

CODEN: LUTFD2/(TFRT-5307)/1-030/(1984)

ETT EXEMPEL PÅ ROBOTPOSITIONERING MED HJÄLP
AV VIDEO KAMERA.

KENNETH JOHANSSON

INSTITUTIONEN FÖR REGLERTEKNIK
LUNDS TEKNISKA HÖGSKOLA

JANUARI 1984

TILLHÖR REFERENSBIBLIOTEKET

UTLANAS EJ

LUND INSTITUTE OF TECHNOLOGY DEPARTMENT OF AUTOMATIC CONTROL Box 725 S 220 07 Lund 7 Sweden	Document name	
	Master Thesis	
	Date of issue	
Author(s)	jan 1984	
	Document number	
	CODEN:LUTFD2/(TFRT-5307)/1-030/(1984)	
Kenneth Johansson	Supervisor	
	Lars Nielsen	
Sponsoring organization		
Title and subtitle		
Ett exempel på robotpositionering med hjälp av videokamera.		
(An example of robot positioning using a video camera.)		
Abstract		
<p>A two wheeled robot is controlled using image information. The problem is to reach an arbitrarily placed goal, from an arbitrary starting position. The hardware consists of a videocamera, image memory with frame grabber, VAX-11/780, DA-converter and the robot (a toy turtle) moving on the floor.</p> <p>The implementation consists of image processing, feature detection, correction of perspective, obstacle detection, route planning and control.</p>		
Key words		
Classification system and/or index terms (if any)		
Supplementary bibliographical information		
ISSN and key title		ISBN
Language	Number of pages	Recipient's notes
Swedish	30	
Security classification		

Distribution: The report may be ordered from the Department of Automatic Control or borrowed through the University Library 2, Box 1010, S-221 03 Lund, Sweden, Telex: 33248 lubbis lund.

INNEHALLSFÖRTECKNING

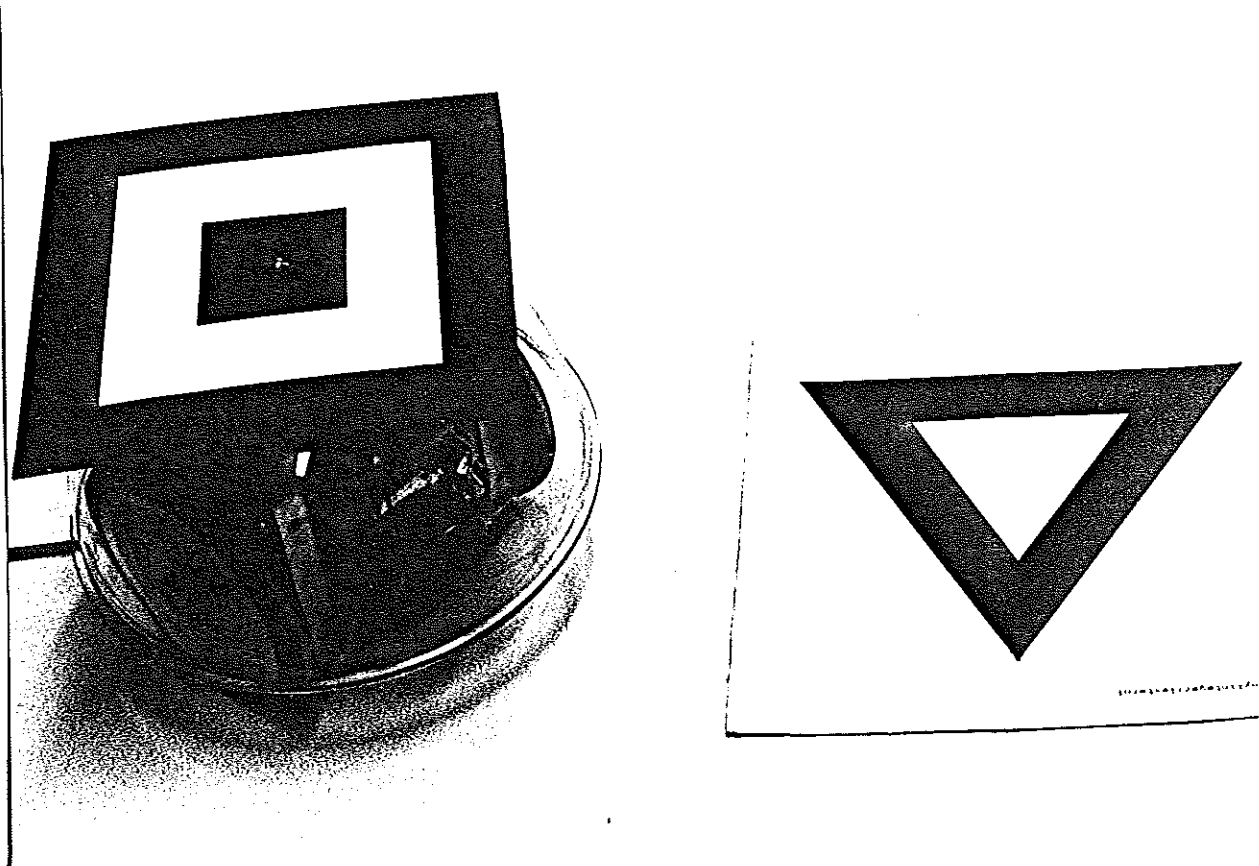
I. Inledning.....	1
II. Översikt.....	3
III. Definitioner.....	9
3.1 Enkla objekt.....	9
3.2 Sammansatta objekt.....	9
IV. Operationer på enkla objekt.....	10
4.1 Klassificering.....	10
4.2 Enkla objekts hörn.....	10
4.3 Tester på enkla objekt.....	12
V. Kantdetektering.....	14
VI. Operationer på sammansatta objekt.....	15
VII. Robotens val av väg till målet.....	16
7.1 Detektering av hinder.....	16
7.2 Robotens val av väg.....	17
7.3 Styrstrategi.....	18
IIIX. Geometriska korrektioner.....	19
8.1 Initialskede.....	19
8.2 Kamerans skalfaktor.....	21
8.3 Positionsbestämning av robot.....	21
IX. Programbeskrivning.....	25
Referenser.....	28

I. INLEDNING

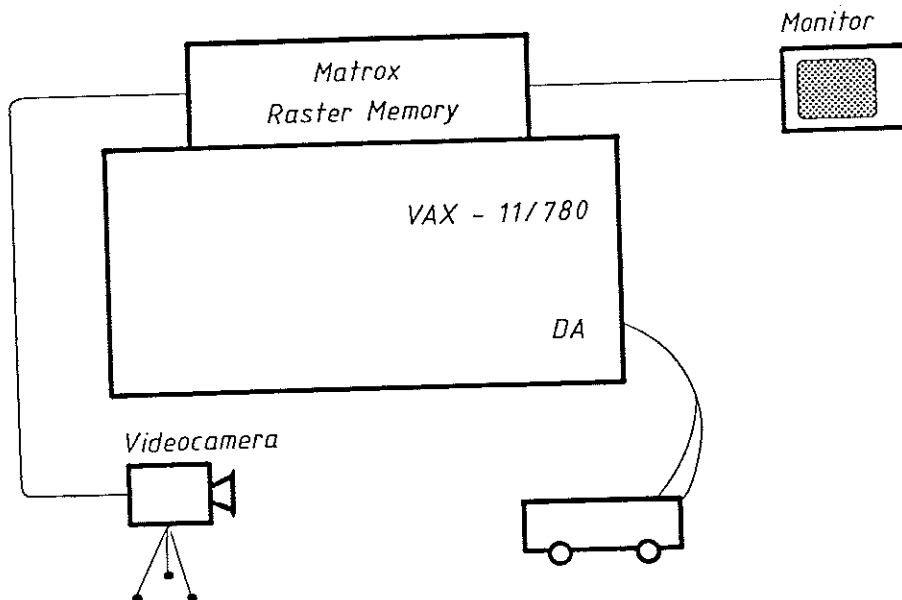
Examensarbetets målsättning är att styra en tvåhjulig robot, se figur 1, genom att använda informationen från bilder av roboten och dess omgivning. Systemet enligt figur 2 används. Uppgiften är dels att styra roboten direkt från en godtycklig utgångsposition, till ett likaså godtyckligt utplacerat mål. Dels att få roboten att undvika ett eventuellt hinder, som befinner sig mellan robotens utgångspunkt och målet. Under robotens gång fram till målet, måste dess riktning och körsträckans längd korrigeras, framförallt på grund av fel i riktnings- och avståndsbestämningen, men också för att robotens hjul slirar mot underlaget. Kameran är fixerad under den tid som roboten behöver för att ta sig till målet. Likaså ändrar inte målet och eventuella hinder sina lägen. Denna inskränkning beror främst på att bildbehandlingen tar avsevärd tid i anspråk, storleksordningen minuter. Arbetets inriktning har varit att ta fram en lösning till problemet och skriva ett fungerande program utan att försöka uppnå snabbhet.

Systemet består av datorn VAX-11/780 med ett Matrox bildminne, en standard svartvit videokamera och en monitor enligt figur 2. En bild är uppbyggd av 512*512 punkter, med en 256-nivåers (8-bitar) gråskala. Videodelen arbetar med en hastighet av 25-bilder/s. Programmet är skrivet i Pascal.

Kapitel 2 ger en översikt av vad programmet utför. I de närmast följande kapitlen, ges en mera ingående beskrivning av den aktuella lösningen till problemet. Enkla objekt och dess parametrar samt sammansatta objekt, definieras i kapitel 3. Sedan följer i kapitel 4 och 6 operationer på enkla respektive sammansatta objekt. Kapitel 5 beskriver kantdetektorn som används för att få fram den binära bilden. Hinderdetektering, robotens vägval och styrstrategin tas upp i kapitel 7. Det därpå följande kapitlet ger en beskrivning av de geometriska korrektionerna som fordras för att korrigera för robotens höjd och kameravinkeln.



Figur 1. Robot och mål.



Figur 2. Schematisk bild av systemet.

II. ÖVERSIKT

Kapitlet innehåller beskrivning av symbolerna som representerar mål och robot samt en översiktlig redogörelse för vad programmet utför.

Identifiering av robot och mål underlättas genom att förse dem med olika symboler, se figur 1. Roboten har en ljus, liksidig ram på mörk botten. Efter kantdetektering fås två likformiga figurer med gemensamt centrum. Målet representeras på liknande sätt av en mörk triangulär ram på ljus botten. Måltriangeln utgör ett referensmått, som används vid beräkning av avstånd mellan kamera och mål, samt vinkeln mellan kamera och triangelns plan.

Roboten och målet placeras ut godtyckligt framför kameran. Positionsbestämningen för objekten startar med att en bild läses in och kanter detekteras. En 'Bug' följer därefter sammanhängande områden och olika objekt detekteras. Målets och robotens symboler identifieras och dess positioner bestäms. Korrektion för robotens höjd beräknas.

Från denna första information kan inga slutsatser dras om robotens riktning. Därför körs den rakt fram en liten bit och en ny positionsbestämning för roboten görs. Nu kan dess vinkel och avstånd direkt till målet beräknas.

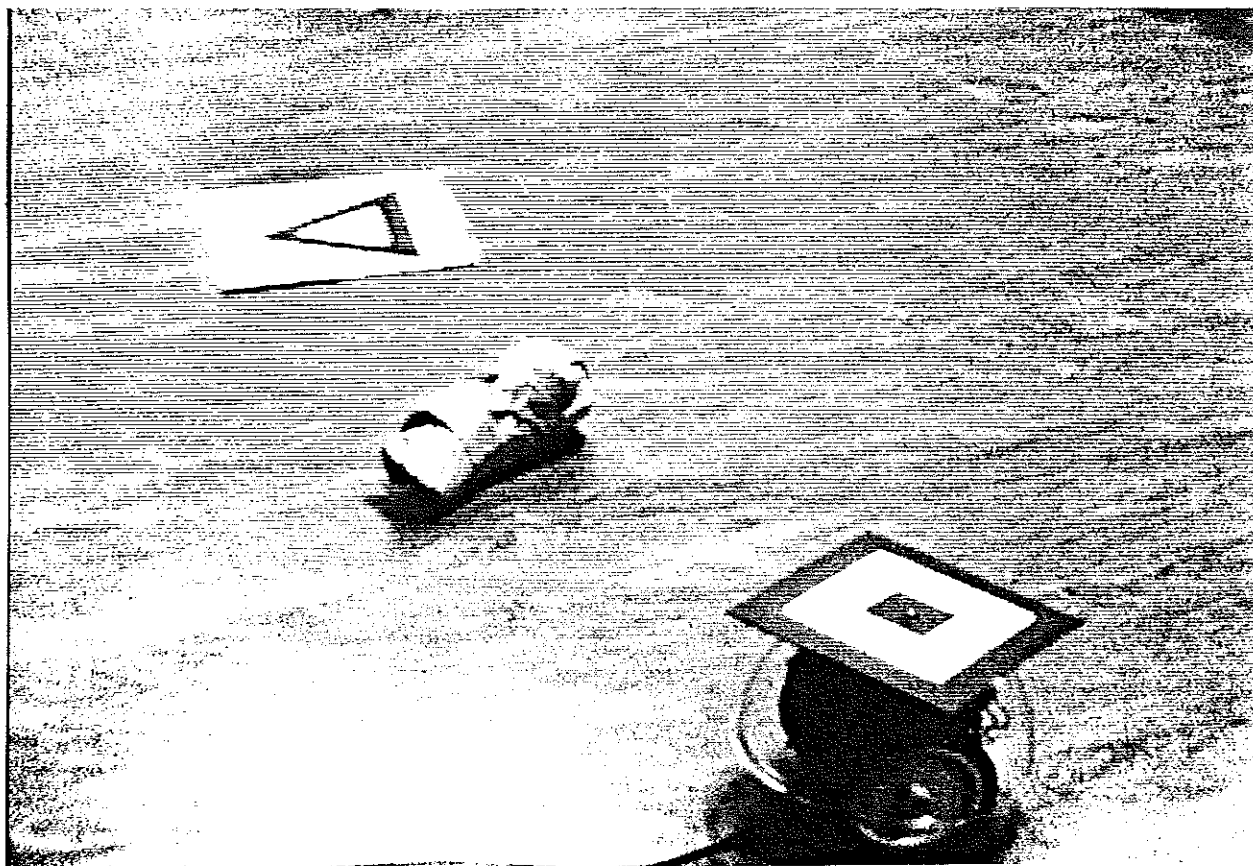
Sträckan mellan robot och mål analyseras för att upptäcka eventuella hinder. Om ett hinder detekteras, väljs en ny väg. Denna nya sträcka analyseras i sin tur och detta pågår tills dess att en hinderfri passage påträffas eller programmet misslyckas med att finna någon väg.

När en hinderfri sträcka har blivit funnen, beräknas robotens vridningsvinkel och avstånd till det första delmålet. Med delmål menas en punkt där vägen ändrar riktning eller måltriangelns position. Sträckan fram till delmålet delas ner i dellängder. Roboten vrider den beräknade vinkeln och kör fram en dellängd. Därefter tas en bild och den delbild behandlas, där roboten uppskattas ha hamnat. Robotens symboler lokaliseras i bilden och sedan korrigeras för dess höjd. Misslyckas lokaliseringen av roboten i delbilden, tas en ny bild och en något större delbild behandlas. Vridningsvinkeln och

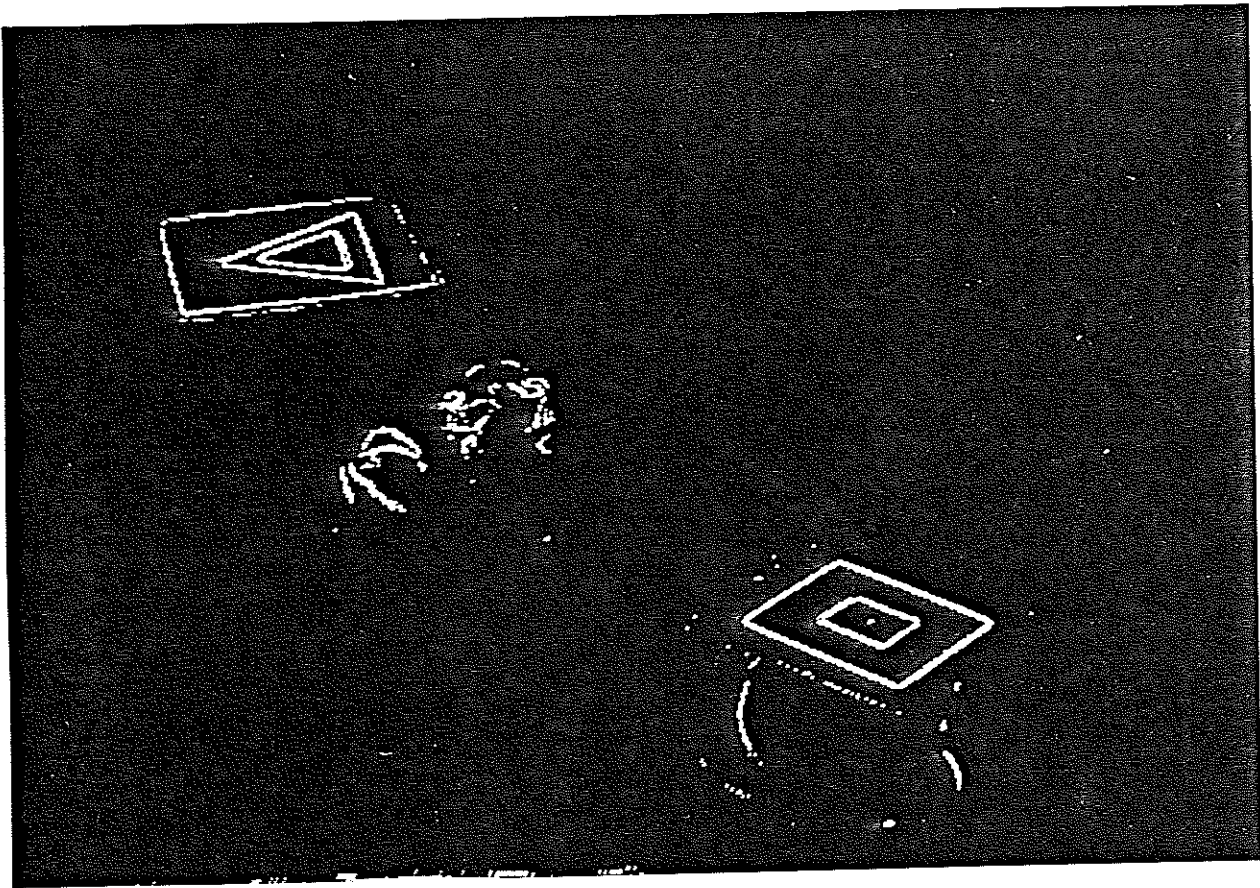
sträckan till delmålet beräknas åter och om roboten inte är framme, vrids den och körs fram en ny dellängd, osv. När roboten har nått fram till delmålet, får den ett nytt mål att köra fram till och detta upprepas tills den når fram till måltriangeln. Antalet delmål kan vara ett eller tre, beroende om något hinder detekterats eller ej.

Bildbehandlingen tar någon eller några minuter i anspråk, beroende på vilka operationer som ska utföras på bilden. Eftersom endast roboten ändrar sin position, minskas tiden för bildbehandlingen avsevärt, genom att endast en mindre del av bilden behöver behandlas. Storleken av delbilden vid robotens positionsbestämning, är beroende av dess avstånd till kameran.

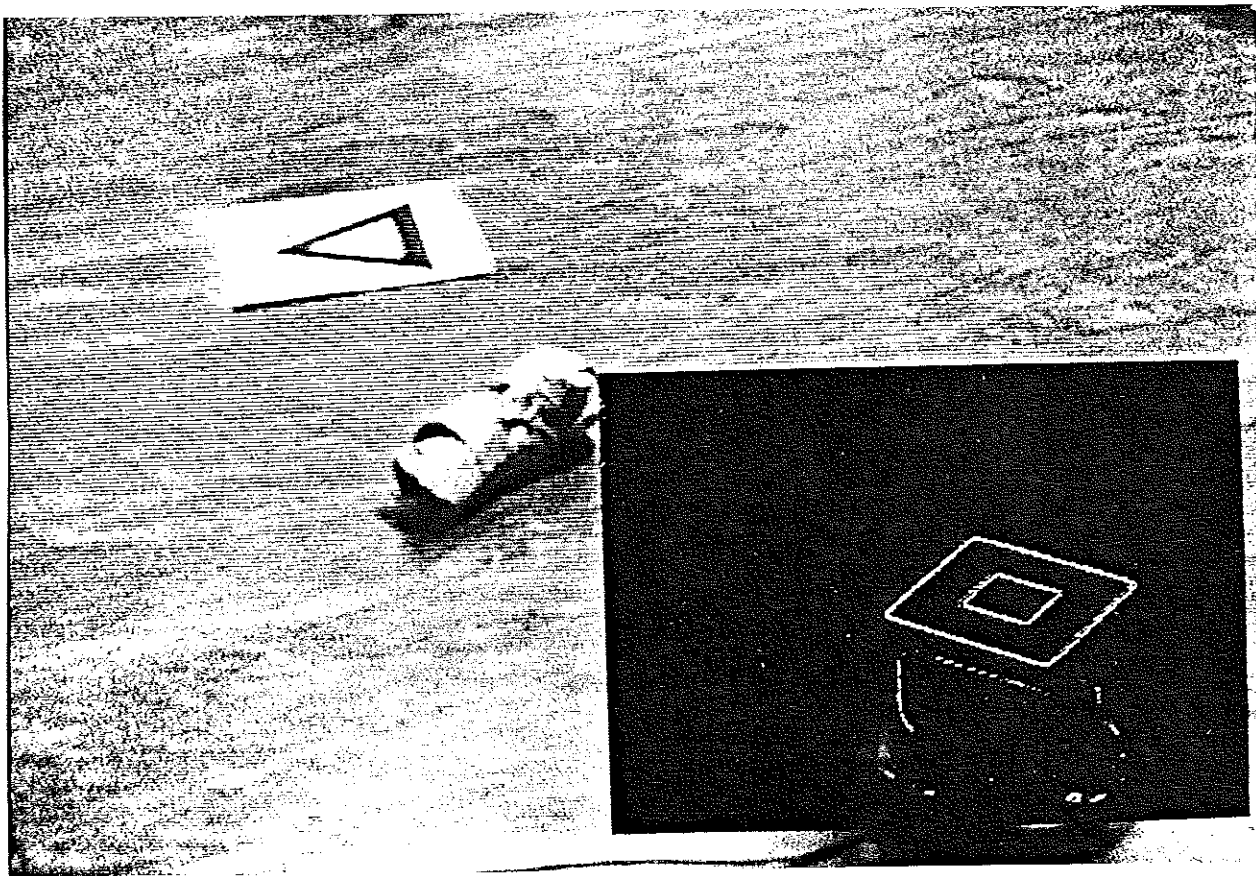
Nedan följer en bildserie av resultatet på monitorn under en körning.



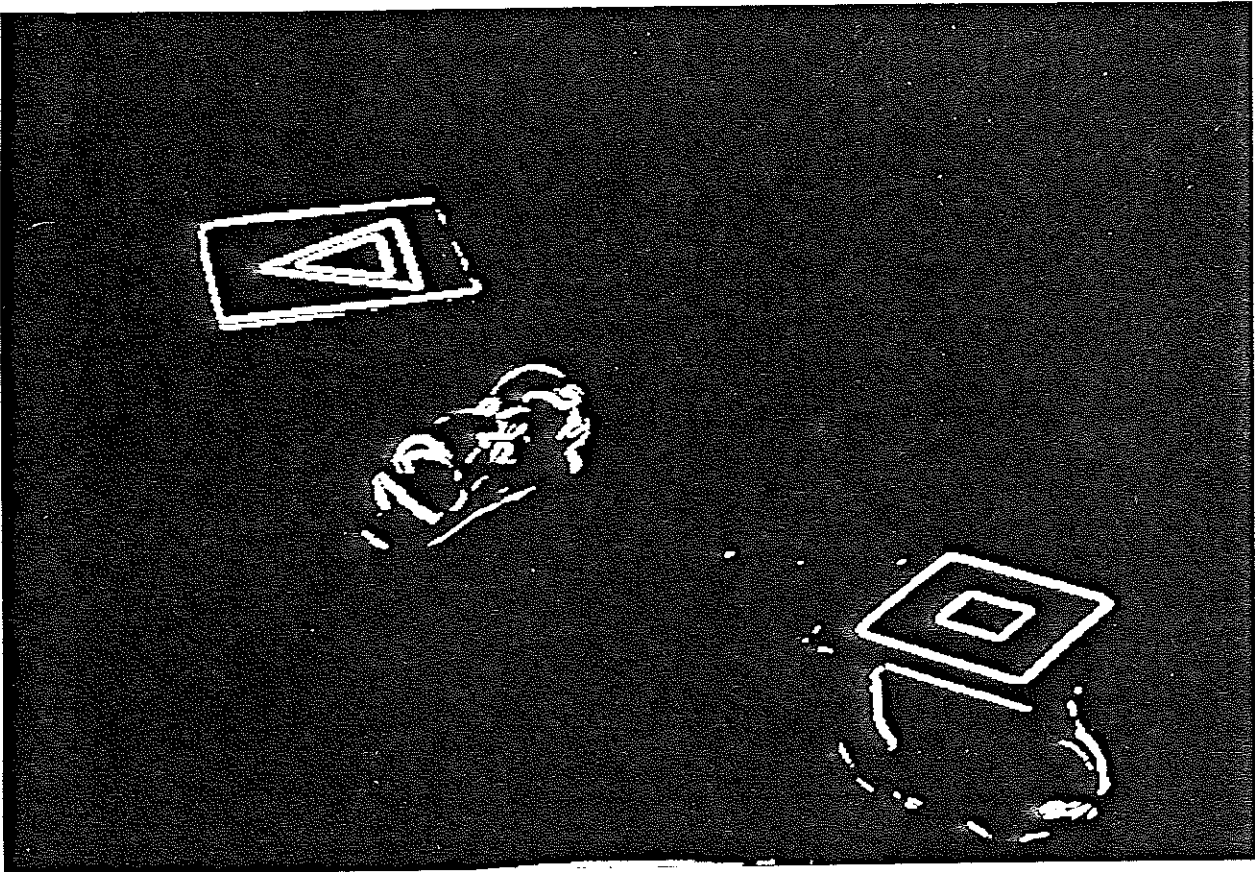
Utgångsposition.



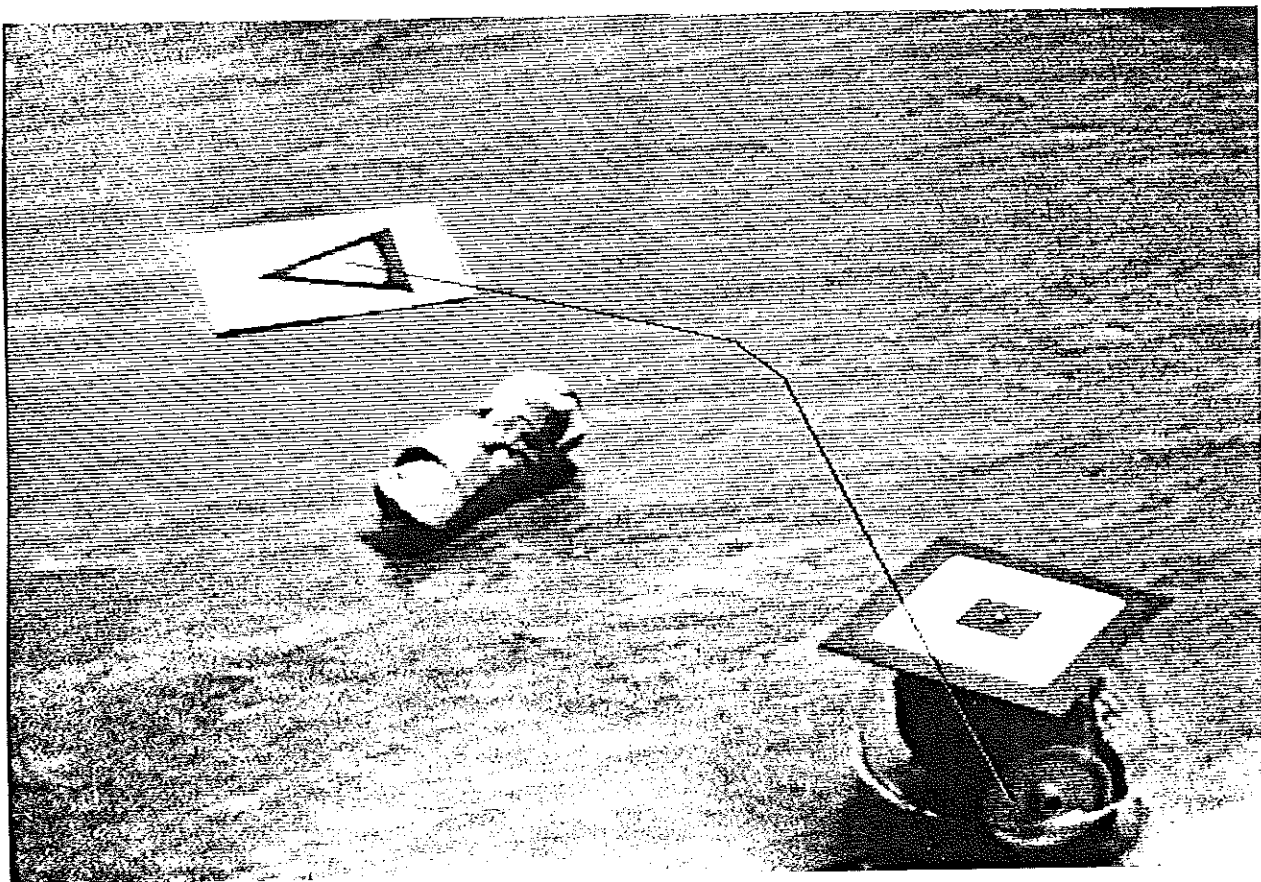
Resultat av kantdetektorn. Både robot och mål ska lokaliseras.



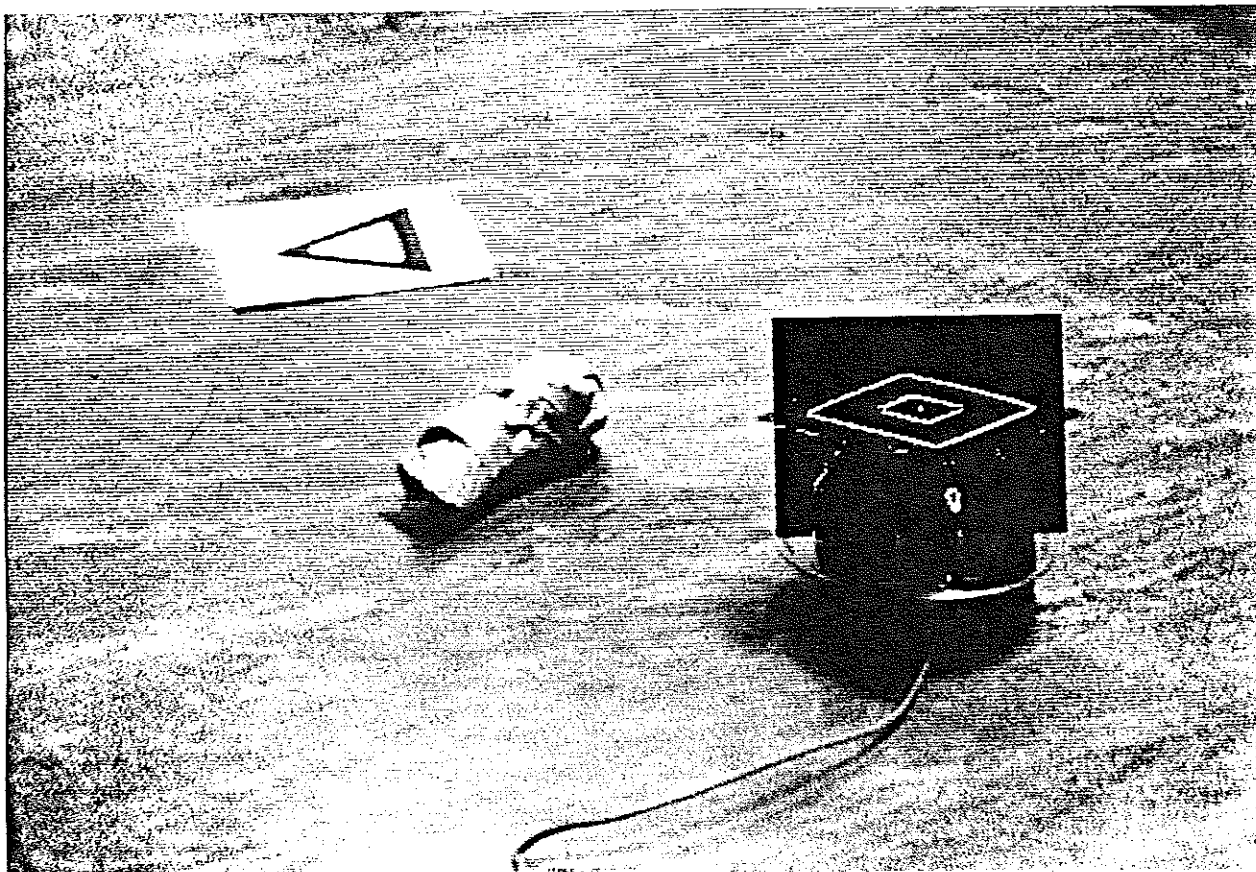
Roboten kör fram en kort fix sträcka för riktningsbestämning.



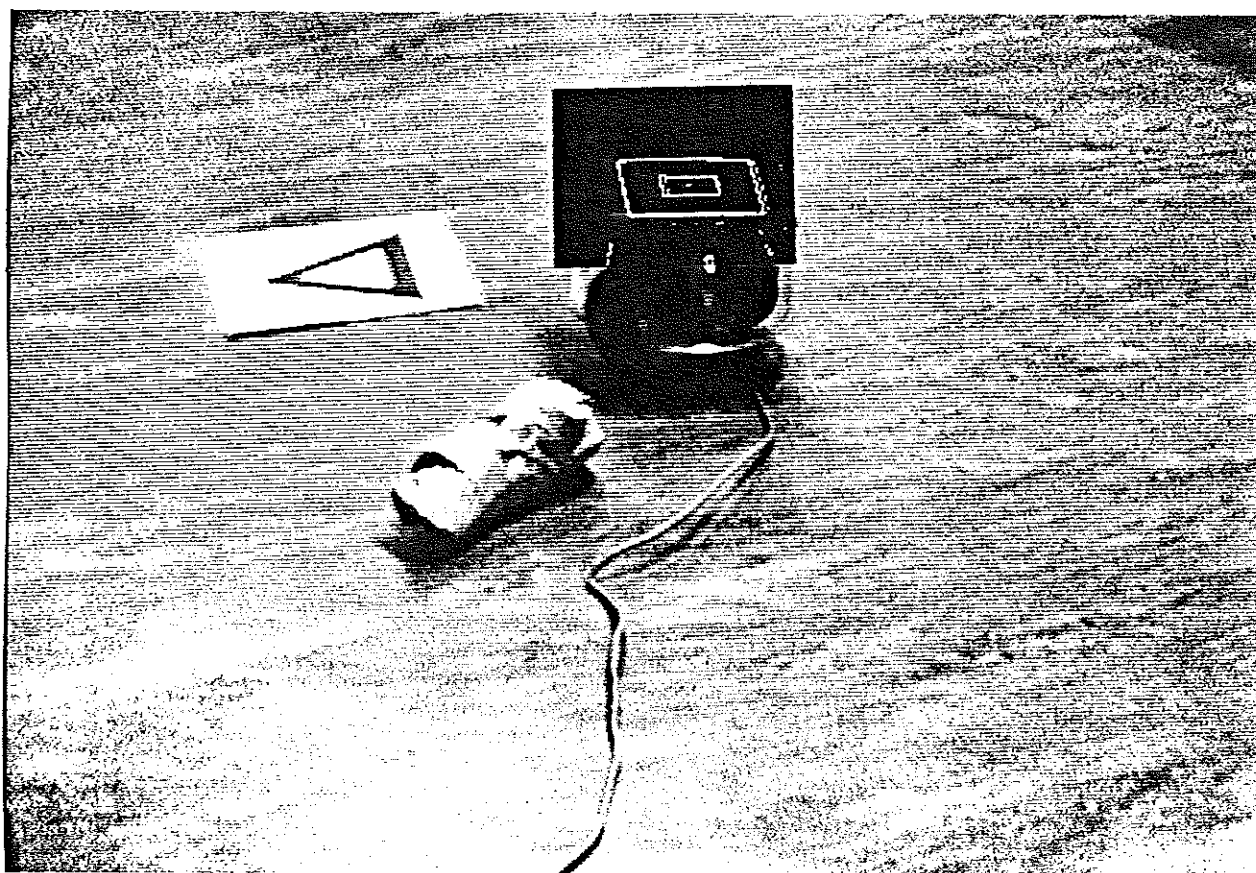
Hinderdetektering.



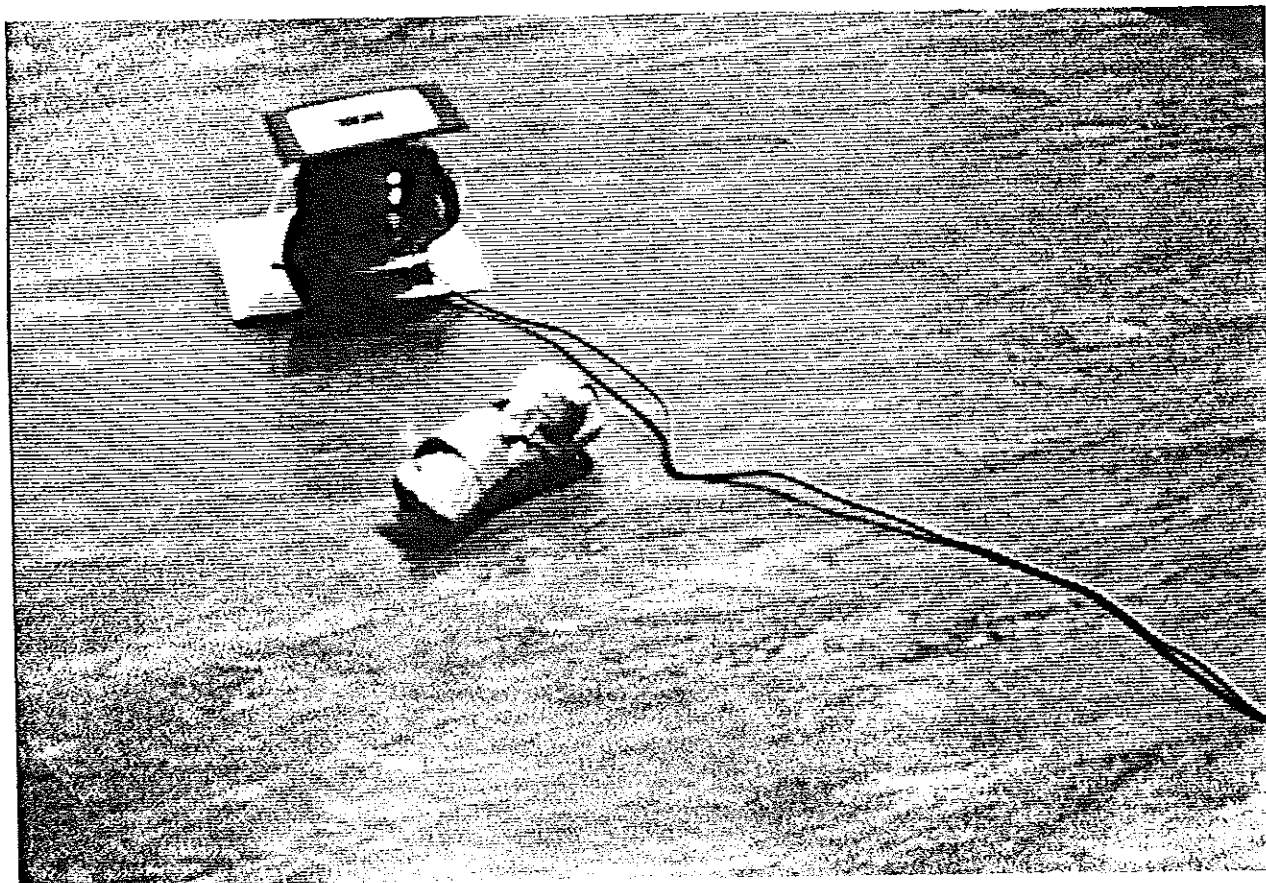
Robotens väg markeras.



Roboten på väg mot första delmålet.



Roboten har svängt efter andra delmålet.



Roboten framme vid måltriangeln.

III. DEFINITIONER

Ett allmänt föremål beskrivs av dess kanter, dvs ett områdes lokala skillnad i gråton relativt omgivningen.

3.1 Enkla objekt: definieras som en mängd sammanhängande vita randpunkter.

3.1.1 Definitioner av enkla objekts parametrar.

a) Centrum (masscentrum): definieras som $\Sigma[\text{rad}(i), \text{kolonn}(i)]/n$, där $(\text{rad}(i), \text{kolonn}(i))$ är områdets i te randpunkt och n antalet punkter. Om varje punkt antas ha massan 1, blir centrum lika med dess masscentrum.

b) Ytenhet: En bildpunkt (pixel) har arean ett. (anm. Både en punkt och en linje har en yta enligt denna definition.)

c) Omkrets: Avståndet mellan två vita grannpunkter är ett om punkterna ligger utmed en horisontell eller vertikal linje. Annars är avståndet $\sqrt{2}$.

3.2 Sammansatta objekt: består av flera enkla objekt, där dessa kan ha någon gemensam egenskap eller gemensam parameter. (Ex. Ett sammansatt objekt kan vara två trianglar med samma centrum och ett visst förhållande mellan areorna.)

IV. OPERATIONER PÅ ENKLA OBJEKT

Kapitlet innehåller en klassificering av enkla objekt och beskriver hur hörnpunkter tas fram för dessa, samt vilka tester som utförs på enkla objekt.

Efter att kanterna har detekterats och bilden har tröskelsatts, erhålls en binär bild. Denna bild innehåller en mängd objekt. Bilden avsökts tills en kantpunkt påträffas som inte behandlats tidigare. Denna utgör startpunkt för ett nytt objekt, vars kantpunkter går igenom av en 'Bug'. Principen för en 'Bug' är att om den befinner sig på en svart pixel vrider den ett steg åt vänster och om den är på en vit pixel tar den av åt höger, tills den når utgångspunkten. Som resultat fås koordinaterna för sammanhängande, vita randpunkter i en lista.

4.1 Klassificering:

Tre typer av enkla objekt definieras: triangel, kvadrat och övrigt.

(anm. Beroende på vinkeln mellan kameran och symbolernas plan, kommer 'kvadraten' att ses som en parallelogram, rektangel eller kvadrat.)

4.2. Enkla objekts hörn.

4.2.1 Approximativ bestämning av hörn, för de definierade objekttyperna.

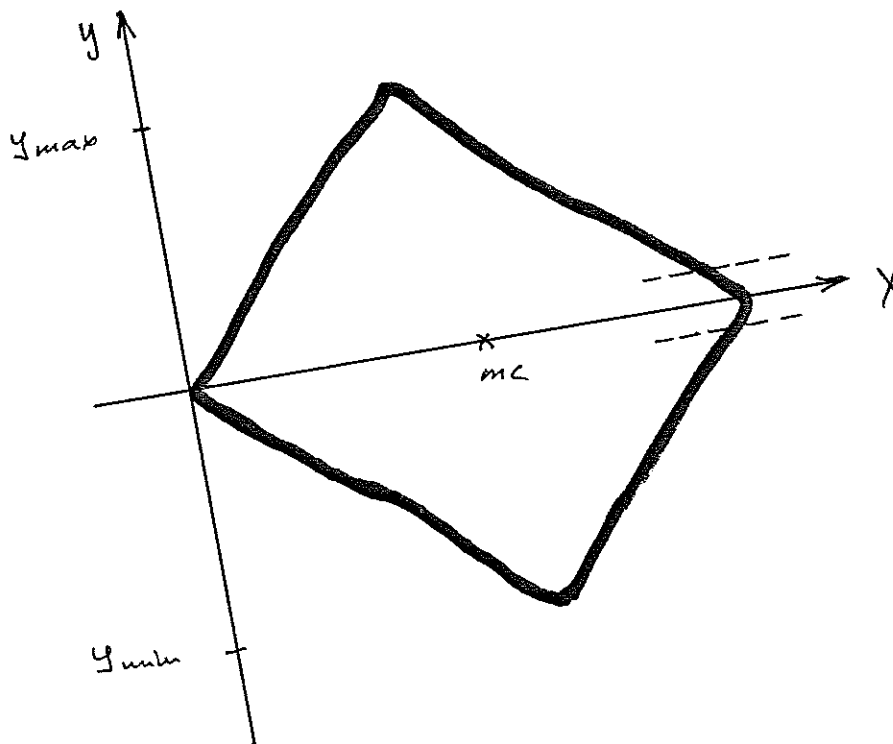
- a) Beräkna randpunkternas masscentrum (m_c).
- b) Sök den punkt som är längst från m_c . Denna punkt kallas primärhörn.
- c) Lägg ett ortogonalt koordinatsystem, med origo i primärhörnet och dess x-axel genom m_c .
- d) Leta upp de punkter som har maximal respektive minimal y-koordinat, samt maximal x-koordinat då y-koordinatens belopp är mindre än ett maxvärde. Kalla diagonalen genom denna sista punkt

för maxdiagonal.

Dessa punkter utgör eventuella approximativa hörn för den aktuella figuren.

4.2.2 Bestämning av antalet hörn.

Längden av maxdiagonalen jämförs med sidorna som har primärhörnet som skärningspunkt. För en parallelogram gäller att kvoten mellan maxdiagonalen och den längsta sidan är större än ett, medan



Figur 3. Hörnbestämning.

motsvarande kvot för triangeln, är mindre än ett.

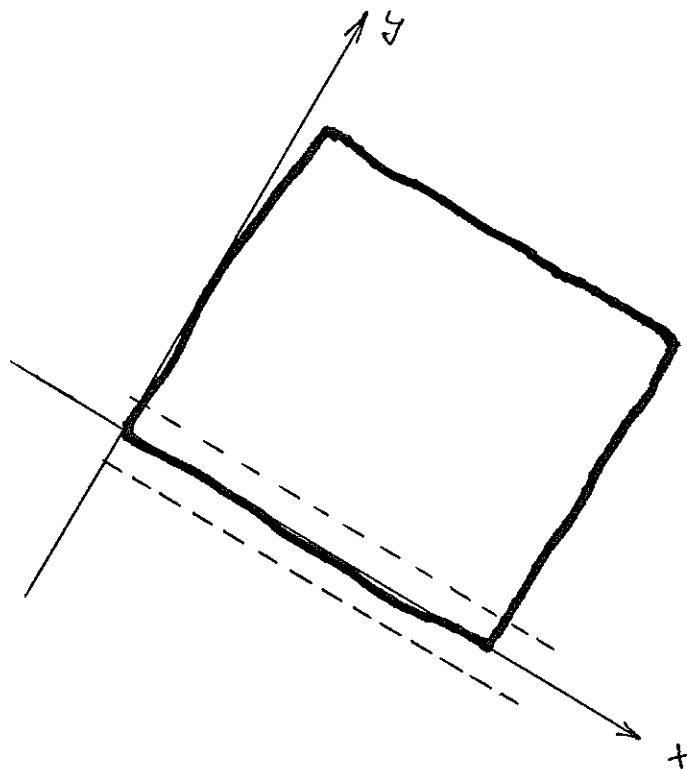
4.2.3 Noggrann bestämning av hörnkoordinaterna.

Hörnen är väsentliga för den klassificering som sedan görs med hjälp av ytmått. Detta kräver en noggrannare bestämning av hörnen än vad 4.2.1 kan ge. När de approximativa hörnpunkterna är givna, sker bestämningen på följande sätt:

x -axeln av ett ortogonalt koordinatsystem läggs utmed en figursida, alltså genom två hörn. Sedan söks de punkter som ligger i ett band längs denna axel och vars y -koordinater är mindre än ett maxtal. Med hjälp av linjär regression, beräknas parametrarna för en rät linje anpassad till dessa punkter. Detta upprepas för figurens alla sidor. Hörnen fås därefter som skärningen mellan dessa linjer. (anm. Antalet punkter som ligger inom bandet utmed en sida, räknas och summan jämförs med sidlängden. För en krökt linje, ligger endast en mindre del av punkterna inom bandet och detta blir en test på att linjen är rak.)

4.3 Tester på enkla objekt.

- a) De sammanhängande områden som analyseras, bör ha en minsta gräns för exempelvis ytan. För små områden är det svårt att dra några slutsatser om dess figurtyp.
- b) En vidare sällning sker med hjälp av omkrets/area kvoten. Denna kvot ska vara mindre än något maxvärde. Exempelvis sällas kurvor som inte är slutna, bort. Detta mått är mycket grovt, eftersom det är beroende av vinkeln mellan kameran och objektets plan.
- c) De från 4.2.3 bestämda hörnpunkterna, ger direkt figurens omkrets. Denna beräknade omkrets jämförs med figurens omkrets från en direkt beräkning på bilden.



Figur 4. Punkterna i ett band längs x-axeln.

- d) Arean för en parallelogram fås som absolutbeloppet av kryssprodukten mellan två icke parallella sidor. För triangeln fås arean på samma sätt om resultatet divideras med två. Det framräknade värdet jämförs med figurens beräknade area direkt på bilden. (anm. Beräkningarna måste justeras med hänsyn till definitionen av en ytenhet, dvs hänsyn tas till att en linje har en yta. Vid beräkningen av en vektor mellan två hörn, korrigeras dess längd.)

Som resultat av operationer på enstaka objekt, fås en lista över de funna objekten med dess parametrar.

V. KANTDETEKTERING

En kant definieras som en gråtonsdifferens mellan två områden, om differensen överstiger ett minimivärde. Detektorn bör ge relativt tunna linjer och markera en kant endast på det ljusare området, för att kanterna inte ska sammanfalla då vinkeln mellan kamera och symbolernas plan är liten, eller då avståndet mellan dem är relativt stort.

Följande kantdetektor används:

1. Givet ett tröskelvärde som anger om en differens ska tolkas som en kant eller ej.
2. Om skillnaden mellan en punkt p och någon av dess åtta grannar är större än tröskelvärdet, är p en kantpunkt, annars inte.

VI. OPERATIONER PÅ SAMMANSATTA OBJEKT

Symbolerna som markerar mål och robot består av två likformiga figurer med samma masscentrum. Ett objekts karakteristika är dess figurtyp, kvoten mellan den inre och yttre figurens area och kvoten mellan deras omkrets.

- 6.1 Först söks de objekt med närliggande masscentrum och som är av samma typ.
- 6.2.a Kvoten mellan det inre och yttre objektets area jämförs med det sökta objektets karakteristiska värde.
- 6.2.b Kvoten mellan det inre och yttre objektets omkrets jämförs med ett typiskt värde för det sökta objektet. (anm. Testerna 6.2.a och 6.2.b är noggranna för de använda figurerna.)

Utfaller testerna positivt, läggs det yttre objektets parametrar och figurtyp in i en lista. Positioneringen i bildplanet är därmed klar.

VII. ROBOTENS VAL AV VÄG TILL MALET

Kapitlet innehåller en beskrivning av hur hinder detekteras, robotens vägval för att undvika eventuella hinder samt den styrstrategi som används.

7.1 Detektering av hinder.

Ett hinder beskrivs av dess kanter samt eventuellt också av dess gråton. Hindrens form antas vara okända. Detekteringen av hinder går dels ut på att söka ett föremål med en gråton som är lika ljus eller ljusare än måltriangelns centrum, dvs ett vitt objekt. Det går dels ut på att detektera ett hinder endast genom att gå efter kantpunkterna, oberoende av dess struktur.

I kantfallet ska två kritiska tröskelvärden bestämmas. En tröskel som definierar en kantpunkt och en som avgör om antalet punkter inom ett område, utgör en kant på ett föremål eller inte. Trösklarna får sättas med hänsyn till bakgrundens gråtonsskiftningar.

Det område som ska undersökas delas upp i ett antal delområden, vars bredd är lite större än robotens diameter och med en konstant längd. Som kriterium på att en kant påträffats, tas att antalet kantpunkter per ytenhet överstiger ett minimivärde. Motsvarande kriterium gäller för antalet vita punkter inom delarean.

Bilden behandlas först med ett lågpasfilter, som medför att mindre färgskiftningar jämnas ut. Kanterna detekteras med hjälp av Sobels kantdetektor och därefter tröskelsätts bilden.

7.2 Robotens val av väg.

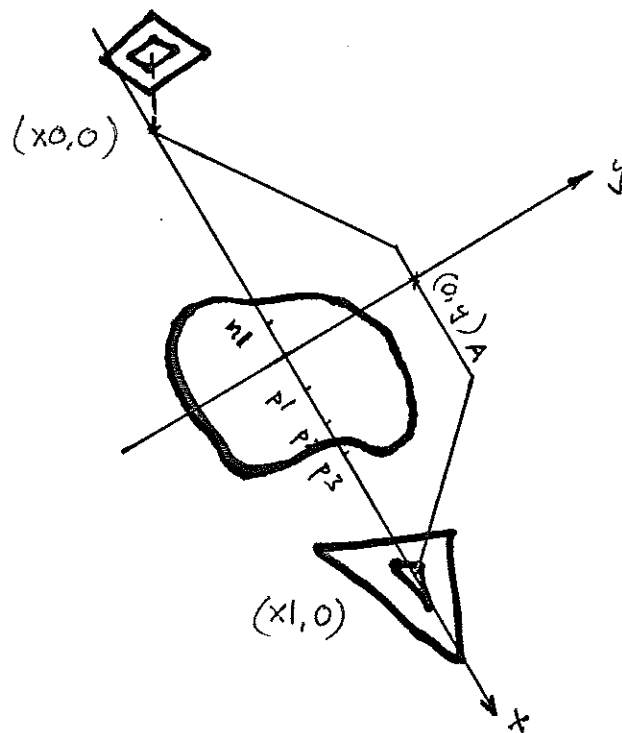
När robotens position och riktning, samt målets koordinater är bestämda, undersöks området mellan robot och mål enligt VII.1. Vägens bredd motsvarar robotens diameter. När ett hinder påträffas, justeras vägen i sidled och i längdled, alternerande till vänster och höger. Denna justering upprepas tills en hinderfri sträcka påträffas, se figur 5.

Programmet förutsätter att en fri väg finns och om ingen fri sträcka detekteras, höjs den tröskel som definierar en kantpunkt och en ny undersökning påbörjas.

7.3 Styrstrategi.

När en hinderfri passage detekterats enligt 7.1 och 7.2, tilldelas roboten det första delmålet. Varje gång roboten tilldelats ett delmål, beräknas dess längd och vridningsvinkel till detta mål. Därefter vrids roboten den beräknade vinkeln och kör fram en kort fix sträcka, varpå dess position åter bestäms och riktningen korrigeras. Robotens körsträcka delas ner tills den är mindre än en bestämd maxlängd, för att roboten inte ska komma för långt ur kurs, framförallt på grund av vinkelfel.

När roboten har nått fram till ett delmål, tilldelas den ett nytt mål och detta upprepas tills den når fram till måltriangeln.



Figur 5. Roboten har koordinaterna $(x_0,0)$ och målet $(x_1,0)$. En yta som motsvarar robotens area undersöks utmed y-axeln, alternerande mellan positiva och negativa y-axeln. När en hinderfri yta detekterats, analyseras sträckorna $(x_0,0)-(n_1,y)$ respektive $(x_1,0)-(p_1,y)$. Påträffas hinder utmed någon av dessa sträckor, förlängs vägen A en robotradie till (p_2,y) . Detta upprepas tills en fri väg detekteras eller sträckan går utanför kamerans synfält, då undersökningen fortsätter på motsatta y-axeln.

IX. GEOMETRISKA KORREKTIONER

I kapitlet redogörs för de korrektioner som fordras för att kompensera för robotens höjd vid positionsbestämning, på grund av att vinkeln mellan kamera och golvplanet är mindre än 90° . Det innehåller också transformationer mellan bild- och golvplan, samt hur approximativa värden beräknas för vinkel och avstånd mellan kamera och en punkt i golvplanet.

Mätningarna utförs på en bild som är en projektion av scenen på bildplanet. Avstånd och vinklar i bildplanet ska transformeras till golvplanet. Två ortogonala koordinatsystem införs enligt figur 6. Ett tredimensionellt koordinatsystem xyz , där robot och måltriangel befinner sig i xy -planet och kameran är placerad någonstans utmed positiva z -axeln. I bildplanet införs ett tvådimensionellt koordinatsystem som benämnes rc -planet. x -axeln är parallell med c -axeln.

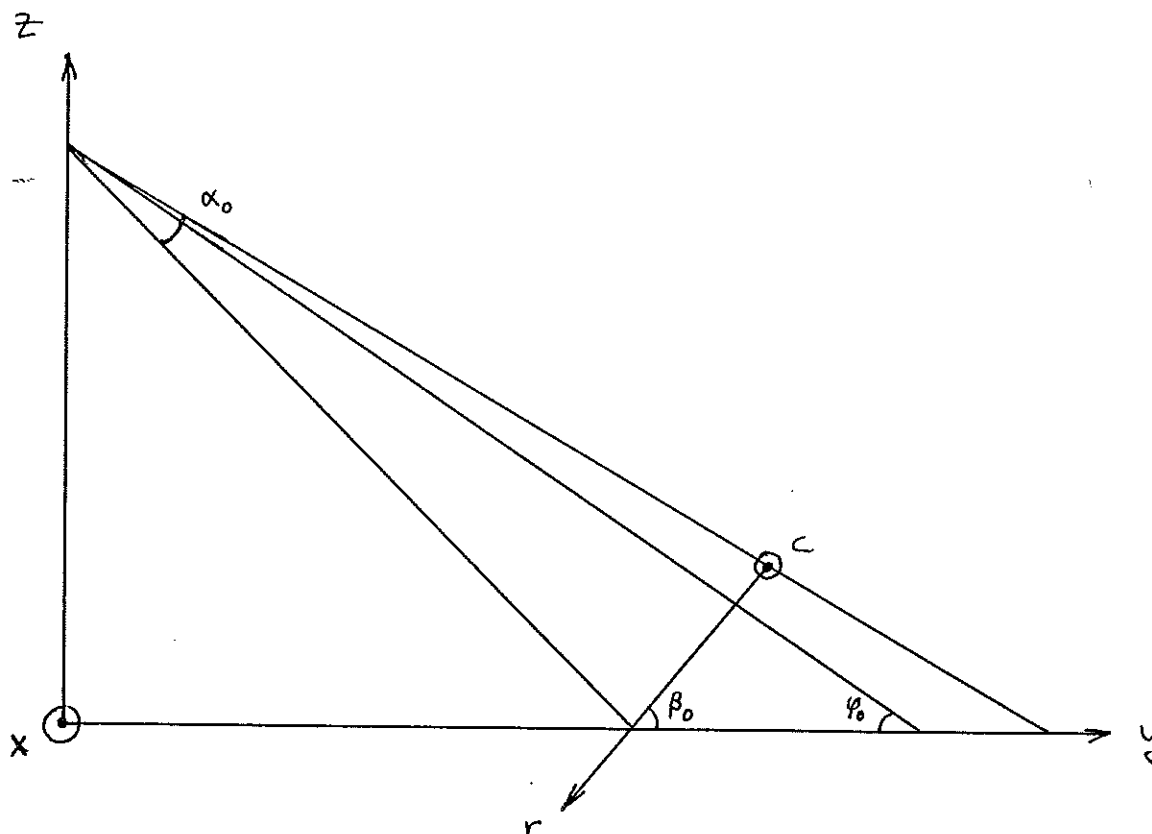
Kamerans höjd över xy -planet (golvplanet) och dess orientering är okänd.

Kamerans objektivvinklar i vertikal och horisontalled har mätts upp. Måltriangelns sidlängder och vinklar är kända. Avståndet mellan kamera och måltriangel fås ur en experimentellt bestämd funktion. I bildplanet kan en approximativ vinkel mellan kamera och måltriangelns plan beräknas från triangeln. Kamerans höjd över xy -planet kan därmed bestämmas.

8.1 Initialskede

1. Avstånd mellan kamera och måltriangel.

Måltriangelns sidor är lika långa och sidlängden är känd. Avståndet R mellan kamera och triangel, är en funktion av triangelns inverterade sidlängd i bildplanet, då sidan är parallell med x -axeln.



Figur 6. φ_0 är vinkeln mellan kameran och måltriangelns plan (xy-planet). Denna vinkel kallas referensvinkel. α_0 är objektivets vinkel i vertikalled, samt β_0 betecknar vinkeln mellan xy- och rc-planet.

Ett antal mätningar på måltriangelns längsta sida i rc-planet är gjorda på olika avstånd mellan kamera och triangel. Denna sida är minst påverkad av vinkeln mellan rc- och xy-planet. Med linjär regression har ett andragsgradspolynom anpassats till dessa mätvärden. Felet i längdbestämningen är mindre än 10% vid typiska kamerauppställningar.

2. Referensvinkel

Vinkeln mellan kamera och xy-planet i måltriangelns centrum, beräknas approximativt från triangelns sidor i bildplanet. Följande empiriska formel används:

$$\varphi_0 = \arcsin\left(\frac{\text{kortaste sidan}}{\text{längsta sidan}}\right) - \text{konst} \quad (8.1)$$

där konstanten har bestämts experimentellt till 7.5° . Onoggrannheten i vinkelbestämningen är till beloppet mindre än 3° , vid typiska kamerauppställningar.

8.2 Kamerans skalfaktor.

Skalfaktorn definieras som kvoten mellan en längd i rc -planet, utmed r - eller c -axeln och motsvarande linje i mm i y - respektive x -led i xy -planet, då objektivplanet är parallellt med xy -planet. (anm. Skalfaktorerna i r - respektive c -led, är olika för den använda kameran.) Skalfaktorn beror av kamera och avståndet mellan kamera och objekt.

Om skalfaktorn multipliceras med avståndet R mellan objekt och kamera, erhålls en konstant, som kallas skalkonstant.

$$\text{Skalkonstant} = \text{Skalfaktor} * R = \frac{p}{2 \tan(\alpha_1/2)} \quad (8.2)$$

där p är längden i bildplanet och α_1 är objektets synvinkel utmed aktuell axel.

Skalfaktorn för en godtycklig punkt i rc -planet, fås genom att avståndet R beräknas och divideras med skalkonstanten.

8.3 Positionsbestämning av robot.

Från rc -koordinaten i bilden för robotens symboler, korrigeras för robotens höjd. Därefter bestäms avstånd och vinkel till aktuellt delmål.

1. Vinkel och avstånd mellan kamera och en godtycklig punkt i xy -planet.

Antag att punktens koordinater i bildplanet är (r,c) och projektionens koordinater är (x,y) i xy -planet. Vinkeln mellan kameran och xy -planet fås approximativt som:

$$\varphi = \varphi_0 + \frac{(r - r_{\text{Tri}}) \alpha_0}{512} \quad (8.3)$$

där r_{Tri} är måltriangelns r -koordinat, φ_0 är referensvinkeln och α_0 är objektivets vertikala vinkel.

Avståndet är

$$R = R_0 \sin(\varphi_0) / \sin(\varphi) \quad (8.4)$$

där R_0 och φ_0 är referensvärde för avstånd respektive vinkel till måltriangeln.

2. Höjdkorrektion.

Positionen av robotens kvadrater är (r_1, c_1) i figur 7, men detta motsvarar en y -koordinat i xy -planet som inte överensstämmer med robotens verkliga position, på grund av dess höjd och vinkeln mellan de båda planen.

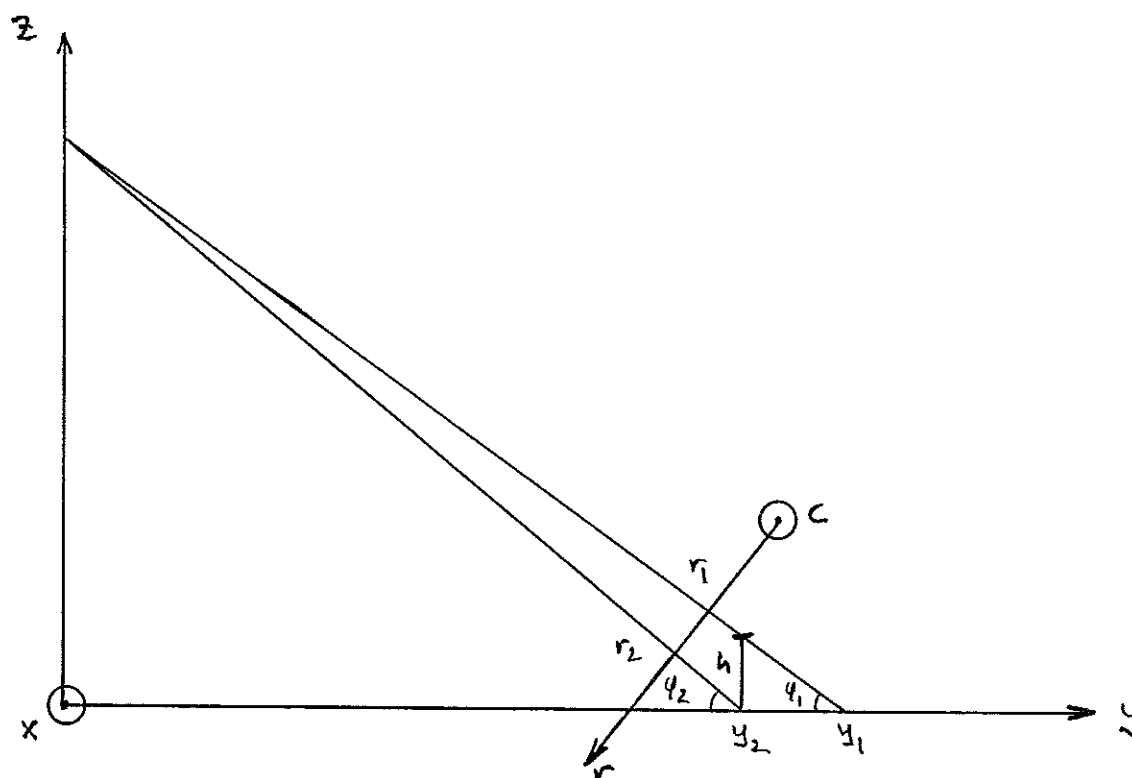
Vinkeln φ_1 beräknas enligt (8.3). Låt φ_2 beteckna vinkeln mellan kameran och xy -planet i robotens verkliga xy -koordinat. φ_2 fås då som

$$\varphi_2 = \arctan\left(\frac{H}{H-h} \tan(\varphi_1)\right) \quad (8.5)$$

där h är robotens höjd och H kamerans höjd över måltriangeln. Korrektionen i rc -planet blir

$$h_{rc} = (\varphi_2 - \varphi_1) R_2 s_2 \quad (8.6)$$

där R_2 är avståndet mellan kameran och robotens verkliga y -koordinat och s_2 är vertikala skalfaktorn i punkten (r_2, c_2) . φ_1 och φ_2 är angivna i radianer.



Figur 7. Robotens symboler har en y-koordinat y_1 , som inte överensstämmer med dess verkliga y-koordinat y_2 .

3. Verkliga längden mellan två punkter i rc-planet.

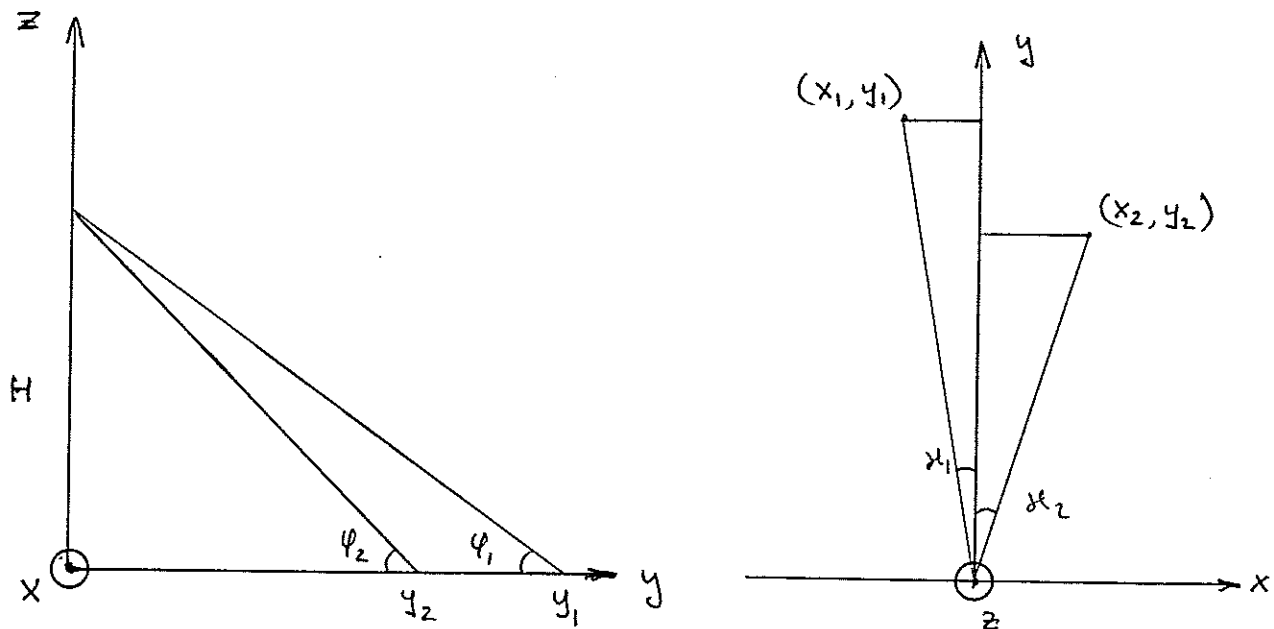
Givet två punkter (r_1, c_1) och (r_2, c_2) i rc-planet. Avståndet mellan dessa punkter i xy-planet söks. Punkterna motsvaras av (x_1, y_1) respektive (x_2, y_2) , se figur 8.

Vinklarna φ_1 , φ_2 , κ_1 och κ_2 definieras i figur 8. Skillnaden i y-led blir approximativt

$$y = H[\cot(\varphi_2)\cos(\kappa_2) - \cot(\varphi_1)\cos(\kappa_1)] \quad (8.7)$$

där H är kamerans höjd över måltriangeln, som är lika med dess z-koordinat.

Skalfaktorn är en funktion av r-koordinaten. Avståndet mellan c-koordinaten och c_{const} beräknas för de båda punkterna. Längden utmed x-axeln blir



Figur 8. Avståndet mellan två punkter (x_1, y_1) och (x_2, y_2) .

$$x = \frac{(c_1 - c_{\text{const}})}{s_1} + \frac{(c_{\text{const}} - c_2)}{s_2} \quad (8.8)$$

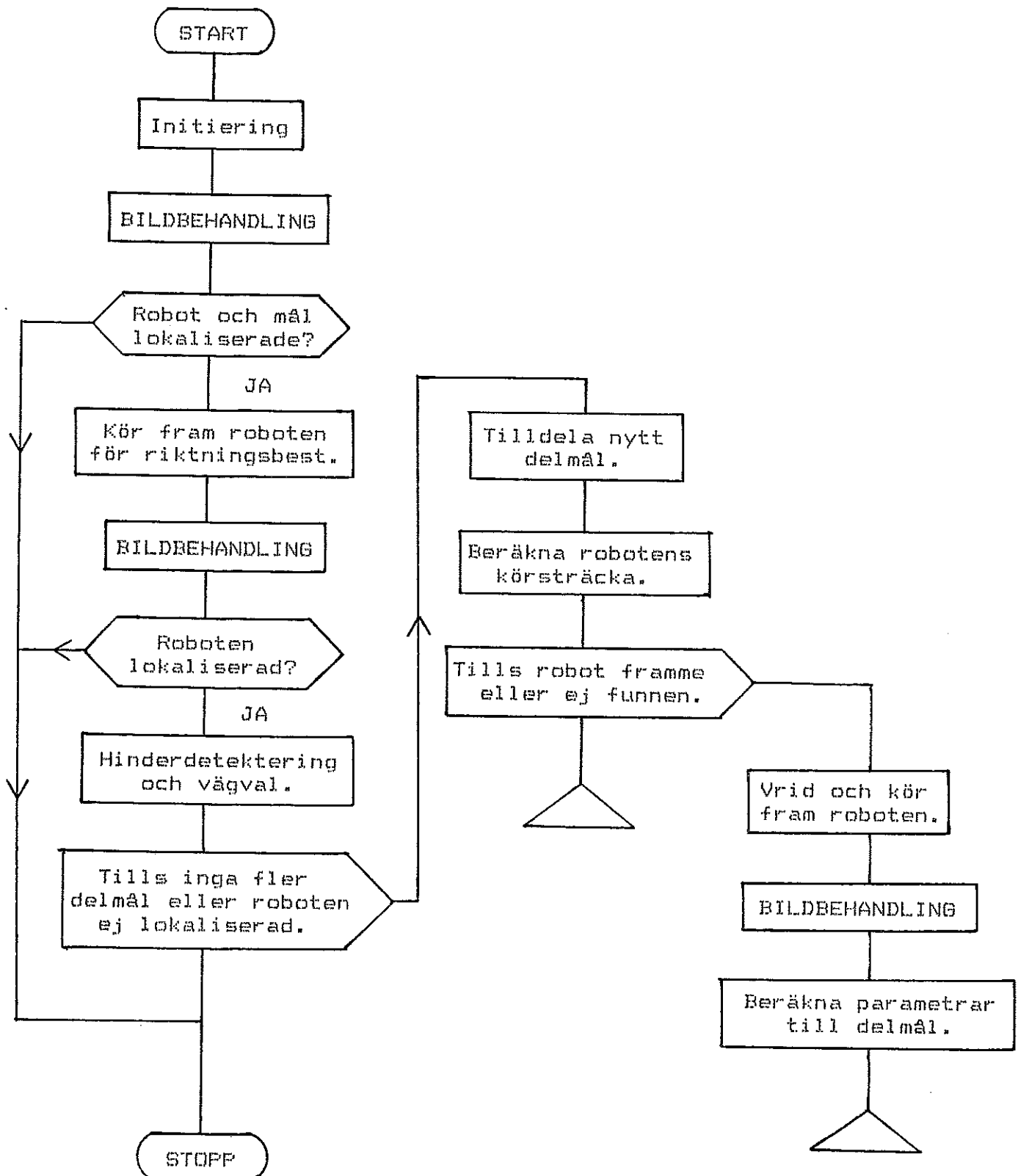
där s_1 och s_2 är respektive punkters horisontella skalfaktorer. (anm. Om antalet pixel i c -led är c_{max} , är $c_{\text{const}} = c_{\text{max}}/2$.)

Totala längden fås som

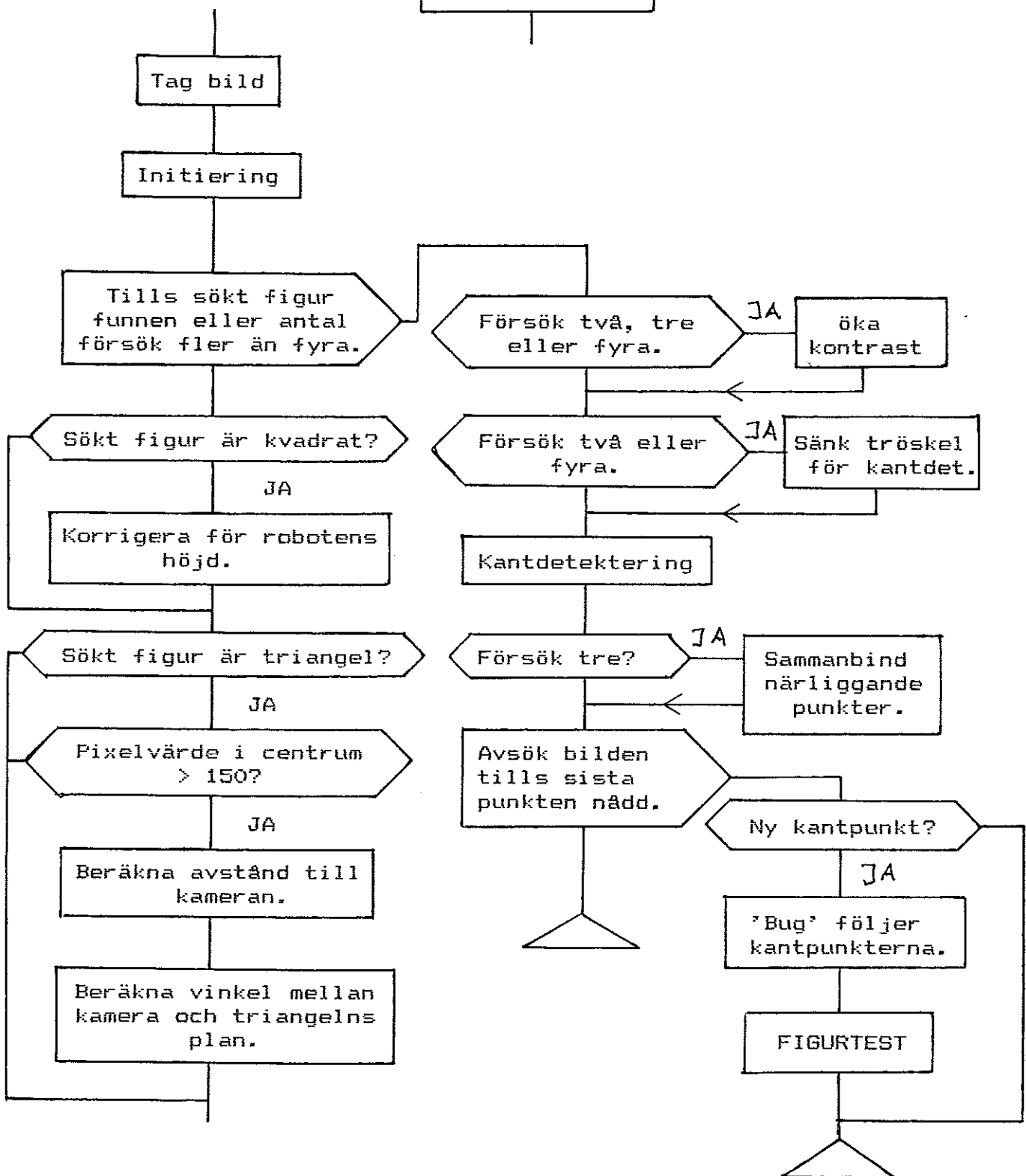
$$L = \sqrt{x^2 + y^2} \quad (8.9)$$

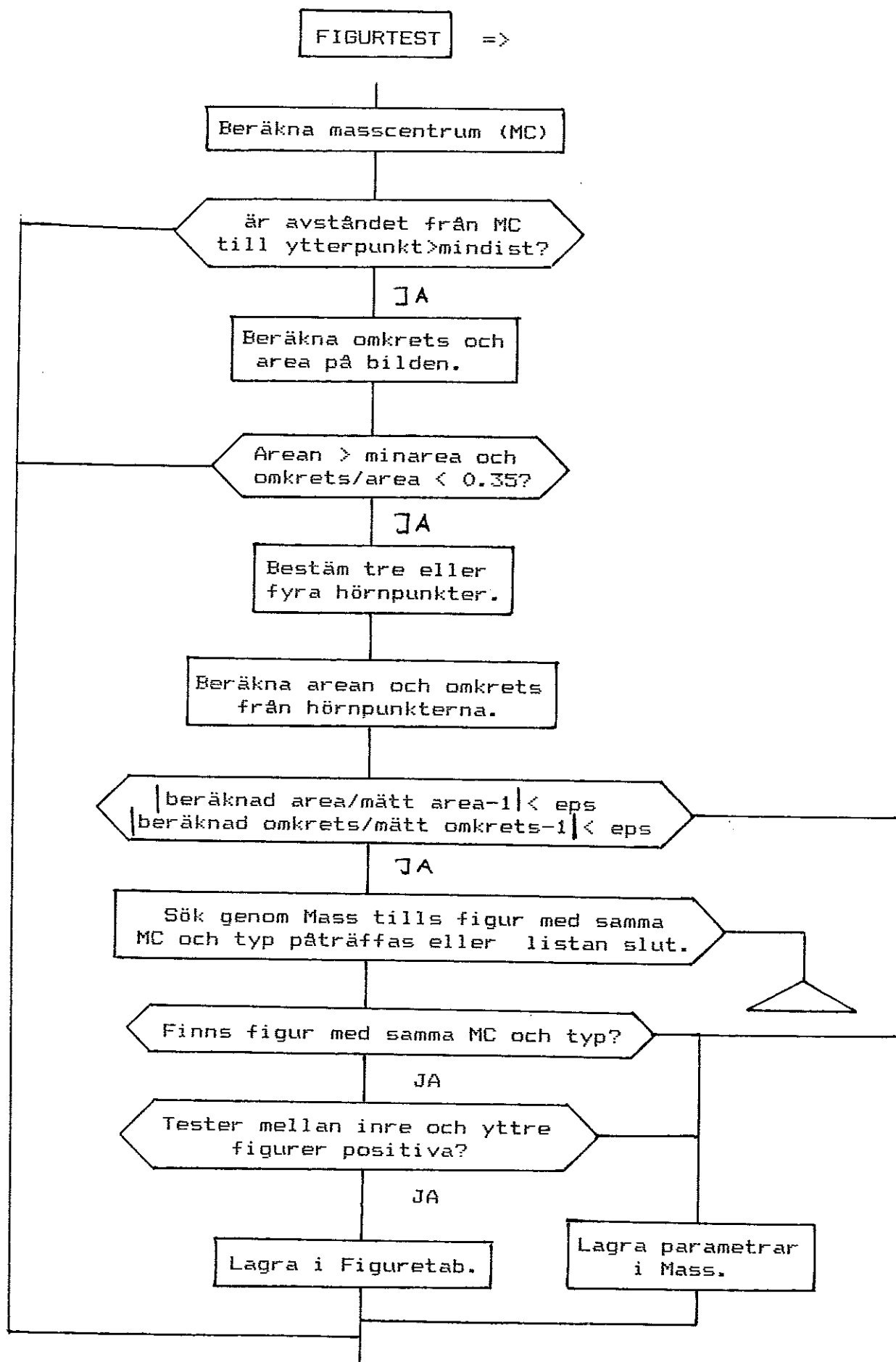
IX. PROGRAMBESKRIVNING

Koden omfattar cirka 5000 rader och innehåller en hel del listhantering. Nedan följer flödesscheman, med de viktigaste delarna.



BILDBEHANDLING =>





Referenser

Pratt William, Digital Image Processing, 1978.

Nielsen Lars, Elmqvist Hilding, An Image Laboratory. CODEN:
LUTFD2/(TFET-7261)/1-051/(1983)

Sandin G., Wullt T, Experiment med datorstyrd kännande robot.
CODEN:LUTFD2/(TFET-5271)