Linköpings universitet
TEKNISKA HÖGSKOLAN

ITN

Projektrapport

Program:

Civilingenjör i Medieteknik

Kurs:

TFYA65 Ljudfysik

Projektets titel:

Läderappen

Medlemmar:	LiU-ID:
Nicholas Frederiksen	nicfr426
Julius Kördel	julko800
Jakob Gunnarsson	jakgu444

Handledarens namn:
Per Sandström och Peter Andersson

Datum:
2017-10-9

1. Bakgrund

Allt fler tillämpningar utvecklas för röststyrning, röstigenkänning och röstförvrängning. Röststyrning kan underlätta för användare med särskilda behov/förutsättningar. Röstigenkänning kan underlätta användning och bidra med säkerhet för röststyrning. Röstförvrängning användas för att t.ex. skydda identiteten av ett vittne i en intervju, eller för att skapa musik.

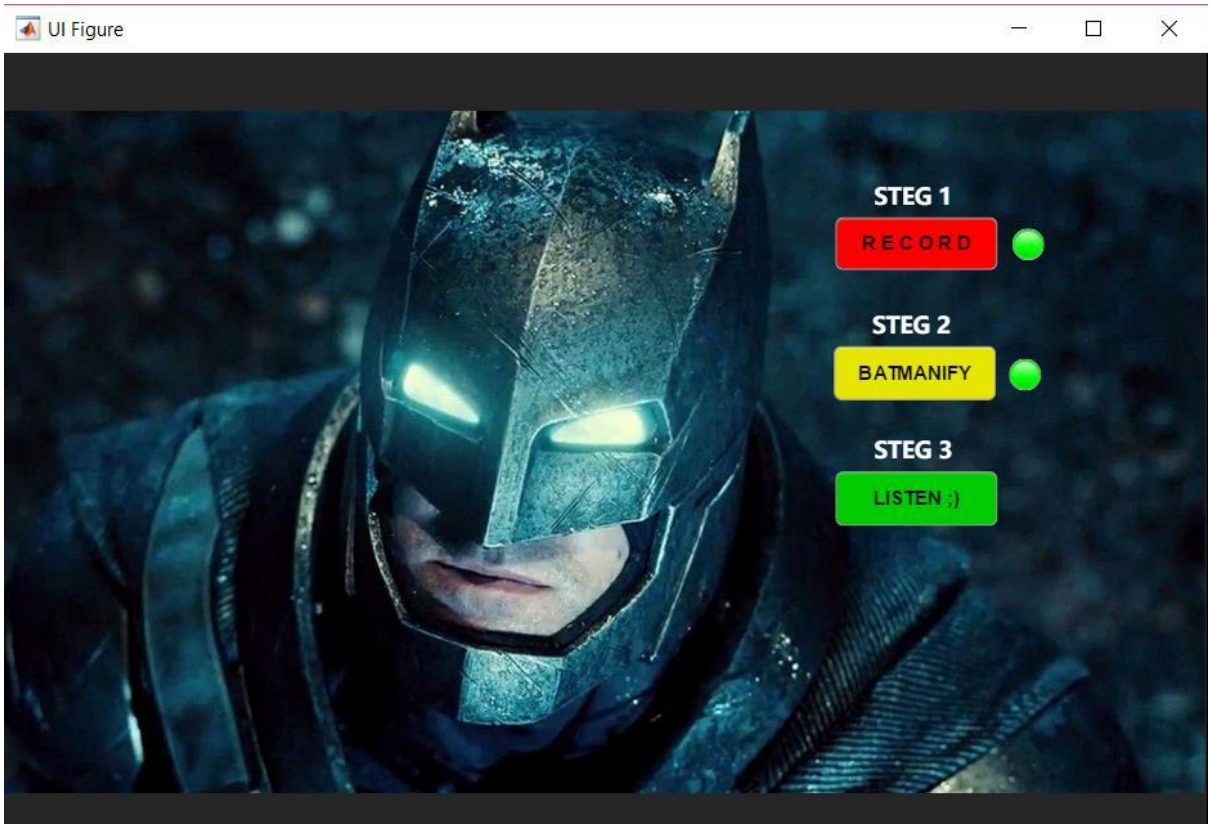
2. Syfte och mål

Syftet med projektet är att fördjupa sig i ljudanalys och ljudbehandling. Samt utveckla förståelsen för de verktyg vilka finns för hantering av signaler, i detta fall ljud i MATLAB, då signalbehandling är en central del i civilingenjör-basen.

Gruppen har målet att skapa en röstmodelator vilket får användarens röst att likna en vald röst, i detta fall en "Batman röst". Där användarens röst inte ska påverka resultatet. Menat att oavsett om rösten är ljus eller mörk skall resultatet fortfarande efterlikna den önskade rösten.

3. Beskrivning av applikation

Användaren talar in en sekvens på 5 sekunder. Sekvensen bearbetas och användaren väljer när den vill spela upp det modifierade ljudet med en uppspelningsknapp. Inställningarna för hur rösten modifieras är gömda för användaren för att underlätta användningen. Vyn användaren möter ses i *Figur 1*.



Figur 1: Bild på användargränssnittet för gruppens MATLAB-applikation

4. Metod

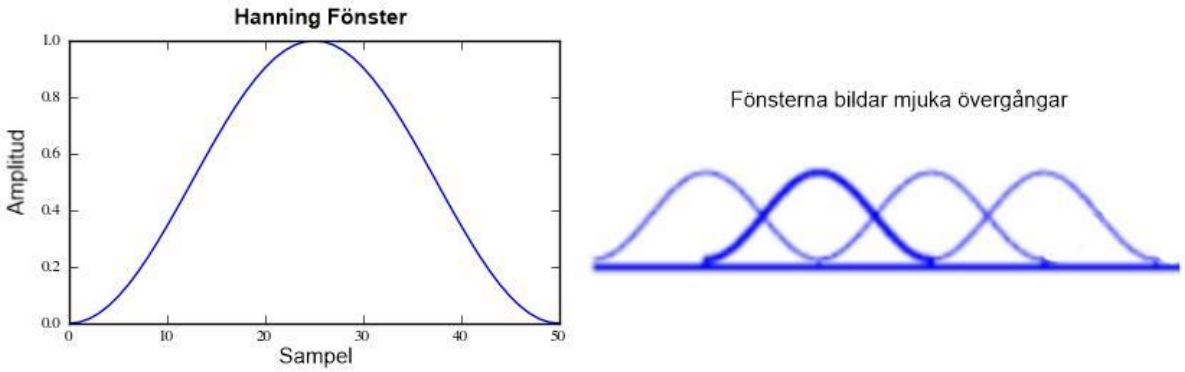
1. Analys

En röst har flera olika egenskaper. Tempo, betoning, ton och styrka är några av de egenskaperna. Manipulationen av rösten i programmet är baserad på en analys av dess ton men även dess styrka. Då en kort beräkningstid eftersträvades behövdes förenklingar ske. I teorin kan varje sampel värde bli korrekt modifierat men detta skulle ta tid. Därför tas medelvärden av bitar istället. Ifall rösten växlar mycket i tonläge bör en noggrannare indelning väljas men om rösten är relativt monotont, dvs liten variation i tonläge, räcker ett medelvärde av hela inspelningen.

Styrka är även typiskt för en röst. En inspelning kan vara svag i jämförelse med den önskade rösten. Då kan inspelningen behöva förstärkas för att det efterlikna det önskade. Att förstärka inspelningen bidrog till upplevelsen men att försvaga gör ej det. Det är mer korrekt i teorin men en sådan korrigering bidrar inte till ett starkare intryck då rösten är i fokus. Styrka och ton är de egenskaper vilka kalibreringarna i resultatet tar hänsyn till.

2. Fönsterhantering

Inom behandling/bearbetning av ljud är det vanligt att använda fönsterfunktioner. Det finns flera olika typer av fönster och de har olika påverkan på resultatet. Vanligt för fönsterfunktioner är att lägga till ett område för övergångarna innan och efter de delar behandlingen skett på. En av dessa typer av fönster är Hanning-fönster. Hanning ger en mjuk övergång mellan delarna sätts ihop till en slutsignal. Slutet av delarna tonas ut och början av delarna tonas in tills de når en övergångspunkt där det ljud som stärkts kommer bli dominant. *Figur 2* visar funktionaliteten hos ett Hanning-fönster av typen periodisk.



Figur 2: Bild över Hanning-fönster.

3. Bearbetnings algorithm

Det utförs två modifieringar av rösten. Den ena för att justera tonläget. Den andra för att skapa en mer förvräng röst likt "Batman". Först tonläget, där används ett en kvot mellan medelfrekvensen dvs medeltonen mellan intalade rösten och den för "Batman". Denna kvot används för att bestämma hur mycket en ton måste ändras för att bli samma ton vilket den önskvärda tonen har. Denna förvrängning kan bli för kraftig vilket medför att det finns störningar i utljudet. Att höja tonläget hos röster under "Batmans" ton gav ett mer rätt resultat matematiskt sett men att istället sänka de likvärdigt motsvarande toner ovanför "Batmans" ton uppfattades mer likt. För att kontrollera dessa kvoter användes en bestämning av kvoten enligt *Figur 3.*

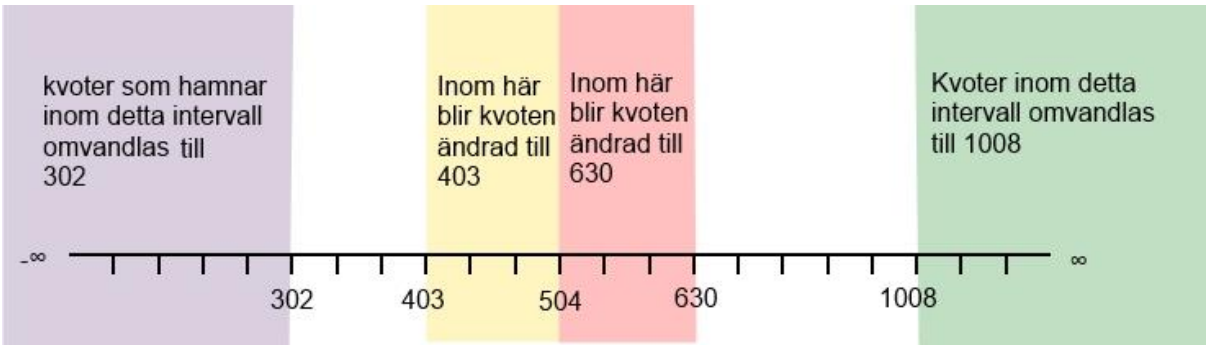


Figure 3: Illustration på hur kvoten bestäms.

Läderappen använder sig av två olika funktioner, en pitch- och en robot-funktion, vilka utför modifieringar för att förvränga användarens röst.

Pitch-funktionen funkar på det sätt att ljudfilen vilket skapades under inspelningen analyseras och ett medelvärde för frekvensen bestäms. Detta medelvärde jämförs mot ett antal förbestämda värden och en kvot "pitch-ratio" tas fram från dessa jämförelser. Det inspelade ljudet delas upp i Hanning-fönster. De uppdelade ljudbitarna görs sedan om till sinusform i frekvensdomänen. Där kvoten "pitch-ratio" multipliceras med ljudbiten i sinusform för att ändra värdena i biten vilket medför att en förvrängning sker. De resulterande bitarna är för stora på grunda av fönster funktionerna och skalas då om innan de sätts ihop. Den slutgiltiga ljudmatrisen vilken kommer ut från funktionen är nu "pitch-skiftad", frekvenserna för det inspelade ljudet har nu flyttats lägre eller högre, (nedåt för Läderappen), vilket resulterar i en mörkare(lägre) eller ljusare(högre) om man var för låg från början.

Den andra funktionen är en robot-funktion. Den gör en robotisering av inspelningen. Denna teknik använder sig av Hanning-fönster likt pitch-funktionen. Där storleken på fönstret påverkar skrovligheten i den modifierade rösten. En mindre fönsterstorlek resulterar i ett mer skrovligt ljud. Annan robotisering kan påverka är hur högt ljudet blir och även lite av tonhöjden. Det man gör är att man lägger nollfasvärden på varje Fouriertransformation innan man rekonstruerar den igen. Effekten applicerar en fast

tonhöjd på ljudet, vilket får det att låta monotont. Då processen tvingar ljudet att vara periodisk, omvandlas oregelbundna variationer i ljudet till "robotliknande" ljud.

Funktionerna i "Läderappen" vilka förvränger användarens röst läggs inte på en efter en på samma ljudfil. Istället utgår alla funktioner från en oredigerad ljudfil och skapar instanser där bara funktionens förvrängning är applicerad. Det sker sedan en addition mellan de skapade instanserna, där går det att specificera hur mycket varje instans bidrar i den ihopsatta ljudfilen, vilken skickas till högtalarna. De tre adderade instanserna är en pitch- och en robot-förvrängning men även det oredigerade ljudet.

4. Utvecklingsverktyg

All utveckling i detta projekt har gjorts i MATLAB, vilket har verktyg för att läsa in och bearbeta ljudsignaler. Det finns bibliotek för hantering av signaler och hur det representerar ljud vilken gruppens medlemmar var bekant med sedan tidigare.

5. Tidsåtgång / Tidsplan

Projektet delades in i 3 faser. Först instudering/påläsning av verktyg och metoder lämpliga för området. Följt av implementering av analys och modifiering. Sista fasen var justering och optimering av programmet. För att skapa en mer önskvärd förvrängning samt en snabbare körtid vilket är en central del i projektet.

5. Resultat

Här presenteras resultatet av det arbete beskrivet i metoden.

1. Hur appen funkar

När användaren möter appen möts den av gränssnitt se *Figur 1*. Appen kräver att användaren talar in en sekvens av ord i fem sekunder. Efter det får användaren sin röst förvrängd till att vara mer lik "Batmans" förvrängda röst. När användaren talat in och bearbetningen är klar kan användaren spela upp resultatet flera gånger.

2. Förvrängning

Den tonförvrängningsfaktor i appen tar hänsyn till vilken ton inkommande rösten har och sedan oavsett ifall den intalade röstens ton är under eller över "Batman" sänker vi tonerna enligt *Figur 3*. Vilket medför att alla inte talar i "Batmans" ton exakt men den förvrängda rösten låter mer likt" Batman".

3. Slutmixning

Mixningen är avgörande för den slutgiltiga förvrängda rösten. Den mixningen i slutprodukten är satt för ett mer generellt bruk. Det betyder alltså att den funkar för en bredare användargrupp men för ett resultat mer likt "Batman" skulle det behövas en manuell korrigering i programmet.

6. Diskussion

Det behövs ofta en bakomliggande analys och förståelse för hur förändringar uppfattas av mottagaren för att skapa ett önskat resultat.

En upptäckt under projektets gång var att om inspelningsrösten hade en lägre medelton än ”Batman” gav det inte alltid ett önskat resultat, då den förvrängda rösten upplevdes pipig. Därför valdes att alltid ändra tonen nedåt även om det är mindre korrekt matematiskt.

Det finns flera olika mixningar vilka skulle ge ett gott resultat. Att få mixningen att styras beroende på den inkommande rösten fanns det ej tid för i projektet. Dock om man går vidare med projektet hade detta varit den viktigaste delen att utveckla vidare, alltså att få en bra mixning oavsett inkommande röst.

Det nämndes flera faktorer under rubriken ‘Analys’. De faktorer vilka ej används i slutprodukten kräver oftast en inlärningsfas men kan ge bättre resultat. Faktorerna kan underlätta beräkningarna och även minska körtiden då man vet mer om användaren innan den förvrängande beräkningen. Alltså genom att utnyttja mer tid innan körning kan man korta ner den totala körningstiden.

Körtiden är en central del i hur funktionell appen är. En användare vill nog inte ha en väntetid vilken överstiger inspelningstiden när de ska lyssna på resultatet. Därför vill man sikta på en sådan kort körtid som möjligt. Anledningen till att man får långa körtider är p.g.a. beräkningarna till förvrängningen. Denna tid går att halvera genom att offra lite av ljudkvaliteten. Man får en liten skillnad i kvalitet men det blir fortfarande ett ganska likvärdigt resultat. Det har varit en ständig avvägning mellan körtid och noggrannhet vid utveckling av ”Läderappen”.

Varje samplevärde kan justeringen beräknas på och det skulle ge ett mer korrekt resultat. Dock resulterar detta i en lång körtid. Därför för att hålla nere körtiden valdes att ta medelvärdet för hela inspelade sekvensen. Det ger ett bra resultat, bara inte inspelningen innehåller en kraftig variation i tonläget. Detta är blir ett av kraven på användaren att inte tala i det lägsta och högst tonläget personen kan i samma inspelning, även om det är ovanligt att en individ gör detta i vanligt tal.

Inspelningen kan delas upp i flera delar, säg ett medelvärde per sekund. Det medför att tonläget kan täcka en större variation i talet men kostar i körtid. För ett vanligt tal upplevdes det inte ge tillräckligt gott resultat för att väga upp för ökningen av körtid och det var basen för beslutet om ett medelvärde per inspelning är grundat i, att spara körtid.

7. Slutsats

Med tanke på den tid projektet var tilldelat och den kunskap gruppen hade från början är gruppen nöjd med resultatet. Den funktionalitet vi eftersträva gick att implementera men att få den i realtid var inte möjligt med den valda bearbetningsmetoden av röster och den datorkraft vilket fanns till förfogande. Vi ser även att det finns flera sätt att vidareutveckla/förbättra produkten enligt det nämnda i diskussionen.