Helsingin yliopisto
Tietojenkäsittelytieteen laitos

Cum laude laboratorio
Ohjelmistotuotantoprojekti Nobel

KÄYTTÖOHJE

Ilkka Forsblom, Risto-Matti Kääriäinen, Jussi Lahti,
Maarit Saaresto & Jarkko Salminen

versio 1.0

Ohjaaja: **Teemu Kurppa**
Tilaaja: **Tapio Elomaa**
jätetty: 3.5.2001

Sisältö

1 Ohjelmiston tarkoitus

Ohjelmisto on erään suunnistus- ja paikanmääritysmenetelmän käytännön toteutus laitoksen Nomad Super Scout II -robotille. Toteutettu menetelmä pohjautuu ennalta annettuun, ympäristöä kuvaavaan karttaan, jonka perusteella voidaan suunnistaa annettuun maalipisteeseen, olettaen että tiedetään nykyinen sijainti. Nykyinen sijainti puolestaan selvitetään ottamalla kuvia ympäristöstä, tunnistamalla kuvista ennaltamääriteltyjä maamerkkejä ja laskemalla niiden perusteella robotin sijainti ns. triangulointimenetelmällä. Lisäksi robotin odometriatietojen perusteella voidaan pysyä ajan tasalla sijainnista, kunhan sijainti on aikaisemmin selvitetty maamerkkien perusteella. Odometria kuitenkin on varsin altis erilaisille kumuloituville virheille, joten paikanmääritys pitää pitkillä matkoilla suorittaa tietyin väliajoin oikeellisen sijaintitiedon ylläpitämiseksi.

Toteutettu suunnistus- ja paikanmääritysmenetelmä on idealtaan varsin yksinkertainen, ja sen pitäisi olla oikein toteutettuna ja käytettynä varsin luotettava. Ajatuksena onkin ollut luoda laitoksen tutkijoille toimiva ja luotettava ohjelmisto, johon muita menetelmiä voi verrata.

Ohjelmistolle annetaan syötteenä kartta, maalipiste, maamerkkien sijainnit ja tukivektorikoneet, joilla maamerkit voi tunnistaa. Ohjelmisto selvittää robotin sijainnin, laskee reitin maalipisteeseen, ja liikuttaa robottia laskettua reittiä pitkin maaliin, vältellen törmäystä esteisiin, ja tarkistaen robotin sijainnin tietyin väliajoin. Ohjelman suoritus päättyy robotin selviydyttyä maalipisteeseen.

2 Ohjelmiston laitteistovaatimukset

Ohjelmisto on suunniteltu ajettavaksi laitoksen Nomad Super Scout II -robotissa. Robottia ohjaa 233 Mhz Intel Pentium MMX -prosessorilla ja 256 MB muistilla varustettu tietokone. Kovalevyn kapasiteetti on 5,2 GB, minkä lisäksi robotti sisältää `Brooktree Bt848` kuvankaappauskortin ja Realtek 8029 –piirillä varustetun verkkokortin. Robotissa on kamera liitettynä pantilt–yksikköön, jonka avulla kameraa voi kääntää robotista riippumatta. Ohjelmistoa ajetaan robotin tietokoneessa radiolinkin kautta radiomodeemin avulla. Käyttöjärjestelmänä robotin tietokoneessa on TKTL:n Red Hat 6.1 -pohjainen CSL2 (Linux version 2.2.17).

3 Ohjelmiston käsittävät tiedostot

Asennettuna ohjelmisto koostuu kolmesta tiedostosta: ajettavasta ohjelmasta (nobel), kartan konvertoijatyökalusta (rmap), sekä käyttöohjeesta (tämä tiedosto).

4 Ohjelmiston ajaminen

4.1 Syötetiedostot

Ohjelma ottaa parametrina yhden tiedostonimen. Parametrina annettu tiedosto on ajotiedosto, joka sisältää muiden ohjelman parametrien ohella myös kolmen muun tiedoston nimet. Nämä tiedostot ovat karttatiedosto, maamerkkitiedosto, ja segmenttiväritiedosto.

4.1.1 Ajotiedosto

Ajotiedosto sisältää kahdenlaisia rivejä: kommenttirivejä ja asetusrivejä. Kommenttirivit alkavat prosenttimerkillä (%). Kommenttirivit ja tyhjät rivit ohjelma jättää huomioimatta. Asetusrivit ovat muotoa 'ASETUKSEN_NIMI = arvo', eli ensin on parametrin nimi, sitten yhtäsuuruusmerkki, ja sen jälkeen haluttu arvo parametrille. Välilyönnit ovat pakollisia yhtäsuuruusmerkin ympärillä, muuten ohjelma ei osaa tulkita riviä. Pituusyksikköparametrit käyttävät mittayksikkönä senttimetriä. Ajotiedoston parametrit on selitetty taulukossa 4.1.

4.1.2 Karttatiedosto

Karttatiedosto on muodoltaan ASCII-rasterikartta. Tiedostossa merkki X tarkoittaa kiellettyä aluetta, jolle robotti ei saa kulkea. Välilyönti tarkoittaa aluetta, jolla robotti saa kulkea (jos ei oteta huomioon ajotiedostossa määriteltyä marginaalia). Kartan leveydeksi tulee (pisimmän rivin sarakkeiden määrä)*SCALE senttimetriä, korkeudeksi (karttatiedoston rivien määrä)*SCALE senttimetriä.

Kartan origo on vasemmassa alakulmassa. Kaikki ohjelmalle parametreina annettavat koordinaatit ovat siis positiivisia. Kartan ulkopuolelle määritellyt koordinaatit aiheuttavat virheilmoituksen.

Halutun muotoisen kartan voi luoda joko tekstieditorilla naputtelemalla ja tallentamalla ASCII-muodossa tai piirtämällä robotin simulaattoriohjelmalla, tallentamalla, ja konvertoimalla tallennetun kartan rmap -työkalulla.

Rmap (Raster MAP) on apuohjema, joka renderoi robotin simulaattoriohjelmalla (Nserver) piirretyn polygonikartan rasterimuotoon. Rmapia käytetään antamalla sille kolme parametria: syötetiedosto (Nserverillä piirretty kartta), tulostiedosto (luotavan rasterikartan tiedostonimi) ja mittakaava. esim.

rmap kartta.map kartta.asc 120

Tämä luo rasterikartan, jonka tiedostonimi on kartta.asc. Mittakaavan tulkinta on seuraava: se ilmoittaa, kuinka monta Nserverin kartan yksikköä mahtuu yhteen tuloskartan yksikköön. Esimerkiksi edellä 120 polygonikartan yksikköä muodostaisi yhden rasterikartan solun. Tällöin 1200x480-kokoinen objekti polygonikartalla kattaisi rasterikartalla 10x4 solua.

Parametri	Arvo	Selitys
MAP_FILE	tiedostonimi	karttatiedoston nimi
LANDMARK_FILE	tiedostonimi	maamerkkitiedoston nimi
LOG_FILE	tiedostonimi	lokitiedosto (jos tämänniminen tiedosto on jo olemassa, lisätään nimen loppuun (jos tiedostonimessä on piste, ennen ensimmäistä pistettä) juokseva numero, kunnes saadaan aikaan tiedosto, jota ei vielä ollut olemassa)
SCALE	desimaaliluku	Kartan mittakaava (yksi ruutu on SCALE * SCALE kokoinen). Oltava vähintään yksi senttimetri.
GOAL_X	desimaaliluku	maalipisteen X-koordinaatti kartalla
GOAL_Y	desimaaliluku	maalipisteen Y-koordinaatti kartalla
OUTPUT_MAP, OUTPUT_MOV, OUTPUT_CON, OUTPUT_LOC	0, 1, 2, tai 3	Kuinka paljon ohjelma tulostaa ruudulle eri osajärjestelmien toiminnasta tietoja. Osajärjestelmät ovat seuraavat: MAP tarkoittaa reitinlaskentaosajärjestelmää, MOV liikkumisosajärjestelmää, CON päätöksentekoosajärjestelmää ja LOC paikanmääritysosajärjestelmää. Tulostustasoista 0 tarkoittaa ei tulostusta, 1 vähäistä tulostusta, 2 runsasta tulostusta ja 3 kaiken teknisenkin tiedon tulostamista.
LOG_MAP, LOG_MOV, LOG_CON, LOG_LOC	0, 1, 2 tai 3	Kuten edellä, mutta määrittelee, miten paljon tulostetaan lokitiedostoon.
WRITE_UYV, WRITE_SEGMENTS	'y' tai 'n'	Tallennetaanko kameran ottamat kuvat (uyv-formaatissa) ja segmentit (ppm-formaatissa) levylle. Parametri 'y' tarkoittaa kummassakin tapauksessa sitä, että kuvat tai segmentit tallennetaan, muussa tapauksessa ei tallenneta.
SONAR_RANGE_0, SONAR_RANGE_1, ..., SONAR_RANGE_15	desimaaliluku	Määrittelee jokaiselle sonarille erikseen etäisyyden, jota lähempänä havaittu este aiheuttaa robotin pysähtymisen. Sonarit on numeroitu kulkusuunnasta lähtien, kasvaen vastapäivään (robotti voi kulkea etu- ja takaperin).
MARGIN	desimaaliluku	Turvamarginaalin koko. Kartalle merkittyjen kiellettyjen alueiden ympärillä on turvamarginaali, jolta robotti pyrkii pysymään poissa.
ETAP_LENGTH	kokonaisluku	Kuinka pitkän matkan robotti voi kulkea, kunnes se yrittää tarkistaa sijaintinsa maamerkkien avulla.
RANDOMVECTOR_COUNT	kokonaisluku	kuinka monta kertaa suoritetaan satunnaisliikkuminen ja paikanmäärityksen uudelleenyritys, jos paikanmäärityksen on onnistuttava
RANDOMVECTOR_LENGTH	kokonaisluku	kuinka pitkän matkan robotti liikkuu suorittaessaan satunnaisliikkumisen
WAIT	kokonaisluku	aika (sekunneissa), jonka jälkeen yritetään jatkaa matkaa, jos robotti on pysähtynyt havaittuaan yllättävän esteen

Taulukko 4.1: Ajotiedoston parametrit

4.1.3 Maamerkkitiedosto

Maamerkkitiedosto sisältää rivejä, jotka ovat muotoa ‘letter_X.svm 69 79‘. Ensimmäinen tieto rivillä on maamerkin tunnistavan tukivektorikoneen (tiedosto)nimi, seuraavana on maamerkin sijainti kartalla (X– ja Y-koordinaatti senttimetreissä). Jos tiedostossa määritellään samanniminen maamerkki kahteen kertaan eri karttapisteissä, ei maamerkkiä hyväksytä eikä siten tunnisteta. Huomioi, että tukivektorikoneita etsitään ohjelman käynnistyshakemiston alihakemistosta SVM.

4.1.4 Segmenttiväritiedosto

Segmentoinnin väritiedostossa määritellään värirajat, joiden väliin jäävät alueet kamerakuvissa hyväksytään maamerkkitunnistusta varten. Segmenttiväritiedosto on seuraavaa muotoa:

[Colors] (200, 0, 0) 0.000000 Punainen

[Thresholds] (80:230, 60:120, 150:220)

Tässä [Colors]-asetus määrittelee, millä värillä hyväksytyt segmentit piirretään ohjelman segmenttitulosteisiin. Tällä asetuksella ei ole vaikutusta segmenttien tunnistamiseen. Asetuksessa “`( R, G, B) x Punainen`” R, G ja B ovat tulostusvärin arvot RGB-väriavaruudessa. Luku x ja merkkijonotunniste ovat ohjelmistossa käyttämättömät.

[Thresholds]-asetus määrittelee segmentoinnin värirajat YUV-väriavaruudessa. Asetuksessa “$(Y_{MIN} : Y_{MAX}, U_{MIN} : U_{MAX}, V_{MIN} : V_{MAX})$” Y_{MIN} on värin Y-komponentin minimiarvo ja Y_{MAX} sen maksimiarvo, samoin vastaavasti U- ja V-komponenteille. Segmentoinnissa ts. hyväksytään sellaiset pikselit, joiden YUV-arvo (Y, U, V) toteuttaa ehdon $\{Y_{MIN} \leq Y \leq Y_{MAX} \ \wedge \ U_{MIN} \leq U \leq U_{MAX} \ \wedge \ V_{MIN} \leq V \leq V_{MAX}\}$.

4.2 Ohjelmiston käynnistäminen

Ohjelma käynnistetään antamalla komento ‘nobel ajotiedosto‘, jossa ajotiedosto ohjelman lukema ajotiedosto.

4.3 Ohjelmiston pysäyttäminen

Ohjelman voi lopettaa kirjoittamalla q ja <ENTER> tai Q ja <ENTER>. Tällöin robotti ei pysähdy ja lopeta toimintaansa heti, vaan vasta kun se on liikkunut loppuun kesken olevan reitinpätkän.

Jos haluaa robotin pysähtyvän välittömästi, on ohjelmassa hätäpysäytys, jota voi käyttää antamalla ohjelmalle signaalin SIGINT (näppäimistöltä annettu keskeytys, saadaan aikaan näppäinyhdistelmällä CTRL+C). Tällöin robotti pysähtyy lähes välittömästi. Viivettä pysähtymiseen aiheuttaa verkkoyhteyden aiheuttama viive näppäinkomennon välittymiseen ohjelmalle. Lisäksi robotin jarrutusmatka riippuu robotin nopeudesta.

Jos ohjelmaa ei lopeteta näppäinkomennolla tai hätäpysäytyksellä, päättyy ajo joko robotin löydettyä maalipisteeseen tai mahdolliseen virhetilanteeseen.

5 Virhetilanteet

5.1 Virhetilanteet, joihin ohjelman suoritus päättyy

5.1.1 Input file x was not found.

Ohjelmalle komentoriviparametrina annettua ajotiedostoa ei löydetty tai sitä ei voitu lukea. Tarkista, että annoit oikean tiedostonimen ja että ohjelmalla on lukuoikeudet tiedostoon.

5.1.2 Incorrect settings file syntax: use = separated with spaces.

Ajotiedostossa on syntaksivirhe. Tarkista, että tiedosto on luvussa 4.1.1 kuvattua muotoa. Erityisesti kiinnitä huomiota, että parametri, yhtäsuuruusmerkki ja parametrin arvo on eroteltu välilyönnein.

5.1.3 Unrecognized setting identifier: setting_id

Ohjelma on yrittänyt lukea ajotiedostosta parametria "setting_id", jota siellä ei ole määritelty. Tarkista, että kaikki luvussa 4.1.1 mainitut parametrit on määritelty ajotiedostossa.

5.1.4 Initialization of BasicSegmentation failed!

Ohjelma ei pystynyt alustamaan segmentointikirjastoa. Syynä voi olla ajotiedostossa määritellyn segmenttiväritiedoston lukemisen epäonnistuminen tai väärä syntaksi.

5.1.5 The scale has to be at least 1 cm. (It was x cm.)

Ajotiedostossa on määritelty SCALE olevan alle yksi senttimetri. Yhtä senttimetriä pienempi SCALE ei ole käytännössä järkevä. Aseta ajotiedostossa SCALE vähintään yhdeksi senttimetriksi.

5.1.6 Mapfile x was not found.

Ohjelma ei löytänyt ajotiedostossa määriteltyä karttatiedostoa tai ei pystynyt lukemaan sitä. Tarkista, että karttatiedoston nimi on ajotiedostossa on oikein ja että ohjelmalla on lukuoikeudet kyseiseen tiedostoon. Huomaa, että jos tiedostonimi ei sisällä absoluuttista polkua, karttatiedostoa etsitään lähtien siitä hakemistosta, josta ohjelma käynnistettiin.

5.1.7 Landmarksfile x was not found.

Ohjelma ei löytänyt ajotiedostossa määriteltyä maamerkkitiedostoa tai ei pystynyt lukemaan sitä. Tarkista, että maamerkkitiedoston nimi on ajotiedostossa on oikein ja että ohjelmalla on lukuoikeudet kyseiseen tiedostoon.

5.1.8 SVM-file x was not found.

Ohjelma ei löytänyt maamerkkitiedostossa määriteltyä SVM-tiedostoa tai ei pystynyt lukemaan sitä. Tarkista, että SVM-tiedoston nimi on maamerkkitiedostossa oikein.

5.1.9 Destination x is invalid!

Maalipiste on joko määritelty olevan kielletyllä alueella (It's in the blocked area.) tai turvamarginaalissa (It's in the margin.), eikä robotti siten voi ikinä päästä maaliin. Tarkista, että piste ei sijaitse kartalle merkityllä kielletyllä alueella eikä marginaalissa (ota huomioon myös parametrit SCALE ja MARGIN).

5.1.10 Navigation failed. Localization is impossible.

Paikanmääritys epäonnistui huolimatta yrityksestä vaeltaa parempaan paikkaan. Tarkista, että ympäristössä on tarpeeksi maamerkkejä näkyvillä niin, että robotti voi ne havaita ja tunnistaa (jotta paikanmääritys onnistuisi, on robotin löydettävä ympäristöstään kolme maamerkkiä, joiden muodostaman kolmion sisällä robotti on). Tarkista myös, että kaikki maamerkit on merkitty maamerkkitiedostoon ja että käytetyt tukivektorikoneet ovat oikeaa muotoa.

5.1.11 There isn't any route to the destination.

Maalipisteeseen ei ole olemassa reittiä. Robotti on kartalla sellaisessa pisteessä, josta ei löydy reittiä maaliin. Tarkista, että maalipisteeseen voi kulkea robotin sijaintipaikasta kartan mukaan. Ongelmana saattaa olla myös liian suuri turvamarginaaliasetus (robotti ei suostu kulkemaan turvamarginaalin läpi).

5.1.12 Initialization of the signalhandler failed.

Signaalinkäsittelijän alustaminen epäonnistui. Signaalinkäsittelijää tarvitaan hätäpysäytyksen toteutuksessa.

5.1.13 System is unable to open stdin in non-blocking mode.

Standardisyötevirran avaaminen ei-blokkaavana epäonnistui, ts. ohjelma ei pysty lukemaan näppäimistöltä (tai sen korvikkeelta) lopetuskomentoa.

5.2 Muut virhetilanteet

5.2.1 Lopetuskomennon lukeminen ei toimi

Jos ohjelma ei pysty enää lukemaan standardisyötevirrasta merkkejä, annetaan varoitus "System is unable to read from stdin. Exit operation (q/Q) won't work.". Ohjelman suoritusta jatketaan, mutta lopetuskomento ei enää toimi. Tarvittaessa ohjelman voi vielä lopettaa hätäpysäytystä käyttämällä.

5.2.2 Robotti ei löydä tarpeeksi maamerkkejä

Jos robotti ei löydä/tunnista tarpeeksi maamerkkejä ympäristöstään, jatkaa robotti reitillään eteenpäin (jos mahdollista). Jos reitillä ei pystytä etenemään, tai robotti ei tiedä sijaintiaan (aloitustilanne), robotti vaeltaa lyhyen matkan suuntaan, jossa on sonareiden mukaan eniten tilaa, ja yrittää paikanmääritystä sen jälkeen uudelleen (asetus RANDOMVECTOR_COUNT määrää kuinka monta kertaa tätä yritetään).

5.2.3 Robotti havaitsee esteen

Jos robotti havaitsee esteen, suoritetaan paikanmääritys. Jos robotti on marginaalissa, liikkuu se suorinta reittiä pois marginaalista. Jos robotti ei ole marginaalissa, se odottaa asetuksen WAIT määräämän ajan ja yrittää sen jälkeen jatkaa matkaa reitillään.

5.2.4 Lokitiedoston luominen epäonnistuu

Jos ohjelma ei pysty luomaan lokitiedostoa, annetaan virheilmoitus "Could not open logfile. Check permissions.". Ohjelman suoritus jatkuu, mutta lokitiedostoon tulostaminen ei toimi. Ruudulle tulostetaan silti normaalisti asetusten OUTPUT_* mukaisesti. Tarkista, että ohjelmalla on oikeus luoda tiedostoja hakemistoon, josta se käynnistetään.