

TTT4280 - Sensorer og instrumentering

Laboppgave optikk: Om utregning av puls

1 Tips til datafangst og prosessering

- Pass på at videofilene navngis på en fornuftig måte så det blir lett for deg å forstå hva de inneholder.
- Vi tracker ikke hvor avbildet objekt befinner seg i kamerabildet, så man bør holde seg i ro. Under prosessering blir du nødt til å manuelt markere hvor i bildet hvor du har hud du vil analysere.
- Når snittsignalet dras ut fra ROI, lagre gjerne dataene til ny fil så du slipper å dra signalet ut igjen på nytt. Automatiser så mye som mulig.
- Plott opp dataene, verifiser at pulssignal er til stede. Kan du telle antall pulsslag i signalet, har du allerede en fasit på hva pulsen skal være.
- Det er lurt å ta en måling, kikke på dataene, evt. endre målesituasjonen og ta ny måling for å bli vant med hvordan det fungerer og hva som må til for å få et godt pulssignal. Da blir det også lettere å implementere prosessering.
- Som lyskilde, bruk for eksempel blitzlampa på smartfonen din.
- Å bruke bare taklys som lyskilde kommer til å fungere dårlig, siden taklyset flimrer. (Gir prosesseringsalgoritmen din samme puls på deg selv som et hvitt ark?)
- Du kan ikke forvente å få et fornuftig estimat på puls ut av dårlige data. **Kjører du inn søppeldata, får du bare søppelresultater ut.**

Det er fritt hvordan dere velger å få ut pulsen fra dataene. Labtekstforfatteren foretrakk å bruke autokorrelasjon, men du kan for eksempel bruke FFT for å finne pulsen. Da vil isåfall pulssignalet komme som en topp i FFT-spekteret. Erfaringer fra akustikklabben kan gjenbrukes for autokorrelasjonsmetoden, mens erfaringer fra radarlabben kan gjenbrukes for FFT-metoden.

Det viktigste er: (1) at du klarer å regne ut en pulsfrekvens i antall slag per minutt og (2) får regnet ut SNR.

1.1 Prosessering

Eksempel på nødvendige prosesseringssteg (i én fargekanal), med hint:

1. Fjerne DC-offset/de saktevarierende delene av signalet. Dette er viktig for å få signalet ned på noe som varierer jevnt på tvers av x-aksen, og ikke har saktevarierende trender i seg som evt. forvirrer de senere delene av prosesseringen.
 - Du vil også gjerne fjerne høye frekvenser, slik som for eksempel taklysfrekvensen, så båndpassfilter kan være passende for denne delen. Sjekk og dokumenter at det hjelper!
 - Tips: Bruk båndpassfilteret du muligens lagde i akustikkklubben for dette, modifisert til å ha de rette frekvensene. Ellers kan du se på MATLAB-eksempelet for band pass butterworth filter.
 - Andre mulige alternativer er å trekke fra moving average med ganske stor vindusstørrelse, evt. bruke et Savitzky-Golay-filter til å regne ut den deriverte av signalet. Det vil isåfall fungere som et slags høypassfilter.

Plott signalet før og etter filtrering, sjekk at ting ble riktig, dokumenter resultatene. Sjekk gjerne også i FFT-rommet.

2. Finn pulsen. :-) Tips til steg basert på autokorrelasjon kommer under. FFT kan også brukes.

- (a) Autokorreler signalet for å finne periodisitetene av signalet ditt.

Tips: Du kan autokorrelere et signal med seg selv i MATLAB med `[autocorrelation, lags] = xcorr(signal, length(signal), 'coeff')`. Plotter du det som `plot(lags, autocorrelation)`, vil første topp være på $x = 0$, siden signalet korrelert med seg selv vil gi høyest respons. Har du et tydelig pulssignal, bør neste topp svare til perioden til pulsen. Om pulsen er rimelig jevn og du ikke har for mye artefakter i signalet ditt, vil også de neste toppene være relatert til pulsen i signalet ditt.

Plott autokorrelasjonen, studér autokorrelasjonen. Når du manuelt finner posisjon for første topp, sjekk at denne faktisk stemmer med det du forventer for puls, og det du kan telle ut av antall pulser i signalet.

- (b) Automatisk finn alle signifikante topper i autokorrelasjonsplottet.

Tips: MATLAB-funksjonen `find_peaks(...)` kan brukes til dette, men er autokorrelasjonen veldig støyete kommer den til å finne posisjon for alle støypeakene dine. Du bør derfor få filtrert det på en eller annen måte, så du kun får ut posisjon for de faktiske pulstoppene på signalet. Ett tips er å bruke smoothing splines til dette (slå opp `csaps`), med en fornuftig smoothing factor, ellers kan det være båndpassfilteret ditt gjør jobben. **Sjekk, plott opp, verifiser metoden du velger.**

- (c) Regn ut snittperioden ved å se på differansen mellom alle toppene dine, og bruk det til å gi et snittestimat på BPM-en. Du kan også få ut et standardavvik. Hva måler standardavviket?

3. Estimer SNR ved hjelp av SNR-scriptene.

Bruker du FFT, kan du også eksperimentere med å kjøre FFT-en på mindre tidsvindu, og se på pulsvariasjoner.