

Robottiohjelmoinnin harjoitustyö

Robodoge

Olli Pitkänen

011830007

ohjpitka@cs.helsinki.fi

11.01.2015

1 Robodogen kuvaus

Robodoge on robottivillakoira, jonka ydin on siinä, että se osaa tunnistaa omistajansa puhetta ja toteuttaa tunnistetut komennot. Robodoge kuuntelee, analysoi ja tallentaa kuulemansa komennot. Se osaa verrata kuulemaansa aiempiin puhekomentoihin ja valita muistiin tallennetuista komennoista parhaiten täsmäävän komennon ja toteuttaa sen. Robodoge osaa myös hylätä liian erilaisen komennon. Robodoge osaa haukkua, etenkin silloin, kun se ei ymmärrä omistajansa komentoa. Robodoge osaa esimerkiksi seuraavia komentoja: liiku eteen, liiku taakse, käänny vasempaan, käänny oikealle, pyöri ympäri, pysähdy ja lopeta. Muitakin komentoja voi opettaa tarpeen vaatiessa.

Robodogen kuulo voisi olla herkempi, se ei ole ihan otollisin puheen tunnistukseen, tai sitten pentu ei vain välitä omistajansa komennoista, kuten eivät aina tavallisetkaan koirat. Äänisensorin tarkkuus ei ole kovin hyvä, esim. paine-ero ei palaudu kovin nopeasti kovista äänistä. Äänisignaalit voidaan kuitenkin erottaa toisistaan perinpohjaisella analyysillä, valitsemalla komentoäänet sopivasti ja käyttämällä automaattista puheentunnistusalgoritmia. Robodoge osaa hyödyntää P. Mermelsteinin (1980) kehittämää menetelmää, jota kutsutaan Mel Frequency Cepstral Coefficient (MFCC):ksi.

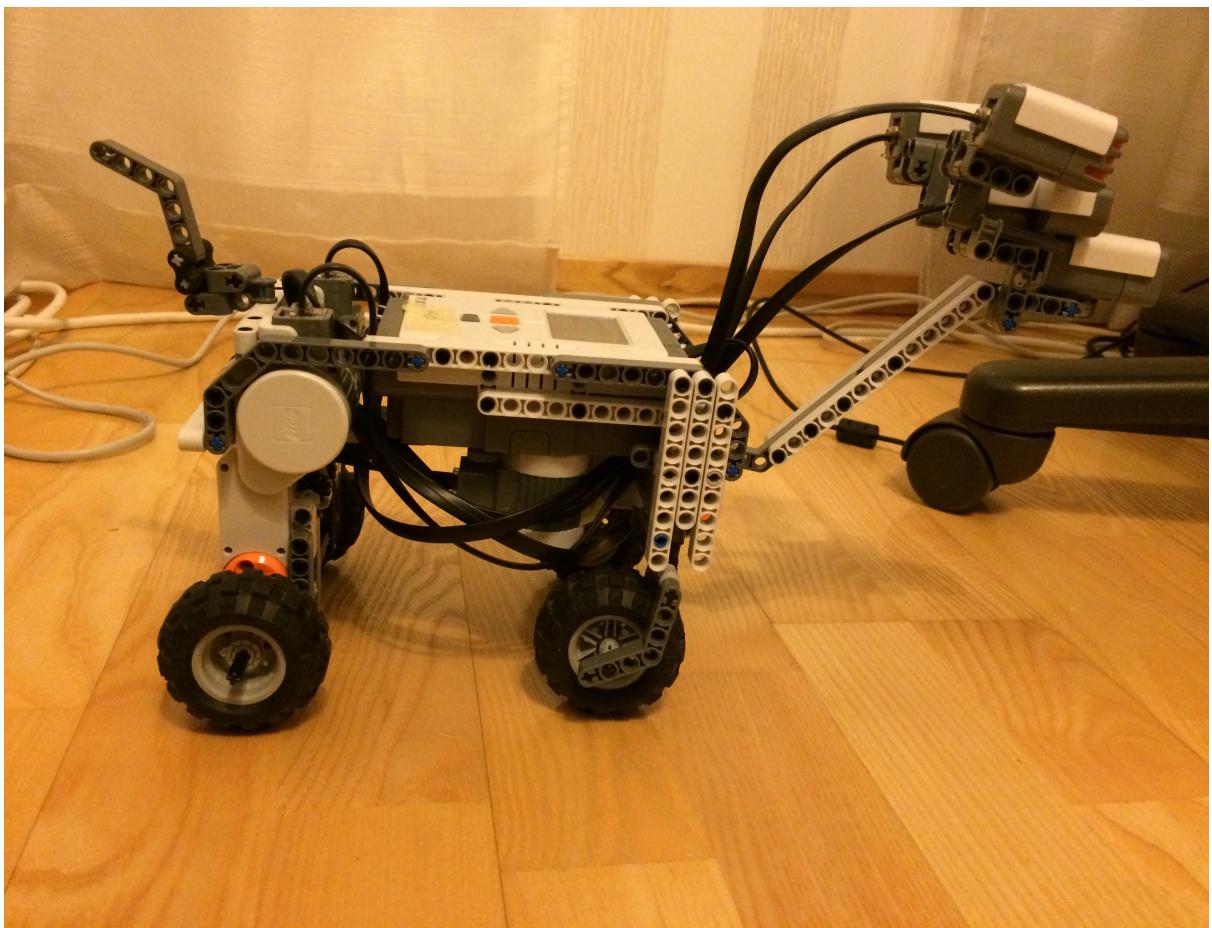
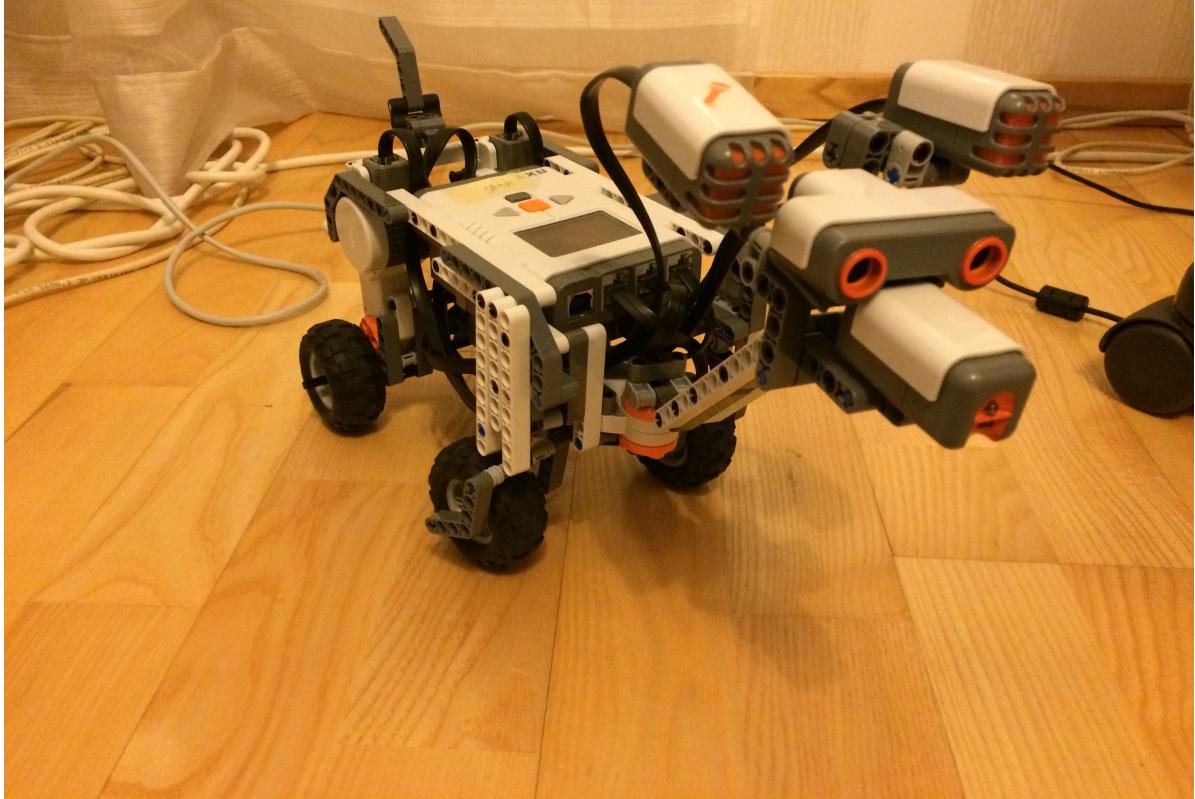
2 Robotin rakenne

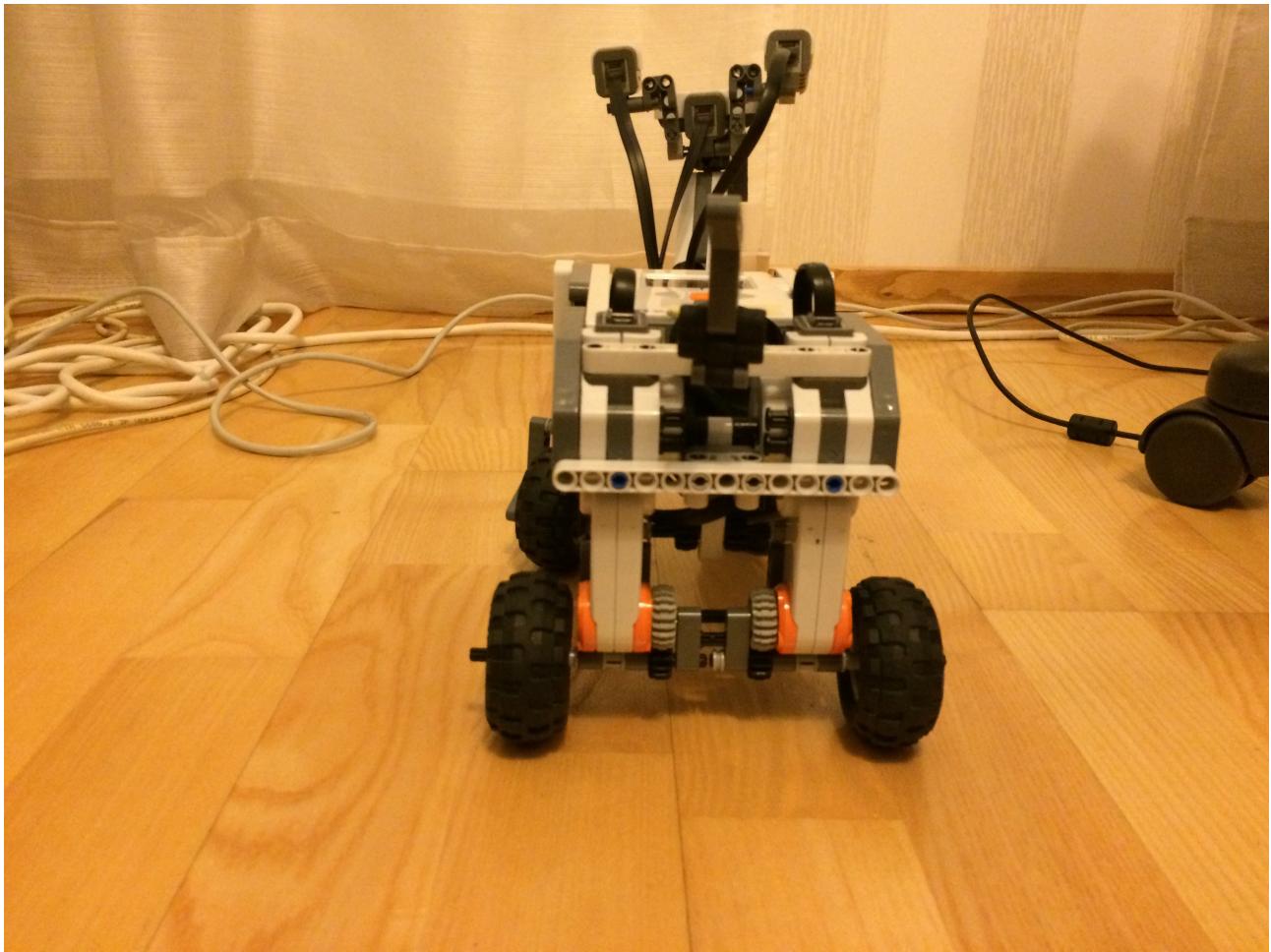
Robodogen muistuttaa etäisesti villakoiraa. Sen päät koostuu sensoreista siten, että äänisensorit ovat sen korvat, ultraäänisensori toimii silminä ja nenänä toimii valosensori. Dogen jalkojen tilalla on renkaat, takaa motorisoidut ja edestä käentyvät. Robodogen voi käntää kolmannen moottorin avulla kaulaa ja päätänsä rajoitetusti hiukan vasemmalle tai oikealle.

Rakennetta on tehty mahdollisimman tukevaksi, mutta kuitenkin ulkoisesti koiranpentua muistuttavaksi. Sensorit ovat sijoitettu muuta robottia korkeammalle päähän n.20 cm korkeuteen. Renkaiden ympärysmitta on 5,6 cm. Äänisensori voi mitata kolmella tavalla: raaka intensiteettiä välillä 1023 -0, missä 1023 on heikoin intensiteetti ja nolla vahvin, normalisoitua (0-100) äänen intensiteetti [dB mode], jossa kaikille taajuuksille annetaan yhtä suuri painoarvo, sekä normalisoitua (0-100) ihmiskuulon mukaan muokattua intensitettiä [dBA mode].

Maksimiintensiteetti, jota sensori pystyy mittamaan on 90 dB.

Alla kuvia robodogesta edestä, sivulta ja takaa.





3 Koodin rakenne

Koodi koostuu viidestä paketista. Aluksi pääpektti DogiPakcakage, joka sisältää pääohjelman luokan Robodoge ja muina luokkina SafeButtonListener, Converter, joka muuntaa tauluja muodosta toiseen, MSE, jonka tehtävänä on laskea keskineliövirheitä ym., sekä SpeechRecognition, joka kokoaa puheentunnistuksessa ja tallennuksessa tarvittavat metodit.

Toisena pakettina toimii DeviceLogics, joka sisältää laitteiden, kuten äänisensorin luokat ListenVoiceCommand, RecordVoiceCommand ja PlaySound. Näillä luokilla ohjataan sensorien ja kaittuminen toimintaa.

Kolmas paketti IOLogics sisältää luokat FileReader ja FileWriter, joissa on työkaluja puheentunnistustiedon lataamiseen ja tallentamiseen.

Neljäs ja ehkä mielenkiintoisin paketti on MFCC, joka sisältää puheentunnistuksessa käytettäviä algoritmeja kolmessa luokassa. Ensimmäinen luokka on DCT eli Discrete Cosine Transformation, toinen luokka on FFT eli Fast Fourier Transform ja kolmas MFCC eli Mel Frequency Cepstral Coefficients, jonka avulla selvitetään puheen tunnisteena käytettävät kertoimet. DCT- ja MFCC-luokkien laatija on Ganesh Tiwari. FFT-luokan laatija on Danny Su.

Viides paketti sisältää Lejos:n esimerkkiluokan PilotParams, jolla robodogen parametrit määritellään pysyvään tiedostoon.

4 Testaus

Robodogen kannalta oleellisinta oli testata sen kuulosensorien sekä puheentunnistusalgoritmin toimintaa tavoitteena saada äänikomennot erotettua toisistaan toistuvasti ja riittävän luotettavasti.

Testitapaus 1:

Laitoin robodogen kuuntelemaan raakaäänidataa tarkoituksesta selvittää onko kahdella eri puhekommennolla eroa äänen intensiteettikuvaajassa. Testi epäonnistui, vaikka eroja löytyi, mutta äänikomentojen toistettavuus ja luotettavuus ei onnistunut johtuen ympäristöstä, äänen kovuuden ja painon vaihtelusta, äänen rytmin vaihtelusta ja etäisyydestä. Ääni vaihtelee kovasti esimerkiksi mielentilan mukaan, sen kovuus, paino tai nopeus voi vihdella niin paljon, ettei sanoja tai lauseita pysty erottamaan Lego-äänisensorin mittaamalla raakadatalla. Testasin äänisignaalien eroja mittaamalla äänen intensiteettiä ajan funktiona ja vertaamalla toistettujen sanojen ja lauseiden intensiteetin keskineliövirhettä.

Testitapaus 2:

Äänen intensiteetin normalisoiminen kovimman mitatun äänenvoimakkuuden mukaan. Tavoitteena selvittää, oliko tällä tekijällä merkittävä vaikutusta puheentunnistukseen. Testi epäonnistui, koska muut äänen vaihtelun tekijät olivat merkittävämpiä.

Testitapaus 3:

Äänisignaalin muunto taajuuksiksi käyttämällä Fourier-muunnosta. Testi epäonnistui. Fourier-muunnoksella intensiteettidatasta sai filteröityä ulos merkittävimmät puheen taajuudet, mutta koska puhe vaihteli niin paljon, en löytänyt eroja toistojen välillä.

Testitapaus 4:

Puheentunnistusalgoritmin käyttö puheen tunnisteiden löytämiseksi. Testi onnistui. Sanojen ja lyhyiden lauseiden toistojen tunnisteet vastasivat toisiaan ja ne erosivat toisista sanoista ja lauseista. Käyttämällä puheentunnistusalgoritmia, puhekommennolla saatati tunnisteksi yksitoista Mel Frequency Cepstral -kerrointa. Kertoimia vertaamalla ja keskineliövirheen laskemalla sain selvitetyä miten paljon toistettavat sanat muistuttivat toisiaan ja miten paljon eri sanat eroavat toisistaan. Löysin merkittäviä ja toistettavia eroja äänianturin heikkoudesta huolimatta. Eroja pystyi tekemään selvemmiksi valikoimalla sopivia sanapareja.

Testaaminen oli aikaa vievää, joten muu robotin toiminnallisuus jäi vähemmälle huomiolle.

5 Rajoitukset ja tulevaisuus

Robodoge on vasta pentu, joka on kasvamassa ja oppimassa uusia toiminnalisuuksia. Robodogen

rajoitteena on sen liikkumistoiminnallisuuden puute, eikä se oikestaan näe testitapausta lukuunottamatta.

Tulevaisuudessa robodogelle voi lisätä liikkumistoiminnallisuutta komentojen mukaisesti. Ultraäänisensoria voi käyttää kohteiden väistämiseen ja omistajan sijainnin selvittämiseen esimerkiksi "tänne Doge"-kutsun toteuttamiseksi. Robodogesta voi kasvaa vielä oiva vahtikoira. Ultraääninäkö voi toimia liikutunnistimena tai lyhyen kantaman tutkana. Robodogesta voi kehittyä myös oikein hyvä kuuntelija, ehkäpä paremman mikrofonin lisääminen auttaisi puheentunnistuksessa ja veisi sen seuraavalle tarkkuustasolle.

Äänikomentojen analysointityökalut voisivat tulla tarpeeseen ja säädäisivät paljon aikaa, jos komentoja tulee paljon. Pienellä parannuksella esim. keskineliövirheet voi laskea rsitiin kaikkien nauhoitettujen komentojen kanssa kerralla, eikä pelkästään yhtä ääntä kerrallaan vertaamalla.

6 Käyttöohje

Robotin kokoamisen jälkeen aja build.xml, kun robotti on kytketty USB-johdolla koneeseen.

1. Ohjelma käynnistyy ja tervehти englannin kielisillä ohjeilla. Puheentunnistustoiminnallisuus vaatii, että puhetta pitää ensin nauhoittaa, analysoida ja tallentaa. Tämä onnistuu painamalla kerran Enter-nappia ja kollmen sekunnin sisällä pitämällä Oikea-nappi pohjassa. Tämä käynnistää puhekomentojen nauhoituksen. Voit nauhoittaa niin monta äänikomentoa kuin muistia riittää. Ohjelma näyttää LCD-näytössä jäljellä olevan muistin määrän.
2. Toimivan puheentunnistuksen takaamiseksi tallennetut ääninäytteet tulee analysoida. Ääninäytötä tallentuvat NXT:n flash-levylle tiedostoon nimellä <sampleIDs.txt>. Tiedosto sisältää rivieroteltuna, Double-arvoina MFC-kertoimet siten, että jokainen 11 numeron kokonaisuus vastaa puhumaasi komentoa. Kannattaa kirjoittaa puhekomennot etukäteen ylös analysoinnin ja toistettavuuden helpottamiseksi. Tiedoston voi ladata koneelle esim. Lejos:n NXT-browserin avulla, kun robodoge on kytketty koneeseen USB-jodolla. Ohjelmassa on valmiina MFC-kertoimien keskineliövirheiden lasku. Samalla voi analysoida joukon äänikomentoja vertaamalla sitä yhteen valittuun referenssiääneen.
3. Kun analyysi näyttää, että erilaiset äänet on löydetty, ne pitää ladata <commandIDs.txt>-nimisenä tiedostona NXT:lle. Tiedosto tulisi olla rivieroteltu ja Double-lukuja sisältävä siten, että jokaista komentoa vastaa 11 MFC-kertoimen joukko.
4. Äänikomentoja pääsee kokeilemaan kun käynnistää ohjelman uudestaan ja painaa kerran Enter:ä. LCD-näytölle tulee toimintaohjeita. Äänikomennot laukaisevat toimintoja, joita tämän päivän versioon on ohjelmoitu seitsemän. Laajennus on helppoa.

7 Lähdetuote

P. Mermelstein (1976), "Distance measures for speech recognition, psychological and instrumental," in *Pattern Recognition and Artificial Intelligence*, C. H. Chen, Ed., pp. 374–388. Academic, New

York.

S.B. Davis, and P. Mermelstein (1980), "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences," in *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(4), pp. 357–366.

<https://code.google.com/p/speech-recognition-java-hidden-markov-model-vq-mfcc/source/browse/trunk/SpeechRecognitionHMM/src/org/?r=3#org%2Fioe%2Ftprsa%2Faudio%2Ffeature>, Ganesh Tiwari, Danny Su, 11.1.2015