

Användarhandledning

Tävlingsbil

Gustaf Sjögemark

Version 1.1



Figur 1: Tävlingsbil med samtliga komponenter.

Version	Datum	Kompletterad	Utförd av	Granskad
1.1	2024-12-19	Kompletterad	Hela gruppen	Gustaf Sjögemark

Innehåll

1 Inledning	3
2 Raspberry Pi mainloop	4
3 Användargränssnitt	5

1 Inledning

Med hjälp av följande användarhandledning kan användaren starta tävlingsbilen och använda den med dess tillhörande användargränssnitt. Användaren kan då navigera bilen manuellt samt sätta på och stänga av autonom navigering genom en bana konstruerad enligt banspecifikation. Ett exempel på en sådan bana kan observeras i Figur 2 nedan.



Figur 2: Bild på produkten körandes en bana

För att starta och använda tävlingsbilen behöver två processer startas. En av dessa startas på Raspberry Pi:n som sitter på bilen medan den andra, som visar användargränssnittet, startas på en extern laptop. De båda processerna kommunicerar sedan över WiFi för att möjliggöra manuell och autonom körning, manipulering av parametrar samt visuell feedback.

2 Raspberry Pi mainloop

För att starta igång bilen måste man först koppla upp till den. För en ny användare kan det vara bra att koppla in Raspberry PI:n till en extern skärm med mus och tangentbord. Väl inloggad på PI:n måste man sätta igång hotspot, vilket görs genom att klicka på WiFi-symbolen i det högra hörnet. Där finns redan nätverket "balg" som startas igång genom att klicka på namnet. När WiFi nätverket är på blir nästa steg att koppla upp en dator till Raspberry PI:n. För att koppla upp datorn börjar man med att ansluta datorn till det trådlösa nätverket "balg". När datorn är uppkopplad till WiFi:t behöver man ta reda på datorns IPv4 adress. Genom att öppna ett terminalfönster på sin dator och skriva detta:

- på macOS: `ipconfig getifaddr en0`
- på Windows: `ipconfig`
- på Linux: `ifconfig`

OBS: Välj "IPv4"-adressen.

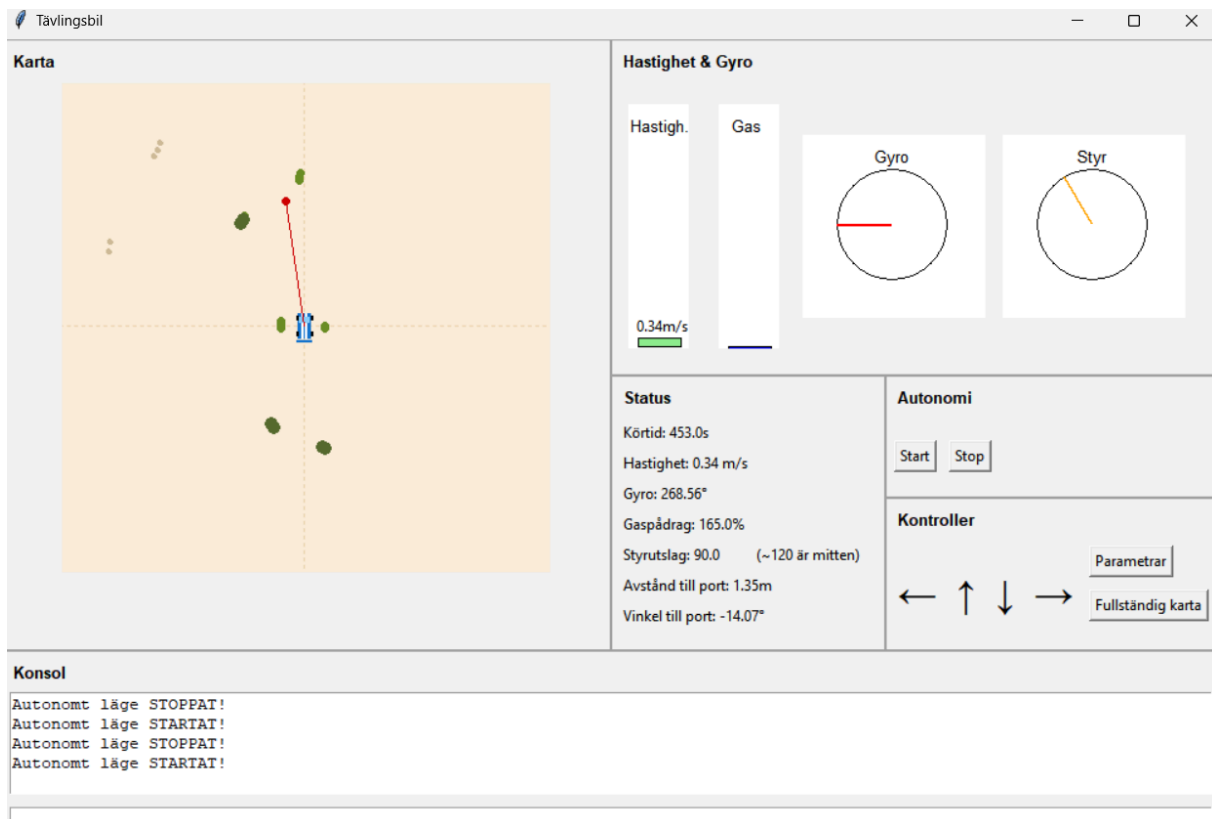
I samma terminalfönster kan man nu skriva "`ssh zootmaster@[IP]`", där IP byts ut mot den adress som dök upp i det tidigare steget. För att få SSH:a in behöver man skriva in lösenordet "balgot". Väl inne i Raspberry PI:n kan man skriva dessa kommandon för att starta bilen i terminalfönstret:

```
cd project_code/Kommunikationsmodul/  
./main.py
```

3 Användargränssnitt

Innan main-filen för GUI:t körs måste datorn anslutas till Raspberry Pi:ns WiFi. Det är även högst lämpligt att öppna config.py (som hittas i project_code/Användargränssnitt i Git-repot) för att bekanta sig med väsentliga inställningar och parametrar. Bland annat bör det säkerställas att värdet tillhörande nyckeln "URI" är en sträng med den korrekta IP-adressen till RPI:en följt av önskad port, exempelvis "ws://10.42.0.1:12307".

Användargränssnittet startas genom att köra main.py (även denna hittas i Användargränssnitt). Efter en kort stund bör kontakt etableras med RPI:n och GUI:t startas. Användaren bör då mötas av ett fönster likt det i Figur 3 nedan.



Figur 3: GUI:ts huvudsakliga fönster där den mesta av funktionaliteten finns.

Gränssnittet är uppdelat i flertalet "frames" med tillhörande rubrik i fetstilt. Nedan följer kortfattade beskrivningar av varje frame:

Karta Visualiserar de portar som finns i bilens närmaste omgivning. Målar även ut punkten som bilen siktar på, och en röd linje från bilen till denna punkt.

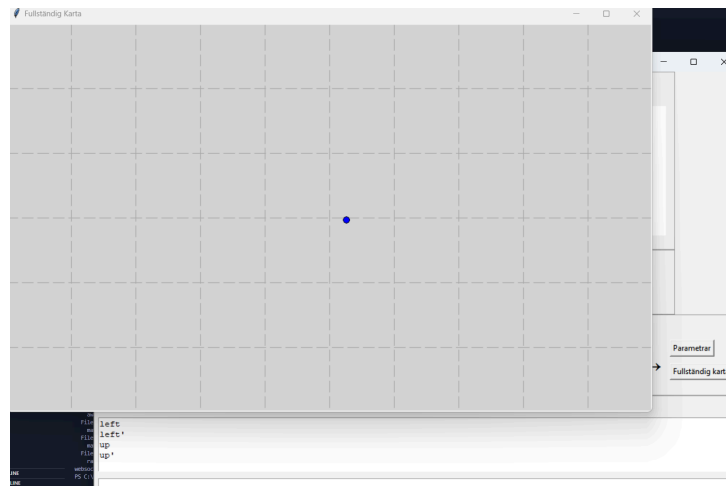
Hastighet & Gyro Visualiserar bilens hastighet, gaspådrag, gyro och styrutslag.

Status Visar diverse drift-information.

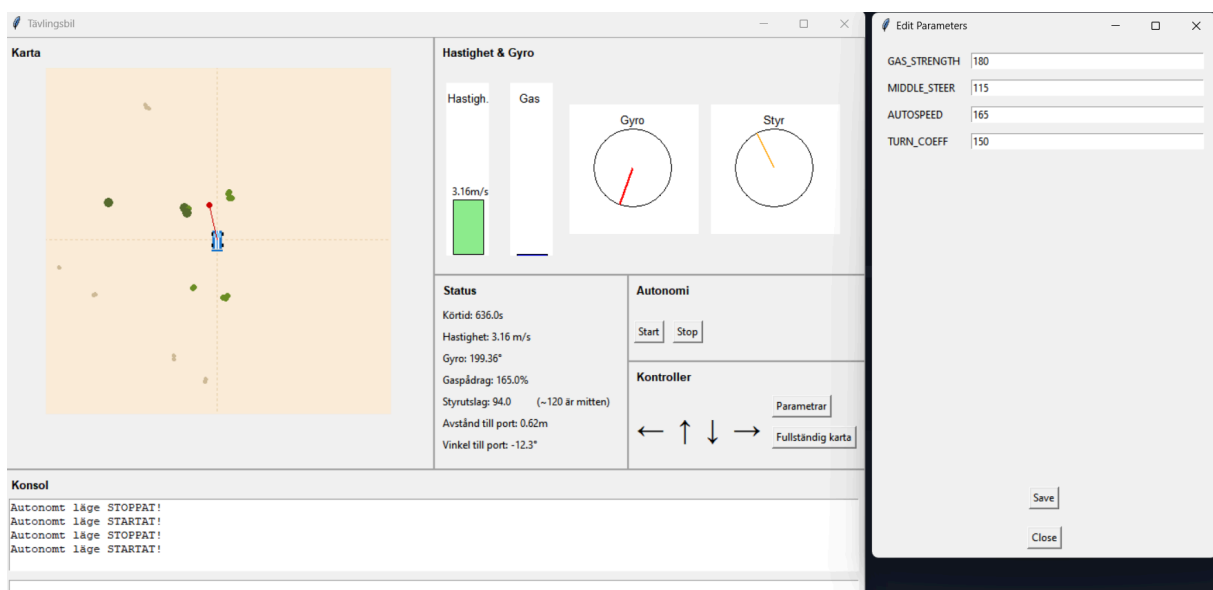
Autonomi Tillåter användaren att växla mellan autonom respektive manuell körning.

Konsol Text-kanal som bekräftar användares olika kommandon och knapptryck. Kan även visa emottagna meddelanden från RPI.

Kontroller Visualiserar användarens inputs vid manuell körning samt innehåller knapp som öppnar fönster för manipulering av parametrar. Det finns även ytterliggare en knapp som var tänkt att öppna ett fönster där en mer fullständig karta kunde visas, men denna har inte utvecklats tillfullo.(se figur 4 och figur 5 nedan)



Figur 4: Fönster som visar prototypen för fönstret där kartläggning av tävlingsbanan skulle ske.



Figur 5: GUI med fönster (till höger) som tillåter användaren att manipulera paramterar såsom gasstyrka. De första två 'parametrarna' avgör gaspådrag och styrning vid manuell körning. De två senare, AUTOSPEED och TURN_COEFF, gör motsvarande men för autonomt läge.

När användaren startar GUI:t är bilen i manUEllt läge, dvs användaren kan direkt, med piltangenterna, börja manövrera bilen. Önskas autonomt läge behöver användaren endast klicka på 'Start'-knappen under **Autonomi**-fliken. Bilen kommer då genast att börja leta efter portar köra mot den mest lämpliga. Under autonom körning kan användaren observera robotens beteende m.h.a kartan och drift-information i **Status** samt dess tillhörande grafiska element i **Hastighet & Gyro**. Vill användaren stoppa självkörningen kan de klicka på 'Stop'-knappen i samma flik, varpå bilen åter ställs i manUEllt läge. Parametrar för både manUEllt och autonomt läge kan justeras utan att användaren behöver starta om GUI:t på nytt. För att autonomi-parametrar ska ta effekt krävs det däremot en omstart av själva autonomin ('Stopp' -> 'Start'). Förändring av parametrar görs genom att klicka på 'Parametrar'-knappen, skriva in önskade värden, och sedan klicka 'Save'.