kast	6.3	21.8	50
12	Glider av, hängandes längs sidan	5.3	13.3	24
13	Tvärnit, vägran vid hinder	5.5	12.2	32*
14	Glider av, hängandes längs sidan	4.8	8.0	153*
15	Glider av, hängandes längs sidan	5.9	7.6	122*
16	Stegring eller liknande fall bakåt	3.1	6.8	19
17	Stegring eller liknande fall bakåt	3.5	18.0	42

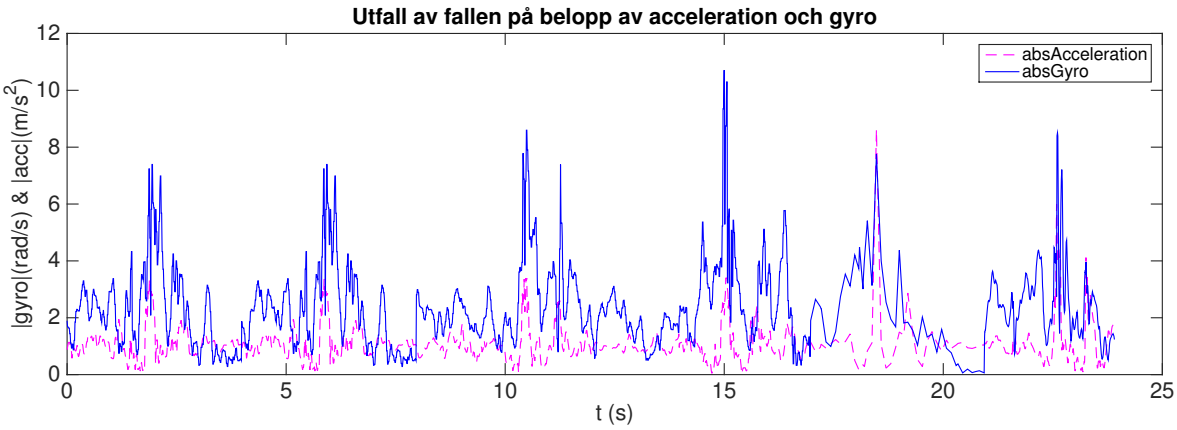
Tabell 7: *Maxbelopp av acceleration och vinkelhastighet samt tidsskillnaden Δt mellan dessa, vid de simulerade olycksfallen. Värdena uppmättes vid kontakt med marken efter fallet. Totalt genomfördes tre fall av typen stegring eller liknande fall bakåt, tio fall av typen bockning eller liknande kast, tre fall av typen glider av och blir hängandes längs sidan samt ett fall av typen tvärnit eller vägran vid hinder. Med liknande fall avses fall som påminner om stegring eller bockning, till exempel fall snett bakåt eller framåt.*

Här ser man att de olika fallen påminner om varandra med avseende på absolutacceleration och absolutgyro; den plötsliga höjningen av dessa två variabler sker ungefär samtidigt i varje fall och dessutom uppnår vinkelhastigheten nästan alltid ett högre värde än accelerationen vid själva fallet.

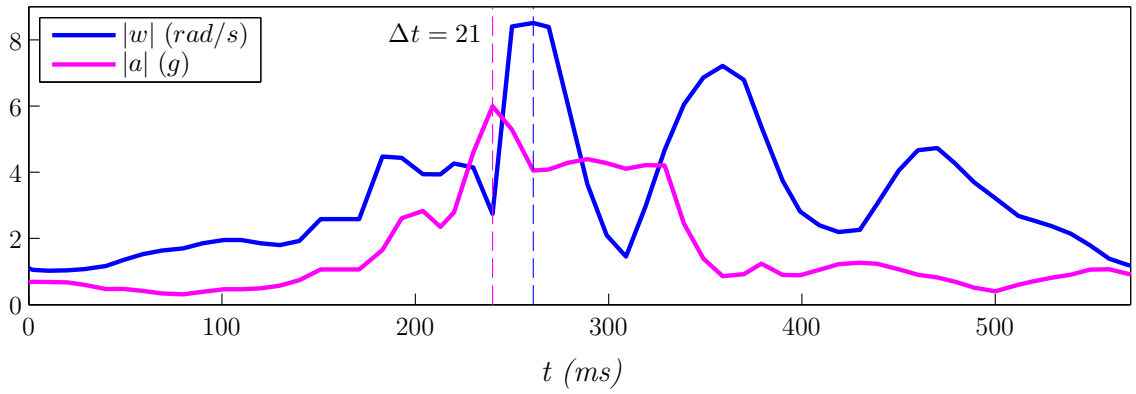
Tidsskillnaden mellan toppen för acceleration och vinkelhastighet vid kontakt med mattan har analyserats för smartmobiler där samplingsfrekvensen är tillräckligt hög. Tidsskillnaden varierar något men är 70 ms eller mindre för samtliga av dessa. Ett exempel på hur värdet mäts ses i Figur 20 där tidsskillnaden är 21 ms.

Vid analys av de simulerade olyckorna ses också att hantering av smartmobilen ger höga värden på framförallt vinkelhastigheten och påminnner mycket om ett fall. Detta kan ses i Figur 21, där absolutbeloppet av både acceleration och vinkelhastighet för simuleringen är plottad mot tiden.

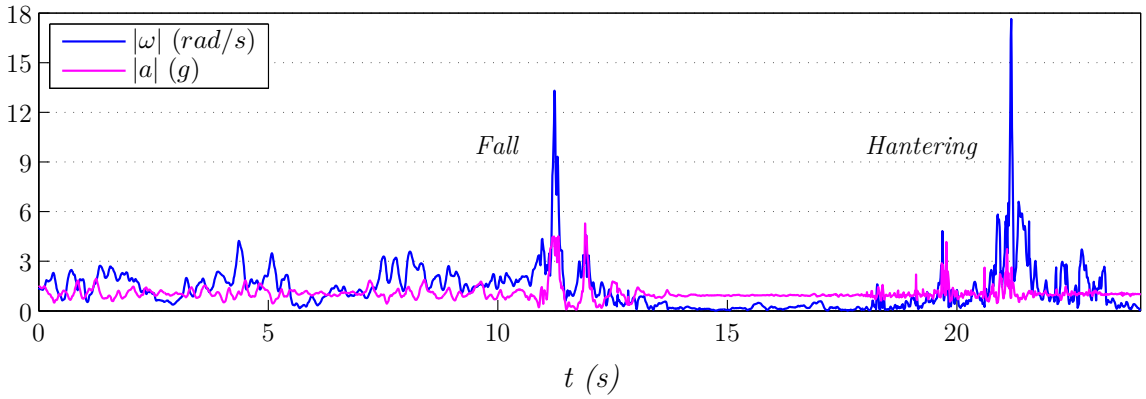
*Osäker samplingsfrekvens vid denna mätning.



Figur 19: *Belopp av acceleration och vinkelhastighet under sex olika fall från tjur. Rosa färg visar beloppet av accelerationen medan blå visar beloppet av vinkelhastigheten.*



Figur 20: *Belopp av acceleration och vinkelhastighet i ögonblicket då personen träffar marken från en av de simulerade olyckorna. Skillnaden är här* 21 ms.



Figur 21: *Absolutbelopp av acceleration och vinkelhastighet under en simulerad olycka. Här ses att vinkelhastigheten är högre vid hantering av smartmobilen än vid själva fallet. Accelerationen når också höga värden vid hanteringen men inte lika höga som vid fallet.*

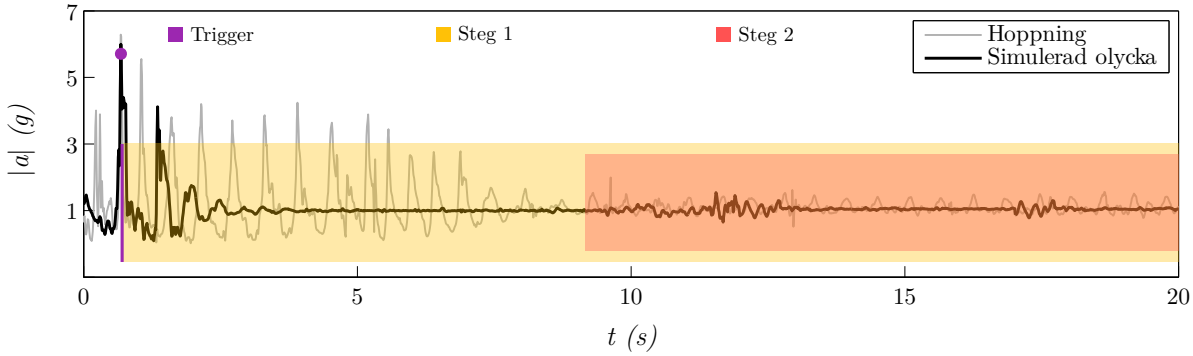
4.2 Algoritm

Det bestämdes utifrån fallplottar samt via inspiration av jalps algoritm [11] att tröskelvärden och triggers skulle användas vid uppbyggnad av algoritmen. Utifrån analys av belopp för acceleration och vinkelhastighet från simulerade olyckor (Tabell 7), upptäcktes att algoritmen som ska detektera olycka ej enbart kan baseras på identifierade tröskelvärden för dessa variabler. Appen skulle isåfall ge falsklarm allt för ofta, eftersom tröskelvärdena överskrids frekvent under galopp (Figur 10) och hoppning men även till viss del vid trav. Algoritmen behöver alltså bestå av fler villkor, som också kan bestämma om en ritt fortfarande pågår enligt det normala. De övriga villkor som anses användbara är rittindikationer (identifiering av gångart) samt rörelseindikationer (påvisning av hastighet eller generell aktivitet).

4.2.1 Utformning

Algoritmens utformning kan förklaras med hjälp av Figur 22. För visualiseringen jämförs här en simulerad olycka med ett hopp över hinder i storleksordningen 100 cm. Dessa jämförs för att visa skillnaden mellan hur algoritmen identifierar en olycka respektive ingen olycka. Endast beloppet av accelerationen visas i figuren för att åskådliggöra händelseförloppen. Vid triggermarkeringen, alltså vid den lila cirkeln, är hoppet och olyckan synkroniserade och båda antar värden över triggervärdet. Detta sker då ekipaget landar i marken efter hoppet respektive personen landar i mattan vid den simulerade olyckan. Efter hoppet tar hästen ett antal galoppsprång med en övergång till skritt vid cirka tio sekunder. I olycksfallet ligger personen relativt stilla en tid efter fallet.

För att algoritmsekvensen för identifiering av olycka skall starta krävs först att en form av trigger hittas, genom att tröskelvärden överskrids som påvisar att en kraftigare stöt har inträffat. Denna detekteras genom en kombination av belopp hos både acceleration och vinkelhastighet. När denna triggerpunkt överskrids startar *Steg 1* där algoritmen letar efter indikationer för att ritten fortfarande pågår. Detta görs genom att jämföra signalen med hur den ser ut vid hästens olika gångarter. Om en gångart identifieras inom tidsrymden för *Steg 1* återställs sekvensen, annars övergår den till *Steg 2* där rörelseindikationer såsom hastighet och generell aktivitet från accelerometern börjar analyseras. Algoritmen letar alltså här efter tecken på rörelse som indikerar att ryttaren inte skadats. Skulle nya triggers upptäckas startar algoritmsekvensen för identifiering av olycka om från tiden där senaste trigger inträffade. Om algoritmsekvensen fullföljs utan att återställas flaggar algoritmen för att en olycka har inträffat.



Figur 22: *Visualisering av algoritmens utformning. Beloppet av accelerationen jämförs mellan ett hopp och en simulerad olycka för att i tidsrummet ge en uppfattning om algoritmens olika steg. När triggerpunktvärden överskrids (lila cirkeln) börjar Steg 1 som innebär sökning efter gångarter som visar att ridturen fortfarande pågår. Skulle detta inte hittas går algoritmen in i Steg 2, där tecken på aktivitet/rörelse eftersöks. Värden på axlar är nödvändigtvis inte exakta för algoritmen, men är av angiven storleksordning.*

4.2.2 Utfall

Algoritmen har testats mot den normaldata som samlats in (Tabell 4) med bra kvalitet (1893 min $\approx$ 32 h) samt alla de 17 simulerade olyckorna (Tabell 7) upprepade gånger med förändring av tidigare nämnda parametrar (såsom triggervärden och tider) mellan testerna för att hitta det optimala resultatet.

Utfallet av den slutgiltiga algoritmen för normaldata, som inte innehåller några olyckor och där algoritmen därför inte skall upptäcka någon, ses i Tabell 8. Totalt utgörs de 32 timmarna av 38 ritter. Av dessa är det 32 som passerar algoritmen felfritt och tre där falsklarm i samband med avsittningar förekommer. Dessa är förmodade avsittningar då larmen inträffar i slutet av respektive loggning. ID 10 och ID 16 skedde vid hoppning i ridhus och innehåller tre respektive ett falsklarm. ID 15 har fem falsklarm och var från en ”ridlekis”-lektion, som innebär lektionsridning för nybörjare som ridit i mindre än ett år.

Algoritmens utfall för de simulerade olyckorna, där en olycka bör upptäckas i varje fall, ses i Tabell 9. Av 17 fall är det 16 där algoritmen identifierar olyckan korrekt men en där algoritmen missar olyckan. Noterbart är att det i detta fall beror på att algoritmen identifierar en gångart efter fallet.

Träffsäkerheten för all data som körts genom algoritmen, både normaldata och olycksdata, kan ses i Figur 23. ”Normal” representerar normaldata där inga falsklarm skall ske medan ”larm” står för där algoritmen skall larma och identifiera olyckor. Algoritmen klassificerar normaldatan korrekt i 84.2 % av fallen medan den identifierar olyckorna i 94.1 %. Den totala träffsäkerheten blir därav 87.3 % (48 av 55 fall korrekt klassificerat).

Sett till antalet timmar normaldata som testats och antalet falsklarm som genereras från denna ges en indikation på hur många falsklarm som inträffar per tidsenhet med nuvarande algoritm. Det totala antalet timmar uteritt ses i Tabell 8 och blir cirka 21 h. Under dessa timmar genereras två falsklarm, från ritt-ID 14 respektive 52. Detta medför att algoritmen ger ett falsklarm ungefär var tionde timma. Görs samma analys för ridning i ridhus finns ca 10 h loggad data med tio genererade falsklarm. Det medför

att algoritmen ger ett falsklarm i timman för ridning inomhus.

Ritt-ID	**Längd** (min)	**Typ av ritt**	**Klassificering** ● Korrekt ● Ger falsklarm
1	28	Uteritt	●
2	16	Uteritt	●
3	19	Uteritt	●
4	67	Uteritt	●
5	43	Uteritt	●
6	46	Uteritt	●
7	33	Ridhus, ridlekis	●
8	60	Ridhus, markarbete	●
9	43	Ridhus, dressyr	●
10	55	Ridhus, hoppning	● 3 falsklarm
11	41	Ridhus, hoppning	●
12	123	Uteritt	●
13	108	Uteritt	●
14	50	Uteritt paddock	● 1 falsklarm, vid avsittning
15	33	Ridhus, ridlekis	● 5 falsklarm
16	55	Ridhus, hoppning	● 1 falsklarm
17	41	Ridhus, hoppning	●
18	50	Ridhus, dressyr	●
19	50	Ridhus, dressyr	●
20	42	Ridhus, dressyr	●
21	0	Simulering upp/avsittning	●
22	0	Simulering upp/avsittning	●
23	33	Uteritt	●
24	27	Uteritt	●
25	35	Uteritt	●
26	56	Ridlektion	● 1 falsklarm, vid avsittning
27	97	Uteritt	●
28	54	Ridlektion	●
29	108	Uteritt	●
30	47	Uteritt paddock	●
31	120	Uteritt	●
32	19	Uteritt	●
33	64	Uteritt	●
34	43	Uteritt	●
35	41	Uteritt	●
36	55	Uteritt	●
37	52	Uteritt	● 1 falsklarm, vid avsittning
38	40	Uteritt	●

Tabell 8: *Algoritmens utfall för normal ridning där inga olyckor förekommit. Grön markering indikerar att algoritmen klassificerar datan korrekt. Falsklarm som beror på avsittningar är förmodade avsittningar, eftersom dessa falsklarm inträffar i slutet av respektive loggning.*

Fall-ID	Typ av fall	Klassificering ● Korrekt ● Missar olyckan
1	Bockning eller liknande kast	●
2	Stegring eller liknande fall bakåt	●
3	Bockning eller liknande kast	●
4	Bockning eller liknande kast	●
5	Bockning eller liknande kast	●
6	Bockning eller liknande kast	●
7	Bockning eller liknande kast	●
8	Bockning eller liknande kast	● Hittar gångart
9	Bockning eller liknande kast	●
10	Bockning eller liknande kast	●
11	Bockning eller liknande kast	●
12	Glider av, hängandes längs sidan	●
13	Tvärnit, vägran vid hinder	●
14	Glider av, hängandes längs sidan	●
15	Glider av, hängandes längs sidan	●
16	Stegring eller liknande fall bakåt	●
17	Stegring eller liknande fall bakåt	●

Tabell 9: *Algoritmens utfall för simulerade fall där en olycka skall detekteras i varje fall. Grön markering indikerar att algoritmen klassificerar datan korrekt.*

		Utfall		
		normal	**larm**	
Faktisk	**normal**	32 58.1 %	6 10.9 %	38 84.2 %
	larm	1 1.8 %	16 29.0 %	17 94.1 %
		33 97.0 %	22 72.7 %	55 87.3 %

Figur 23: *Visar träffsäkerheten i algoritmen vid* 55 *olika ritter som är loggade med LogYard. Av de* 38 *normalritterna klassificeras* 32 *korrekt* (84.2 %)*. Av de* 17 *simulerade olyckorna identifieras* 16 *av dem* (94.1 %)*. Totalt klassificeras* 48 *av* 55 *korrekt* (87.3 %)*.*

		Utfall					
		Galopp	Trav	Skritt	Promenad	Stillastående	
Faktisk	Galopp	280 2.3 %	0 0 %	2 0 %	0 0 %	0 0 %	282 99.3 %
	Trav	3 0 %	2991 24.5 %	28 0.2 %	0 0 %	0 0 %	3022 99.0 %
	Skritt	0 0 %	0 0 %	4530 37.1 %	2 0 %	30 0.3 %	4562 99.3 %
	Promenad	58 0.5 %	31 0.3 %	224 1.8 %	3190 26.1 %	17 0.1 %	3520 90.6 %
	Stillastående	0 0 %	0 0 %	26 0.2 %	0 0 %	795 6.5 %	821 96.8 %
		341 82.1 %	3022 99.0 %	4810 94.2 %	3192 99.9 %	842 94.4 %	12 207 96.6 %

Figur 24: *Visar träffsäkerheten för gångartsbestämningen för* 12 207 *mätningar. Av de* 282 *galoppmätningarna klassificeras* 280 *korrekt* (99.3 %). *Av de* 3022 *travmätningarna klassificeras* 2991 *korrekt* (99.0 %). *Av de* 4562 *skrittmätningarna klassificeras* 4530 *korrekt* (99.3 %).

En del av olycksdetekteringsalgoritmen består också av en igenkänningsalgoritm för gångarter. Denna igenkänningsalgoritm används för att avbryta olycksdetekteringen om en gångart identifieras. Utöver att den ingår i olycksdetekteringsalgoritmen sparar igenkänningsalgoritmen även data om hur många tidsenheter av varje gångart som identifierats under ridturen. Detta kan senare användas för att ange hur stor andel av varje gångart som ridturen bestått av.

Träffsäkerheten för igenkänningsalgoritmen ses i Figur 24, där all verifierad normaldata testas mot algoritmen. Figuren visar hur många procent av den faktiska datan för varje gångart som identifieras via algoritmen. Kolumnen längst ut till höger anger träffsäkerheten för alla separata gångarter samt den sammanlagda. Raden längst ned i figuren anger andel av det algoritmen identifierat som respektive gångart som faktiskt är korrekt.

4.3 Apputveckling

4.3.1 Första utkast av app

En första svartvit grovskiss över appen kan ses i Figur 25–27, där valet av funktioner är baserat utifrån resultatet på enkäten (Avsnitt 4.1.1). I Figur 25a visas startskärmen, den skärm användaren möter då appen öppnas. I mitten av skärmen finns en cirkel med sammanfattning över föregående ritt. Cirkeln är uppdelad i tre olika färgfält. Dessa fält kommer efter avslutad ritt visa hur stor procentuell del av ridturen som ägnats åt skritt, trav respektive galopp. Inuti cirkeln sammanfattas total tid senaste ritt pågick samt hur lång turen var.

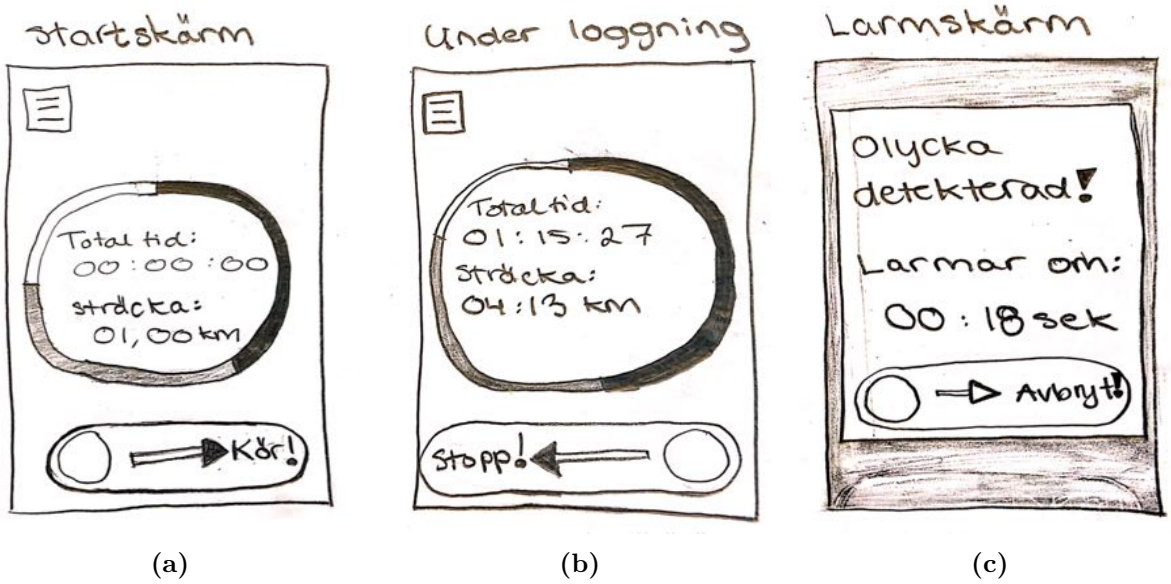
Den skärm som kommer att vara synlig under pågående ridtur ses i Figur 25b. Inuti cirkeln visas nu en tickande klocka samt ökande sträcka, som förväntas visa användaren att loggning pågår. En menyknapp finns uppe i vänstra hörnet, väl igenkänd från andra

appar på Androids telefoner. Med hjälp av menyknappen kan användaren navigera sig till menyn i Figur 26b.

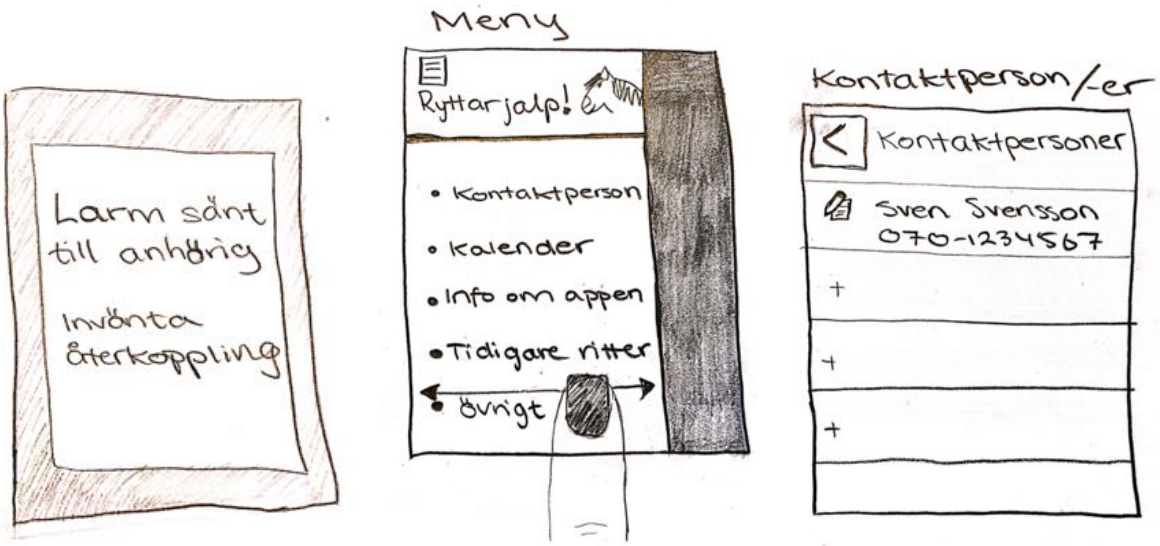
Vid tillfälle då en olycka detekteras av appens algoritm visas skärmen i Figur 25c. En klocka kommer att ticka ner och ett larm kommer ljuda samtidigt som det finns möjlighet att avbryta det väntande larmet. Larmet skickas ej om användaren väljer att avbryta, till exempel vid eventuellt falsklarm. Om larmet avbryts återgår appen till huvudvyn och loggningen fortsätter. Att ha en svepning istället för ett enkelt tryck för att avbryta larmet minskar risken för att det avbryts av misstag. Larmskärmen kommer att skilja sig från start- respektive loggskärmen genom att växla färg till rött, detta ska tydligt signalera att allt inte är som det ska. Avbryts inte larmet innan klockan tickat ner till noll kommer skärmvyn i Figur 26a att visas. Den talar om för användaren att appen skickat iväg ett larm till anhörig, som listats under kontaktpersoner, och uppmanar användaren att invänta återkoppling.

Utöver de olika loggvyerna finns även andra funktioner och därmed vyer. Med hjälp av menyn i Figur 26b kan användaren navigera sig till önskad funktion. Överst i menyn kommer appens namn samt logotyp att visas och under den följer en lista på möjliga val som användaren kan göra. Genom att klicka på en av dessa rubriker kommer en ny sida med innehåll att glida in från höger i skärmen. För att återgå till huvudvyn klickar användaren på tillbaka-pilen längst upp till vänster som framträder i den nya vyn, alternativt trycker på Androids inbyggda ”back”-knapp.

Genom att från menyn klicka på rubriken ”Kontaktperson” kommer vy i Figur 26c visa sig. Denna sida gör det möjligt att välja kontaktpersoner som appen ska kontakta om en olycka skulle detekteras. På skärmen finns olika fält, dessa är till en början tomma men efter att ha skapat kontakter kommer de att synas så som Sven Svensson. Genom att klicka på ett plustecken kan en ny kontakt läggas till i listan. Tanken är att appen ska



Figur 25: *De tre olika loggvyerna startskärm, under loggning och larmskärm. Längst ner på skärmen finns ett ”spår” med en rund knapp placerad på vänstra sidan av spåret. För att starta loggningen, dras denna runda knapp till höger. Den dras sedan tillbaka till ursprungsläget vid ridturens slut. Liknande knapp uppkommer vid detekterad olycka och fungerar på samma sätt.*



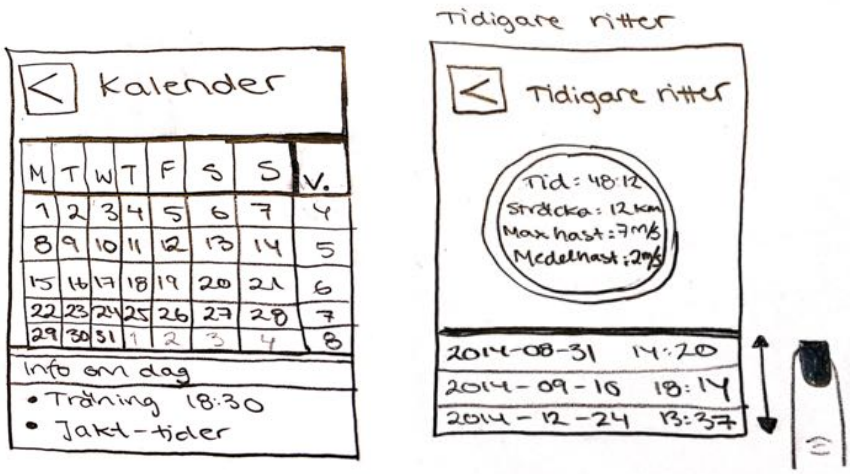
(a) (b) (c)

Figur 26: *Figur (a) visar vyn som syns när larm skickats till anhörig. Appens huvudmeny-vy syns i Figur (b), där användaren kan navigera sig vidare till menyns underrubriker. I Figur (c) syns vyn för att fylla i kontaktpersoner*

kontakta alla personer i kontaktlistan vid en eventuell olycka, antingen alla på en gång eller i turordning. Genom att klicka på pilen uppe i vänstra hörnet återgår skärmen till menyn och användaren kan på nytt välja rubrik.

Under rubriken kalender kommer vy i Figur 27a att dyka upp. Detta är en kalender som visar den aktuella månaden och markerar det aktuella datumet. Längst ner på skärmen kommer det att visas specifik information för dagen, exempelvis fasta händelser som att det är jakttid, men också egna tillagda händelser som träningspass eller skoning för hästen. Under rubriken ”Info om appen” kommer information om samt bakgrund till varför appen skapats.

Slutligen kommer det under rubriken ”Tidigare ritter” vara möjligt för användaren att titta tillbaka på tidigare ritter, denna vy kan ses i Figur 26c. I nedre delen av denna skärm visas en lista med datum och tider. Dessa kommer att representera tidigare ridturer och genom att klicka på dem får användaren upp aktuell data för respektive ridtur i övre delen av skärmen. Datan innefattar hur lång tid ritten varade, hur lång sträcka ridturen var samt max- respektive medelhastighet under passet.



(a) (b)

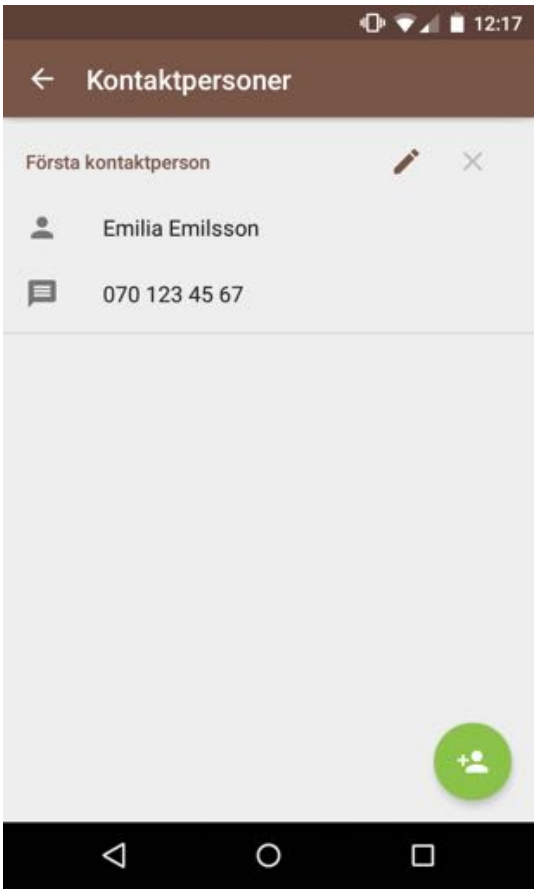
Figur 27: *Två olika vyer som kan väljas i menyn. I Figur (a) visas kalendervyn, som ska innehålla en kalender med både fasta händelser, såsom jakttider, men också egna tillagda händelser. Historikvyn för tidigare ritter syns i Figur (b) och ska visa en sammanfattning av vald tidigare ritt samt en rullmeny med samtliga tidigare ritter.*

4.3.2 Realiserad app

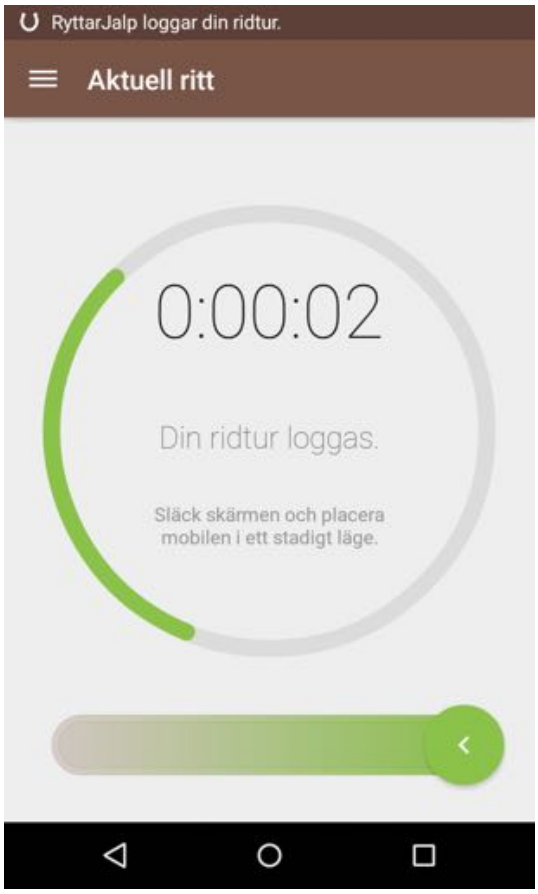
Den slutgiltiga appens gränssnitt har byggts upp utifrån skisserna i Avsnitt 4.3.1. Gränssnittet är till stor del likt skisserna men innehåller även ändringar som presenteras nedan.

Vid första uppstart av appen visas kontaktvyn där användaren ombeds att lägga till minst en kontaktperson. Denna vy ses i Figur 28a. Kontaktpersoner läggs till genom att välja ur smartmobilens befintliga telefonbok. Detta valdes för att minimera risken att ett felaktigt telefonnummer registreras. Maximalt kan tre kontaktpersoner läggas till listan, vilka i tur och ordning meddelas om appen detekterar en olycka. Minst en kontaktperson krävs för att det skall vara möjligt att använda appen.

I Figur 28b ses appens huvudvy efter att användaren startat loggningen. En uppmaning visas om att släcka skärmen och placera mobilen stabilt under ritten. Ingen annan information än tidtagaruret är synlig under inspelningen, till skillnad från första utkast, där även sträckan syntes. Detta är för att minimera mobilhanteringen under ritten, som enligt analys ger upphov till kraftiga sensorutslag. Figur 29a visar huvudvyn efter att loggningen avslutats, precis på samma sätt som i första utkast. Förutom fördelningen av gångarter, som syns i figuren, ändras fältet i cirkeln efter bestämd tid till att visa sträcka samt max- och medelhastighet. Det finns också möjlighet att visa en terrängkarta med rittens registrerade GPS-positioner, markerade med färg beroende på vilken hastighet som hölls vid varje position.



(a) *Kontaktvyn där minst en kontaktperson läggs till. Kontaktpersoner meddelas i tur och ordning om appen detekterar en olycka.*



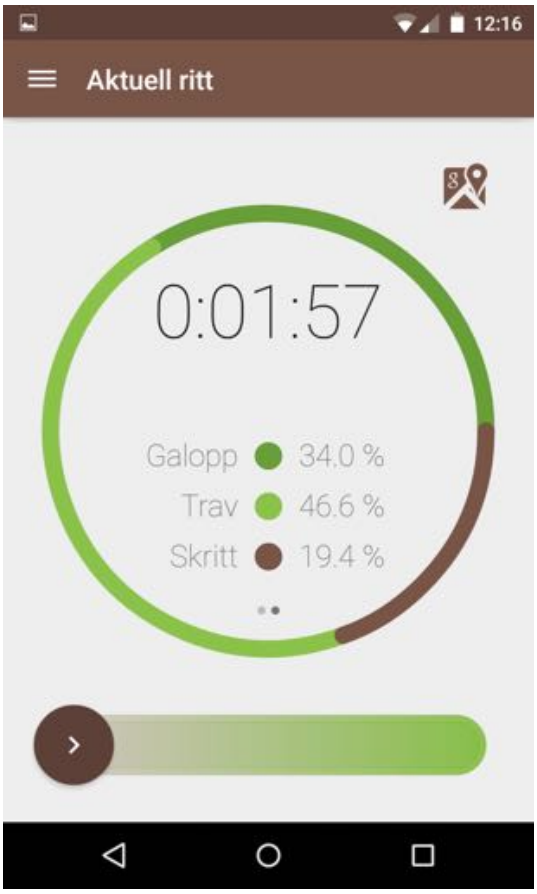
(b) *Appens huvudvy under loggning. Tidtagaruret räknar och den gröna delen av cirkeln roterar för att indikera att loggningen är aktiv.*

Figur 28: *Två av appens vyer: Vy för kontaktlista respektive vyn som syns under pågående ritt.*

Vid detekterad olycka startar larmvyn som ses i Figur 29b. Under nedräkning tills dess att kontaktperson kontaktas vibrerar telefonen och spelar en alarmsignal med ökande volym. Detta görs för att uppmärksamma ryttaren om att ett larm är aktivt.

Det automatiska sms som når första kontaktperson, om ryttaren inte avbryter föregående skärm, inkluderar information om ryttarens senast kända position samt vid vilken tid den registrerades. Det visar också vilken noggrannhet som kan förväntas på ryttarens position. Denna hämtas från Android-systemet, där det antas att GPS-avvikelser är slumpvisa och följer en normalfördelning. Det är då 68 % sannolikt (en standardavvikelse) att korrekt position finns inom en cirkel med angiven radie, där GPS-koordinaten är cirkelns centrum [27]. Ett exempel på mottaget sms ses i Figur 30a. Om användaren lagt till fler personer kontaktlistan meddelas dessa i turordning med fem minuters mellanrum.

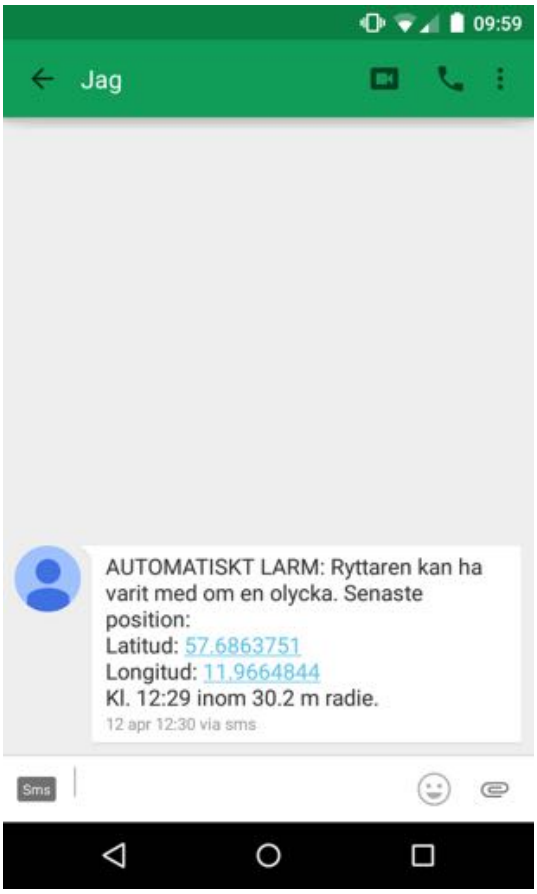
Allt eftersom ritter avslutas lagras information från dessa lokalt i telefonen. I vyn ”Tidigare ritter” som ses i Figur 30b når användaren alla ritter som genomförts, sorterade efter datum. Genom att klicka på datumet expanderas ritten och en sammanställning visas. Karta och GPS-positioner nås via kartikonen på samma sätt som i huvudvyn. Rittinformationen kan lätt raderas om så önskas genom att klicka på respektive kryss.



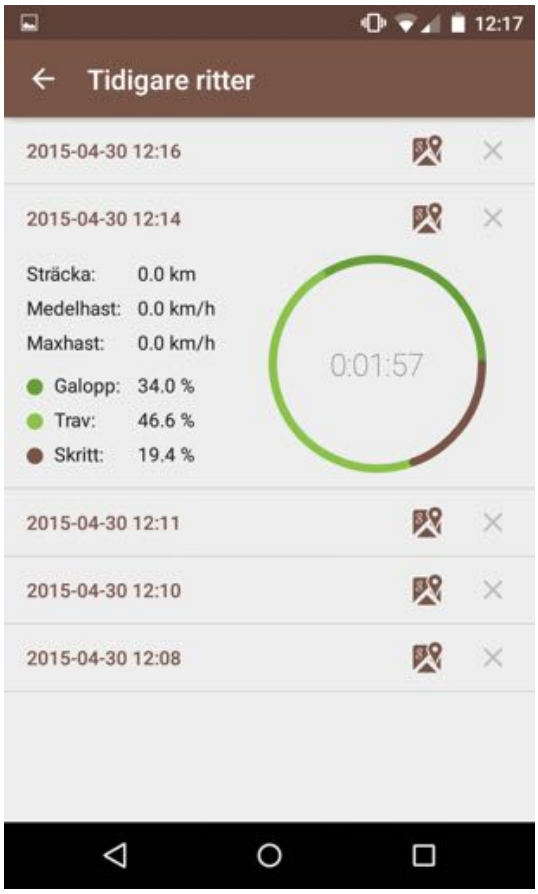
(a) *Appens huvudvy efter avslutad ritt. Här syns fördelningen av gångarter. Kartikonen ger användaren möjlighet att se en terrängkarta med GPS-koordinater från ritten.*

(b) *Larmvyn där användaren under en minut har möjlighet att avbryta larmet innan kontaktperson meddelas.*

Figur 29: *Två av appens vyer: Appens huvudvy efter avslutad ritt respektive vy vid detekterad olycka.*



(a) *Exempel på hur sms:et kan se ut i kontaktpersonens sms-app. Positionsinformation anges för att snabbt kunna lokalisera ryttaren.*



(b) *Vyn "Tidigare ritter" där varje avslutad ritt lagras sorterad efter datum. Användaren kan här lätt jämföra genomförda ritter.*

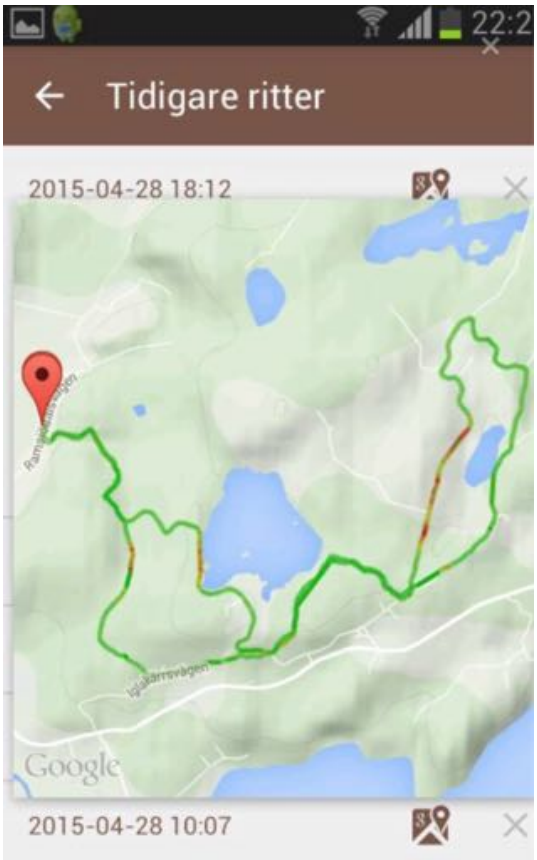
Figur 30: *Två av appens vyer: exempel på sms till kontaktperson respektive sammanställningsvy över tidigare ritter.*

För att kontrollera hur väl appen fungerar testades en betaversion av den under ritt med algoritmen implementerad. Skillnaden mellan denna version och den ovan beskrivna slutgiltiga appen var att sms-funktionen var avstängd. Denna stängdes av eftersom olycksdetekteringsalgoritmens funktion inte kunde säkerställas innan testerna och därför inte skulle skicka ut information vid eventuella falsklarm till kontaktpersoner.

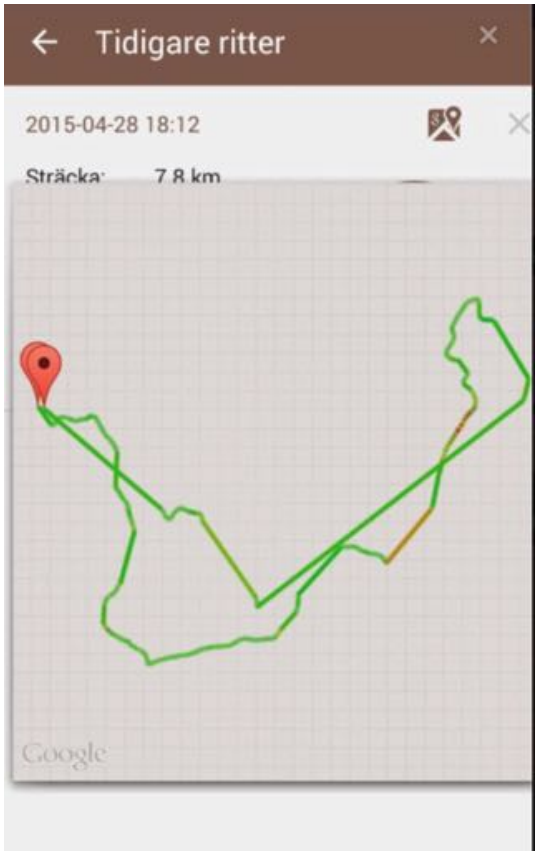
En ridtur i skogsområde på cirka 90 min genomfördes där appen användes av två ryttare. Samma typ av ridning genomfördes då ryttarna red tillsammans under hela turen. Trots ridning i kuperad terräng och delvis utmanande ridning erhölls inga falsklarm.

Kartor med GPS-data från ritten ses i Figur 31. Den ena telefonen har registrerat tätare GPS-positioner än den andra vilket också påverkat den totala uppmätta sträckan, som ses i Tabell 10. Viktigt att poängtera är att GPS-mottagningen inte påverkar hur väl olycksdetekteringsalgoritmen jobbar. Att det i den ena smartmobilen alltid finns täta GPS-positioner visar dock på att appen har varit aktiv och inte avbrutits av operativsystemet under någon del av ritten. Andelen gångarter stämde relativt väl överens mellan de två ryttarna, men skritt är den gångart som skiljer mest. "Annat" i Tabell 10 avser allt utöver de tre gångarterna, såsom promenad, stillastående eller oregelbunda rörelser som inte matchar gångartsalgoritmen.

Ryttarna uppfattade att andelen gångarter under ridturen stämde någorlunda bra med sammanställningen i appen, med undantag för galopp, som upplevdes vara aningen för låg. I övrigt var de nöjda med appen och blev positivt överraskade över det faktum att inga falsklarm genererades.



(a) *Kartläggning av ritt med Samsung Galaxy S2. Mätpunkter har tagits regelbundet med korta intervall vilket gett ett sanningsenligt resultat. Den totala ridsträckan uppges här till att vara* 9.7 km.



(b) *Kartläggning av samma ritt men med annan telefonmodell, Samsung Galaxy S3. Här har GPS-mottagningen varit sämre under en del av sträckan vilket har resulterat i färre mätpunkter. Den totala ridsträckan uppges vara här till att vara* 7.8 km.

Figur 31: *Här ses två kartvyer, en för respektive ryttare vid test med betaversion av app.*

Mobil	Sträcka km	Medelhast km/h	Maxhast km/h	Skritt %	Trav %	Galopp %	Annat %
S2	9.7	4.2	25.2	80.5	0.7	0.7	18.1
S3	7.8	4.6	24.3	90.2	0.2	1.5	8.1

Tabell 10: *Appens sammanställning från två olika ryttare under samma ritt i skogsområde.*

5 Diskussion

I detta avsnitt diskuteras först de delområden som resultatet delats upp i. Efter dessa följer en generell diskussion över hela arbetet samt möjligheter till vidareutveckling.

5.1 Datainsamling

5.1.1 Enkät

Antalet tillfrågade i enkäten var 53 personer. Detta ansågs vara tillräckligt antal för att gruppen skulle kunna basera sitt arbete på resultatet av enkäten. Det hade dock varit önskvärt med ännu fler svar på enkäten. Detta för att om möjligt få svar från fler målgrupper av ryttare, som skulle genererat ett ännu mer statistiskt korrekt svar. Eftersom appens målgrupp förväntas vara många olika typer av ryttare hade det varit önskvärt att flertalet av dessa funnits representerade bland de tillfrågade.

En fråga som hade varit önskvärd att ställa i enkäten hade därmed varit vilken typ av ridning den tillfrågade ägnar sig åt. Därmed skulle en bättre uppfattning om spridningen hos de tillfrågade erhållas som hade gett gruppen bra bild om olika målgrupper har olika åsikter. En kategori ryttare som är mycket utsatta för skador är fälttävlansryttare, som rider i terräng under svåra förhållanden. Dessa känner troligtvis oro under ridning oftare än andra typer av ryttare, som eventuellt ägnar sig åt lugnare ridning. Därmed skulle troligen svaret på frågan angående hur ofta ryttaren känner oro i samband med ridning fått ett annat utfall och behovet av appen Ryttarjalp hade förstärkts ytterligare.

En fråga i enkäten avser hur lång tid användaren skulle vara villig att lägga på uppstart och stopp av ett hjälpmedel som kan öka säkerheten vid ridning. Bland svarsalternativen kan som mest en minut väljas, vilket ger de tillfrågade aningar om att en snabb och lättillgänglig produkt vill utvecklas, vilket därmed kan bli ledande. Svarsalternativen skulle kunna haft större tidsomfattning för att visa mer generellt hur lång tid ryttare är beredda att lägga på ett sådant hjälpmedel.

30 % av de tillfrågade använder en Android-telefon, vilket är en indikation på att det är generellt fler som är iPhone-användare. Appen är idag utvecklad för Android av skäl som återfinns i Avsnitt 1.4. Detta är ej helt optimalt då fler skulle kunna utnyttja appen om den anpassades till iPhone, eftersom denna modell enligt enkäten är vanligare. Det är ändå positivt att se ett relativt stort antal Android-användare. iPhone har stundtals varit ledande inom denna bransch, men Android utvecklas ständigt och blir mer eftertraktat.

5.1.2 Verifierad normaldata

Vid ridning inomhus är GPS-signalen mestadels obefintlig. Ingen logging av hastighet och position kan då genomföras vid användning av LogYard. Dessa två variabler hade varit intressanta att känna till för att kunna koppla ihop gångart med tillhörande hastighet bättre från besöket på ridskolan. Även om information om hastigheter för gångarter kunde hämtas från en ritt som loggats utomhus, hade fler sådana kopplingar varit önskvärda då en större skara ryttare och hästar då funnits representerade. Under ridskolebesöket hämtades data från framförallt ponnyer som generellt håller ett lägre

tempo än större hästar. Därmed borde dess medelhastighet vara något lägre. Denna begränsning i GPS-signal innebär också att att appen inte fungerar lika väl för ridning inomhus som för ridning utomhus. För ritten som loggades utomhus ses också stora variationer i mätningarna. Trots att väderförhållandena var goda och ritten pågick i öppet landskap rapporterade GPS:en fluktuerande värden, främst vid promenad och galopp. Det är därför säkrast att för hastighetsmätningar i appen, beräkna medelvärde över en tid, och inte lita allt för mycket på enstaka mätningar för denna sensor.

Vid studie av samplingsfrekvens kunde det utläsas att denna varit mycket varierande mellan olika smartmobilers loggningar. Detta upptäcktes i början av arbetet och löstes senare genom en uppdatering av appen LogYard. Därefter loggade de flesta telefoner med önskvärd samplingsfrekvens som är 100 Hz.

Samplingsfrekvensen är av stor vikt vid frekvensanalysen, som används för att hitta gångarter hos hästen. Därför användes enbart data med 100 Hz i samplingsfrekvens vid denna analys, för att få ett så exakt resultat som möjligt. Detta medför att vissa gångarter är något underrepresenterade, framförallt galopp. Mycket skritt och trav kunde erhållas från insamlingen av verifierad normaldata, men galopp var svårare då detta inte är en lika ofta förekommande gångart under ett normalt ridpass. Större delen av passet brukar ägnas åt skritt och trav, medan galopp oftast sker under korta sträckor. Eftersom det också är den snabbaste gångarten, sker den också under kortare tid jämfört med trav och skritt. Här skulle därför data från fler studerade ritter vara värdefullt för att öka precisionen. Trots kort tid av galopp så är den insamlad från fem olika hästar vilket stärker särskiljningsalgoritmens trovärdighet. Det upptäcktes också att hästens galopp ger liknande frekvens som vid vanlig promenad och behöver därför ytterligare villkor för att särskiljas. Andra aktiviteter såsom löpning eller cykling har inte analyserats då det bedöms att dessa inte är vanligt förekommande i samband med ridning. Med mer tid hade det dock varit intressant att se hur fler aktiviteter förhåller sig till hästens gångarter.

5.1.3 Simulerade olyckor

I fallögonblicket vid simulering av olyckor, landade den aktuella gruppmedlemmen i en mjuk matta som gav upphov till en viss ”studseffekt”. Detta syntes i data från dessa insamlingar, där accelerationen pendlade frekvent upp och ner under en kortare stund efter fallet. I verkligheten, vid ett fall från hästrygg, landar ryttaren troligtvis på ett hårdare underlag som inte sviktar tillbaka. Därför borde värdena för acceleration och vinkelhastighet vara annorlunda i verkligheten. Det som ger högsta utslag för sensorerna är just i ögonblicket när personen träffar mattan eller marken. Detta beror på att smartmobilen vid den kraftiga stöten börjar ”skaka” och ger upphov till snabba men små orienteringsförändringar. På grund av detta blir acceleration och vinkelhastighet förmodligen högre i verkligheten än i simuleringen på grund av mattan, som hjälper till att dämpa fallen. Det kan därför inte garanteras att de simulerade fallen motsvarar verkliga fall. Eftersom de uppmätta värdena vid simulering är åt det lägre hållet ökar dock säkerheten i att detektera en olycka i verkligheten, snarare än minskar.

Detta är inte det enda som avviker från ett verkligt fall. Storleken på tjuren motsvarade inte en stor häst och den aktuella gruppmedlemmen föll därför inte från en lika hög höjd som från en hästrygg. Ett högre fall kan leda till förändring i acceleration och vinkelhastighet, då ryttaren skulle kunna genomföra ett annorlunda rörelsemönster vid

ett fall från en högre höjd. Dessutom har en mekanisk tjur ingen rörelse framåt och hastigheten innan fallet blir därför obefintlig. Detta är något som också kan påverka den genererade kraften i fallet samt utfall på acceleration och vinkelhastighet. För att simulera mer verkliga fall hade det därför varit önskvärt med fall från högre höjd.

Tröskelvärdena för acceleration och vinkelhastighet skulle kunna justeras noggrannare genom analys av fler insamlade fall, som motsvarar verkligheten på ett bättre sätt. Då hade troligen parametrar såsom acceleration och vinkelhastighet nått högre värden som avvek mer från normal ridning. Dock får det inte riskeras att tröskelvärdena väljs på ett sådant sätt, att lugnare verkliga fall från hästrygg inte genererar någon olycksdetektion av appen. Därför valdes att hålla tröskelvärdena relativt låga, då dessa inte är det enda kriteriet för detektion av olycka i algoritmen. Det krävs en kombination av flera parametrar för att utlösa larmet.

För de simulerade olyckorna användes flertalet smartmobiler med olika accelerometrar. Vissa kunde mäta 8 g medan andra bara klarade 2 g i varje axel. Spridningen i acceleration beror därför i många fall på att accelerationen nått sitt maximala värde vid de lägre värdena. När det gäller vinkelhastigheterna har det inte kunnat ses att något tak har nåtts, utan här beror skillnaderna förmodligen mycket på hur personen landat efter fallet. Värdena för vinkelhastigheten är något lägre än vad gruppen förväntat sig, då det i *jalp*:s tester observerades vinkelhastigheter upp mot 20 rad/s i samtliga kraschtester [11]. Troligtvis beror skillnaden på att testpersonerna här landat på en mjuk matta, jämfört med i *jalp*:s fall där en docka föll handlöst mot fast mark.

Det förekommer även problem i de simulerade olyckorna med att samplingsfrekvensen är varierande under mätningens gång (10–100 Hz). Detta gäller för två smartmobiler av typen iPhone 5s. Eftersom huvudsyftet med simuleringarna är att få en uppfattning om hur höga utslag på accelerometer och gyroskop som inträffar vid fallet har dessa ändå valts att använda. Det är möjligt att just den högsta toppen i utslaget inte detekteras om denna topp skulle inträffa mellan två samplingar, men är isåfall ett problem i rätt riktning då syftet var att hitta de lägsta utslag som förekommer.

Vid hantering av smartmobilen innan och efter själva simuleringen av fallet, erhölls stora sensorutslag bland annat när testpersonen placerade smartmobilen i fickan. Dessa utslag blir ibland större än vid själva fallet och riskerar därför att utlösa falsklarm. I appen har detta lösts genom att tillfälligt stänga av olycksdetekteringen om användaren aktivt använder smartmobilen, samt under en kort tid efteråt.

5.2 Algoritmen

Gruppen är generellt nöjd med träffsäkerheten i algoritmen. Det är många timmar av normal ridning som har testats rakt av, där algoritmen passerar utan falsklarm. Problem med falsklarm finns som väntat framförallt vid avsittningar och vid hoppning över höga hinder. Även lägen med plötsliga rörelser följt av stillastående riskerar att ge falsklarm, som är fallet med ID 15 (Tabell 8) där flera larm förekommer. Detta var en ”ridlekis”-lektion med ryttare i ålder 5–7 år där ekipaget var stilla mer än i rörelse, som därför skiljer sig något mot resten av ritterna. Det är ändå två av fyra hopplektioner som algoritmen klassificerat rätt (ryttarna hoppade i tur och ordning med mycket stillastående emellan) och majoriteten av ritterna ger inte falsklarm vid avsittningar.

Det fall i olyckssimuleringen (Tabell 9) som inte upptäcks har analyserats noggrannare genom att manuellt undersöka datan och jämföra med film som samtidigt spelades in. Det som gör att en gångstil upptäcks (skritt) och därmed avbryter detekteringen, är att mattan som testpersonen landar i studsar med liknande frekvens som vid skritt. Detta fortsatte i cirka 10 s efter fallet och bör normalt inte ske vid verkliga olyckor och kan därför anses vara ett specialfall.

Ett scenario där ett falsklarm kan tänkas ske, med befintlig algoritm, är om ryttaren sitter av tillfälligt under pågående ritt för att exempelvis ordna något med hästen. Kraftfulla avsittningar kan generera sensordata som överskrider tröskelvärdena. Ingen gångart hittas direkt efteråt eftersom ryttaren står still samt hastigheten och generella aktiviteten är för låg, vilket kan göra att algoritmen tolkar det som en olycka. I detta läge skulle ytterligare villkor krävas för att avhjälpa ett sådant larm.

5.3 App

Vid detektion av rörelse som kan påvisa fall i appen informeras användaren om detta genom att en annan vy visas i appen med texten ”Är du ok?”. Med risk för falsklarm vid framförallt avsittningar under ritt skulle det då istället kunna stå ”Har du suttit av hästen?” eller liknande. Detta för att ryttaren då får en bekräftelse på att säkerhetsfunktionen i appen faktiskt fungerar och känner av när ryttaren gör någon avvikande rörelse. Dessutom blir det en bra påminnelse att stänga av appen efter avslutad ritt. Det är vanligt att man under ritt enbart sitter upp och sitter av en gång, därmed skulle detta meddelande komma lägligt i slutet av ridturen. Om ryttaren råkar ut för en olycka under ritt, ställs samma fråga men då ryttaren inte stänger av larmet skickas sms till anhöriga.

Eftersom appen fungerar bäst för smartmobil-modeller med hög samplingsfrekvens, 100 Hz, kan detta vara ett krav för installation. Appen kan under sin uppstart kontrollera samplingsfrekvensen och meddela användaren om den underskrider en viss gräns. Informationen kan ges i form av en varning med meddelandet att appen inte kan garanteras fungera optimalt för just denna smartmobil-modell eller liknande.

5.4 Generell diskussion

Olycksdetekteringen är baserad på den normaldata gruppen fått in under arbetet. Specifika fall förekommer förmodligen i verkligheten, där både ryttare och häst rör sig på andra sätt än i normalfallet. Här är svårt att säga något om hur resultatet blir. Gruppen anser ändå att många ridtyper finns representerade i normaldatan och att appen därför kan anses vara tillförlitlig i de flesta fall. Att larmfunktionen fungerar beror också på andra parametrar såsom att mobiltäckning finns, smartmobilen är i helt skick och att det finns pengar för att skicka sms, vilket har tagits för givet i detta arbete.

Olycksdetekteringen baseras i viss mån på hur aktiv ryttaren är efter en misstänkt olycka. Det är därför i första hand de allvarliga olyckorna som kan förväntas ge larm, då ryttaren är såpass skadad att den inte kan röra sig eller ringa på hjälp själv. De fall som kan ske där ryttaren sitter upp på hästen igen och fortsätter rida, kommer alltså med stor sannolikhet inte ge larm.

De extra funktioner förutom algoritmen som implementerats i appen förväntas bidra till att ryttare faktiskt använder den. Andelen gångstilar som visas i appen kan inte förväntas vara hundraprocentig i det specifika fallet, men gruppen är nöjd med utfallet för den data som samlats in. GPS-positioner och kartfunktionen fungerar väl men beror främst på hur bra täckning som finns där ryttaren befinner sig.

I appen tas ingen hänsyn till storlek på häst eller var smartmobilen placeras. Funktionen är densamma oavsett och samma kriterium gäller för olycksdetekteringsalgoritmen. Detta är något gruppen har funderat på, om skillnaden är såpass stor att olika varianter av algoritmen bör kunna väljas för användaren innan ritt. Anledningen till att det inte studerats vidare är att det dels varit för liten spridning av häststorlekar i den verifierade datan. Vid loggningar från frivilliga testryttare har de också ombetts notera var smartmobilen har varit placerad under ritten, men detta har i många fall inte gjorts. Det är därför svårt att dra en slutsats om variationerna beror på häststorlek eller var smartmobilen varit placerad. Vi misstänker dock att variationer främst beror på hur stadigt smartmobilen sitter mot kroppen, då datan i vissa verifierade ritter skiljt sig mellan relativt jämnstora hästar och ryttare.

5.5 Vidareutveckling

Även om appen anses redo för lansering på marknad för appar vid arbetets slut finns fortfarande förbättringsmöjligheter och visioner som gruppen inte hunnit implementera i arbetet på grund av tidsbrist.

iPhone är en vanligt förekommande smartmobil som är ledande inom marknaden. Sannolikheten är stor att den fortsätter sin framgång och att användarantalet är högt. Därför är det önskvärt att utveckla appen även för denna typ av smartmobil, för att flera ska få tillgång och kunna använda appen.

Under arbetet har flera typer av dagens smartmobiler använts för att samla in data med. Många av dem använder sig fortfarande av accelerometrar som har en maxbegränsning på 2 g i varje axel (iPhone 4, Samsung Galaxy S2, Samsung Galaxy S5, HTC One X). Eftersom dessa kan nå taket på maximal acceleration även vid normal ridning är det därför svårt att särskilja det normala från kraftigare stötar som förekommer vid fall från hästrygg. Detta är en stor anledning till att ytterligare villkor behöver användas för algoritmen. Det har också påträffats smartmobiler som klarar upp till 8 g (iPhone 5s, iPhone 6). Därför kan det vara önskvärt att vid senare tillfälle, när bättre accelerometrar blir vanligare i fler smartmobiler på marknaden, att genomföra ytterligare studier för att öka träffsäkerheten i algoritmen.

Algoritmen kan antagligen optimeras ytterligare, genom att göra den mer komplex. Något som skulle kunna vara ett stöd till nuvarande lösning vid hantering av smartmobilen, är ”närhetssensorn” som finns i de flesta av dagens smartmobiler. Med hjälp av denna skulle det gå att avgöra om smartmobilen exempelvis lyfts upp ur fickan, och på så sätt avaktivera olycksdetekteringen för att undvika falsklarm. Enstaka tester gjordes med en av gruppens smartmobiler, vilket föll ut väl. Det är dock osäkert om alla närhetssensorer fungerar på samma sätt, samt vad som händer om smartmobilen placeras i exempelvis en ryggsäck. Detta måste isåfall studeras närmare innan det införs.

5.5.1 Ytterligare funktioner

För att göra appen ännu mer attraktiv för användarna kan ytterligare funktioner implementeras i den. I Tabell 3 ses de funktioner som enligt enkäten var mest önskvärda utöver de som redan är inkluderade i appen. I tabellen ses hur många röster funktionerna fått samt hur stor del de utgör av totalt antal röster.

Ett introduktions-sms skulle kunna skickas till valda anhöriga i appen med information om att ryttaren valt just denna person som kontaktperson vid en eventuell olycka under ridning. Då blir det kanske inte en lika stor chock om ett sms angående olycka från ryttaren når den anhöriga om detta skulle ske.

5.5.2 Marknadsföring

För att nå ut till användare och skapa intresse för appen, behöver först och främst information spridas om att ett nytt hjälpmedel existerar. Det mest effektiva sättet att nå ut till aktiva inom ridsporten är att göra reklam på populära hemsidor såsom Facebook, Bukefalos, Hästnet och Svenska Ridsportförbundet.

Ett annat sätt att nå ut till användare på är att använda sig av bloggar. I området Bohuslän och Göteborg finns många stora hästbloggar med över 100 000 unika läsare i veckan. Att då antingen få göra reklam via en sådan blogg eller låta bloggaren själv testa och utvärdera appen hade varit ett bra sätt att skapa intresse för produkten.

6 Slutsats

Arbetet har resulterat i en smartmobil-app utvecklad för Android, med olycksdetekteringalgoritm implementerad. Appen kan detektera olyckor vid fall från hästrygg och sänder om så inträffar, larm till tre förvalda kontaktpersoner. Gruppen anser att appen är mogen för lansering på Google Play.

Skillnaden i sensordata mellan normal ridning och eventuell olycka är tillräckligt stor för att kunna skiljas åt med hjälp av en algoritm. Detta görs utan att behöva anpassa till storlek på ryttare/häst, eller var smartmobilen är placerad. Den framtagna algoritmen kan skilja på simulerade olyckor och den normaldata som samlats in med hög tillförlitlighet. De simulerade olyckorna uppskattas till att motsvara mindre allvarliga olyckor eftersom testpersonerna landat på en mjuk matta efter fallet. Algoritmen bör därför vara mer effektiv på att upptäcka verkliga olyckor eftersom dessa troligtvis ger större utslag i sensordata än vid simuleringarna.

Appens användbarhet anses vara god då ryttaren endast behöver starta loggningen före ritt och stänga av efteråt för att den ska fungera korrekt. Om ett falsklarm trots allt skulle inträffa informeras ryttaren först, som då har möjlighet att enkelt avbryta larmet, innan information sänds ut till anhöriga. De extra funktioner som implementerats (sammanställning av gångarter, hastigheter och kartfunktion) gör appen mer intressant för användaren och bidrar till att appen används i högre grad än om den enbart skulle fungera som en säkerhetsapp.

Om algoritmen går att applicera vid alla typer av ridning kan inte kontrolleras med säkerhet, men vi bedömer att den mängd data som testats och blivit godkänd representerar majoriteten av ridtyper som normalt förekommer. Alltså anser gruppen sig vara nöjda med produkten som arbetet genererat och är förväntansfulla inför vad denna kan göra för framtiden inom ridsporten.

Referenser

[1] Myndigheten för samhällsskydd och beredskap. (6 dec. 2013). *Olyckor med häst* [Online]. Tillgänglig: http://www.dinsakerhet.se/Global/pdf/Fritid_resor/Olyckor%20med%20h%C3%A4st.pdf?epslanguage=sv

[2] Svenska Ridsportförbundet. (20 febr. 2012). *Två av tre ridolyckor kan förebyggas med rätt kunskap och träning* [Online]. Tillgänglig: http://www3.ridsport.se/templates/Pages/NewsPage.aspx?id=27748&epslanguage=sv

[3] J. P. Chitnavis *et al.*, "Accidents with horses: What has changed in 20 years?", *Injury*, vol. 27, nr 2, s. 103–105, mars 1996.

[4] J. Altgärde *et al.*, "Horse-related trauma in children and adults during a two year period", *Scand J Trauma Resusc Emerg Med*, juli 2014.

[5] T. Jagodzinski och G. DeMuri, "Horse-related injuries in children: A review", vol. 104, nr 2, febr. 2005.

[6] Svenska Ridsportförbundet. (26 sept. 2013). *Ryttarens utrustning* [Online]. Tillgänglig: http://www3.ridsport.se/Hastkunskap/Ryttarens-utrustning/Sakerhetsvast/

[7] C. G. Ball *et al.*, "Equestrian injuries: Incidence, injury patterns, and risk factors for 10 years of major traumatic injuries", *The American Journal of Surgery*, nr 193, s. 636–640, 2007.

[8] R. M. Hasler *et al.*, "Protective and risk factors in amateur equestrians and description of injury patterns: A retrospective data analysis and a case - control survey", *Journal of Trauma Management & Outcomes*, nr 5:4, 2011. DOI: 10.1186/1752-2897-5-4.

[9] Google. (2015). *Se var dina vänner är på en karta* [Online]. Tillgänglig: https://support.google.com/plus/answer/3302509

[10] Apptivity. (2012). *Horse Rider SOS* [Online]. Tillgänglig: http://www.horseridersos.com/

[11] R. Andersson *et al.*, "Jalp!: Ett smartphonebaserat hjälpmedel vid cykelolyckor", Kandidatarbete, Chalmers tekniska högskola, Göteborg, 2013.

[12] Myndigheten för samhällsskydd och beredskap, Statens folkhälsoinstitut, *Statistik och analys, Fysisk aktivitet och skador*. Karlstad: Myndigheten för samhällsskydd och beredskap, 2010.

[13] H. N. Kuhl *et al.*, "Concussion history and knowledge base in competitive equestrian athletes", *Sports Health*, nr 6(2), s. 136–138, mars 2014.

[14] H. Stigson *et al.* (2014). *Folksams ridhjälmstest* [Online]. Tillgänglig: https://www.folksam.fi/sites/default/files/rapport_-_folksams_ridhjalmtest_2014.pdf

[15] C. L. Phillips, J. M. Parr och E. A. Riskin, *Signals, Systems and Transforms*, 4. utg. Upper Saddle River, NJ: Pearson Prentice Hall, 2007.

[16] Android Developers. (2015). *Processes and Threads* [Online]. Tillgänglig: http://developer.android.com/guide/components/processes-and-threads.html

[17] Android Developers. (2015). *Communicating with the UI Thread* [Online]. Tillgänglig: https://developer.android.com/training/multiple-threads/communicate-ui.html

[18] Android Developers. (2015). *Local Broadcast Manager* [Online]. Tillgänglig: http://developer.android.com/reference/android/support/v4/content/LocalBroadcastManager.html

[19] Android Developers. (2015). *Services* [Online]. Tillgänglig: http://developer.android.com/guide/components/services.html

[20] Android Developers. (2015). *Logcat* [Online]. Tillgänglig: http://developer.android.com/tools/help/logcat.html

[21] ICEdot. (2015). *ICEdot, We are an emergency identification and notification system* [Online]. Tillgänglig: https://icedot.org/site/

[22] S. Candefjord *et al.*, ”Using smartphones to monitor cycling and automatically detect accidents - towards ecall functionality for cyclists”, presenterad vid International Cycling Safety Conference 2014, Göteborg, 19 nov. 2014.

[23] FitnessKeeper Inc. (4 febr. 2015). *RunKeeper - GPS Följ Spring Gå* [Online]. Tillgänglig: https://play.google.com/store/apps/details?id=com.fitnesskeeper.runkeeper.pro&hl=sv

[24] M. Berntman och B. Modén, ”Socialstyrelsens slutenvårdsregister avseende trafikskador - ett komplement till den officiella statistiken?”, Lunds Tekniska Högskola, 2006.

[25] European Commission. (28 april 2015). *ECall in all new cars from April 2018* [Online]. Tillgänglig: https://ec.europa.eu/digital-agenda/en/news/ecall-all-new-cars-april-2018

[26] Google. (2 febr. 2015). *Platform Versions* [Online]. Tillgänglig: http://developer.android.com/about/dashboards/index.html

[27] Android Developers. (2015). *Location—getAccuracy()* [Online]. Tillgänglig: http://developer.android.com/reference/android/location/Location.html#getAccuracy()

Bilaga A Enkätfrågor

Frågeformulär Säkerhetsapplikation

Hur gammal är du?
Markera endast ett val.

- ❏ 0-9
- ❏ 10-15
- ❏ 16-20
- ❏ 20-40
- ❏ Över 40

Hur länge har du varit aktiv inom ridsporten?
Markera endast ett val.

- ❏ 0-2 år
- ❏ 2-5 år
- ❏ 6-10 år
- ❏ Mer än 10 år

Har du någon gång känt dig orolig för att ramla av och skada dig i samband med ridning?
Markera endast ett val.

- ❏ Ja, nästan alltid
- ❏ Ibland, vid vissa tillfällen (t.ex. uteritt, hoppning osv)
- ❏ Nej, inte alls
- ❏ Vet ej
- ❏ Övrigt:

Om det fanns ett hjälpmedel som kunde öka säkerheten i samband med ridning, hur mycket tid skulle du vara beredd att lägga på det precis innan/efter ridturen?
T.ex. att starta/stoppa hjälpmedlet
Markera endast ett val.

- ❏ högst 10 sekunder
- ❏ högst 20 sekunder
- ❏ högst 40 sekunder
- ❏ Upp till en minut
- ❏ Ingen tid alls

Vilka av dessa funktioner skulle du tycka var intressant om hjälpmedlet kunde visa för dig?
Du kan markera flera alternativ, välj max 5.

- ❏ Max- och medel-hastighet under ridturen
- ❏ Total tid man skrittat, travat och galopperat under ridturen
- ❏ Temperatur
- ❏ Vilken tur man ridit och hur lång den var

- ❏ Att kunna ansluta det till sina kompisar och "tävla" mot dem.
- ❏ Antal kalorier man förbrukat under ridturen/ridpasset
- ❏ Mål eller uppdrag som man kunde uppfylla inom hjälpmedlet med din ridtur (likt ett spel)
- ❏ Höjdskillnader, för att se hur mycket backträning man gjort
- ❏ Om det är jakt-tider
- ❏ När solen går upp/ned, för att t.ex. veta om du behöver reflexer
- ❏ Kalender som visar när det är dags för träning, skoning eller vaccination
- ❏ Övrigt:

Brukar du ha med din mobil under en ridtur?
Markera endast ett val.

- ❏ Ja
- ❏ Nej

Om Ja, var brukar du placera din mobil under en ridtur?
Här kan du kan markera flera. Om du svarade Nej på frågan ovanför, välj då bara alternativet "Jag brukar inte ha den med mig".

- ❏ Jackfickan
- ❏ Byxfickan
- ❏ I en mobilhållare på armen
- ❏ I ridstrumpan
- ❏ I bh:n
- ❏ I midjeväska/bälte
- ❏ Jag brukar inte ha den med mig
- ❏ Övrigt:

Hur ofta brukar du ramla av?
Markera endast ett val.

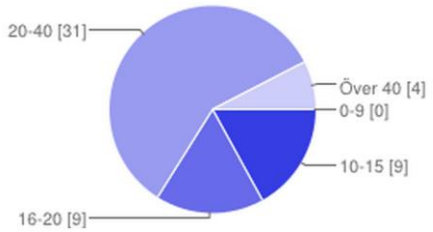
- ❏ 1-2 gånger per år
- ❏ 1-2 gånger i halvåret
- ❏ 1 gång i månaden
- ❏ Oftare än en gång i månaden
- ❏ Aldrig

Vilken mobiltelefon använder du?
Markera endast ett val.

- ❏ Android-telefon (Samsung, Sony m.m.)
- ❏ Windows-telefon
- ❏ Iphone
- ❏ Jag använder ingen smartphone
- ❏ Övrigt:

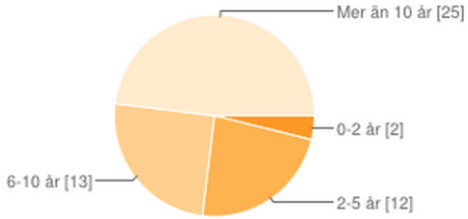
Bilaga B Enkätsvar

Hur gammal är du?



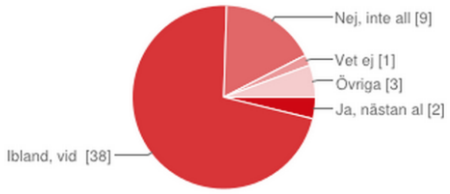
0-9	**0**	0 %
10-15	**9**	17 %
16-20	**9**	17 %
20-40	**31**	58 %
Över 40	**4**	8 %

Hur länge har du varit aktiv inom ridsporten?



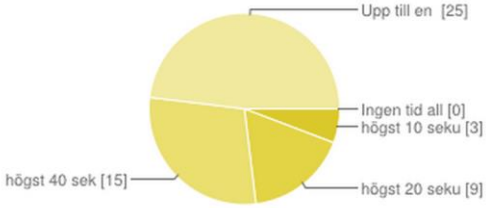
0-2 år	**2**	4 %
2-5 år	**12**	23 %
6-10 år	**13**	25 %
Mer än 10 år	**25**	47 %

Har du någon gång känt dig orolig för att ramla av och skada dig i samband med ridning?



Ja, nästan alltid	**2**	4 %
Ibland, vid vissa tillfällen (t.ex. uteritt, hoppning osv)	**38**	72 %
Nej, inte alls	**9**	17 %
Vet ej	**1**	2 %
Övriga	**3**	6 %

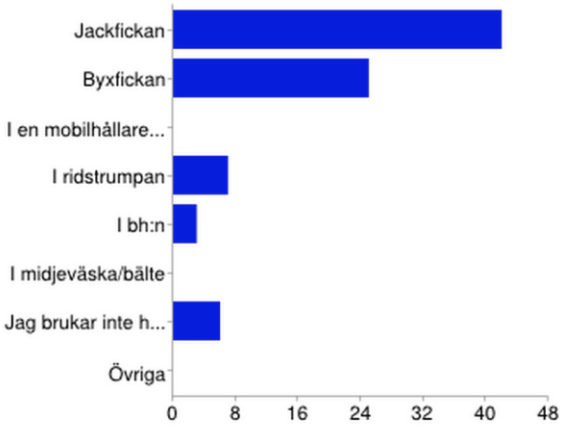
Om det fanns ett hjälpmedel som kunde öka säkerheten i samband med ridning, hur mycket tid skulle du vara beredd att lägga på det precis innan/efter ridturen?



högst 10 sekunder	**3**	6 %
högst 20 sekunder	**9**	17 %
högst 40 sekunder	**15**	28 %
Upp till en minut	**25**	47 %
Ingen tid alls	**0**	0 %

Max- och medel-hastighet under ridturen	**25**	47 %
Total tid man skrittat, travat och galopperat under ridturen	**27**	51 %
Temperatur	**7**	13 %
Vilken tur man ridit och hur lång den var	**34**	64 %
Att kunna ansluta det till sina kompisar och "tävla" mot dem.	**6**	11 %
Antal kalorier man förbrukat under ridturen/ridpasset	**13**	25 %
Mål eller uppdrag som man kunde uppfylla inom hjälpmedlet med din ridtur (likt ett spel)	**8**	15 %
Höjdskillnader, för att se hur mycket backträning man gjort	**9**	17 %
Om det är jakt-tider	**17**	32 %
När solen går upp/ned, för att t.ex. veta om du behöver reflexer	**17**	32 %
Kalender som visar när det är dags för träning, skoning eller vaccination	**14**	26 %
Övriga	**5**	9 %

Om Ja, var brukar du placera din mobil under en ridtur?

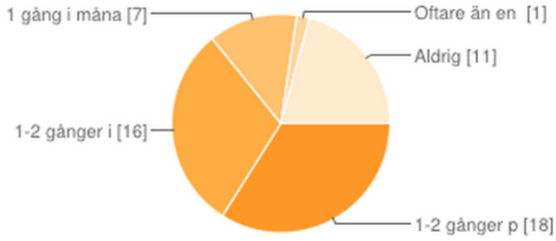


Jackfickan	**42**	79 %
Byxfickan	**25**	47 %
I en mobilhållare på armen	**0**	0 %
I ridstrumpan	**7**	13 %
I bh:n	**3**	6 %
I midjeväska/bälte	**0**	0 %
Jag brukar inte ha den med mig	**6**	11 %
Övriga	**0**	0 %

Brukar du ha med din mobil under en ridtur?

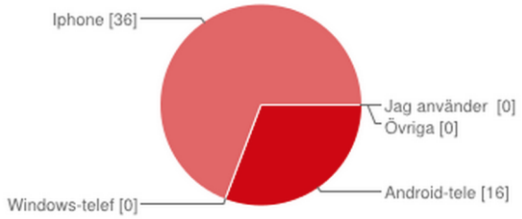
Ja	**46**	87 %
Nej	**7**	13 %

Hur ofta brukar du ramla av?



1-2 gånger per år	**18**	34 %
1-2 gånger i halvåret	**16**	30 %
1 gång i månaden	**7**	13 %
Oftare än en gång i månaden	**1**	2 %
Aldrig	**11**	21 %

Vilken mobiltelefon använder du?



Android-telefon (Samsung, Sony m.m.)	**16**	30 %
Windows-telefon	**0**	0 %
Iphone	**36**	68 %
Jag använder ingen smartphone	**0**	0 %
Övriga	**0**	0 %