

# **CONTRIBUȚII LA UTILIZAREA ROBOȚILOR ÎN AGRICULTURĂ PENTRU APLICAȚII DE COMBATERE A BURUIENILOR**

Teză destinată obținerii  
titlului științific de doctor inginer  
la  
Universitatea „Politehnica” din Timișoara  
în domeniul INGINERIE MECANICĂ  
de către

**Ing. dipl. Mihaela Tilneac**

Conducător științific:

Referenți științifici:

Prof.univ.dr.ing. Valer Dolga

Prof.univ.dr.ing. Radu Bălan

Prof.univ.dr.ing. Doru Talabă

Prof.univ.dr.ing. Arjana Davidescu

Ziua susținerii tezei: 27.02.2012

Seriile Teze de doctorat ale UPT sunt:

- |                        |   |
|------------------------|---|
| 1. Automatică          | 7. Inginerie Electronică și Telecomunicații |
| 2. Chimie              | 8. Inginerie Industrială                    |
| 3. Energetică          | 9. Inginerie Mecanică                       |
| 4. Ingineria Chimică   | 10. Știința Calculatoarelor                 |
| 5. Inginerie Civilă    | 11. Știința și Ingineria Materialelor       |
| 6. Inginerie Electrică |   |

Universitatea „Politehnica” din Timișoara a inițiat seriile de mai sus în scopul diseminării expertizei, cunoștințelor și rezultatelor cercetărilor întreprinse în cadrul școlii doctorale a universității. Seriile conțin, potrivit H.B.Ex.S Nr. 14 / 14.07.2006, tezele de doctorat susținute în universitate începând cu 1 octombrie 2006.

Copyright © Editura Politehnica – Timișoara, 2012

Această publicație este supusă prevederilor legii dreptului de autor. Multiplicarea acestei publicații, în mod integral sau în parte, traducerea, tipărirea, reutilizarea ilustrațiilor, expunerea, radiodifuzarea, reproducerea pe microfilme sau în orice altă formă este permisă numai cu respectarea prevederilor Legii române a dreptului de autor în vigoare și permisiunea pentru utilizare obținută în scris din partea Universității „Politehnica” din Timișoara. Toate încălcările acestor drepturi vor fi penalizate potrivit Legii române a drepturilor de autor.

România, 300159 Timișoara, Bd. Republicii 9,  
tel. 0256 403823, fax. 0256 403221  
e-mail: editura@edipol.upt.ro

## Cuvânt înainte

Teza de doctorat este rezultatul activității prestate în perioada stagiului ca doctorand cu frecvență în cadrul Departamentului de Mecatronică, Facultatea de Mecanică, Universitatea "Politehnica" din Timișoara.

Adresez mulțumiri deosebite conducătorului de doctorat prof. dr. ing. Valer Dolga, pentru sfaturile și suportul acordat în toate problemele apărute pe parcursul elaborării tezei. Mulțumesc d-lui prof. dr. rer. nat. Arno Ruckelshausen, sub îndrumarea căruia am efectuat un stagiul de cercetare în străinătate la „Fachhochschule Osnabrück” din Germania. Adresez mulțumiri d-nei șef. lucr. dr. ing. Sanda Grigorescu și d-lui ing. dipl. Victor Paléologue, pentru colaborările în domeniul controlului roboților prin vedere artificială.

Îmi exprim întreaga considerație față de membrii comisiei de doctorat, care au răspuns solicitării de a face parte din comisia de analiză a tezei, pentru observațiile făcute și pentru timpul acordat lucrării.

De asemenea mulțumesc tuturor celor care m-au ajutat cu diverse sfaturi și indicații utile finalizării tezei de doctorat.

Timișoara, februarie 2012

Ing.dipl. Mihaela Tilneac

Teza de doctorat a fost realizata cu sprijin partial din grantul strategic POSDRU/6/1.5/S/13, ID6998, cofinantat din Fondul Social European "Investeste in oameni", in cadrul Programului Operational Sectorial Dezvoltare Resurse Umane 2007-2013

Tilneac, Mihaela

**Contribuții la utilizarea roboților pentru agricultură în aplicații de combatere a buruienilor**

Teze de doctorat ale UPT, Seria 9, Nr. 112, Editura Politehnica, 2012, 228 pagini, 130 figuri, 34 tabele.

ISSN: 1842-4937

ISBN: 978-606-554-455-0

Cuvinte cheie:

Roboți, Agricultură, Navigare, Recunoaștere plante, Combatere buruieni, Calibrare camere video, Sisteme stereoscopice

Rezumat,

Teza de doctorat aduce contribuții în dezvoltarea roboților pentru agricultură. Obiectivul ce stă la baza cercetării este crearea unui robot pentru combaterea buruienilor de pe terenurile agricole utilizând sisteme mecanice. Spre deosebire de combaterea buruienilor prin erbicidare, combaterea mecanică aduce avantaje de natură ecologică. Principalele funcții pe care robotul trebuie să le îndeplinească sunt următoarele: (1) navigarea printre rândurile de culturi; (2) recunoașterea plantelor; (3) Înlăturarea mecanică a buruienilor.

În teza de doctorat sunt prezentate rezultatele cercetărilor efectuate în scopul perfecționării funcțiilor enumerate anterior. Referitor la funcția de navigare, au fost efectuate studii privind navigarea robotului prin ghidare automată de-a lungul rândurilor de culturi, detectate printr-un sistem vizual format din camere 3D-ToF (Time of Flight). Cercetările privind utilizarea camerelor 3D-ToF au fost începute în perioada stagiului de cercetare la „Fachhochschule Osnabrück” din Germania, sub îndrumarea d-lui prof. dr. rer. nat. Arno Ruckelshausen și a d-lui ing. dipl. Ralph Klose. Pentru recunoașterea speciilor de plante, au fost investigate diverse metode. Unele cercetări au avut ca scop recunoașterea speciilor de plante pe baza culorilor. Alte cercetări au avut ca obiectiv recunoașterea speciilor de plante pe baza caracteristicilor geometrice ale frunzelor. Un alt studiu a constatat în corelarea imaginilor prelevate de la un sistem optic format dintr-o cameră 3D-ToF și o cameră 2D-color. Prin acest procedeu pot fi obținute simultan informații despre culorile plantei, respectiv distanțele la care se află punctele care aparțin suprafeței plantei. Corelarea imaginilor implică cercetări ample privind calibrarea camerelor video și utilizarea sistemelor de vedere stereoscopică. Autoarea tezei a dezvoltat o metodă originală de calibrare intrinsecă, și două metode originale de calibrare extrinsecă a camerelor video. În lucrare este prezentată o metodă de diferențiere buruienă/plantă pe baza valorilor înălțimilor determinate prin utilizarea unui sistem stereoscopic. În teza de doctorat este prezentat un concept de robot pentru agricultură dotat cu dispozitiv mecanic de înlăturare a buruienilor.



# Cuprins

Cuprins.....	V
Notății, abrevieri, acronime .....	VIII
Lista de tabele.....	XVI
Lista de figuri .....	XVII
1 Introducere .....	1
1.1 Motivația abordării tematicii alese .....	1
1.2 Structura tezei de doctorat .....	3
2 Sinteze bibliografice privind tematica tezei de doctorat .....	5
2.1 Stadiul actual de utilizare a roboților mobili în aplicații de combatere a buruienilor .....	5
2.1.1 Introducere .....	5
2.1.2 Navigarea robotului printre rândurile de culturi .....	6
2.1.3 Recunoașterea speciilor de plante .....	8
2.1.4 Vedere stereo, modelare 3D și fuziune senzorială .....	11
2.1.5 Combaterea buruienilor utilizând sisteme automate .....	13
2.1.6 Prototipuri de roboți pentru protecția culturilor de plante .....	15
2.1.7 Concursul internațional „Field Robot Event” .....	19
2.1.8 Sisteme de securitate în timpul funcționării robotului.....	20
2.1.9 Concluzii .....	20
2.2 Calibrarea camerelor video .....	21
2.2.1 Generalități .....	21
2.2.2 Calibrarea intrinsecă .....	21
2.2.3 Calibrarea extrinsecă.....	22
2.2.4 Metode de calibrare OFF-LINE (sinteză bibliografică) .....	23
2.2.5 Metode de calibrare ON-LINE (sinteză bibliografică) .....	25
2.2.6 Concluzii .....	26
3 Obiectivele și planul de cercetare .....	27
3.1 Obiectivele cercetării .....	27
3.2 Planul de cercetare .....	28
4 Contribuții la recunoașterea speciilor de plante pe baza culorilor .....	29
4.1 Introducere .....	29
4.2 Standul experimental.....	29
4.3 Analiza experimentală a culorilor frunzelor în condiții variabile de iluminare .....	31
4.4 Programul de procesare a culorilor .....	47
4.5 Concluzii .....	50
5 Contribuții la recunoașterea speciilor de plante pe baza formelor frunzelor .....	53
5.1 Introducere .....	53

5.2	Experiment 1 - Identificarea culorilor și a formelor prin procesare de imagini .....	53
5.3	Experiment 2 - Recunoașterea semnăturilor frunzelor .....	55
5.4	Experiment 3 - Recunoașterea formelor frunzelor .....	60
5.5	Selectarea frunzelor în cadrul unei aplicații robotizate .....	62
5.6	Programe de recunoaștere a frunzelor .....	64
5.7	Concluzii .....	66
6	Localizarea obiectelor în spațiul tridimensional .....	67
6.1	Introducere .....	67
6.2	Măsurarea distanței prin utilizarea unui cerc .....	67
6.3	Măsurarea distanțelor utilizând vederea stereo.....	69
6.4	Programul de măsurare a distanțelor .....	74
7	Calibrarea camerelor video .....	75
7.1	Calibrare intrinsecă .....	75
7.1.1	Introducere .....	75
7.1.2	Algoritmi matematici .....	75
7.1.3	Descrierea experimentului.....	80
7.1.4	Rezultatele experimentului.....	81
7.1.5	Concluzii.....	85
7.2	Calibrare extrinsecă prin metode analitice .....	86
7.2.1	Introducere .....	86
7.2.2	Modelul matematic.....	86
7.2.3	Descrierea experimentului.....	102
7.2.4	Rezultatele experimentului .....	106
7.3	Calibrare extrinsecă utilizând modele CAD .....	107
7.4	Concluzii .....	113
8	Diferențierea buruiană/plantă pe bază de măsurări optice tridimensionale ....	115
8.1	Introducere .....	115
8.2	Localizarea planului solului .....	116
8.3	Determinarea ecuației planului pe baza mai multor imagini .....	119
8.4	Diferențierea buruiană/plantă .....	121
8.5	Concluzii .....	122
9	Utilizarea camerei 3D-ToF (Time of Flight) pentru aplicații în agricultură .....	125
9.1	Introducere .....	125
9.2	Analiza plantelor utilizând un sistem stereo format dintr-o cameră 3D-ToF (Time of Flight) și o cameră 2D-color.....	125
9.3	Utilizarea camerelor 3D-ToF (Time of Flight) la navigarea robotului mobil .....	130
9.4	Concluzii .....	134

10	Proiectarea și realizarea robotului <i>GardenerRob</i> pentru aplicații în agricultură .....	135
10.1	Introducere .....	135
10.2	Structura dispozitivului de ghidare (DG) și a efectorului final (EF) .....	135
10.3	Modelul geometric direct și invers al robotului serial .....	136
10.4	Modul de funcționare al robotului autonom <i>GardenerRob</i> .....	142
10.5	Concluzii .....	144
11	Concluzii finale, contribuții originale și direcții viitoare de cercetare.....	145
11.1	Concluzii finale.....	145
11.2	Contribuții originale .....	145
11.3	Direcții viitoare de cercetare .....	146
	Lista publicațiilor personale .....	147
	Bibliografie.....	149
	Anexe.....	169
	Anexa A; Programe MATLAB pentru procesarea culorilor din imagine .....	169
	Anexa B; Programe MATLAB pentru procesarea formelor din imagine .....	171
	Anexa C; Programe MATLAB pentru calibrarea camerelor video.....	178
	Anexa D; Programe MATLAB utilizate la diferențierea buruiană/plantă pe baza înălțimilor.....	195
	Anexa E; Programe MATLAB pentru corelarea imaginilor .....	198
	Anexa F; Programe utilizate la sistemul de navigare al robotului .....	201
	Anexa G; Măsurarea distanțelor utilizând vederea stereo .....	203
	Curriculum vitae .....	207

## Notații, abrevieri, acronime

Simbol	Unitate de măsură	Descriere
$\alpha$	[grad]	Rotație în jurul axei x.
$\beta$	[grad]	Rotație în jurul axei y.
$\gamma$	[grad]	Rotație în jurul axei z.
CG	-	Centru de greutate
d_cerck_world	[mm]	Diametrul cercului k aflat în spațiul scenei.
d_elipsak_im	[pixel]	Lungimea axei mari a elipsei ce reprezintă proiecția cercului k pe planul imagine.
d_elipsak_im1	[pixel]	Lungimea axei mari a elipsei ce reprezintă proiecția cercului k în imaginea 1.
d_elipsak_im2	[pixel]	Lungimea axei mari a elipsei ce reprezintă proiecția cercului k în imaginea 2.
d_FPk_im	[pixel]	Distanța dintre focarul F și punctul Pk_im.
d_FPk_world	[mm]	Distanța dintre focarul F și punctul Pk_world.
d_F1Pk_im1	[pixel]	Distanța dintre focarul F1 și punctul Pk_im1.
d_F1Pk_im1_x	[pixel]	Distanța dintre focarul F1 și punctul Pk_im1, utilizând f1_x.
d_F1Pk_im1_y	[pixel]	Distanța dintre focarul F1 și punctul Pk_im1, utilizând f1_y.
d_F2Pk_im2	[pixel]	Distanța dintre focarul F2 și punctul Pk_im2.
d_F2Pk_im2_x	[pixel]	Distanța dintre focarul F2 și punctul Pk_im2, utilizând f2_x.
d_F2Pk_im2_y	[pixel]	Distanța dintre focarul F2 și punctul Pk_im2, utilizând f2_y.
d_F1Pk_world	[mm]	Distanța dintre focarul F1 și punctul

d_F2Pk_world	[mm]	Pk_world. Distanța dintre focarul F2 și punctul Pk_world.
d_P1P2_world	[mm]	Distanța dintre centrul cercului 1 și centrul cercului 2.
F	-	Punct focal.
f	[pixel]	Distanța focală. Distanța dintre focar și planul imaginii.
f_x	-	Distanța focală corespunzătoare axei x.
f_y	[pixel]	Distanța focală corespunzătoare axei y.
F1	-	Punctul focal al camerei 1.
f1	[pixel]	Distanța focală 1. Distanța dintre focarul F1 și planul imaginii 1.
f1_x	[pixel]	Distanța focală 1 corespunzătoare axei x.
f1_y	[pixel]	Distanța focală 1 corespunzătoare axei y.
F2	-	Punctul focal al camerei 2.
f2	[pixel]	Distanța focală 2. Distanța dintre focarul F2 și planul imaginii 2.
f2_x	[pixel]	Distanța focală 2 corespunzătoare axei x.
f2_y	[pixel]	Distanța focală 2 corespunzătoare axei y.
k	-	Numărul punctelor de control. (Valoarea lui k este egală cu numărul cercurilor.)
M_im	-	Punctul principal. (Centrul imaginii.)
M_im1	-	Punctul principal al imaginii 1. (Centrul imaginii 1.)
M_im2	-	Punctul principal al imaginii 2. (Centrul imaginii 2.)
M_Rotatie	-	Matrice de rotație utilizată pentru a realiza trecerea din sistemul de referință cam1 în sistemul de referință cam2.
M_Translatie	-	Vector de translație utilizat pentru a realiza trecerea din sistemul de referință cam1 în sistemul de referință cam2.

**X** Notatii, abrevieri, acronime

---

(O_cam, x_cam, y_cam, z_cam)	-	Sistemul de referință al camerei.
(O_cam1, x_cam1, y_cam1, z_cam1)	-	Sistemul de referință al camerei 1. Prescurtarea este notată cam1.
(O_cam2, x_cam2, y_cam2, z_cam2)	-	Sistemul de referință al camerei 2. Prescurtarea este notată cam2.
(O_im, x_im, y_im, z_im)	-	Sistemul de referință al imaginii.
(O_im1, x_im1, y_im1, z_im1)	-	Sistemul de referință al imaginii 1. Prescurtarea este notată im1.
(O_im2, x_im2, y_im2, z_im2)	-	Sistemul de referință al imaginii 2. Prescurtarea este notată im2.
(O_offset_cam1, x_offset_cam1, y_offset_cam1, z_offset_cam1)	-	Sistemul de referință rezultat prin translatarea originii sistemului de referință cam1 în punctul P4_world. Prescurtarea este notată offset_cam1.
(O_offset_cam2, x_offset_cam2, y_offset_cam2, z_offset_cam2)	-	Sistemul de referință rezultat prin translatarea originii sistemului de referință cam2 în punctul P4_world. Prescurtarea este notată offset_cam2.
P	-	Punct.
P_im	-	Proiecția unui punct oarecare P_world pe planul imagine, raportat la sistemul de referință al imaginii.
P_im_cam	-	Proiecția unui punct oarecare P_world pe planul imagine, raportat la sistemul de referință al camerei.
P_im1	-	Coordonatele proiecției punctului oarecare P_world pe planul imaginii 1, raportate la sistemul de referință al imaginii 1.
P_im1_cam1	-	Coordonatele proiecției punctului oarecare P_world pe planul imaginii 1, raportate la sistemul de referință al camerei 1.
P_im2	-	Coordonatele proiecției punctului oarecare P_world pe planul imaginii 2, raportate la sistemul de referință

		al imaginii 2.
P_im2_cam2	-	Coordonatele proiecției punctului oarecare P_world pe planul imaginii 2, raportate la sistemul de referință al camerei 2.
P_world	-	Punct oarecare din scenă.
P_world_cam	-	Punct oarecare aflat în spațiul scenei, raportat la sistemul de referință al camerei.
P_world_cam1	-	Punct oarecare aflat în spațiul scenei, raportat la sistemul de referință al camerei 1.
P_world_cam2	-	Punct oarecare aflat în spațiul scenei, raportat la sistemul de referință al camerei 2.
Pk	-	Punctul k.
Pk_im	-	Proiecția punctului Pk_world pe planul imagine, raportat la sistemul de referință al imaginii.
Pk_im_cam	-	Proiecția punctului Pk_world pe planul imagine, raportat la sistemul de referință al camerei.
Pk_im1	-	Coordonatele proiecției punctului Pk_world pe planul imaginii 1, raportate la sistemul de referință al imaginii 1.
Pk_im1_cam1	-	Coordonatele proiecției punctului Pk_world pe planul imaginii 1, raportate la sistemul de referință al camerei 1.
Pk_im2	-	Coordonatele proiecției punctului Pk_world pe planul imaginii 2, raportate la sistemul de referință al imaginii 2.
Pk_im2_cam2	-	Coordonatele proiecției punctului Pk_world pe planul imaginii 2, raportate la sistemul de referință al camerei 2.
Pk_world	-	Centrul cercului k.
Pk_world_cam	-	Punctul k aflat în spațiul scenei, raportat la sistemul de referință al camerei.
Pk_world_cam1	-	Centrul cercului k, raportat la sistemul de referință al camerei 1.

## **XII** Notatii, abrevieri, acronime

---

Pk_world_cam2	-	Centrul cercului $k$ , raportat la sistemul de referință al camerei 2.
Pk_world_offset_cam1	-	Centrul cercului $k$ , raportat la sistemul de referință offset_cam1.
Pk_world_offset_cam2	-	Centrul cercului $k$ , raportat la sistemul de referință offset_cam2.
R_x	-	Matrice de rotație în jurul axei $x$ .
R_y	-	Matrice de rotație în jurul axei $y$ .
R_z	-	Matrice de rotație în jurul axei $z$ .
sk_cam	[mm / pixel]	Factor de scalare corespunzător obiectului $k$ .
sk_cam1	[mm / pixel]	Factor de scalare corespunzător camerei 1, ce reprezintă raportul dintre diametrul real al cercului $k$ și proiecția acestuia în imaginea 1.
sk_cam1_x	[mm / pixel]	Factor de scalare pe axa $x$ corespunzător camerei 1, ce reprezintă raportul dintre diametrul real al cercului $k$ și proiecția acestuia în imaginea 1.
sk_cam1_y	[mm / pixel]	Factor de scalare pe axa $y$ corespunzător camerei 1, ce reprezintă raportul dintre diametrul real al cercului $k$ și proiecția acestuia în imaginea 1.
sk_cam2	[mm / pixel]	Factor de scalare corespunzător camerei 2, ce reprezintă raportul dintre diametrul real al cercului $k$ și proiecția acestuia în imaginea 2.
sk_cam2_x	[mm / pixel]	Factor de scalare pe axa $x$ corespunzător camerei 2, ce reprezintă raportul dintre diametrul real al cercului $k$ și proiecția acestuia în imaginea 2.
sk_cam2_y	[mm / pixel]	Factor de scalare pe axa $y$ corespunzător camerei 2, ce reprezintă raportul dintre diametrul real al cercului $k$ și proiecția acestuia în imaginea 2.
size_im	[pixel x pixel]	Dimensiunea imaginii.
size_im1	[pixel x pixel]	Dimensiunea imaginii 1.
size_im2	[pixel x pixel]	Dimensiunea imaginii 2.



t_x	[mm]	Translație de-a lungul axei x.
t_y	[mm]	Translație de-a lungul axei y.
t_z	[mm]	Translație de-a lungul axei z.
x_M_im	[pixel]	Coordonata punctului M_im pe axa x, raportat la sistemul de referință al imaginii.
x_M_im1	[pixel]	Coordonata punctului M_im1 pe axa x, raportată la sistemul de referință al imaginii 1.
x_M_im2	[pixel]	Coordonata punctului M_im2 pe axa x, raportată la sistemul de referință al imaginii 2.
x_P	-	Proiecția punctului P pe axa x.
x_P_im	[pixel]	Proiecția punctului P_im pe axa x.
x_P_im_cam	[pixel]	Proiecția punctului P_im_cam pe axa x.
x_P_im1	[pixel]	Proiecția punctului P_im1 pe axa x.
x_P_im1_cam1	[pixel]	Proiecția punctului P_im1_cam1 pe axa x.
x_P_im2	[pixel]	Proiecția punctului P_im2 pe axa x.
x_P_im2_cam2	[pixel]	Proiecția punctului P_im2_cam2 pe axa x.
x_P_world	[mm]	Proiecția punctului P_world pe axa x.
x_P_world_cam	[mm]	Proiecția punctului P_world_cam pe axa x.
x_P_world_cam1	[mm]	Proiecția punctului P_world_cam1 pe axa x.
x_P_world_cam2	[mm]	Proiecția punctului P_world_cam2 pe axa x.
x_Pk	-	Proiecția punctului Pk pe axa x.
x_Pk_im	[pixel]	Proiecția punctului Pk_im pe axa x.
x_Pk_im_cam	[pixel]	Proiecția punctului Pk_im_cam pe axa x.
x_Pk_im1	[pixel]	Proiecția punctului Pk_im1 pe axa x.
x_Pk_im1_cam1	[pixel]	Proiecția punctului Pk_im1_cam1 pe

#### **XIV** Notatii, abrevieri, acronime

---

$x_{Pk\_im2}$	[pixel]	Proiecția punctului $Pk\_im2$ pe axa x.
$x_{Pk\_im2\_cam2}$	[pixel]	Proiecția punctului $Pk\_im2\_cam2$ pe axa x.
$x_{Pk\_world}$	[mm]	Proiecția punctului $Pk\_world$ pe axa x.
$x_{Pk\_world\_cam}$	[mm]	Proiecția punctului $Pk\_world\_cam$ pe axa x.
$x_{Pk\_world\_cam1}$	[mm]	Proiecția punctului $Pk\_world\_cam1$ pe axa x.
$x_{Pk\_world\_cam2}$	[mm]	Proiecția punctului $Pk\_world\_cam2$ pe axa x.
$x_{Pk\_world\_offset\_cam1}$	[mm]	Proiecția punctului $Pk\_world\_offset\_cam1$ pe axa x.
$x_{Pk\_world\_offset\_cam2}$	[mm]	Proiecția punctului $Pk\_world\_offset\_cam2$ pe axa x.
$x\_size\_im$	[pixel]	Dimensiunea imaginii pe axa x.
$x\_size\_im1$	[pixel]	Dimensiunea imaginii 1 pe axa x.
$x\_size\_im2$	[pixel]	Dimensiunea imaginii 2 pe axa x.
$y\_M\_im$	[pixel]	Coordonata punctului $M\_im$ pe axa y, raportat la sistemul de referință al imaginii.
$y\_M\_im1$	[pixel]	Coordonata punctului $M\_im1$ pe axa y, raportată la sistemul de referință al imaginii 1.
$y\_M\_im2$	[pixel]	Coordonata punctului $M\_im2$ pe axa y, raportată la sistemul de referință al imaginii 2.
$y\_P$	-	Proiecția punctului P pe axa y.
$y\_P\_im$	[pixel]	Proiecția punctului $P\_im$ pe axa y.
$y\_P\_im\_cam$	[pixel]	Proiecția punctului $P\_im\_cam$ pe axa y.
$y\_P\_im1$	[pixel]	Proiecția punctului $P\_im1$ pe axa y.
$y\_P\_im1\_cam1$	[pixel]	Proiecția punctului $P\_im1\_cam1$ pe axa y.
$y\_P\_im2$	[pixel]	Proiecția punctului $P\_im2$ pe axa y.
$y\_P\_im2\_cam2$	[pixel]	Proiecția punctului $P\_im2\_cam2$ pe axa y.
$y\_P\_world$	[mm]	Proiecția punctului $P\_world$ pe

		axa y.
y_P_world_cam	[mm]	Proiecția punctului P_world_cam pe axa y.
y_P_world_cam1	[mm]	Proiecția punctului P_world_cam1 pe axa y.
y_P_world_cam2	[mm]	Proiecția punctului P_world_cam2 pe axa y.
y_Pk	-	Proiecția punctului Pk pe axa y.
y_Pk_im	[pixel]	Proiecția punctului Pk_im pe axa y.
y_Pk_im_cam	[pixel]	Proiecția punctului Pk_im_cam pe axa y.
y_Pk_im1	[pixel]	Proiecția punctului Pk_im1 pe axa y.
y_Pk_im1_cam1	[pixel]	Proiecția punctului Pk_im1_cam1 pe axa y.
y_Pk_im2	[pixel]	Proiecția punctului Pk_im2 pe axa y.
y_Pk_im2_cam2	[pixel]	Proiecția punctului Pk_im2_cam2 pe axa y.
y_Pk_world	[mm]	Proiecția punctului Pk_world pe axa y.
y_Pk_world_cam	[mm]	Proiecția punctului Pk_world_cam pe axa y.
y_Pk_world_cam1	[mm]	Proiecția punctului Pk_world_cam1 pe axa y.
y_Pk_world_cam2	[mm]	Proiecția punctului Pk_world_cam2 pe axa y.
y_Pk_world_offset_cam1	[mm]	Proiecția punctului Pk_world_offset_cam1 pe axa y.
y_Pk_world_offset_cam2	[mm]	Proiecția punctului Pk_world_offset_cam2 pe axa y.
y_size_im	[pixel]	Dimensiunea imaginii pe axa y.
y_size_im1	[pixel]	Dimensiunea imaginii 1 pe axa y.
y_size_im2	[pixel]	Dimensiunea imaginii 2 pe axa y.

## Lista de tabele

Tab. 3. 1. Plan de cercetare – obiective, activități și rezultate.....	28
Tab. 4. 1. Obiectivele urmărite în experimentele 1, 2 și 3 .....	31
Tab. 4. 2. Semnificația parametrilor și unitatea de măsură.....	32
Tab. 4. 3. Imaginile utilizate în experimentul 1 .....	33
Tab. 4. 4. Valorile parametrilor la care au fost prelevate imaginile. ....	35
Tab. 4. 5 Valorile RGB corespunzătoare frunzelor de roșie, ardei și vânăță. ....	35
Tab. 4. 6. Valorile RGB corespunzătoare frunzelor de roșie, ardei și vânăță. ....	36
Tab. 4. 7. Valorile RGB corespunzătoare frunzelor de roșie, ardei și vânăță. ....	37
Tab. 4. 8. Valorile RGB corespunzătoare frunzelor de roșie, ardei și vânăță. ....	38
Tab. 4. 9. Imaginile utilizate în experimentul 2 .....	40
Tab. 4. 10. Semnificația simbolurilor utilizate în experimentul 2.....	41
Tab. 4. 11. Diferențierea speciilor de roșie, ardei și vânăță .....	44
Tab. 4. 12. Imaginile utilizate în experimentul 3 .....	45
Tab. 4. 13. Analiză comparativă a concluziilor desprinse din experimentele 1, 2 și 3 .....	52
Tab. 5. 1. Intervalele de circularitate corespunzătoare formelor: pătrat, cerc și triunghi .....	54
Tab. 5. 2. Explicarea funcțiilor programului .....	54
Tab. 5. 3. Explicarea principalelor funcții ale programului.....	57
Tab. 5. 4. Numărul punctelor de inflexiune corespunzătoare semnăturilor frunzelor de roșie, ardei și vânăță .....	58
Tab. 5. 5. Explicarea funcțiilor utilizate la comanda robotului.....	63
Tab. 7. 1. Tipurile de camere și caracteristicile principale .....	81
Tab. 7. 2. Imaginile utilizate în experiment .....	82
Tab. 7. 3. Camera 1 – Erorile distanțelor focale. ....	83
Tab. 7. 4. Camera 2 – Erorile distanțelor focale. ....	83
Tab. 7. 5. Camera 3 – Erorile distanțelor focale. ....	84
Tab. 7. 6. Camera 4 – Erorile distanțelor focale. ....	84
Tab. 7. 7. Camera 5 – Erorile distanțelor focale. ....	85
Tab. 7. 8. Avantajele și dezavantajele metodei de calibrare extrinsecă .....	85
Tab. 7. 9. Etapele de elaborare a modelului matematic.....	88
Tab. 7. 10. Tipurile de camere și caracteristicile principale .....	106
Tab. 8. 1. Determinarea ecuației planului pe baza mai multor imagini .....	119
Tab. 8. 2. Variante de stabilire a semnelor coeficienților .....	120
Tab. 8. 3. Verificarea preciziei .....	120
Tab. 9. 1. Detectarea și urmărirea rândului de culturi .....	132
Tab. 10. 1. Tabelul Denavit-Hartenberg .....	139

## Lista de figuri

Fig. 1. 1. Specii de buruieni cu rădădini adânci .....	2
Fig. 1. 2. Concept de robot pentru combaterea mecanică a buruienilor cu rădăcini adânci .....	2
Fig. 2. 1. Structura roboților mobili pentru agricultură .....	5
Fig. 2. 2. Camera stereo ECO-DAN [Möller, 2010] .....	6
Fig. 2. 3. Cinci imagini prelevate din diferite unghiuri [Wang, 2009] .....	12
Fig. 2. 4. Modelarea punctelor omoloage prin constrângeri epipolare [Wang, 2009] .....	12
Fig. 2. 5. „Light-curtain” [Ruckelshausen, 2007] .....	12
Fig. 2. 6. Prășitoare pentru cereale, ghidată prin vedere artificială [Home, 2002] ..	14
Fig. 2. 7. Disc rotativ prevăzut cu o secțiune pentru protejarea platelor cultivate [Tillett, 2007] .....	14
Fig. 2. 8. Prășitoare cicloidală „Querhacke” concepută la Universitatea din Osnabrück, Germania [Griepentrog, 2007] .....	14
Fig. 2. 9. Robot autonom pentru combaterea buruienilor conceput la Universitatea din Halmstad, Suedia [*** Img, Lukas] .....	16
Fig. 2. 10. Robot autonom pentru combaterea buruienilor conceput la Institutul Danez de Științe Agricole, Danemarca [Slaughter, 2008] .....	16
Fig. 2. 11. Robot autonom pentru combaterea buruienilor conceput la Universitatea din Wageningen, Olanda [Bakker, 2011] .....	17
Fig. 2. 12. Robot autonom pentru protejarea culturilor agricole conceput la Universitatea din Aarhus, Danemarca [*** Img, Hortibot] .....	17
Fig. 2. 13. Robot autonom pentru combaterea buruienilor „Weedy” conceput la Universitatea din Osnabrück, Germania [*** Img, Weedy] .....	18
Fig. 2. 14. Robot autonom pentru agricultură conceput la Universitatea din Osnabrück în colaborare cu firmele Amazone și Bosch [*** Img, BoniRob] .....	18
Fig. 2. 15. Robotul „Helios” conceput la Universitatea din Braunschweig, Germania	19
Fig. 2. 16. Robotul „Amaizeing” conceput la Universitatea din Osnabrück, Germania .....	19
Fig. 2. 17. Robotul „4M” conceput la Universitatea din Helsinki, Finlanda .....	19
Fig. 2. 18. Robotul „Eyesonic” conceput la Universitatea din Wageningen, Olanda .	19
Fig. 2. 19. Robotul „Eduro Maxi” conceput la Universitatea din Praga, Cehia .....	19
Fig. 2. 20. Robotul „Optimaize Prime” conceput la Universitatea din Osnabrück, Germania .....	19
Fig. 2. 21. Geometria camerei pinhole .....	21
Fig. 2. 22. Corelația între sistemul de referință <i>Ref1</i> și sistemul de referință <i>Ref2</i> .	23
Fig. 4. 1. Conversia din intensitate luminoasă în tensiune [Tilneac, 2009] .....	29
Fig. 4. 2. Schema standului experimental .....	30
Fig. 4. 3. Standul experimental .....	30
Fig. 4. 4. Eșantionarea valorilor RGB [Tilneac, 2010a] .....	32
Fig. 4. 5. Grafice de corelare a valorilor R, G, B cu tensiunile fotodiodei corespunzătoare datelor din Tab.4.4. ....	35
Fig. 4. 6. Grafice de corelare a valorilor R, G, B cu tensiunile fotodiodei corespunzătoare datelor din Tab.4.5. ....	36

## **XVIII** Lista de figuri

---

Fig. 4. 7. Grafice de corelare a valorilor R, G, B cu tensiunile fotodiodei corespunzătoare datelor din Tab.4.6. ....	37
Fig. 4. 8. Grafice de corelare a valorilor R, G, B cu tensiunile fotodiodei corespunzătoare datelor din Tab.4.7. ....	38
Fig. 4. 9. Analiza influenței luminii asupra componentei de culoare „R” .....	41
Fig. 4. 10. Analiza influenței luminii asupra componentei de culoare „G” .....	42
Fig. 4. 11. Analiza influenței luminii asupra componentei de culoare „B” .....	42
Fig. 4. 12. Analiza influenței luminii asupra componentei de culoare „H” .....	43
Fig. 4. 13. Analiza influenței luminii asupra componentei de culoare „S” .....	43
Fig. 4. 14. Analiza influenței luminii asupra componentei de culoare „V” .....	44
Fig. 4. 15. Analiza influenței luminii asupra componentei de culoare „R” .....	46
Fig. 4. 16. Analiza influenței luminii asupra componentei de culoare „G” .....	46
Fig. 4. 17. Analiza influenței luminii asupra componentei de culoare „B” .....	46
Fig. 4. 18. Interfața grafică a programului de procesare a culorilor.....	47
Fig. 4. 19. Încărcarea imaginii .....	48
Fig. 4. 20. Segmentarea imaginii utilizând spațiul de culoare HSV .....	49
Fig. 4. 21. Afișarea imaginilor RGB și HSV; afișarea spațiului de culoare HSV .....	49
Fig. 4. 22. Selectarea frunzei și calculul valorii medii a componentelor de culoare a pixelilor corespunzători frunzei .....	49
Fig. 4. 23. Interfață grafică de afișare a rezultatelor statistice .....	50
Fig. 4. 24. Experiment 1 – Studiul eficienței de recunoaștere a plantelor pe baza componentelor de culoare R, G, B .....	50
Fig. 4. 25. Experiment 2 – Studiul eficienței de recunoaștere a plantelor pe baza componentelor de culoare R, G, B sau H, S, V .....	51
Fig. 4. 26. Experiment 3 – Recunoașterea frunzei de ardei pe baza componentei „G” .....	51
Fig. 5. 1. Structura aplicației MATLAB pentru recunoașterea obiectelor .....	53
Fig. 5. 2. Obiectele utilizate în experiment .....	54
Fig. 5. 3. Explicarea funcției detectare_culoare_forma .....	54
Fig. 5. 4. Procesarea imaginii pentru detectarea culorilor și a formelor.....	55
Fig. 5. 5. Tipuri de margini ale frunzelor [*** Img, Leaf] .....	55
Fig. 5. 6. Frunze de roșie, ardei și vânăță (de la stânga la dreapta).....	56
Fig. 5. 7. Distanțele de la centrul de greutate până la fiecare pixel care corespunde marginii frunzei în imagine. Detalii în anexa B. ....	56
Fig. 5. 8. Structura aplicației MATLAB (concept aflat în fază de studiu) .....	56
Fig. 5. 9. Etapele de determinare a semnăturii frunzei .....	57
Fig. 5. 10. Semnături specifice conturului frunzelor de roșie .....	58
Fig. 5. 11. Semnături specifice conturului frunzelor de ardei .....	59
Fig. 5. 12. Semnături specifice conturului frunzelor de vânăță .....	59
Fig. 5. 13. Recunoașterea frunzei de roșie.....	60
Fig. 5. 14. Recunoașterea frunzei de ardei .....	61
Fig. 5. 15. Recunoașterea frunzei de vânăță .....	61
Fig. 5. 16. Detectarea buruienilor.....	61
Fig. 5. 17. Schema standului experimental.....	62
Fig. 5. 18. Aplicație robotizată de selectare a frunzelor .....	63
Fig. 5. 19. Interfața grafică a programului de învățare a formelor .....	64
Fig. 5. 20. Procedeele de determinare a parametrilor $D_n$ ( $D_1 \div D_{10}$ ) .....	65
Fig. 5. 21. Setarea parametrilor F1, F2, F3, F4, F5 corespunzători frunzelor de roșie, ardei, respectiv vânăță .....	65
Fig. 5. 22. Procesarea imaginii.....	66

Fig. 5. 23. Recunoașterea frunzelor.....	66
Fig. 6. 1. Exemple de proiecții ale cercului. Lungimea axei mari a elipsei rămâne constantă, dacă centrul cercului este deplasat în același plan, care este perpendicular pe axa optică a camerei video. ....	67
Fig. 6. 2. Reprezentare geometrică simplificată a relațiilor imagine-scenă .....	69
Fig. 6. 3. Reprezentare geometrică simplificată a unui sistem stereo binocular ....	70
Fig. 6. 4. Interfața grafică a programului utilizat la măsurarea distanțelor prin vedere stereo .....	74
Fig. 7. 1. Obiecte geometrice utilizate pentru calibrarea intrinsecă .....	75
Fig. 7. 2. Modelul geometric pentru determinarea distanței focale .....	76
Fig. 7. 3. Interfața grafică .....	80
Fig. 7. 4. Obiecte geometrice utilizate pentru calibrare .....	86
Fig. 7. 5. Modelul geometric pentru calibrarea extrinsecă [Tilneac, 2011b].....	87
Fig. 7. 6. Imagine prelevată de la camera 1 (a); imagine prelevată de la camera 2 (b).....	102
Fig. 7. 7. Interfață grafică - START .....	102
Fig. 7. 8. Interfața grafică – STEP1 .....	103
Fig. 7. 9. Interfața grafică – STEP2 .....	103
Fig. 7. 10. Interfața grafică – STEP3.....	104
Fig. 7. 11. Corespondență puncte - STEP4 (camera 1).....	104
Fig. 7. 12. Corespondență puncte - STEP4 (camera 2).....	105
Fig. 7. 13. Rezultatele calibrării extrinseci .....	105
Fig. 7. 14. Verificarea preciziei de calibrare extrinsecă .....	106
Fig. 7. 15. Imagini prelevate de la camerele video 1 și 2.....	107
Fig. 7. 16. Calibrarea extrinsecă a camerelor video utilizând trei cercuri de dimensiune cunoscută .....	107
Fig. 7. 17. Câmpul vizual al camerei Logitech C200 .....	108
Fig. 7. 18. Câmpul vizual al camerei 3D-ToF pmd-ifm.....	108
Fig. 7. 19. Procesarea imaginii 1 .....	109
Fig. 7. 20. Procesarea imaginii 2 .....	110
Fig. 7. 21. Măsurarea parametrilor extrinseci în programul CATIA [Tilneac, 2011a] .....	112
Fig. 7. 22. Verificarea preciziei de calibrare .....	113
Fig. 7. 23. Calibrarea extrinsecă a camerelor video Logitech C200 în mediul CATIA V5 .....	114
Fig. 7. 24. Calibrarea extrinsecă a camerelor 3D-ToF și Logitech C200 în mediul CATIA V5.....	114
Fig. 7. 25. Calibrarea extrinsecă a camerelor video 3D-ToF în mediul CATIA V5 ..	114
Fig. 8. 1. Stand de măsurare a înălțimilor prin vedere stereo.....	115
Fig. 8. 2. Procesarea imaginii în mediul MATLAB .....	117
Fig. 8. 3. Interfața grafică de setări a programului <i>ECUATIA_PLANULUI</i> .....	118
Fig. 8. 4. Interfața grafică pentru procesarea imaginii și afișarea valorilor coeficienților corespunzători ecuației planului .....	118
Fig. 8. 5. Procesarea imaginilor în procesul de diferențiere buruiună/plantă.....	123
Fig. 8. 6. Rezultatele experimentului de diferențiere buruiună/plantă cultivată....	124
Fig. 9. 1. Corelarea imaginilor prelevate de la camera 2D-color cu imagini prelevate de la camera 3D-ToF .....	126

## XX Lista de figuri

---

Fig. 9. 2. Imagini prelevate de la camera 2D color (a) și imagini prelevate de la camera 3D-ToF (b) și (c) .....	126
Fig. 9. 3. Interfața grafică pentru lansarea programului .....	127
Fig. 9. 4. Fereastra de lucru pentru setarea parametrilor de intrare.....	127
Fig. 9. 5. Analiza unei plante de roșie .....	128
Fig. 9. 6. Analiza unei plante de ardei.....	128
Fig. 9. 7. Analiza unei plante de vânăță .....	129
Fig. 9. 8. Concept de robot pentru agricultură dotat cu un sistem de camere video format din o cameră 2D-color și o cameră 3D-ToF .....	129
Fig. 9. 9. Camerele 3D amplasate pe robotul „Optimaize Prime” (Hochschule Osnabrück) .....	130
Fig. 9. 10. Camera 3D-ToF (dreapta) și o imagine prelevată de la această cameră .....	130
Fig. 9. 11. Geometria raportului imagine-scenă corespunzător camerei 3D-ToF...	131
Fig. 9. 12. Imagini prelevate de la sistemul de camere 3D-ToF .....	133
Fig. 9. 13. Imagini prelevate în timpul experimentului efectuat la „Hochschule Osnabrück” .....	133
Fig. 10. 1. Structura sistemului mecanic a robotului [Kovacs, 2001].....	135
Fig. 10. 2. Structura dispozitivului de ghidare a robotului [Kovacs, 2001] .....	135
Fig. 10. 3. Schema cinematică și spațiul de lucru al robotului .....	136
Fig. 10. 4. Schema cinematică a mecanismului de orientare .....	136
Fig. 10. 5. Situația robotului în spațiul de lucru .....	137
Fig. 10. 6. Sistemele de coordonate atașate axelor robotului .....	139
Fig. 10. 7. Robotul <i>GardenerRob</i> modelat în mediul CATIA V5 .....	142
Fig. 10. 8. Dimensiunile de gabarit ale robotului <i>GardenerRob</i> .....	143
Fig. 10. 9. Cursele dispozitivului de translație pe verticală și orizontală .....	143



# 1 Introducere

## 1.1 Motivația abordării tematicii alese

Aspectele ecologice din agricultură reprezintă o problemă majoră de actualitate. Ideea ce stă la baza acestei cercetări este aceea de a aduce contribuții în dezvoltarea sistemelor mecatronice destinate combaterii automate a buruienilor de pe terenurile agricole. În multe zone ale lumii, lucrătorii agricoli sunt nevoiți să muncească în condiții de muncă precară, cauzate de condițiile meteo nefavorabile, de suprasolicitarea fizică, și de mediul toxic la care lucrătorii sunt expuși din cauza utilizării abundente a pesticidelor. Erbicidele sunt utilizate în proporție de 72% din totalul de pesticide utilizate în culturile agricole [Meyer, 2011].

La nivel mondial sunt căutate soluții pentru creșterea productivității în agricultura ecologică. În acest scop este adus aportul automatizării și robotizării proceselor în domeniul agricol. În anul 2006, Pedersen et al. au făcut un studiu orientat pe fezabilitatea economică de utilizare a vehiculelor autonome robotizate în comparație cu sistemele convenționale. Rezultatele au arătat că aplicațiile robotizate sunt mult mai economice [Pedersen, 2006]. Combaterea buruienilor este o problemă importantă în producția culturilor agricole [Slaughter, 2008]. Pagubele provocate de către buruieni plantelor de cultură sunt aproape egale cu cele cauzate de boli și dăunători [Manea, 2002]. În publicația „Agrotehnică și Herbologie” [Manea, 2002], autorul a clasificat aspectele privind pagubele aduse agriculturii de către buruieni. Buruienile contribuie la diminuarea cantitativă a recoltelor din următoarele cauze [Manea, 2002]:

- Buruienile consumă apa din stratul arabil și subarabil, concurând cu plantele de cultură și amplificând acțiunea secetei.
- Buruienile consumă mari cantități de elemente nutritive, în detrimentul plantelor cultivate.
- Buruienile stânjenesc creșterea rădăcinilor plantelor cultivate cu care conviețuiesc.
- Buruienile parazite își extrag hrana prin intermediul haustoriilor din tulpinile sau rădăcinile plantelor de cultură.
- Buruienile umbresc plantele de cultură și solul.
- Înfășurându-se în jurul tulpinilor, unele specii de buruieni stânjenesc creșterea și dezvoltarea plantelor.
- Buruienile, prin dezvoltarea lor, stânjenesc procesele microbiologice din sol.
- Buruienile sunt plante gazdă pentru multe boli și mulți dăunători.

Plivirea manuală a buruienilor durează între 50-300 de ore pe hectar, atunci când această activitate este efectuată de către un grădinar [Jørgensen, 2007]. Comercializarea unor mașini automate pentru combaterea buruienilor, ar aduce beneficii atât prin diminuarea efortului fizic al grădinarilor, cât și prin reducerea costurilor [Jørgensen, 2007]. Din această cauză sunt efectuate cercetări pentru găsirea unor soluții cât mai eficiente pentru combaterea acestora.

## 2 Introducere - 1



*Amaranthus retroflexus*



*Setaria glauca*



*Setaria verticillata*

Fig. 1. 1. Specii de buruieni cu rădăcini adânci

Autoarea tezei de doctorat și-a propus să realizeze un robot (Fig. 1. 2) pentru combaterea mecanică a buruienilor care au rădăcini adânci, precum știrul (*amaranthus retroflexus*) și mohorul (*setaria glauca* și *setaria verticillata*). Aceste două specii de buruieni sunt foarte răspândite pe terenurile agricole din România (Fig. 1. 1). Robotul trebuie să aibă capacitatea de a face distincția între planta cultivată și buruiană.

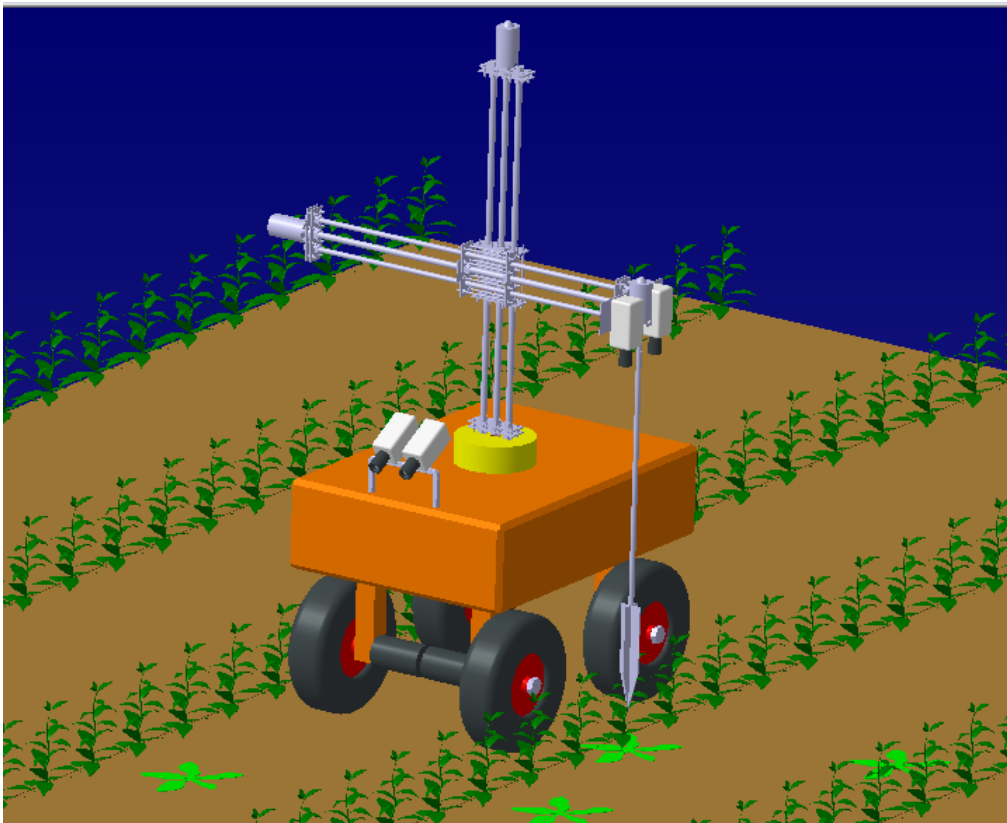


Fig. 1. 2. Concept de robot pentru combaterea mecanică a buruienilor cu rădăcini adânci

Robotul pentru combaterea buruienilor trebuie să îndeplinească trei funcții de bază: (1) navigarea autonomă pe terenurile agricole; (2) detectarea buruienilor; (3) distrugerea mecanică a buruienilor. Din analiza critică a literaturii de specialitate rezultă că fiecare dintre aceste trei funcții este insuficient dezvoltată. Din această cauză, la momentul actual nu există nici un robot comercial pentru combaterea mecanică a buruienilor. Prototipurile unor astfel de roboți sunt încă în fază de testare în institute de cercetare și universități. Există, totuși, câteva mașini agricole, parțial automatizate, care au fost deja comercializate [\*\*\* Img, T&H]. Principalul obstacol în dezvoltarea acestor tipuri de roboți constă în incapacitatea de procesare eficientă a imaginilor în timp real. Pe terenurile agricole există o varietate foarte mare de forme. Imaginile prelevate de la camerele video urmează a fi procesate digital. Procesarea imaginilor prelevate de pe terenurile agricole este o operație extrem de dificilă din următoarele cauze: formele plantelor sunt observate din perspectivă; plantele sunt parțial vizibile în imagine; uneori plantele sunt denaturate de mediu sau de insecte; iluminarea nu este constantă; apar umbre; vizibilitatea este îngreunată de ceață sau praf. Din acest motiv, teza de doctorat este focalizată pe vedere artificială.

## 1.2 Structura tezei de doctorat

În lucrare sunt prezentate rezultatele unor experimente de recunoaștere a speciilor de plante pe baza culorilor și a formelor frunzelor. În teză este prezentat un studiu de stabilire a interdependenței culoare-luminozitate pentru frunze aparținând unor specii diferite de plante (roșie, ardei, vânăță). Acest studiu este util pentru adaptarea pragului de binarizare a imaginii în funcție de luminozitatea din mediul ambiant. Stereoscopia este tema cea mai des abordată în teza de doctorat, deoarece prin această tehnică pot fi obținute simultan informații despre culoare și distanță. Aceste informații sunt necesare atât pentru a face diferențierea între planta cultivată și buruiiană, cât și pentru detectarea rândurilor de culturi. Calibrarea camerelor este prima etapă în utilizarea sistemelor stereoscopice. În lucrare sunt prezentate trei metode originale de calibrare a camerelor video.

Capitolul 1, *Introducere*, prezintă motivația abordării tematicii alese și structura tezei de doctorat.

Capitolul 2, *Sinteze bibliografice privind tematica tezei de doctorat*, este structurat în două subcapitole. Primul subcapitol prezintă o sinteză bibliografică privind stadiul actual de utilizare a roboților mobili în aplicații de combatere a buruienilor. Al doilea subcapitol prezintă o sinteză bibliografică privind metodele de calibrare a camerelor video.

Capitolul 3, *Obiectivele și planul de cercetare*, prezintă obiectivele cercetării și planul de cercetare.

Capitolul 4, *Contribuții la recunoașterea speciilor de plante pe baza culorilor*, prezintă rezultatele unor studii privind influența luminii asupra culorilor plantelor. Sunt descrise două experimente prin care au fost determinate interdependențele între valorile tensiunilor unei fotodiode și valorile componentelor de culoare în spațiile RGB, respectiv HSV.

Capitolul 5, *Contribuții la recunoașterea speciilor de plante pe baza formelor*, prezintă aplicații de recunoaștere a speciilor de plante pe baza caracteristicilor geometrice ale frunzelor.

Capitolul 6, *Localizarea obiectelor în spațiul tridimensional*, prezintă două tehnici de localizare a unui punct din scenă în raport cu sistemul de referință al camerei video. Prima tehnică prezintă o modalitate de a localiza poziția centrului

#### 4 Introducere - 1

---

unui cerc al cărui diametru este cunoscut. A doua tehnică prezintă o modalitate de localizare a unui punct din scenă, utilizând un sistem de vedere stereo binocular. Tot în acest capitol este prezentată o metodă de determinare a ecuației planului corespunzător solului.

Capitolul 7, *Calibrarea camerelor video*, prezintă contribuții în dezvoltarea a trei metode originale de calibrare a camerelor video. Prima metodă este utilizată la calibrarea intrinsecă. Celelalte două metode sunt utilizate la calibrarea extrinsecă a camerelor video. Metodele dezvoltate au avantajul că pot fi utilizate atât în laborator, cât și pe terenurile agricole.

Capitolul 8, *Diferențierea buruiană/plantă pe bază de măsurări optice tridimensionale*, prezintă o tehnică de măsurare a înălțimii plantelor prin utilizarea unui sistem de vedere stereo binocular. Diferențierea buruiană/plantă este realizată pe baza diferențelor de înălțime.

Capitolul 9, *Utilizarea camerei 3D-ToF (Time of Flight) pentru aplicații în agricultură*, prezintă exemple de aplicabilitate a camerei 3D-ToF. Prin corelarea unei camere 3D-ToF cu o cameră 2D-color pot fi obținute simultan informații despre culoarea și distanța la care se află un punct de pe suprafața unei plante. Rândurile de culturi pot fi detectate pe baza informațiilor de distanță prelevate de la două camere 3D-ToF.

Capitolul 10, *Proiectarea și realizarea robotului GardenerRob pentru aplicații în agricultură*, prezintă un concept de robot mobil pentru aplicații de combatere a buruienilor prin sisteme mecanice.

Capitolul 11 prezintă concluziile finale, contribuțiile personale, și direcțiile viitoare de cercetare.

## 2 Sinteze bibliografice privind tematica tezei de doctorat

### 2.1 Stadiul actual de utilizare a roboților mobili în aplicații de combatere a buruienilor

#### 2.1.1 Introducere

Evoluțiile tehnologice din ultimii ani permit dezvoltarea de sisteme robotizate pentru aplicații în agricultură. Astfel de sisteme aduc avantaje datorită posibilității de diminuare a timpului de expunere a lucrătorilor agricoli la praf, zgomot și pesticide [Griepentrog, 2010]. Automatizarea mașinilor agricole este văzută ca un mijloc de reducere a costurilor de cultivare a terenurilor agricole [Bakker, 2009b]. O succintă trecere în revistă a studiilor efectuate în domeniul roboților pentru agricultură arată că interesul pentru asemenea realizări este în continuă creștere. După decenii de cercetare și dezvoltare, a fost atins un nivel ridicat de automatizare.

Literatura de specialitate este bogată în metode de vedere artificială pentru identificarea speciilor de plante [Meyer, 2011]. Controlul automat al buruienilor este un domeniu de înaltă tehnologie [Oberndorfer, 2006]. Slaughter et al. au publicat în anul 2008 o sinteză a stadiului actual privind utilizarea sistemelor automate în aplicații de combatere a buruienilor [Slaughter, 2008]. Meyer et al. (2001) au publicat o clasificare a metodelor de recunoaștere a frunzelor [Meyer, 2011]. Sørensen et al. (2009) au publicat o structură conceptuală de proiectare a roboților utilizați în aplicații de protecție a plantelor [Sørensen, 2009]. Chiar și în România, în anul 1971 a existat un interes pentru un astfel de mediu tehnic, manifestat de doi ingineri, care însă nu au putut edifica un astfel de sistem din cauză că soluția a fost cotelată ca fiind de domeniul fanteziei [Tănăsioiu, 1995]. Fig. 2. 1 prezintă structura unui robot mobil pentru agricultură

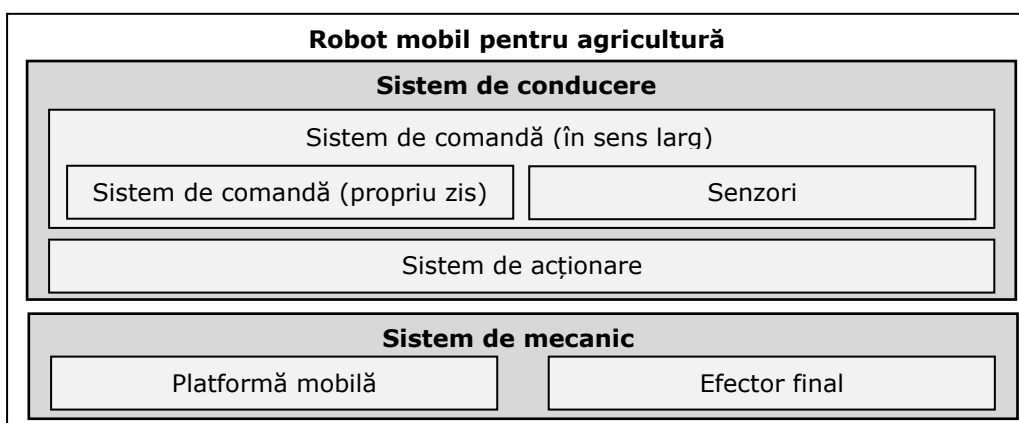


Fig. 2. 1. Structura roboților mobili pentru agricultură

### 2.1.2 Navigarea robotului printre rândurile de culturi

Senzorii cei mai des utilizați la navigarea robotului sunt: senzorii optici și GPS-ul. Numeroase cercetări au fost concentrate pe tehnica SLAM (Simultaneous Localisation And Mapping) utilizată la navigarea roboților mobili pentru prelevarea de informații perceptuale și pentru construirea sau actualizarea unor hărți locale în timpul navigării [Griepentrog, 2010], [Billingsley, 2008].

**Navigare prin vedere artificială.** În cele mai multe cazuri, sistemul de navigare prin vedere artificială are la bază procesarea digitală a imaginilor color. În anul 1998, Tian și Slaughter [Tian, 1998] au investigat metode pentru îmbunătățirea performanței tehnicilor de vedere artificială pentru operarea în diferite condiții de iluminare naturală. În anul 1999, Slaughter et al. [Slaughter, 1999], [Slaughter, 2000] au prezentat un vehicul ghidat prin vedere artificială utilizând segmentarea culorilor. Eroarea de poziționare a sistemului a fost de la 7 mm (în caz de densități reduse ale buruienilor) până la 12 mm (în caz de densități ridicate ale buruienilor), la viteze de deplasare de până la 16 km/h. În anul 2000, Tillett and Hague [Hague, 2000] au publicat o lucrare despre un vehiculul autonom pentru protecția plantelor. Kise et al. (2005) au dezvoltat un sistem de ghidare cu vedere stereo în infraroșu apropiat [Kise, 2005]. Eroarea laterală a sistemului a fost de 30-50mm în funcție de viteză și de curbura rândului. Metoda necesita existența unor zone fără buruieni, pentru a permite sistemului stereoscopic să detecteze punctele de navigare. Tot în anul 2005, Astrand și Baerveldt [Astrand, 2005] au dezvoltat un sistem de ghidare prin vedere artificială, care a fost robust la diverse înălțimi ale plantelor, și la densități mari ale buruienilor (200 buruieni/m<sup>2</sup>). Bakker et al. (2007) au publicat o metodă de detectare a rândurilor pe baza contrastului dintre plante și sol, utilizând imagini alb-negru [Bakker, 2007]. Jeon et al. (2011) au utilizat o metodă robustă de segmentare a plantelor și a buruienilor în condiții de iluminare necontrolată [Jeon, 2011].

În multe sisteme de navigare este utilizată vederea stereoscopică. Andersen et al. (2005) au utilizat vederea stereo pentru detectarea obstacolelor, introducând metoda PQ (*Projection and Quantification*) de reprezentare a norului de puncte corespunzător scenei [Andersen, 2005b]. Problema detectării obstacolelor a fost tratată și în [Zhao, 2008]. Kise et al. (2005) au atașat la tractor un sistem de vedere stereo pentru detectarea rândurilor [Kise, 2005]. Prin acest sistem, tractorul a fost ghidat în mod automat. De asemenea, Ortiz et al. (2006) și Blas et al. (2010) au utilizat vederea stereo pentru ghidare automată de-a lungul rândurilor de culturi [Ortiz, 2006], [Blas, 2010]. Camera stereo ECO-DAN (Fig. 2. 2) este una dintre primele camere stereo, special proiectate pentru aplicații agricole [Sørensen, 2005], [Möller, 2010]. Camera conține două chip-uri CMOS cu o rezoluție de 752x480 pixeli, amplasate la o distanță de 12 cm între axele optice ale ocularilor.

În sistemul de navigare pot fi utilizate nivelurile de gri ale imaginii pentru



Fig. 2. 2. Camera stereo ECO-DAN [Möller, 2010]

detectarea rândurilor [Astrand, 2002]. Benson et al. (2003) au utilizat o singură cameră video monocromă montată pe cabina combinei [Benson, 2003].

În cele mai multe cazuri, rândurile de culturi sunt detectate prin transformata Hough. Bakker et al. (2007) au utilizat o metodă de recunoaștere a rândurilor, care se bazează pe fuzionarea imaginilor procesate prin transformata Hough cu imagini pe nivele de gri, rezultând o îmbunătățire considerabilă a vitezei de procesare a imaginii [Bakker, 2007]. Hemming et al. (2002) au utilizat transformata Hough și o metodă de clustering pentru detectarea plantelor [Hemming, 2002]. Leemans et al. (2006) au utilizat clustere de detectare a liniilor folosind o variantă a transformatei Hough [Leemans, 2006]. Tellaèche et al. (2007) au utilizat imagini binare în care materialul vegetal (buruieni și culturi) este de culoare albă, iar restul (sol și pietre) este de culoare neagră. Rândurile de culturi din imagine au fost detectate folosind transformata Hough [Tellaèche, 2007]. Metode similare au fost utilizate de către Jiang et al. și Astrand et al. [Jiang, 2010] [Astrand, 2004].

Søgaard și Olsen (2003) au utilizat metoda regresiei liniare pentru determinarea poziției și orientării rândului [Søgaard, 2003]. Bossu et al. (2008) au studiat două tehnici de detectare a rândurilor de culturi (transformata wavelat și filtrul Gabor) [Bossu, 2008]. Guijarro et al. (2010) au utilizat o metodă de segmentare automată a texturilor relevante din imaginile terenurilor agricole [Guijarro, 2010]. Ericson și Astrand (2010) au detectat rândurile de pe terenul agricol, utilizând o cameră omnidirecțională [Ericson, 2010].

**Navigare prin GPS.** În aplicații de navigare a robotului este des utilizat senzorul RTK-GPS (Real-Time Kinematic Global Positioning Systems), care poate oferi un anumit grad de precizie de poziționare laterală de-a lungul rândului [Ciolac, 2003]. Utilizând RTK-GPS, unele tractoare automate pot naviga cu o precizie de 25 mm [Slaughter, 2008]. Ruckelshausen et al. (2009) utilizează RTK-DGPS pentru determinarea fenotipului plantelor, iar conceptul de navigare se bazează pe robotica probabilistică [Ruckelshausen, 2009].

Comparativ cu utilizarea vederii artificiale, utilizarea GPS-ului este avantajoasă prin faptul că nu depinde de aspectul vizual al culturilor, nu este afectat negativ de densitatea buruienilor, nu este afectat de umbre, nu este afectat de discontinuitățile rândului de culturi sau alte condiții care degradează performanțele sistemelor de ghidare prin vedere artificială. Utilizarea GPS-ului are dezavantajul că acesta poate fi utilizat doar în zone cu semnale radio. Pentru atingerea unei performanțe ridicate este necesar ca cel puțin patru sateliți să fie disponibili. De asemenea, distribuția sateliților pe cer influențează performanțele GPS-ului.

**Navigare prin utilizarea altor tipuri de senzori.** În ultimii ani au fost implementate tehnologii noi, bazate pe imagistica 3D. Feldkämper et al. (2010) au utilizat două camere 3D-ToF (Time of Flight) la navigarea robotului între rândurile de culturi [Feldkämper, 2010]. Weiss și Biber (2011) au utilizat senzor LIDAR pentru detectarea și cartografierea plantelor [Weiss, 2011], [Weiss, 2010a].

În publicațiile anuale ale concursului internațional „Field Robot Event” sunt prezentate diverse concepte de roboți mobili pentru agricultură. Publicațiile concursului pot fi accesate pe internet [\*\*\* FRE, 2004], [\*\*\* FRE, 2005], [\*\*\* FRE, 2006], [\*\*\* FRE, 2007], [\*\*\* FRE, 2008], [\*\*\* FRE, 2009], [\*\*\* FRE, 2010]. Din aceste lucrări științifice rezultă că senzorii cei mai utilizați pentru navigare sunt: senzori de distanță cu infraroșu, senzori de distanță ultrasonici, busole, giroscopae și senzori laser.



### 2.1.3 Recunoașterea speciilor de plante

Variatatea caracteristicilor vizuale utilizate pentru identificarea speciilor de plante este divizată în trei categorii: caracteristici de formă, caracteristici de culoare, și caracteristici de textură.

**Recunoașterea plantelor pe baza caracteristicilor de culoare și reflexie spectrală.** Numeroase studii au investigat utilizarea culorii și a reflexiei spectrale pentru recunoașterea speciilor de plante. Unul dintre cele mai mari avantaje ale acestor tehnici este că identificarea pe baza culorii este robustă la acoperirea parțială [Slaughter, 2008]. În plus, procesarea digitală a culorilor este mai rapidă decât procesarea formelor. Deseori, valorile cromatice sunt utilizate la segmentarea plantelor de fundalul solului. Multe studii de recunoaștere a plantelor pe baza formelor utilizează segmentarea culorilor ca primă etapă, pentru a face distincția între plante și sol. În anul 2001, Hemming și Rath au analizat eficiența unui sistem de vedere artificială în aplicații de recunoaștere a plantei de varză, a morcovilor și a buruienilor [Hemming, 2001]. Procesarea imaginilor a fost realizată atât pe baza culorii cât și a formei, dar clasificarea plantelor a fost realizată pe baza formei. Ei au observat că în culturile de morcov adăugarea de caracteristici de culoare nu prezintă avantaje clare, în timp ce în culturile de varză au existat unele beneficii pozitive [Slaughter, 2008]. Una dintre cele mai de succes mașini cu vedere artificială care au folosit culorile pentru diferențierea între plante și buruieni, a fost robotul autonom dezvoltat de Astrand și Baerveldt [Astrand, 2002]. În anul 2002, cei doi cercetători au publicat rezultatele obținute pe terenuri cultivate cu sfeclă de zahăr. Watchareeruetai et al. (2007) au utilizat informații de culoare pentru a detecta buruienile de pe gazon în timpul iernii [Watchareeruetai, 2007]. Burgos et al. (2010) au prezentat o metodă de diferențiere a buruienilor, bazată pe utilizarea unui sistem de procesare rapidă a imaginilor, și a unui sistem de detectare robustă a rândurilor de culturi. Cel de-al doilea sistem compensează erorile primului sistem. Această metoda este eficientă în condiții necontrolate de iluminare, și funcționează în timp real [Burgos, 2010]. Lee (1998) a utilizat informațiile de culoare pentru a detecta buruienile din culturile de roșii [Lee, 1998]. Nieuwenhuizen (2009) a utilizat culorile pentru detectarea buruienilor din culturile de sfeclă de zahăr [Nieuwenhuizen, 2009]. Slaughter et al. (2000) au utilizat metoda MDF (*Minimum Distance Function*). Această metodă s-a dovedit a fi eficientă chiar și în cazul suprapunerii plantelor, permițând robotului să se deplaseze cu o viteză constantă 1m/s. Prin metoda MDF se poate face distincția între plantele cu frunze late și plantele cu frunze înguste cu o precizie medie de 87% în câmpurile cultivate cu bumbac [Slaughter, 2000]. Pérez (2000) a utilizat informațiile de culoare și formă pentru detectarea buruienilor din culturile de cereale [Pérez, 2000]. Tema recunoașterii plantelor pe bază de culoare este abordată și în [Jafari, 2004], [Kadir, 2011a], [Kebapci, 2010].

Vrindts (2002) a utilizat reflexia pentru a detecta buruienile [Vrindts, 2002]. Dacă reflexia plantelor cultivate, a buruienilor și a solului diferă în lungimi de undă vizibile în infraroșu apropiat, există posibilitatea de a face diferențierea pe baza reflexiei în diferite lungimi de undă. O modalitate, prin care se ține cont de variațiile condițiilor din mediu, este utilizarea unui filtru homomorfic care separă iluminarea de reflexie, luând astfel în considerare doar efectele reflexiei, iar efectele iluminării sunt înlăturate [Tellaèche, 2008].

Recent, au fost efectuate numeroase studii de recunoaștere a speciilor de plante utilizând imagistica hiperspectrală.



**Recunoaștere plantelor pe baza caracteristicilor de formă.** Majoritatea cercetărilor de vedere artificială cu privire la identificarea speciilor de plante au fost făcute la nivelul geometriei frunzei; iar unele la nivelul întregii plante. O serie de studii de recunoaștere a speciilor de plante pe baza morfologiei biologice au fost efectuate de către Guyer et al. în anul 1986 [Guyer, 1986]. Aceste studii au investigat o gamă largă de caracteristici de formă a marginilor frunzelor (curbură sau lobare), și ansamblul formei frunzei sau a plantei (arie, lungime, lățime, perimetru, moment, raporturile adimensionale, etc). În general, sunt obținute rate ridicate de recunoaștere în condiții ideale. Aceste studii demonstrează fezabilitatea utilizării vederii artificiale în aplicații de recunoaștere a speciilor de plante atunci când întreaga frunză este vizibilă, iar frunzele sunt intacte. În anul 2005, Søggaard [Søggaard, 2005] a dezvoltat o metodă de clasificare a speciilor de buruieni pe baza modelului *Active Shape*. A fost construită o bază de date pentru clasificarea celor mai importante 19 specii de buruieni existente pe terenurile agricole din Danemarca.

Una dintre principalele provocări în domeniul roboților autonomi pentru combaterea buruienilor este dezvoltarea unor tehnici care să permită o vizualizare optimă a formei frunzelor. Operatorii umani pot schimba cu ușurință poziția corpului pentru a găsi o orientare care le permite să minimizeze acoperirea vizuală, în scopul de a maximiza calitatea vederii [Slaughter, 2008]. În prezent, majoritatea echipamentelor de procesare a imaginilor utilizează o singură cameră, aceasta însemnând că planta este observată dintr-un singur punct de vedere. Acest lucru este dezavantajos din cauză că informațiilor vizuale sunt limitate, iar procesul de recunoaștere a plantelor devine mai sensibil la problemele cauzate de acoperire. Pentru rezolvarea acestei probleme, roboții pentru agricultură pot fi dotați cu sisteme formate din mai multe camere video. O altă modalitate ar fi fixarea camerei video pe un braț robotizat cu multiple grade de libertate. Lee et al. (1999) au evaluat o metodă de recunoaștere a formelor frunzelor, care s-a dovedit a fi ceva mai robustă la acoperire în aplicații de diferențiere buruienă/plantă [Lee, 1999]. În timp ce un număr mare de metode pentru recunoașterea formei plantelor au demonstrat un potențial ridicat în condiții ideale, există un deficit de metode robuste pentru rezolvarea problemelor legate de acoperire și denaturări ale frunzelor pe terenurile agricole. Rezolvarea acestor probleme rămâne o provocare majoră pentru comercializarea roboților pentru agricultură. Beghin et al. (2010) au analizat semnătura conturului frunzei, pentru a face deosebirea între frunze lobate și frunze nelobate [Beghin, 2010]. Hemming și Rath (2001) au calculat opt caracteristici morfologice diferite și trei caracteristici de culoare ale fiecărui obiect, pentru a construi spațiul caracteristic comun [Hemming, 2001]. Oberndorfer (2006) a utilizat, pentru clasificarea plantelor, câteva atribute geometrice: grosimea, compactitatea, și alungirea [Oberndorfer, 2006]. În multe studii este utilizată transformata Fourier pentru analiza formei frunzei. Tema recunoașterii plantelor pe bază de transformată Fourier este abordată în [Hâruta, 2011], [Kadir, 2011b], [Nejati, 2008], [Neto, 2006], [Ulissi, 2011]. Kebapci et al. (2010) au raportat că recunoașterea plantelor a fost realizată cu succes prin utilizarea transformatei SIFT (Scale Invariant Feature Transform) [Kebapci, 2010]. Persson și Astrand (2008), precum și Søggaard (2005) au utilizat tehnica de „Active Shape Models” pentru clasificarea plantelor și a buruienilor [Persson, 2008] [Søggaard, 2005]. Tema recunoașterii formelor plantelor este abordată și în [Kadir, 2011a], [Lee, 1998], [Shrestha, 2005], [Guyer, 1986] [Pérez, 2000]. Ghazali et al. (2008) au identificat buruienile prin tehnicile de recunoaștere a formei: GLCM (Gray Level Co-Occurrence Matrix), FFT (Fast Fourier Transform) și SIFT (Scale Invariant Feature Transform) [Ghazali, 2008].

Foarte multe studii de recunoaștere a speciilor de plante sunt axate pe rețele neuronale artificiale, logică fuzzy, clasificatoare Bayes, algoritmi genetici și alte tipuri de tehnici avansate. Tema recunoașterii plantelor prin rețele neuronale artificiale a fost abordată în multe lucrări științifice, printre care: [Chaki, 2011], [Jeon, 2011], [Kadir, 2011a], [Nieuwenhuizen, 2009], [Singh, 2010], [Tang, 2003], [Vioix, 2002], [Wang, 2006], [Wu, 2007]. Hemming și Rath (2001) au utilizat logica fuzzy pentru clasificarea plantelor. Depinzând de stadiul de creștere, densitatea buruienilor și metoda de calcul, au fost clasificate corect între 51% și 95% din plante [Hemming, 2001]. Pentru a clasifica plantele și buruienile, Noguchi et al. (1998) au utilizat o combinație între logica fuzzy și algoritmi genetici [Noguchi, 1998]. Nieuwenhuizen (2009) a dezvoltat o metodă adaptivă de clasificare bayesiană pentru clasificarea buruienilor din culturile de sfeclă de zahăr [Nieuwenhuizen, 2009]. Buruienile sunt distribuite neregulat pe terenurile agricole. Mai mult decât atât, buruienile și culturile au semnături spectrale și texturi similare. Acesta însemnând că diferențiere lor este dificilă. Pentru a rezolva această problemă, Tellaèche et al. (2007) au proiectat o strategie care presupune doi pași: segmentarea imaginii și luarea de decizii. Luarea deciziilor este realizată prin intermediul tehnicii Bayes [Tellaèche, 2007]. Burgos et al. (2009) au utilizat algoritmi genetici [Burgos, 2009]. Pentru a recunoaște frunzele, Gu et al. (2005) au utilizat o combinație între transformata *Wavelet* și interpolarea *Gauss* [Gu, 2005].

În unele studii sunt descrise metode de adaptare la condițiile de mediu. Tian et al. (1998) a dezvoltat un algoritm de segmentare adaptivă a imaginilor plantelor. Experimentele au arătat că algoritmul se poate adapta la condițiile de mediu, cum ar fi schimbările de temperatură ale sursei de lumină solară în timpul zilei [Tian, 1998]. Burgos et al. (2008) au utilizat raționamentul în funcție de caz (*Case-Based Reasoning*), care stabilește metoda de procesare optimă pentru fiecare imagine, în mod automat și în timp real [Burgos, 2008]. Du et al. (2006) au dezvoltat o metodă care permite recunoașterea frunzelor parțiale, distorsionate sau suprapuse. [Du, 2006].

Kebapci et al. (2010) și Kadir et al. (2011) au utilizat texturile frunzelor ca al treilea set de caracteristici; primele seturi de caracteristici fiind culorile și formele [Kebapci, 2010] [Kadir, 2011a].

Hemming și Rath (2002), precum și Jelacic (2006), au utilizat metode de *clustering* (grupare) pentru recunoașterea plantelor [Hemming, 2002] [Jelacic, 2006].

**Recunoaștere plantelor pe bază de cartografiere.** Griepentrog et al. (2006) au investigat precizia cu care unele rotații poate combate buruiana pe bază de GPS [Griepentrog, 2006]. Eficiența cu care plantele pot fi recunoscute de către roboți, poate fi îmbunătățită dacă în prealabil sunt cunoscute pozițiile plantelor. Astfel de informații pot fi obținute prin cartografierea pozițiilor plantelor în timpul semănării. Determinarea poziției, în timpul cultivării plantei, este realizată cu RTK-GPS (Real Time Kinematic Global Positioning System). Prin acest sistem locația plantei este determinată cu o precizie de câțiva centimetri [Vinstrup, 2005]. Sistemul de vedere artificială identifică conturul și punctele centrale ale plantelor de cultură printr-un procedeu de *Active Shape Model* și instruieste echipamentul de plivit să înlăture toată vegetația verde, care nu este identificată ca fiind cultură [Vinstrup, 2005]. Ruckelshausen et al. (2009) utilizează RTK-DGPS pentru determinarea fenotipului plantelor [Ruckelshausen, 2009].

Unele tehnici de cartografiere utilizează vederea stereo a imaginilor prelevate de la înălțime (din elicopter). Acest procedeu a fost utilizat de către Rovira et al. (2005) în lucrările [Rovira, 2005] și [Rovira, 2006]. Aceeași echipă de

cercetare a cartografiat terenuri agricole utilizând camere stereo amplasate pe un vehicul care se deplasa pe câmp. Acest procedeu este publicat în lucrările [Rovira, 2007] și [Rovira, 2011]. Lin et al. (2008) au dezvoltat un sistem GIS de realitate virtuală utilizând imagini stereo prelevate de pe terenurile agricole. În acest sistem, tridimensional, scena panoramică este reconstruită din perechi de imagini multiple prelevate prin rotirea camerei [Lin, 2008].

#### 2.1.4 Vedere stereo, modelare 3D și fuziune senzorială

**Vedere stereo.** Vederea stereo binoculară este utilizată în aplicații de măsurări optice tridimensionale. Andersen et al. (2005) au investigat posibilitatea utilizării vederii stereo binoculare în analize tridimensionale ale plantelor pentru estimarea atributelor geometrice, cum ar fi înălțimea și aria frunzei [Andersen, 2005a]. Henten et al. (2002) au publicat o lucrare despre un robot autonom utilizat în sere la aplicații de recoltare a castraveților. Robotul se deplasează pe șine. Camerele video ale robotului, prin alunecare, prelevează imagini din perspective diferite, ceea ce permite reconstrucția 3D a scenei prin tehnica de triangulație [Henten, 2002]. Tehnica de prelevare a imaginilor utilizată de către Henten et al. (2002) este echivalentă cu tehnica stereoscopică. Huang și Lee (2009) au utilizat vederea stereo pentru localizarea punctului de prindere a plantelor [Huang, 2009]. Plantele trebuie manipulate prin prindere de tulpină sau rădăcină. Frunzele sunt prea fragile, astfel încât gripper-ul robotului le-ar putea deteriora. Huang și Lee (2009) au dezvoltat un algoritm de procesare a imaginilor pentru localizarea punctului optim de prindere a plantei. Sistemul stereo binocular a fost aplicat pentru calculul coordonatelor 3D ale punctului de prindere. Kise și Zhang (2007) au creat imagini panoramice utilizând sisteme stereo multispectrale. Imagini stereo multispectrale au fost utilizate și de către Piron et al. (2009) pentru detectarea buruienilor [Piron, 2009]. Sistemul a fost utilizat pentru urmărirea stadiului de creștere a plantelor. Meyer (2011) a obținut o precizie de câțiva centimetri în localizarea rândurilor de culturi și a buruienilor. El a folosit o camera stereo ECO-DAN; una dintre primele camere stereo pentru aplicații agricole [Meyer, 2011]. Piron et al. (2008) au determinat înălțimea plantelor și au detectat buruienile utilizând imagini stereoscopice [Piron, 2008] [Piron, 2010]. Rath et al. (2002) au utilizat vederea stereo pentru determinarea distanței la care se află planta [Rath, 2002]. Shrestha et al. (2002) au determinat înălțimea plantelor timpurii prin vedere stereo [Shrestha, 2002]. Ericson și Astrand (2008) au utilizat vederea stereo pentru a dezvolta un sistem de odometrie vizuală, care nu este sensibil la terenuri accidentate [Ericson, 2008].

**Modelare 3D.** Fotogrametria este o tehnică aplicată pe scară largă la modelarea 3D. De obicei, acest procedeu este utilizat în arhitectură la modelarea clădirilor. Plantele sunt foarte greu de modelat 3D, din cauza complexității geometrice și a varietăților de aparență. Fotogrametria este o soluție promițătoare pentru modelarea 3D a plantelor individuale. Mai mult decât atât, achiziționarea unei camere video obișnuite implică costuri mult mai reduse decât achiziționarea unei camere 3D sau a unui sistem LIDAR. Imaginile captate prin tehnici fotogrametrice sunt abundente în informații utile pentru reconstrucția modelului complex al plantei [Wang, 2009]. O astfel de tehnică a fost aplicată de către Wang et al. (2009) pentru reconstrucția modelului plantei de porumb (Fig. 2. 3 și Fig. 2. 4). Tema reconstrucției 3D a plantei de porumb prin fotogrametrie a fost abordată și de către Zhang et al. (2008) [Zhang, 2008b]. Tehnici 3D pentru determinarea structurii plantelor individuale, au fost utilizate și de către Biskup (2009) [Biskup,

2009]. Lin et al. (2008) au dezvoltat un program GIS de realitate virtuală, care utilizează informațiile prelevate de la un sistem stereoscopic [Lin, 2008]. Klose et al. (2009) a introdus un concept de modelare 3D pe baza informațiilor prelevate de la un sistem de camere 3D-ToF (Time of Flight) amplasate în jurul plantei [Klose, 2009]. Alenya et al. (2011) și Song et al. (2011) au corelat imaginile prelevate de la sisteme formate din camere 3D-ToF și camere 2D-color. Prin această metodă au creat modele 3D parțiale care au culorile suprafețelor identice cu culorile plantelor [Alenya, 2011] [Song, 2011]. Weiss et al. (2010) au clasificat și cartografiat speciile de plante utilizând un sistem senzorial 3D de tip LIDAR [Weiss, 2010b] [Weiss, 2011].



Fig. 2. 3. Cinci imagini prelevate din diferite unghiuri [Wang, 2009]



Fig. 2. 4. Modelarea punctelor omoloage prin constrângeri epipolare [Wang, 2009]

**Fuziune senzorială.** Ruckelshausen et al. (2010) au utilizat fuziunea senzorială pentru a detecta caracteristicile morfologice și spectrale ale plantelor [Ruckelshausen, 2010b]. În acest scop au fost utilizate camere 3D-ToF, camere spectrale, *light-curtain* (Fig. 2. 5) și camere color [Ruckelshausen, 2010b], [Ruckelshausen, 2009], [Klose, 2010]. Busemeyer et al. (2010) au publicat o lucrare în care prezintă două tipuri de platforme robotizate utilizate la determinarea fenotipului plantelor în aer liber. Una dintre platforme este concepută pentru a fi utilizată în culturi agricole cu densitate mică de plante (culturi de porumb), iar a doua platformă este utilizată în culturi cu densitate mare de plante (culturi de cereale). Platformele sunt dotate cu diferite sisteme de senzori pentru determinarea caracteristicilor morfologice și spectrale [Busemeyer, 2010].

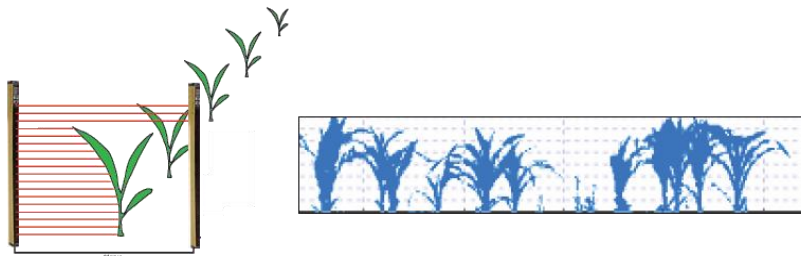


Fig. 2. 5. „Light-curtain” [Ruckelshausen, 2007]

### 2.1.5 Combaterea buruienilor utilizând sisteme automate

Combaterea automată a buruienilor poate fi realizată prin diverse metode:

- Combatere prin erbicidare selectivă
- Combatere prin sisteme mecanice
- Combatere prin flacără
- Combatere prin jet de apă fierbinte
- Combatere prin laser
- Combatere prin microunde
- Combatere prin infraroșu
- Combatere prin descărcări electrice
- Combatere prin presiune de aer
- Combatere prin jet de apă
- Combatere prin înghețare

*Combatere prin erbicidare selectivă.* Detectarea buruienilor este un domeniu studiat intensiv, deoarece utilizarea de erbicide are un impact negativ asupra economiei agricole și a mediului [Vioix, 2002]. Scopul final este de a reduce cantitatea de erbicid pulverizată. Adică, pulverizarea să aibă loc doar în zona în care este prezentă buruiana [Tellaeché, 2010] [Tellaeché, 2008].

Lee et al. (1999) au dezvoltat un sistem robotizat de combatere a buruienilor în timp real, care aplică selectiv erbicide pe buruienile din culturile de roșii [Lee, 1999]. Cercetări privind erbicidarea cu precizie, au fost efectuate și de către Slaughter et al. (2000), Nieuwenhuizen et al. (2009) și de către Jeon și Tian (2009) [Slaughter, 2000], [Nieuwenhuizen, 2009], [Jeon, 2009].

*Combatere prin sisteme mecanice.* Încă din anul 1960, au fost comercializate câteva sisteme de rărire automată a plantelor [Slaughter, 2008]. Buruienile pot fi tăiate sau smulse din pământ cu ajutorul unui actuator mecanic [Bakker, 2009b]. În anul 2002, Astrand și Baerveldt [Astrand, 2002] au prezentat o prășitoare rotativă pentru înlăturarea buruienilor. Prășitoarea a fost atașată la un robot utilizat la combaterea buruienilor în culturile de sfeclă de zahăr. Home et al. (2002), de la Universitatea Cranfield din Anglia, au efectuat un studiu privind poziționarea cu precizie a uneltelor de prășit, utilizând vederea artificială (Fig. 2. 6).

Tillett et al. (2007) au efectuat un experiment în care au utilizat o prășitoare ghidată, echipată cu palete pentru combaterea mecanică a buruienilor. Paletele, acționate hidraulic, sunt rotite în jurul axei verticale (Fig. 2. 7). Fiecare paletă are câte o secțiune care permite ca unealta să ocolească plantele cultivate, pentru a nu le distruge [Tillett, 2007]. Cercetări similare au fost efectuate de către Dedousis (2007) și Gobor (2007) [Dedousis, 2007] [Gobor, 2007].

Ruckelshausen et al. (2006) au conceput o prășitoare cicloidală pentru combaterea mecanică a buruienilor din interiorul rândurilor de culturi [Ruckelshausen, 2006]. Prășitoarea cicloidală din Fig. 2. 8 a fost construită la Universitatea din Osnabrück, Germania. Griepentrog et al. (2006) au atașat prășitoarea cicloidală la un tractor [Griepentrog, 2006].





Fig. 2. 6. Prășitoare pentru cereale, ghidată prin vedere artificială [Home, 2002]



Fig. 2. 7. Disc rotativ prevăzut cu o secțiune pentru protejarea platelor cultivate [Tillett, 2007].



Fig. 2. 8. Prășitoare cicloidală „Querhacke” concepută la Universitatea din Osnabrück, Germania [Griepentrog, 2007]

*Combatere prin flacără.* În unele cazuri au fost utilizate sisteme termice pentru combaterea selectivă a buruienilor. Tehnica de plivire cu flacără poate fi utilizată la combaterea selectivă a buruienilor doar în cazul în care există o barieră termică (cortină de aer sau jet de apă) pentru a proteja plantele dorite [Slaughter, 2008]. Este preferabilă utilizarea tehnicii de combatere termică cu jet de apă fierbinte, deoarece această metodă nu prezintă riscuri de producere a incendiilor.

*Combatere prin jet de apă fierbinte.* Combaterea buruienilor este realizată prin jet de apă fierbinte, cu temperatură de minim 150°C [Bakker, 2009b], [Slaughter, 2008].

*Combatere prin laser.* Laserul poate fi utilizat ca dispozitiv de tăiere a tulpinii buruienilor, provocând oprirea sau întârzierea creșterii. Prin laser nu pot fi tăiate porțiunile de plantă care se află sub suprafața solului. Prin urmare laserul are efect scăzut asupra anumitor specii de buruieni. Pentru a ajunge la o performanță acceptabilă, este nevoie de laser de putere mare. Aceasta implică costuri ridicate [Bakker, 2009b].

*Combatere prin microunde.* Buruienile sunt distruse prin expunere la radiații cu microunde [Bakker, 2009b].

*Combatere prin infraroșu.* O altă metodă termică de combatere a buruienilor este expunerea la radiații cu infraroșu [Bakker, 2009b].

*Combatere prin descărcări electrice.* În trecut au fost efectuate studii de combatere a buruienilor prin sisteme electrice de înaltă tensiune (15-60 kV) [Slaughter, 2008]. Prin această metodă, fiecare plantă trebuie atinsă cu un dispozitiv electric; acesta fiind un mare dezavantaj.

*Combatere prin presiune de aer.* Combaterea buruienilor este realizată prin introducerea în sol a unor duze de aer sub presiune orientate spre direcția buruieni [Bakker, 2009b].

*Combatere prin jet de apă.* Tulpina plantei poate fi tăiată prin jet de apă sub presiune mare. Însă, această metodă este inefficientă atunci când tulpina plantei depășește grosimea de 3mm [Bakker, 2009b].

*Combatere prin înghețare.* Combaterea buruienilor este realizată prin înghețare [Bakker, 2009b].

### **2.1.6 Prototipuri de roboți pentru protecția culturilor de plante**

În acest subcapitol sunt prezentate exemple de platforme autonome pentru protecția culturilor de plante. Probabil, aceste platforme robotizate vor fi disponibile pe piață în viitorul apropiat [Griepentrog, 2010].

Robotul Lukas (Fig. 2. 9) a fost conceput la Universitatea Halmstad din Suedia. Informații referitoare la acest robot au fost publicate de către Bjorn Astrand și Albert-Jan Baerveldt în lucrările [Astrand, 2002] și [Astrand, 2004]. Lukas este un robot mobil pentru agricultură, utilizat la combaterea mecanică a buruienilor de pe terenurile agricole. Robotul este dotat cu două sisteme vizuale. Primul sistem funcționează pe bază de nivele de gri, și este utilizat la recunoașterea rândurilor de culturi și ghidarea robotului de-a lungul rândului. Al doilea sistem, bazat pe vederea color, este capabil să identifice plantele de cultură printre buruieni. Cel de-al doilea sistem vizual controlează o unealtă de înlăturare a buruienilor din rândul de culturi.



Fig. 2. 9. Robot autonom pentru combaterea buruienilor conceput la Universitatea din Halmstad, Suedia [\*\*\* Img, Lukas]

Robotul API (Fig. 2. 10) a fost conceput la Universitatea Aalborg din Danemarca [Xie, 2005]. Primele studii au început în anul 2000, la inițiativa Institutului Danez pentru Agricultură. Robotul API este utilizat în aplicații de cartografiere a plantelor și erbicidare selectivă [Xie, 2005].



Fig. 2. 10. Robot autonom pentru combaterea buruienilor conceput la Institutul Danez de Științe Agricole, Danemarca [Slaughter, 2008]

Un alt prototip de robot pentru agricultură a fost conceput la Universitatea Wageningen din Olanda (Fig. 2. 11). Informații referitoare la acest robot au fost publicate de către Bakker et al. în lucrările [Bakker, 2009a] și [Bakker, 2011]. Robotul permite investigarea unui spectru larg de opțiuni de cercetare privind detectarea buruienilor și acționarea actuatorului utilizat la înlăturarea acestora [Bakker, 2009a]. Robotul are implementat un sistem de navigare autonomă pe bază de RTK-DGPS (Real Time Kinematic Differential Global Positioning System) care, în combinație cu un sistem vizual, permite cartografierea rândurilor de culturi [Bakker, 2011].





Fig. 2. 11. Robot autonom pentru combaterea buruienilor conceput la Universitatea din Wageningen, Olanda [Bakker, 2011]

Robotul autonom HortiBot (Fig. 2. 12) a fost dezvoltat la Universitatea Aarhus din Danemarca, pentru a fi utilizat în aplicații agricole [Jørgensen, 2007]. Acesta este echipat cu o cameră care permite navigarea prin urmărirea rândurilor de culturi. În zonele fără rânduri, robotul este poziționat printr-un sistem RTK-GPS. Sistemul de pulverizare cu erbicide este dotat cu un set de camere video care prelevează imagini ale suprafeței solului [Griepentrog, 2010]. Imaginile sunt analizate pentru a detecta buruienile. Atunci când una sau mai multe buruieni sunt detectate în imagine, informațiile despre locația acestora sunt salvate [Griepentrog, 2010].



Fig. 2. 12. Robot autonom pentru protejarea culturilor agricole conceput la Universitatea din Aarhus, Danemarca [\*\*\* Img, Hortibot]

Robotul Weedy (Fig. 2. 13) a fost dezvoltat la Universitatea Osnabrück din Germania, pentru a fi utilizat în aplicații de combatere a buruienilor [Klose, 2008]. Robotul este un sistem mecatronic pe bază de fuziuni senzoriale, pentru combaterea selectivă a buruienilor [Klose, 2008].



Fig. 2. 13. Robot autonom pentru combaterea buruienilor „Weedy” conceput la Universitatea din Osnabrück, Germania [\*\*\* Img, Weedy]

Robotul BoniRob (Fig. 2. 14) a fost dezvoltat la Universitatea Osnabrück din Germania, în colaborare cu firmele Amazone și Bosch. În anul 2009, Ruckelshausen et al. (2009) au publicat o lucrare despre robotul autonom BoniRob [Ruckelshausen, 2009].



Fig. 2. 14. Robot autonom pentru agricultură conceput la Universitatea din Osnabrück în colaborare cu firmele Amazone și Bosch [\*\*\* Img, BoniRob]

Proiectul interdisciplinar BoniRob este focalizat pe dezvoltarea unei platforme autonome pentru analiza fenotipului plantelor de porumb în diferite etape de creștere [Griepentrog, 2010]. În scopul de a redetecta plantele individuale în diferite etape de creștere, este utilizat un sistem RTK-DGPS de rezoluție înaltă [Fender, 2006]. Caracterizarea plantelor este realizată prin fuziune multisenzorială având ca scop detectarea plantelor și a buruienilor din rândurile de porumb cu ajutorul mai multor tipuri de tehnologii senzoriale: imagistică CCD, imagistică spectrală VIS/NIR, camere 3D-ToF (Time-of-Flight), senzori fotoelectrici, și senzori de distanță [Griepentrog, 2010]. Robotul are incluse sisteme de securitate. Fiecare

## 2.1 - Stadiul actual de utilizare a roboților în aplicații de combatere a buruienilor 19

plantă poate fi marcată individual pentru a fi redetectată în timpul măsurătorilor efectuate în diferite etape de creștere. Robotul permite diverse opțiuni de mișcare, cum ar fi: rotirea, schimbarea înălțimii, și deplasarea de-a lungul a două sau patru rânduri simultan [Griepentrog, 2010]. Navigarea se bazează pe metode de robotică probabilistică, ținând cont de incertitudinile care apar în timpul aplicațiilor. Metoda SLAM utilizează date măsurate pentru a asigura navigarea corectă. Pozițiile plantelor de porumb, marginile câmpului sau curbura rândurilor pot fi utilizate pentru orientare [Griepentrog, 2010].

### 2.1.7 Concursul internațional „Field Robot Event”

Concursul internațional „Field Robot Event” a fost fondat de către Universitatea din Wageningen (Olanda) în 2003 pentru a orienta generațiile viitoare de studenți spre domeniul roboților pentru agricultură. Evenimentul are rolul de a facilita schimbul de informații tehnice între studenți, experți și public, precum și de a face schimb de cunoștințe interdisciplinare și experiențe cu privire la roboții pentru agricultură. Roboții trebuie să îndeplinească două funcții majore:

- Navigarea complet autonomă în culturile de porumb
- Detectarea buruienilor



Fig. 2. 15. Robotul „Helios” conceput la Universitatea din Braunschweig, Germania [\*\*\* Img, Helios]



Fig. 2. 16. Robotul „Amaizeing” conceput la Universitatea din Osnabrück, Germania [\*\*\* Img, Amaizeing]



Fig. 2. 17. Robotul „4M” conceput la Universitatea din Helsinki, Finlanda [Backman, 2008]



Fig. 2. 18. Robotul „Eyesonic” conceput la Universitatea din Wageningen, Olanda [\*\*\* Img, Eyesonic]



Fig. 2. 19. Robotul „Eduro Maxi” conceput la Universitatea din Praga, Cehia [Kroulik, 2010]



Fig. 2. 20. Robotul „Optimaize Prime” conceput la Universitatea din Osnabrück, Germania [\*\*\* Img, Optimaize Prime]



### 2.1.8 Sisteme de securitate în timpul funcționării robotului

Securitatea are o importanță majoră din cauza faptului că roboții pentru agricultură se deplasează într-un mediu deschis, în care pot să apară obstacole necunoscute (de exemplu: animale sau ființe umane) în direcția de mișcare a robotului [Nielsen, 2005]. În cazul roboților pentru agricultură, implementarea sistemelor de siguranță este mai greu de realizat decât în cazul roboților industriali.

Un grup de cercetători de la Universitatea Okayama din Japonia au propus implementarea unui sistem de siguranță format din trei subsisteme:

- *Subsistemul 1:* Crearea unui sistem de sesizare a prezenței umane în apropiere, și a mișcărilor acestuia incluzând poziția, direcția și viteza [Monta, 2005].
- *Subsistemul 2:* Sistem de calcul al gradului de pericol pe baza informațiilor obținute de la senzorul de detectare a prezenței umane. De exemplu; gradul de pericol este mare atunci când distanța dintre robot și lucrător este mică; iar viteza relativă este mare [Monta, 2005].
- *Subsistemul 3:* În funcție de gradul de pericol, robotul va primi o comandă de accelerare, decelerare, oprire sau evitare. Sistemul de sesizare a prezenței umane constă din patru perechi de senzori cu ultrasunete și infraroșu, care sunt dispuși la 90°. Sistemul de scanare acționează pe toate direcțiile în două spații dimensionale paralele cu solul [Monta, 2005].

### 2.1.9 Concluzii

Utilizarea roboților în agricultură ar putea aduce avantaje de natură economică, productivă, și ecologică. Din cauza complexității funcțiilor, până în prezent, nici un robot pentru protecția culturilor agricole nu este disponibil în comerț [Griepentrog, 2010].

Majoritatea problemelor sunt legate de recunoașterea plantelor. Acest lucru este cauzat de faptul că în prezent nu există nici un sistem adecvat cu putere mare de calcul care să facă distincția între plantă și buruiană [Oberndorfer, 2006]. În recunoașterea speciilor de plante apare problema găsirii setului de caracteristici specifice fiecărei plante [Andersen, 2002]. Majoritatea problemelor au următoarele cauze:

- Condiții meteo nefavorabile (umbre, vânt, ceață, precipitații, înnorări).
- Denaturări ale plantelor, cauzate de mediu sau insecte.
- Schimbarea culorilor și a morfologiei plantelor în funcție de stadiul de creștere și de anotimp.
- Aglomerări ale plantelor, ceea ce determină suprapunerea totală sau parțială în imagine.
- Dificultăți de recunoaștere a plantelor prin procesarea imaginilor prelevate din perspectivă.
- Prezența prafului în atmosferă.

## 2.2 Calibrarea camerelor video

### 2.2.1 Generalități

Calibrarea camerelor este un pas necesar în vederea tridimensională computerizată, în scopul de a extrage informații metrice din imagini 2D. Au fost făcute numeroase cercetări în fotogrametrie, și mai recent, în computer vision. Putem clasifica aceste tehnici aproximativ în două categorii [Zhang, 2008]:

- calibrare fotogrametrică (OFF-LINE)
- autocalibrare (ON-LINE)

Prima categorie include metodele clasice de calibrare, bazate pe imagini ale unui obiect de dimensiuni cunoscute [Toma, 2005]. Fotogrametria (calibrarea OFF-LINE) este o tehnică utilizată pentru înregistrarea obiectelor bi-dimensionale sau tri-dimensionale prin suprapunerea parțială a fotografiilor efectuate în diferite locații. Acest procedeu este numit vedere stereo. Marele avantaj al acestei metode este obținerea unei precizii ridicate a valorilor parametrilor. Dezavantajul este că atunci când este schimbat unul dintre parametri, cum ar fi "zoom-ul", calibrarea camerelor trebuie să fie reefectuată. A doua categorie, autocalibrarea (calibrare ON-LINE) conține metode care utilizează doar punctele de corelație dintre imagini, fără a fi necesară cunoașterea scenei. Avantajul constă în faptul că aceste metode permit calibrarea în mod automat în timpul funcționării obișnuite a camerei. Dezavantajul este precizia scăzută și numărul mare de erori ce apar în timpul funcționării. Modelul camerei reprezintă relația matematică dintre coordonatele 3D ale punctului și proiecțiile acestuia în planul 2D al imaginii. Din cauza proprietăților lentilelor, modelele camerelor se divid în două categorii [Toma, 2005]:

- modele ideale (modele fără distorsiuni cauzate de lentile)
- modele reale (modele cu distorsiuni cauzate de lentile)

Calibrarea camerelor implică parcurgerea a două grupe principale de calibrarea:

- calibrarea intrinsecă
- calibrarea extrinsecă

### 2.2.2 Calibrarea intrinsecă

Termenul de „calibrare intrinsecă” se referă la determinarea valorilor parametrilor intrinseci: distanța focală, punctul principal, oblicitate (*skew*), și distorsiunile lentilelor [Bouguet].

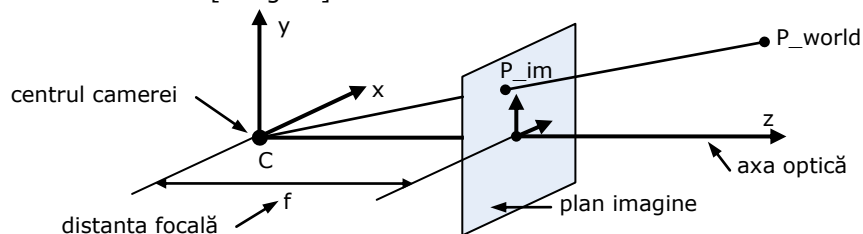


Fig. 2. 21. Geometria camerei pinhole

În acest caz este utilizat modelul *pinhole* (Fig. 2. 21). Acest model este utilizat în cazul camerelor *pinhole*. Camerele de tip *pinhole* sunt camere fără lentile. Avantajul utilizării acestor camere constă în faptul că, neavând lentile, nu apar distorsiuni ale imaginilor.

### 2.2.3 Calibrarea extrinsecă

Termenul de „calibrare extrinsecă” se referă la determinarea valorilor pozițiilor relative dintre două sau mai multe camere video, fiind necesară cunoașterea matricei de orientare și a vectorului de poziție [Sutton, 2009] a sistemului de referință atașat camerei 2, raportat la sistemul de referință atașat camerei 1. Prin calibrarea extrinsecă a camerelor sunt determinate valorile parametrilor  $\alpha, \beta, \gamma, t_x, t_y, t_z$ ; unde  $\alpha, \beta, \gamma$  reprezintă unghiurile de rotație în jurul axelor  $x, y$  respectiv  $z$ , iar  $t_x, t_y, t_z$  reprezintă translațiile de-a lungul axelor  $x, y$  respectiv  $z$ . Pe baza acestor parametri este calculată matricea de rotație (matricea de orientare), respectiv matricea de translație (matricea de poziție) a sistemului de referință 2 în raport cu sistemul de referință 1. Parametrii determinați prin procedeul de calibrare extrinsecă sunt necesari pentru a face corespondența între mai multe sisteme de referință. În algoritmiile de calcul prezentate mai jos, matricea de rotație este notată cu  $M\_Rotatie$  iar matricea de translație este notată cu  $M\_Translatie$ .

$$M\_Rotatie = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} * \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix} * \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.1)$$

$$M\_Translatie = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad (2.2)$$

Având un punct  $P$  din spațiul 3D, poziția acestui punct poate fi raportată relativ la mai multe sisteme de referință. Considerând că avem două sisteme de referință  $Ref1$  respectiv  $Ref2$ , corespondența coordonatelor punctului  $P$  raportat la sistemele de referință  $Ref1$ , respectiv  $Ref2$ , este dată prin relația:

$$\begin{bmatrix} x\_P\_Ref2 \\ y\_P\_Ref2 \\ z\_P\_Ref2 \end{bmatrix} = M\_Rotatie^{-1} * \begin{bmatrix} x\_P\_Ref1 \\ y\_P\_Ref1 \\ z\_P\_Ref1 \end{bmatrix} - M\_Translatie \quad (2.3)$$

Valorile parametrilor  $t_x, t_y, t_z$  sunt de fapt valorile coordonatei originii sistemului de referință  $Ref2$  raportat la sistemul de referință  $Ref1$ . Parametrii  $\alpha, \beta, \gamma$  reprezintă rotațiile succesive pe care sistemul de referință  $Ref1$  trebuie să le parcurgă pentru a rezulta orientarea sistemului de referință  $Ref2$  ().

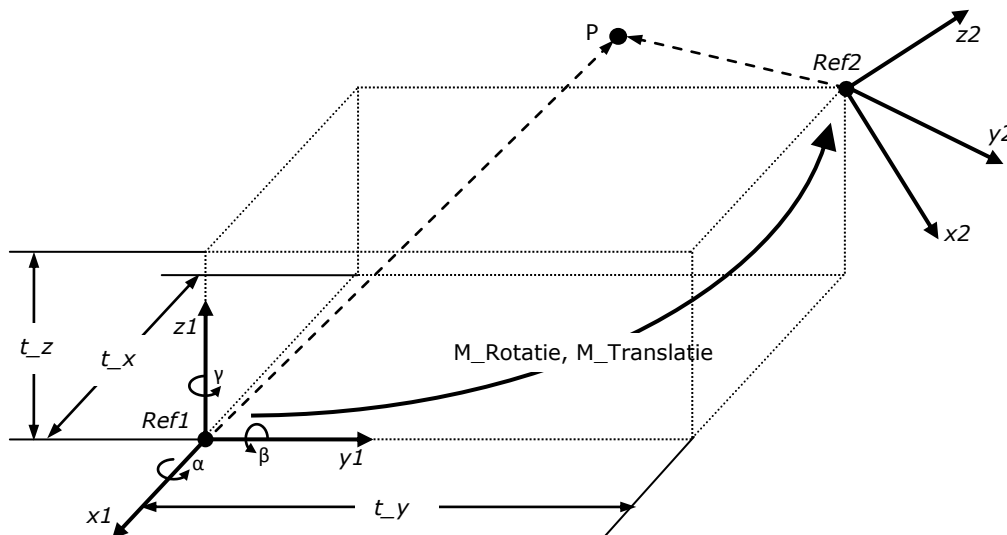


Fig. 2. 22. Corelația între sistemul de referință *Ref1* și sistemul de referință *Ref2*

#### 2.2.4 Metode de calibrare OFF-LINE (sinteză bibliografică)

Metodele de calibrare OFF-LINE fac parte din grupa metodelor clasice de calibrare a camerelor. Prin aceste metode pot fi obținute rezultate foarte precise. În cazul metodelor OFF-LINE este necesar să fie cunoscute unele informații despre scenă. În literatura de specialitate sunt întâlnite diverse metode de calibrare OFF-LINE; în fiecare dintre acestea fiind utilizat cel puțin un obiect de dimensiuni cunoscute. Obiectele pot fi de tip 3D (ex: cuburi sau plane ortogonale), 2D (ex: model planar tip tablă de șah), sau 1D (ex: puncte aliniate pe o dreaptă).

În majoritatea cazurilor, calibrarea camerelor se realizează prin utilizarea unor puncte de control aflate pe o suprafață plană de tip „tablă de șah”. Cel puțin două imagini ale modelului sunt necesare pentru realizarea calibrării. Cu cât numărul perechilor de imagini prelevate este mai ridicat, cu atât rezultatele calibrării sunt mai exacte. Modelul tip „tablă de șah” permite localizarea colțurilor pătratelor, și determinarea distorsiunilor lentilelor. Tsai (1987) a fost publicată o metodă de calibrare foarte precisă, bazată pe tehnica *two-stage* [Tsai, 1987]. Tehnica *two-stage* vizează calculul eficient al poziției externe a camerei și a orientării relative în raport cu sistemul de referință al piesei, permițând calculul distanței focale, a distorsiunilor radiale a lentilelor, și a parametrilor imaginii. Tehnica *two-stage* are avantaje în ceea ce privește precizia, viteza, și adaptabilitatea. Weng et al. (1992), au utilizat camere stereo pentru a estima parametrii interni și externi ai fiecărei camere [Weng, 1992]. Folosind acești parametri, poziția 3D a unui punct din scenă poate fi determinată prin metoda triangulației. Zhang (1999) a prezentat o metodă de calibrare care poate fi utilizată fără a avea cunoștințe speciale de geometrie 3D sau computer vision [Zhang, 1999], [Zhang, 2008]. Prin această tehnică, sunt analizate imaginile unui model planar văzut din diferite perspective. Shah și Aggarwal (1996), precum și Ramalingam et al. (2005) au conceput metode de calibrare a camerelor cu deformații mari cauzate de lentile (ex: camere cu lentile de tip "fisheye" sau cu lentile catadioptrale) [Shah, 1996], [Ramalingam, 2005]. Shih et

al. (1993), au introdus modelul punctelor circulare de control ca o nouă alternativă a modelului tip „tablă de șah” [Shih, 1993]. Douchamps și Chihara (2008) au prezentat o soluție legată de problema măsurării corecte a poziției unui marker într-o imagine [Douchamps, 2008]. Markerele sunt subiectul unui compromis fundamental, deoarece:

- trebuie să fie mici, pentru a limita efectul non-linear de distorsiune
- trebuie să fie mari, pentru a limita efectul de zgomot și de discretizare

În [Douchamps, 2008] este prezentată o nouă tehnică de calibrare a camerelor ce exploatează modelul complet al camerei în timpul localizării punctelor de control, eliminând astfel compromisul dimensiunii markerului. Datta et al. (2009) au descris o metodă de calibrare în care sunt utilizate modele de calibrare planare sub formă de pătrat, cerc sau inel [Datta, 2009]. A fost efectuat un set de experimente din care au rezultat erori cu 50% mai puține decât cele din "OpenCV Camera Calibration Toolbox". În [Sutton, 2009], sistemul de vedere stereo este calibrat folosind o suprafață cu puncte uniform distanțate.

În numeroase cazuri calibrarea este realizată prin utilizarea liniilor și punctelor de fugă. În [Wang, 1991], liniile de fugă sunt generate de laturile unui hexagon observat din perspectivă. În [Beardsley, 1992] sistemul de calibrare constă dintr-o cameră statică ce preia imaginii ale unui plan de calibrare aflat în mișcare de rotație în jurul unei axe fixe. În fiecare imagine sunt determinate liniile și punctele de fugă. Wilczkowiak et al. (2001) au utilizat paralelipipedul ca obiect pentru calibrare [Wilczkowiak, 2001]. Chen et al. (2003) au dezvoltat o metodă de calibrare a unei rețele hibride, formate din camere cu catadioptru și camere cu perspectivă [Chen, 2003]. Relația dintre imaginile multiple capturate de la diferite camere, poate fi determinată prin sistemul de coordonate al camerei cu catadioptru. Fung et al. (2003) au propus o metodă de calibrare în care sunt utilizate proprietățile geometrice ale liniilor de fugă corespunzătoare marcajelor rutiere [Fung, 2003].

Un alt element des utilizat pentru calibrarea camerelor, este cercul. Kim et al. (2002) au utilizat imagini ale unor cercuri concentrice observate din diverse perspective [Kim, 2002]. În [Meng, 2002] este utilizat un cerc prin originea căruia sunt trasate mai multe linii. Cei cinci parametri intrinseci pot fi determinați liniar prin analiza imaginilor cercului. Imaginile cercului au fost prelevate de la camere a căror orientare este necunoscută. Chen et al. (2004) au descris o metodă de calibrare prin care pot fi determinați parametrii extrinseci și distanța focală a camerei, prin utilizarea unei singure imagini a două cercuri coplanare cu diametre de dimensiune arbitrară [Chen, 2004].

Un alt element, des utilizat pentru calibrarea camerelor, este sfera. Teramoto și Xu (2002) au prezentat o tehnică de calibrare a camerelor, în care parametrii extrinseci pot fi determinați dintr-o singură imagine a trei mingi [Teramoto, 2002]. Agrawal și Davis (2003), precum și Zhang et al. (2007), au utilizat proiecțiile conturilor a trei sfere, în scopul determinării parametrilor intrinseci și a localizării sferelor [Agrawal, 2003], [Zhang, 2007].

Heikkila et al. (2000) au utilizat un obiect de calibrare tridimensional, format din plane ortogonale [Heikkila, 2000]. Fiecare dintre planele ortogonale conține un „șablon” format din figuri geometrice amplasate echidistant.

Calibrarea poate fi realizată și prin utilizarea unui obiect de calibrare unidimensional. În [Wu, 2004] este arătat că obiectul de calibrare 1D (unidimensional), analizat în două poziții distincte, este în esență echivalentul unui obiect de calibrare 2D (bidimensional). În [Zhang, 2004] este propusă o metodă de calibrare utilizând un obiect 1D, care constă din mai multe puncte aliniate pe o



dreaptă. Calibrarea poate fi realizată doar dacă unul dintre punctele obiectului 1D rămâne într-o poziție fixă.

În [Do, 1999] sunt studiate tehnicile stereoscopice de măsurare a pozițiilor 3D, utilizând rețele neuronale. Rețelele neuronale sunt utilizate pentru corectarea erorilor metodei proiecției lineare și pentru transformarea imaginilor distorsionate în imagini ideale. În [Memony, 2001] este propusă utilizarea rețelelor neuronale artificiale (RNA) pentru a elimina procesul de calibrare a camerelor. Prin RNA, sistemul este instruit astfel încât atunci când în sistem există o pereche de puncte corelate; să fie calculate în mod automat coordonatele din lumea reală ale punctului corespunzător. Utilizând acest concept, pot fi obținute rezultate foarte bune atunci când este aplicat unui robot mobil autonom dotat cu sistem de vedere stereo. În [Ji, 2001] și [Savii, 2004] sunt descrise metode de calibrare bazate pe algoritmi genetici.

Wong et al. (2003) au introdus o tehnică de calibrare, care se bazează pe simetria imaginii suprafețelor de revoluție [Wong, 2003]. Utilizarea unor suprafețe de revoluție (ex: boluri și vase), simplifică procesul de calibrare, ca urmare a costurilor reduse și a accesibilității la obiectele de calibrare.

Koterba et al. (2005), au utilizat fața umană ca și obiect de calibrare [Koterba, 2005]. Trei imagini au fost prelevate simultan de la trei camere amplasate în stânga, centru, respectiv dreapta. Cao și Shah (2005), au demonstrat că parametrii camerei și orientarea sursei de lumină se pot determina din două imagini de perspectivă asupra unei scene în care se află două linii verticale și umbrele corespunzătoare ale acestora [Cao, 2005]. Marita et al. (2006) au dezvoltat o metodă de calibrare a sistemelor stereo cu domeniu vizual larg [Marita, 2006]. Această metodă a fost concepută pentru a fi aplicată la sistemele vizuale de asistență în conducerea autovehiculelor rutiere.

În [Fiala, 2010] este prezentată o metodă automată de calibrare utilizând markere ARTag. Prin această metodă, în scurt timp pot fi calibrate un număr mare de camere fără intervenție manuală.

În [Joochim, 2010], calibrarea camerelor este realizată cu ajutorul programului "Camera Calibration Toolbox for Matlab", accesibil pe internet la adresa: [http://www.vision.caltech.edu/bouguetj/calib\\_doc/](http://www.vision.caltech.edu/bouguetj/calib_doc/). Utilizând acest program, pot fi determinate cu precizie valorile parametrilor intrinseci ai camerei.

### 2.2.5 Metode de calibrare ON-LINE (sinteză bibliografică)

Faugeras et al. (1992) au calibrat camerele video pe baza punctelor de legătură din secvențele imaginilor unei camere aflate în mișcare [Faugeras, 1992]. Nu este necesar să fie cunoscute deplasările camerei. Xu și Shah (2000) au dezvoltat o metodă de autocalibrare a camerei video, în care sunt utilizate două matrici homografice corespunzătoare unor plane din spațiu [Xu, 2000]. Deutscher et al. (2002) a prezentat o metodă de calibrare complet automată în care este utilizată o singură imagine din Manhattan [Deutscher, 2002]. În [Drummond, 2002] sunt abordate două probleme. Prima este urmărirea a unui obiect tridimensional de dimensiune cunoscută din câmpul vizual al unei camere. Output-ul este o actualizare continuă a poziției obiectului vizualizat. A doua problemă este urmărirea parametrilor interni ai camerei. Rezultatele sistemului de urmărirea sunt utilizate pentru a închide bucla într-un sistem de control al robotului, în scopul ghidării brațului robotizat într-o locație țintă învățată anterior. Svoboda (2003) au prezentat o metodă de calibrare complet automată a mai multor camere simultan. Singurul

element hardware utilizat în această metodă a fost un pointer laser [Svoboda, 2003]. În [Rekleitis, 2005] este prezentată o metodă de determinare a pozițiilor 3D ale camerelor într-un sistem de referință comun, cu ajutorul unui robot mobil. Lv et al. (2006) au prezentat o metodă de autocalibrare în care parametrii intrinseci și extrinseci sunt determinați utilizând segmente de linii verticale de aceeași înălțime [Lv, 2006].

### **2.2.6 Concluzii**

Literatura științifică este foarte bogată în metode de calibrare. Cu toate acestea, autoarea tezei nu a găsit nici o metodă care să poată fi eficient aplicată în agricultură. Din acest motiv, autoarea tezei a dezvoltat trei metode originale de calibrare a camerelor video. Aceste metode au aplicabilitate mai ridicată pe terenurile agricole decât metodele existente în documentația studiată.

## 3 Obiectivele și planul de cercetare

### 3.1 Obiectivele cercetării

Pornind de la analiza critică asupra cercetărilor actuale din domeniul roboților pentru agricultură, lucrarea de față are drept obiectiv principal **detectarea automată a buruienilor de pe terenurile agricole în vederea combaterii acestora prin sisteme mecanice.**

Cercetările efectuate pe perioada studiilor doctorale au fost focalizate, în principal, pe dezvoltarea și implementarea unor tehnici de vedere artificială în scopul recunoașterii speciilor de plante. De asemenea, au fost făcute cercetări cu privire la navigarea robotului printre rândurile de culturi. La finalul tezei este prezentat un concept de robot autonom pentru combaterea buruienilor prin sisteme mecanice. Pentru atingerea obiectivului principal, au fost stabilite trei obiective majore:

- Obiectiv 1 - Recunoașterea speciilor de plante
- Obiectiv 2 - Navigarea printre rândurile de culturi
- Obiectiv 3 - Înlăturarea mecanică a buruienilor

Pentru atingerea acestor obiective au fost efectuate următoarele activități:

- Activitate 1 - Procesări de imagini în scopul recunoașterii speciilor de plante pe baza culorilor și a formelor frunzelor.
- Activitate 2 - Aplicații robotizate pentru selectarea frunzelor anumitor specii de plante.
- Activitate 3 - Dezvoltarea unor metode de calibrare a camerelor video, care să fie utilizabile pe terenurile agricole.
- Activitate 4 - Corelarea punctelor omoloage ale imaginilor prelevate de la un sistem de vedere stereo format dintr-o cameră 3D-ToF (Time of Flight) și o cameră 2D color.
- Activitate 5 - Dezvoltarea unui sistem de navigare a robotului printre rândurile de culturi utilizând camere 3D-ToF (Time of Flight).
- Activitate 6 - Diferențiere buruiană/plantă pe baza diferențelor de înălțime.
- Activitate 7 - Proiectarea și realizarea unui robot pentru agricultură.

### 3.2 Planul de cercetare

Obiective	Activități	Rezultate
Obiectiv 1	Activitate 1	<ul style="list-style-type: none"> <li>▪ Recunoașterea plantelor pe baza culorilor în condiții diferite de iluminare.</li> <li>▪ Recunoașterea frunzei de ardei pe baza componentei „R” a spațiului de culoare RGB.</li> <li>▪ Recunoașterea speciei de plantă pe baza geometriei frunzei.</li> </ul>
	Activitate 6	<ul style="list-style-type: none"> <li>▪ Detectarea buruienilor pe baza culorilor și a diferențelor de înălțime.</li> </ul>
Obiectiv 2	Activitate 5	<ul style="list-style-type: none"> <li>▪ Detectarea rândurilor de culturi pe baza diferențelor de înălțime determinate prin procesarea datelor prelevate de la camera 3D-ToF.</li> </ul>
Obiectiv 1 Obiectiv 2	Activitate 3	<p>Dezvoltarea a trei metode originale de calibrare a camerelor video:</p> <ul style="list-style-type: none"> <li>▪ Metodă de calibrare intrinsecă</li> <li>▪ Metodă de calibrare extrinsecă prin utilizare de modele CAD</li> <li>▪ Metodă analitică de calibrare extrinsecă</li> </ul>
	Activitate 4	<ul style="list-style-type: none"> <li>▪ Corelarea pixelilor din imaginea camerei 3D-ToF cu pixelii omologi din imaginea camerei 2D-color.</li> </ul>
Obiectiv 3	Activitate 2	<ul style="list-style-type: none"> <li>▪ Selectarea frunzelor a trei specii de plante (roșie, ardei și vânăță) prin utilizarea unui robot la care este atașat un sistem vizual.</li> </ul>
	Activitate 7	<ul style="list-style-type: none"> <li>▪ Concept pe robot pentru agricultură.</li> </ul>

Tab. 3. 1. Plan de cercetare – obiective, activități și rezultate

## 4 Contribuții la recunoașterea speciilor de plante pe baza culorilor

### 4.1 Introducere

Recunoașterea speciilor de plante este una dintre cele mai dificile probleme existente în domeniul roboților pentru agricultură. Schimbările de luminozitate cauzează dificultăți majore de recunoaștere a plantelor. Pentru a rezolva această problemă, teza de doctorat prezintă rezultatele unor studii privind dezvoltarea unei metode de recunoaștere a plantelor cu eficiență ridicată la schimbările condițiilor de luminozitate. În acest scop, au fost efectuate experimente pentru determinarea influențelor luminii asupra culorilor plantelor. Fiecare specie de plantă are o anumită nuanță de verde și o anumită textură prin care se deosebește de celelalte specii de plante. Identificarea individuală a plantelor poate fi realizată prin procesarea digitală a imaginilor. În experiment au fost utilizate modelele de culoare RGB și HSV.

Modelul de culoare RGB este un model de culoare aditiv, în care roșu, verde și albastru sunt combinate în diferite moduri pentru a reproduce o gamă largă de culori. Numele modelului provine de la inițialele celor trei culori primare; R (roșu), G (verde) și B (albastru). Modelul de culoare HSV este derivat din modelul RGB, iar culorile sunt definite de componentele H (nuanță), S (saturație) și V (valoare).

Intensitatea luminoasă influențează puternic valorile componentelor culorilor. Pentru a mări eficiența procesului de recunoaștere a plantelor pe baza culorilor, autoarea a corelat valorile componentelor RGB sau HSV cu valorile tensiunilor unei fotodiode amplasate în apropierea plantei. În experimentele ce urmează a fi prezentate, fotodioda este utilizată ca și instrument de estimare a intensității luminii.

În cadrul capitolului sunt prezentate rezultatele unor cercetări care vizează recunoașterea speciilor de plante pe baza culorilor.

### 4.2 Standul experimental

În Fig. 4. 1 este prezentată schema electrică, în care  $i$  reprezintă intensitatea curentului alternativ,  $I$  reprezintă intensitatea luminoasă, și  $u$  reprezintă tensiunea fotodiodei. Folosind o fotodiodă, intensitatea luminii este convertită în tensiune. Tensiunea este măsurată cu un voltmetru.

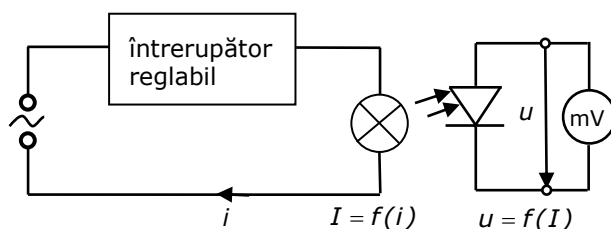


Fig. 4. 1. Conversia din intensitate luminoasă în tensiune [Tilneac, 2009]

### 30 Contribuții la recunoașterea speciilor de plante pe baza culorilor - 4

Experimentul a fost efectuat în laborator pe timpul nopții, din cauză că în timpul zilei intensitatea luminoasă este foarte neuniformă. Din cauza condițiilor meteo sau a poziției soarelui, luminozitatea diurnă este instabilă.

Fig. 4. 2 prezintă standul experimental. Acesta este format din următoarele componente: sursă de lumină (bec mat 100W), întrerupător reglabil, fotodiodă, voltmetru, cameră video, și un suport prevăzut cu dispozitiv de translație pe verticală și dispozitiv de reglare a unghiului.

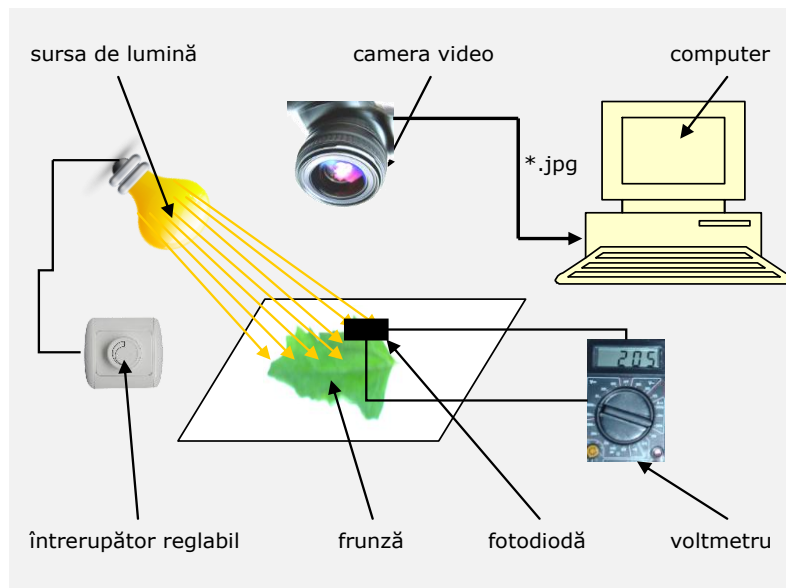


Fig. 4. 2. Schema standului experimental

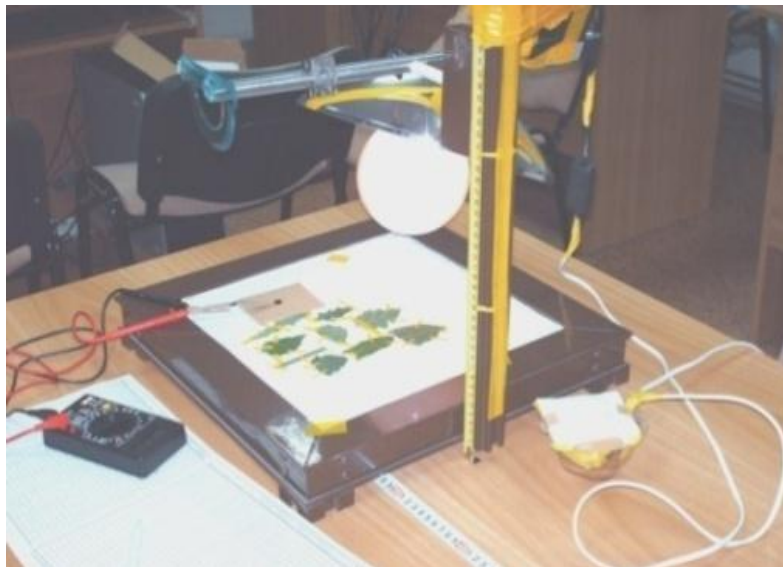


Fig. 4. 3. Standul experimental

### 4.3 Analiza experimentală a culorilor frunzelor în condiții variabile de iluminare

Analiza experimentală a fost desfășurată pentru frunze ce aparțin unor specii diferite de plante (roșie, ardei și vânăță). Activitatea a fost structurată în trei experimente. Experimentele au fost efectuate pe standul experimental prezentat anterior (Fig. 4. 3). Obiectivele urmărite în experimentele 1, 2 și 3 sunt prezentate în Tab. 4. 1.

Experiment	Obiectivele urmărite
Experiment 1	<ul style="list-style-type: none"> <li>▪ Verificarea ipotezei de recunoaștere a frunzelor pe baza culorilor în condiții variabile de iluminare.</li> <li>▪ Analiza modului în care valorile componentelor R, G, B (specifice plantelor de roșie, ardei și vânăță) se modifică în funcție de nivelul de iluminare estimat prin valoarea tensiunii fotodiodei.</li> <li>▪ Vizualizarea grafică a rezultatelor prin corelarea valorilor tensiunilor fotodiodei cu fiecare dintre componentele de culoare R, G, B specifice frunzelor de roșie, ardei și vânăță (Fig. 4. 5, Fig. 4. 6, Fig. 4. 7 și Fig. 4. 8).</li> </ul>
Experiment 2	<ul style="list-style-type: none"> <li>▪ Determinarea intervalelor valorilor componentelor de culoare RGB și HSV (specifice plantelor de roșie, ardei și vânăță) în funcție de nivelul de iluminare estimat prin valoarea tensiunii fotodiodei.</li> <li>▪ Vizualizarea grafică a intervalelor rezultate și corelarea valorilor tensiunilor fotodiodei cu fiecare dintre componentele de culoare R, G, B și H, S, V specifice frunzelor de roșie, ardei și vânăță (Fig. 4. 9, Fig. 4. 10, Fig. 4. 11, Fig. 4. 12, Fig. 4. 13 și Fig. 4. 14).</li> </ul>
Experiment 3	<ul style="list-style-type: none"> <li>▪ Verificarea rezultatelor experimentelor 1 și 2 prin repetarea acestora utilizând alte eșantioane de frunze de roșie, ardei și vânăță, amplasate în locații diferite față de sursa de lumină.</li> </ul>

Tab. 4. 1. Obiectivele urmărite în experimentele 1, 2 și 3

#### Experimentul 1

Utilizând standul experimental din Fig. 4. 3, au fost analizate frunze ce aparțin unor specii diferite de plante (roșie, ardei și vânăță). Tab. 4. 2 prezintă simbolurile parametrilor utilizați în experiment și semnificația acestora. Cele 27 de

## 32 Contribuții la recunoașterea speciilor de plante pe baza culorilor - 4

imagini, prelevate de la o cameră video, au fost descărcate pe computer în format JPEG. Imaginile sunt prezentate în Tab. 4. 3. Fișierele în format JPEG au fost importate în mediul de programare MATLAB. De pe fiecare frunză au fost extrase manual zece valori RGB distincte, așa cum este prezentat în Fig. 4. 4. Media aritmetică a celor zece valori distincte a fost considerată ca fiind valoarea R, G, respectiv B, specifică plantei în acea imagine. Rezultatele experimentale au fost procesate în programul MATLAB.

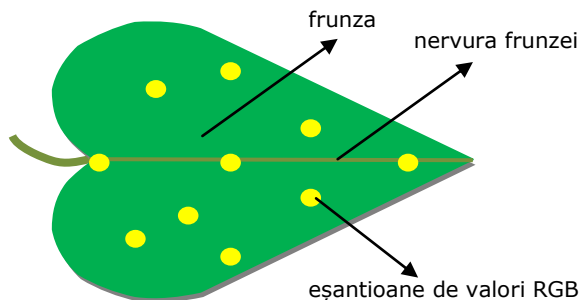
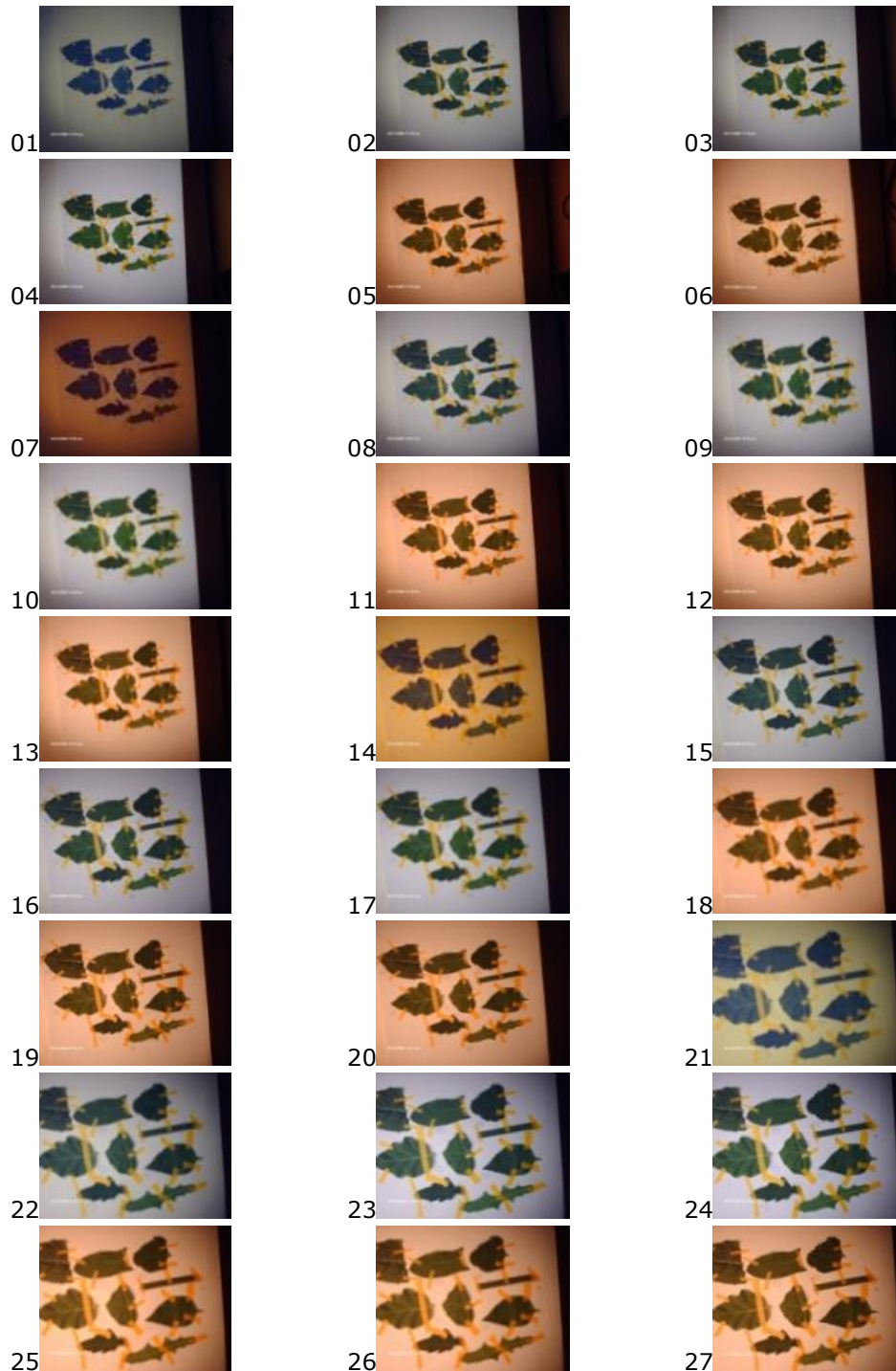


Fig. 4. 4. Eșantionarea valorilor RGB [Tilneac, 2010a]

Parametru	Semnificația	Unitatea
1*	Tensiunea fotodiodei	[mV]
2*	Distanța între camera video și plantă (vertical)	[mm]
3*	Distanța între camera video și plantă (orizontal)	[mm]
4*	Distanța între sursa de lumină și plantă (vertical)	[mm]
5*	Distanța între sursa de lumină și plantă (orizontal)	[mm]
6*	Distanța între sursa de lumină și fotodiodă (vertical)	[mm]
7*	Distanța între sursa de lumină și fotodiodă (orizontal)	[mm]
8*	Unghiul dintre axa optică a camerei video și planul frunzei	[°]
T (Tomato)	Frunză de roșie	-
P (Pepper)	Frunză de ardei	-
A (Aubergine)	Frunză de vânăță	-
R (Red)	Componenta „roșu” corespunzătoare modelului de culoare RGB	-
G (Green)	Componenta „verde” corespunzătoare modelului de culoare RGB	-
B (Blue)	Componenta „albastru” corespunzătoare modelului de culoare RGB	-

Tab. 4. 2. Semnificația parametrilor și unitatea de măsură





Tab. 4. 3. Imaginile utilizate în experimentul 1

### 34 Contribuții la recunoașterea speciilor de plante pe baza culorilor - 4

Relațiile (4. 1), (4. 2) și (4. 3) sunt utilizate pentru calculul mediei aritmetice a celor zece valori R, G, B distincte, prelevate de pe aceeași frunză din aceeași imagine.

$$R = \frac{R_1 + R_2 + \dots + R_{10}}{10} \quad (4. 1)$$

$$G = \frac{G_1 + G_2 + \dots + G_{10}}{10} \quad (4. 2)$$

$$B = \frac{B_1 + B_2 + \dots + B_{10}}{10} \quad (4. 3)$$

Rezultatele experimentale sunt prezentate în următoarele tabele și figuri:

- Tab. 4. 4 prezintă valorile parametrilor la care au fost prelevate cele 27 de imagini. Semnificațiile parametrilor 1\*, 2\*, 3\*, 4\*, 5\*, 6\*, 7\* și 8\* sunt prezentate în Tab. 4. 2.
- Tab. 4. 5, Tab. 4. 6, Tab. 4. 7 și Tab. 4. 8. prezintă valorile componentelor R, G, și B pentru fiecare frunză (roșie, ardei, vânăță), raportate la tensiunile fotodiodei.

Imagine	1*	2*	3*	4*	5*	6*	7*	8*
01	268	450	0	300	150	300	150	0
02	342	450	0	300	150	300	150	0
03	367	450	0	300	150	300	150	0
04	382	450	0	300	150	300	150	0
05	390	450	0	300	150	300	150	0
06	395	450	0	300	150	300	150	0
07	260	395	0	250	150	250	150	0
08	347	395	0	250	150	250	150	0
09	370	395	0	250	150	250	150	0
10	388	395	0	250	150	250	150	0
11	400	395	0	250	150	250	150	0
12	403	395	0	250	150	250	150	0
13	404	395	0	250	150	250	150	0
14	306	345	0	200	150	200	150	0
15	354	345	0	200	150	200	150	0
16	380	345	0	200	150	200	150	0
17	395	345	0	200	150	200	150	0
18	407	345	0	200	150	200	150	0
19	413	345	0	200	150	200	150	0
20	414	345	0	200	150	200	150	0
21	272	265	0	140	130	140	130	0
22	355	265	0	140	130	140	130	0

4.3 - Analiza experimentală a culorilor frunzelor în condiții variabile de iluminare **35**

23	386	265	0	140	130	140	130	0
24	401	265	0	140	130	140	130	0
25	410	265	0	140	130	140	130	0
26	417	265	0	140	130	140	130	0
27	418	265	0	140	130	140	130	0

Tab. 4. 4. Valorile parametrilor la care au fost prelevate imaginile.

Imagine	1*	R			G			B		
		T	P	A	T	P	A	T	P	A
01	268	21	27	25	38	44	44	70	78	72
02	342	29	34	41	43	51	60	46	51	54
03	367	29	33	39	45	54	62	39	40	45
04	382	31	36	46	48	58	67	37	41	41
05	390	55	69	81	37	47	52	14	16	15
06	395	56	72	78	39	50	54	14	17	16

Tab. 4. 5 Valorile RGB corespunzătoare frunzelor de roșie, ardei și vânăță.

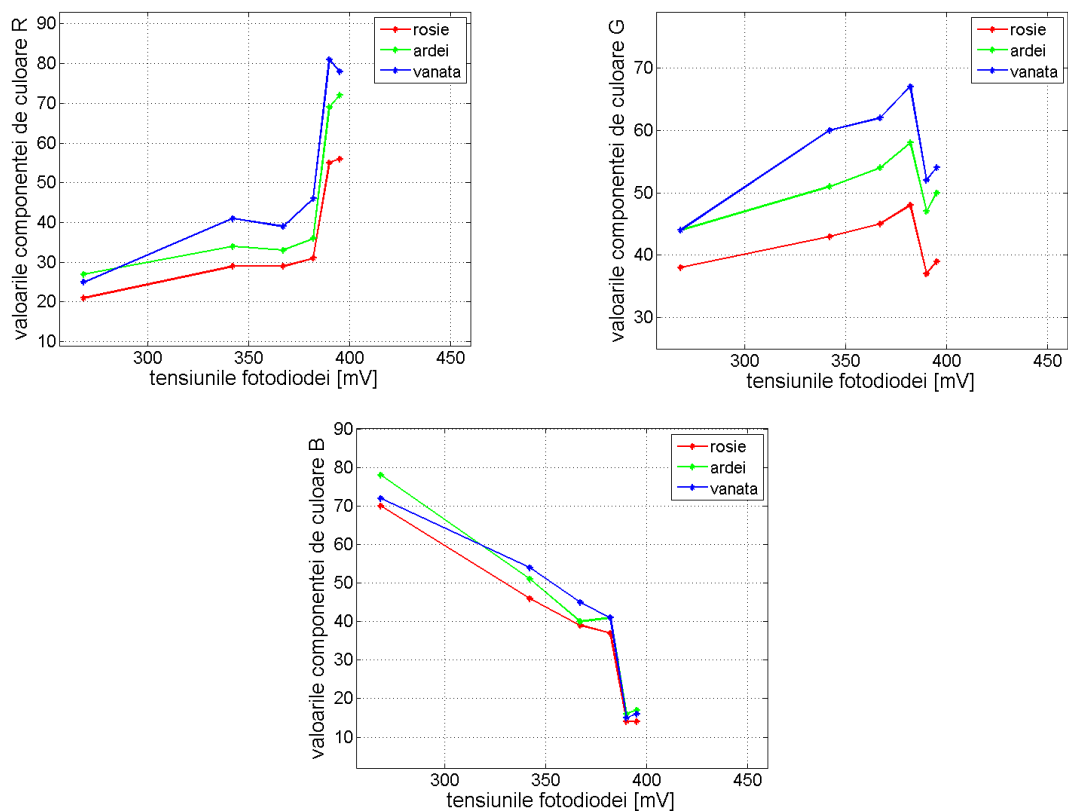


Fig. 4. 5. Grafice de corelare a valorilor R, G, B cu tensiunile fotodiodei corespunzătoare datelor din Tab.4.4.

Imagine	1*	R			G			B		
		T	P	A	T	P	A	T	P	A
07	260	28	27	41	19	21	27	50	44	48
08	347	35	35	52	49	52	69	55	52	66
09	370	37	31	56	50	52	78	48	42	59
10	388	39	59	59	56	83	83	44	51	51
11	400	71	72	107	47	50	70	15	14	22
12	403	71	69	105	49	50	71	15	14	24
13	404	71	76	103	48	54	72	16	16	23

Tab. 4. 6. Valorile RGB corespunzătoare frunzelor de roșie, ardei și vânăță.

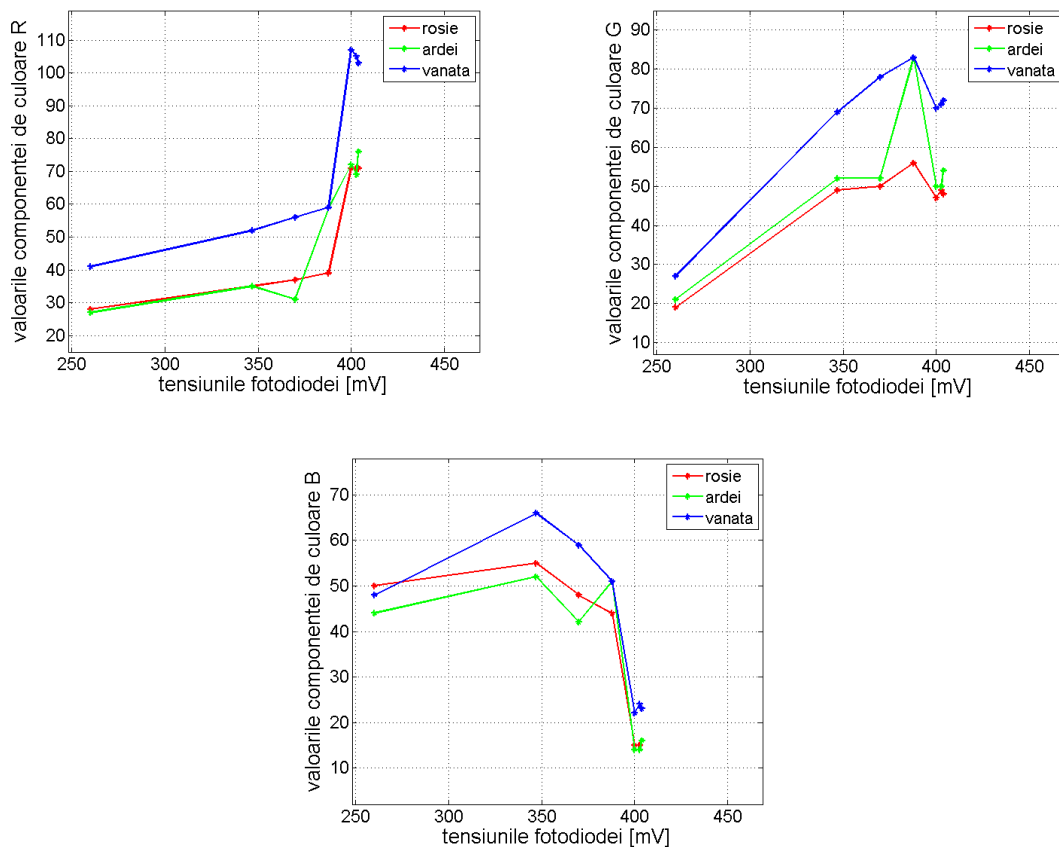


Fig. 4. 6. Grafice de corelare a valorilor R, G, B cu tensiunile fotodiodei corespunzătoare datelor din Tab.4.5.

4.3 - Analiza experimentală a culorilor frunzelor în condiții variabile de iluminare **37**

Imagine	1*	R			G			B		
		T	P	A	T	P	A	T	P	A
14	306	64	52	87	48	41	65	53	42	51
15	354	37	33	56	53	49	74	60	51	73
16	380	38	31	56	53	47	76	51	40	59
17	395	37	29	58	55	49	83	44	34	54
18	407	72	71	109	49	48	72	17	16	23
19	413	70	62	103	47	43	70	16	13	23
20	414	63	59	95	42	40	64	15	10	21

Tab. 4. 7. Valorile RGB corespunzătoare frunzelor de roșie, ardei și vânăță.

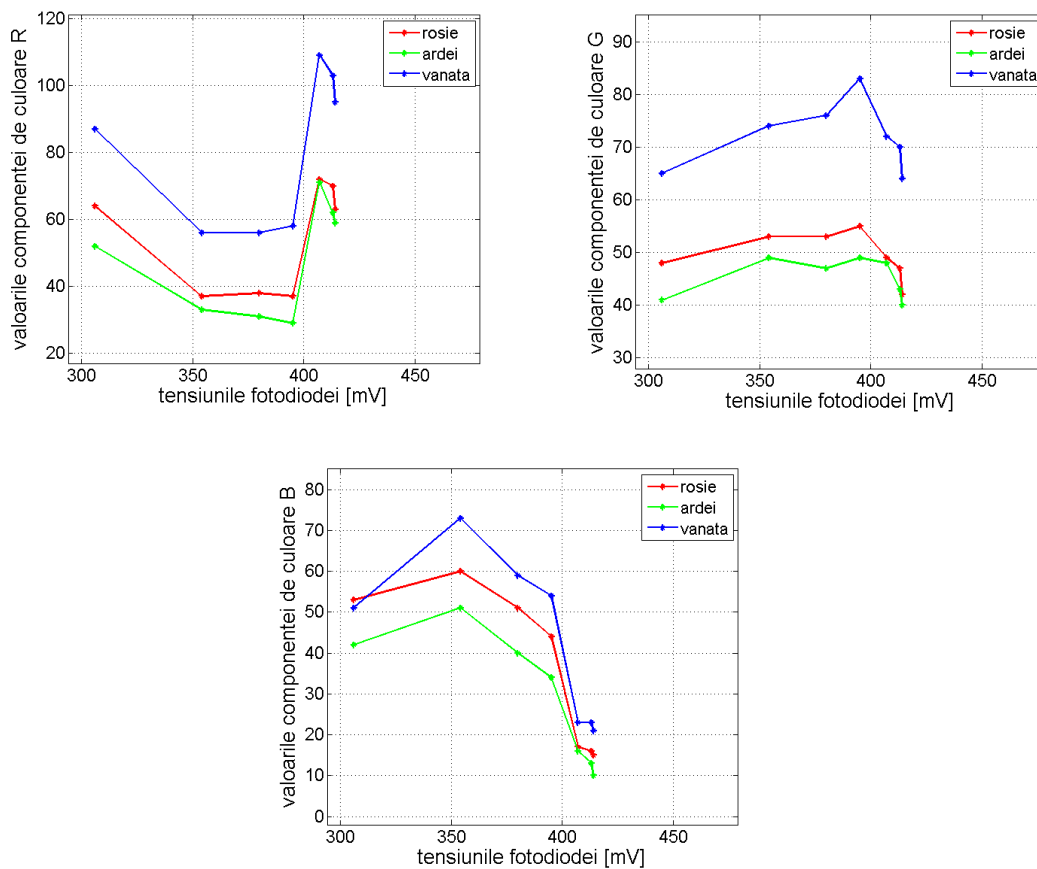


Fig. 4. 7. Grafice de corelare a valorilor R, G, B cu tensiunile fotodiodei corespunzătoare datelor din Tab.4.6.

Imagine	1*	R			G			B		
		T	P	A	T	P	A	T	P	A
21	272	52	35	75	63	50	91	87	67	102
22	355	42	28	64	58	45	83	64	46	79
23	386	75	34	75	94	51	94	84	50	84
24	401	43	31	72	60	48	93	55	41	72
25	410	86	69	129	57	47	87	20	13	31
26	417	80	62	108	53	43	78	19	14	26
27	418	76	59	116	51	41	79	19	12	28

Tab. 4. 8. Valorile RGB corespunzătoare frunzelor de roșie, ardei și vânăță.

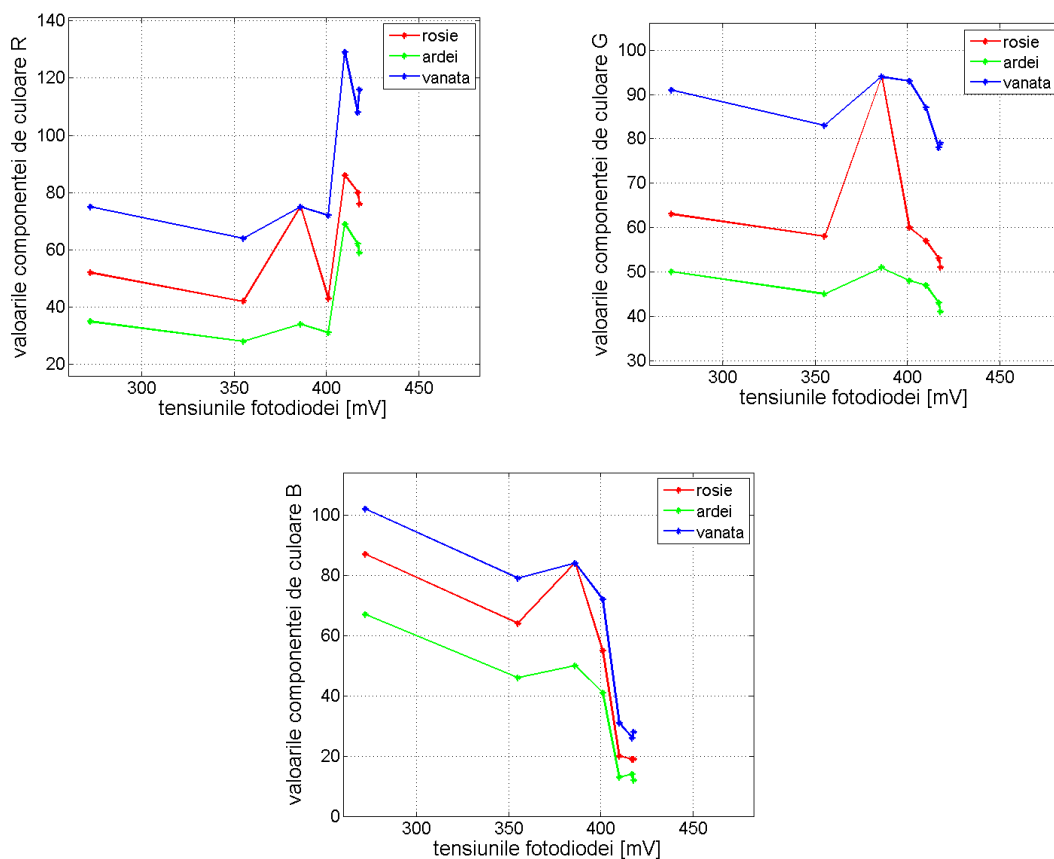


Fig. 4. 8. Grafice de corelare a valorilor R, G, B cu tensiunile fotodiodei corespunzătoare datelor din Tab.4.7.

### Concluzii desprinse din experimentul 1:

Graficele de corelare a valorilor RGB cu tensiunile fotodiodei diferă în funcție de specia de plantă. Analizând graficele (prezentate în Fig. 4. 5, Fig. 4. 6, Fig. 4. 7 și Fig. 4. 8) poate fi observat că, în majoritatea cazurilor, curbele corespunzătoare speciilor de roșie, ardei și vânăță, nu se intersectează. Aceasta însemnând că diferențierea plantelor pe baza culorilor este posibilă.

### Experimentul 2

Acest experiment are ca scop determinarea intervalelor de culoare corespunzătoare frunzelor a trei specii de plante (roșie, ardei, respectiv vânăță) în condiții diferite de iluminare.

De la o cameră video (Logitech C200) au fost prelevate, în condiții diferite de iluminare, imagini ale unor seturi de frunze (Tab. 4. 9). Utilizând un întrerupător reglabil, au fost create cinci nivele diferite de luminozitate. La fiecare nivel de luminozitate au fost înregistrate valorile tensiunilor unei fotodiode amplasate în apropierea frunzelor. Imaginile prelevate de la camera video au fost importate și analizate într-un program MATLAB, special dezvoltat pentru această aplicație. Detalii privind model de funcționare al programului urmează a fi discutate în subcapitolul 4.6. Programul este utilizat pentru a determina valorile componentelor RGB, respectiv HSV ale fiecărei frunze din imagine. Prin relațiile (4. 4) ÷ (4. 9) sunt determinate valorile mediei aritmetice ale componentelor de culoare (R, G, B, respectiv H, S, V), luând în calcul valorile fiecărui pixel care aparține obiectului selectat în imagine. Programul conține un modul de analiză statistică prin care sunt determinate intervalele în care se situează valorile componentelor RGB, respectiv HSV, corespunzătoare speciei de plantă.

$$R\_medie\_frunza = \frac{(R_1 + R_2 + \dots + R_{n\_pixel})}{n\_pixel} \quad (4. 4)$$






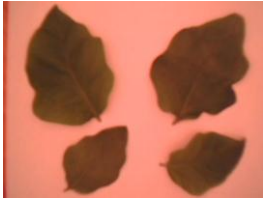









$$G\_medie\_frunza = \frac{(G_1 + G_2 + \dots + G_{n\_pixel})}{n\_pixel} \quad (4. 5)$$

$$B\_medie\_frunza = \frac{(B_1 + B_2 + \dots + B_{n\_pixel})}{n\_pixel} \quad (4. 6)$$

$$H\_medie\_frunza = \frac{(H_1 + H_2 + \dots + H_{n\_pixel})}{n\_pixel} \quad (4. 7)$$

$$S\_medie\_frunza = \frac{(S_1 + S_2 + \dots + S_{n\_pixel})}{n\_pixel} \quad (4. 8)$$

$$V\_medie\_frunza = \frac{(V_1 + V_2 + \dots + V_{n\_pixel})}{n\_pixel} \quad (4. 9)$$

Imagini		
Frunze de roșie	Frunze de ardei	Frunze de vânăță
 U=332 [mV]	 U=334 [mV]	 U=335 [mV]
 U=358 [mV]	 U=360 [mV]	 U=359 [mV]
 U=373 [mV]	 U=374 [mV]	 U=374 [mV]
 U=383 [mV]	 U=383 [mV]	 U=382 [mV]
 U=387 [mV]	 U=387 [mV]	 U=386 [mV]
U – reprezintă tensiunea fotodiodei exprimată în mV		

Tab. 4. 9. Imaginile utilizate în experimentul 2



4.3 - Analiza experimentală a culorilor frunzelor în condiții variabile de iluminare **41**

Simbol	Semnificație
n_pixel	Valoarea numărului de pixeli ce aparțin imaginii frunzei.
Rmin, Gmin, Bmin Hmin, Smin, Vmin	Valorile minime ale parametrilor: <i>R_medie_frunza, G_medie_frunza, B_medie_frunza, H_medie_frunza, S_medie_frunza, V_medie_frunza</i>
Rmed, Gmed, Bmed Hmed, Smed, Vmed	Valorile medii ale parametrilor: <i>R_medie_frunza, G_medie_frunza, B_medie_frunza, H_medie_frunza, S_medie_frunza, V_medie_frunza</i>
Rmax, Gmax, Bmax Hmax, Smax, Vmax	Valorile maxime ale parametrilor: <i>R_medie_frunza, G_medie_frunza, B_medie_frunza, H_medie_frunza, S_medie_frunza, V_medie_frunza</i>

Tab. 4. 10. Semnificația simbolurilor utilizate în experimentul 2

Graficele prezentate în Fig. 4. 9 ÷ Fig. 4. 14 indică intervalele RGB și HSV corespunzătoare frunzelor de roșie, ardei, respectiv vânăță, în condiții diferite de iluminare.

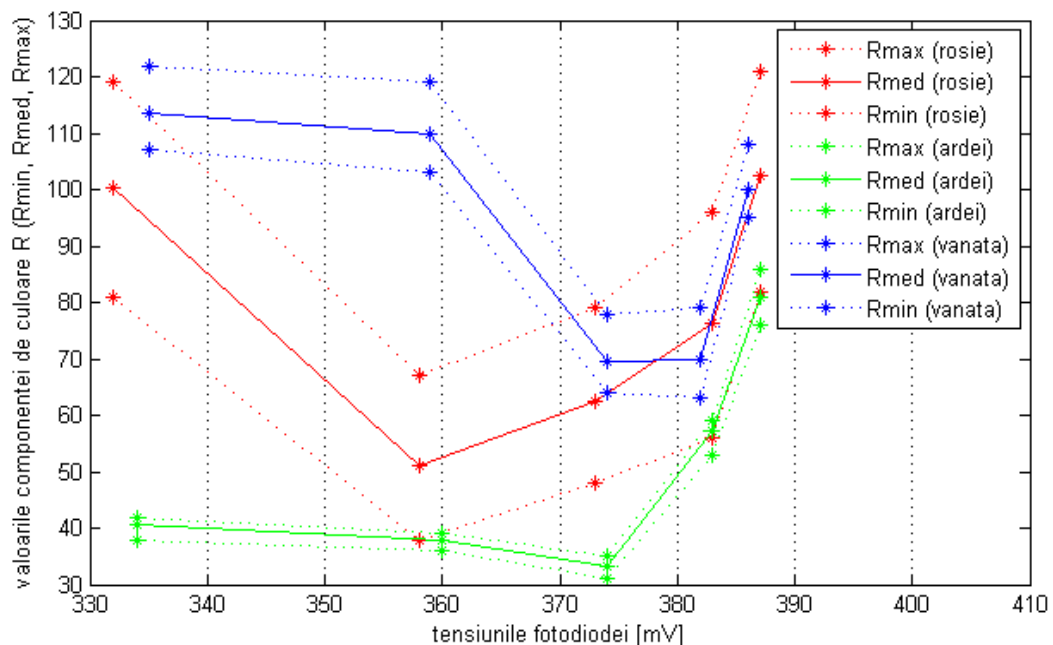


Fig. 4. 9. Analiza influenței luminii asupra componentei de culoare „R”

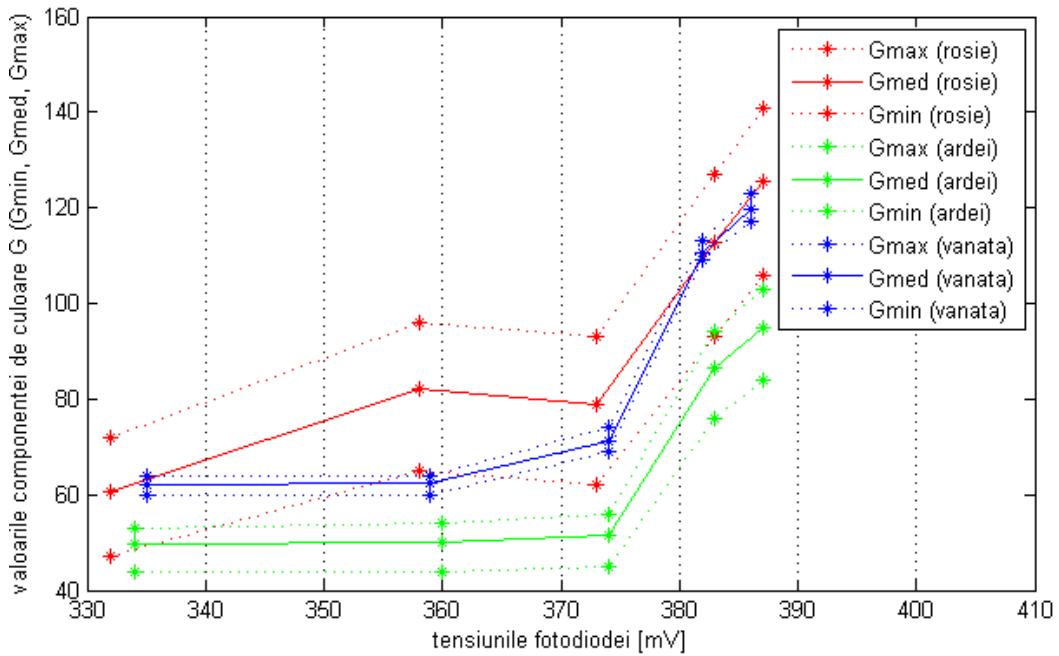


Fig. 4. 10. Analiza influenței luminii asupra componentei de culoare „G”

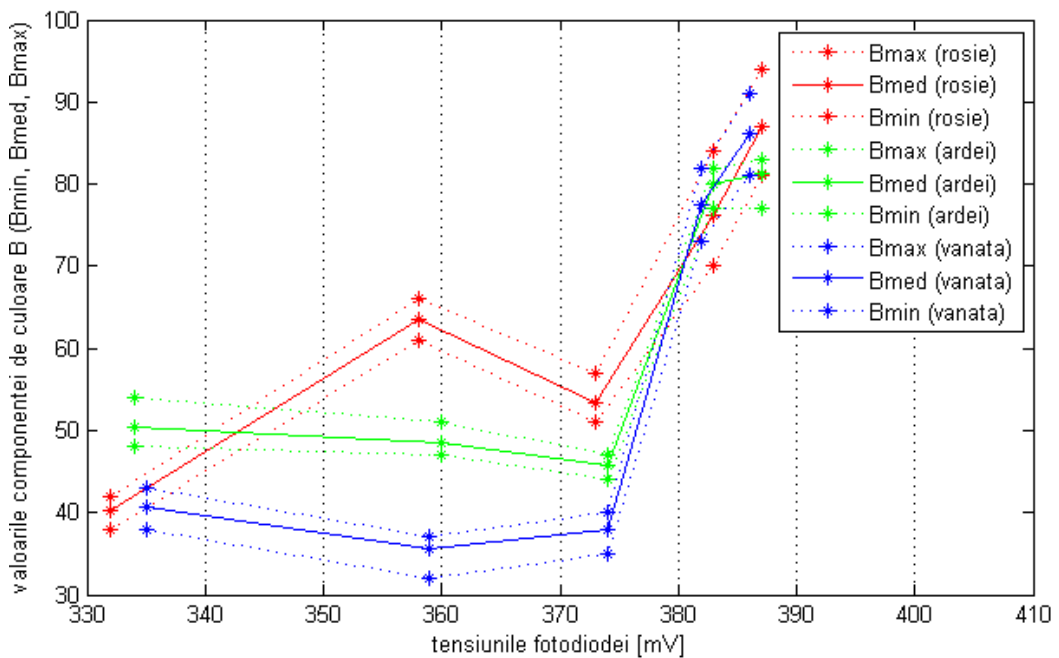


Fig. 4. 11. Analiza influenței luminii asupra componentei de culoare „B”

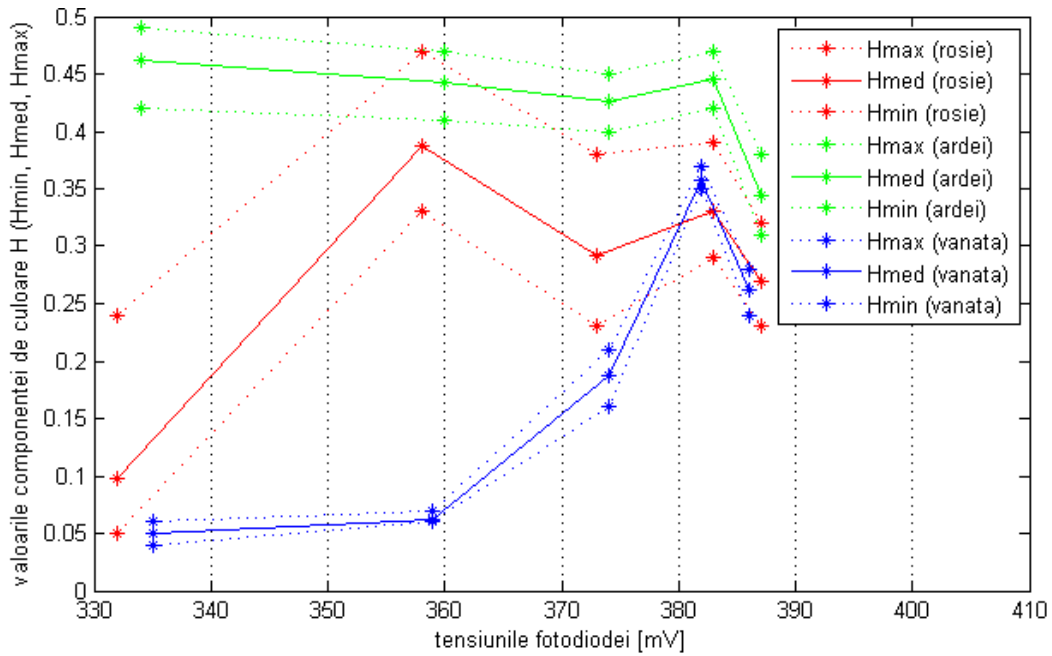


Fig. 4. 12. Analiza influenței luminii asupra componentei de culoare „H”

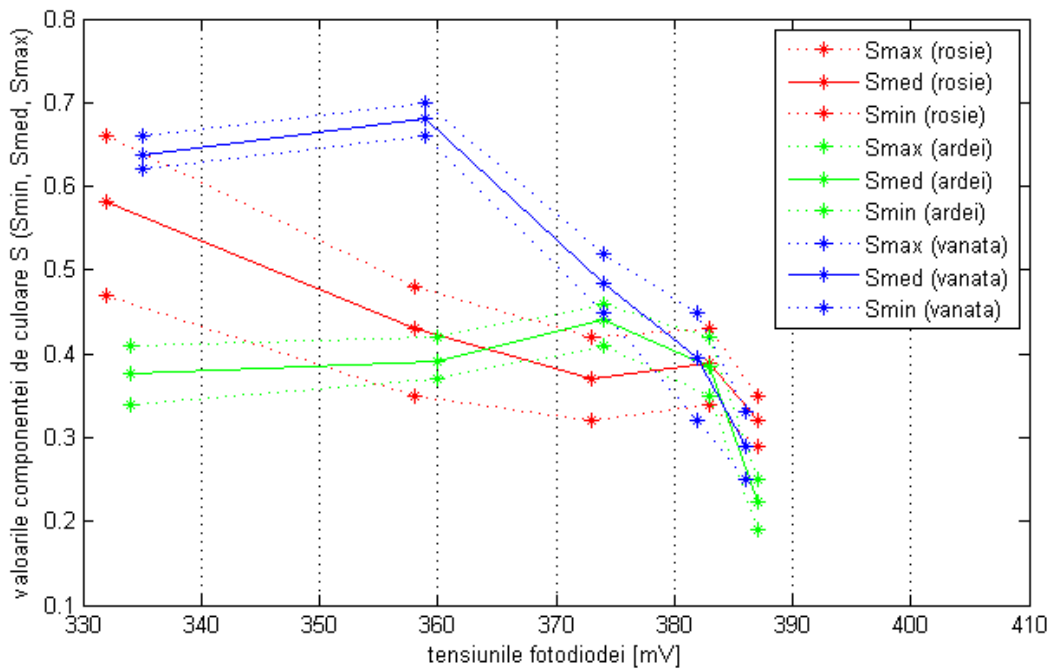


Fig. 4. 13. Analiza influenței luminii asupra componentei de culoare „S”

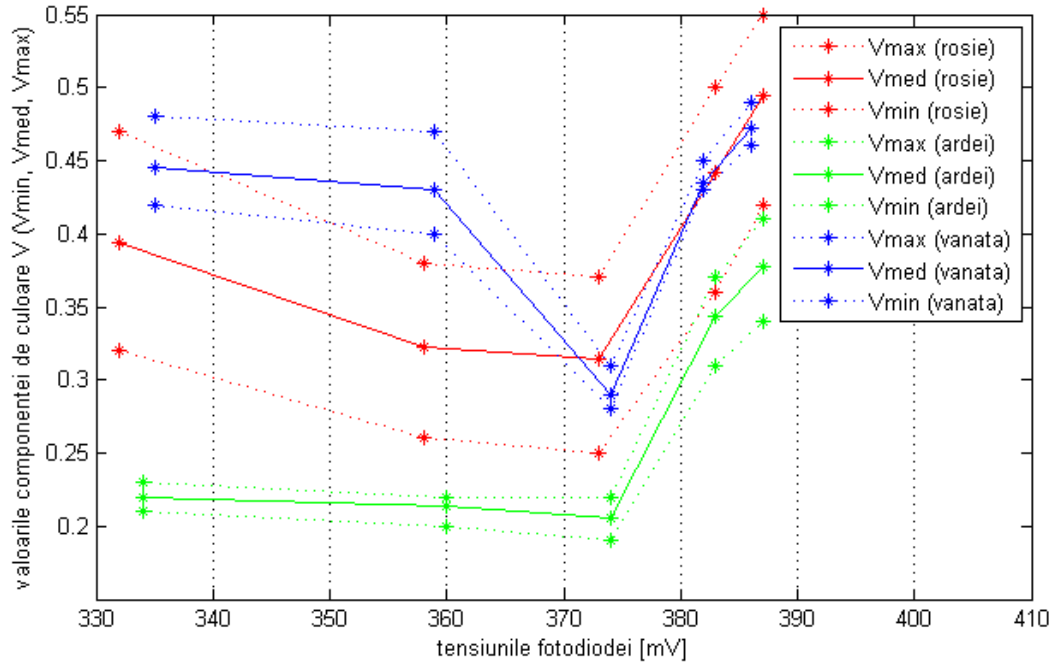


Fig. 4. 14. Analiza influenței luminii asupra componentei de culoare „V”

**Concluzii desprinse din experimentul 2:**

În Tab. 4. 11 poate fi observat faptul că diferențierea frunzelor ardei-vânăță poate fi realizată cu succes pe bază de culoare. Pătratele de culoare gri (Tab. 4. 11) indică faptul că diferențierea între speciile plante poate fi realizată în intervalul de tensiune corespunzător.

Componenta de culoare	Diferențiere ardei-roșie	Diferențiere ardei-vânăță	Diferențiere roșie-vânăță
R	■ ■ □ ■ □ □	■ ■ ■ ■ ■ ■	□ □ ■ ■ □ □
G	■ ■ ■ ■ ■ □	■ ■ ■ ■ ■ ■	□ □ ■ □ □ □
B	□ □ ■ ■ □ □	■ ■ ■ ■ □ □	□ □ ■ ■ □ □
H	■ □ □ □ □ □	■ ■ ■ ■ ■ ■	□ ■ ■ ■ □ □
S	□ □ □ □ □ □	■ ■ ■ ■ □ □	□ □ ■ ■ □ □
V	■ ■ ■ ■ ■ □	■ ■ ■ ■ ■ ■	□ □ ■ □ □ □

Semnificația pătratelor 1 2 3 4 5 6:

- Pătratul 1 reprezintă tensiunea fotodiodei în intervalul 330 – 340 mV.
- Pătratul 2 reprezintă tensiunea fotodiodei în intervalul 340 – 350 mV.
- Pătratul 3 reprezintă tensiunea fotodiodei în intervalul 350 – 360 mV.
- Pătratul 4 reprezintă tensiunea fotodiodei în intervalul 360 – 370 mV.
- Pătratul 5 reprezintă tensiunea fotodiodei în intervalul 370 – 380 mV.
- Pătratul 6 reprezintă tensiunea fotodiodei în intervalul 380 – 390 mV.



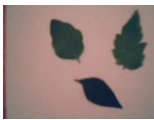






















Tab. 4. 11. Diferențierea speciilor de roșie, ardei și vânăță

#### 4.3 - Analiza experimentală a culorilor frunzelor în condiții variabile de iluminare 45

- Diferențierea ardei-roșie poate fi realizată cel mai eficient pe baza componentelor G și V în intervalele de tensiune 330 – 380 mV.
- Diferențierea ardei-vânăță poate fi realizată cel mai eficient pe baza componentelor R, G, H și V în intervalele de tensiune 330 – 390 mV.
- Diferențierea roșie-vânăță poate fi realizată cel mai eficient pe baza componentei H în intervalul de tensiune 340 – 370 mV.

### Experimentul 3

Experimentul a avut ca scop verificarea rezultatelor obținute în experimentele anterioare. În timpul desfășurării experimentelor 1 și 2 s-a constatat faptul că becul mat nu emite lumină suficient de difuză. Din această cauză unele frunze au fost mai puternic iluminate. A mai fost observat faptul că lumina este instabilă în anumite condiții, ceea ce s-ar putea să fi generat unele rezultate eronate. Pentru a demonstra faptul că într-adevăr recunoașterea plantelor pe baza culorilor poate fi realizată, a fost efectuat experimentul 3.

Imagini				
Set 1	Set 2	Set 3	Set 4	Set 5
 U=337 [mV]	 U=332 [mV]	 U=336 [mV]	 U=330 [mV]	 U=327 [mV]
 U=362 [mV]	 U=359 [mV]	 U=360 [mV]	 U=357 [mV]	 U=353 [mV]
 U=375 [mV]	 U=372 [mV]	 U=372 [mV]	 U=369 [mV]	 U=366 [mV]
 U=385 [mV]	 U=381 [mV]	 U=380 [mV]	 U=378 [mV]	 U=375 [mV]
 U=387 [mV]	 U=383 [mV]	 U=384 [mV]	 U=382 [mV]	 U=380 [mV]
U – reprezintă tensiunea fotodiodei exprimată în mV				

Tab. 4. 12. Imaginile utilizate în experimentul 3

Un număr de cîte cinci eşantioane de frunze (roşie, ardei şi vînată) au fost analizate în condiții diferite de iluminare şi poziție. În experiment au fost determinate valorile componentelor de culoare R,G,B ale fiecărei frunze din imaginile prezentate în Tab. 4. 12. Rezultatele cumulative sunt reprezentate grafic în Fig. 4. 15 (componentele de culoare R) , Fig. 4. 16 (componentele de culoare G) şi Fig. 4. 17 (componentele de culoare B).

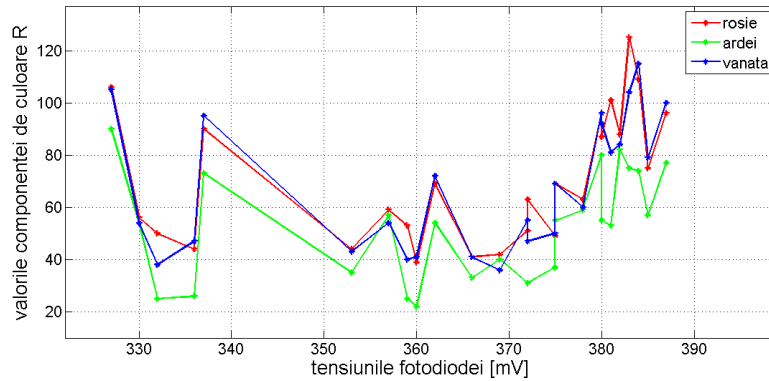


Fig. 4. 15. Analiza influenței luminii asupra componentei de culoare „R”

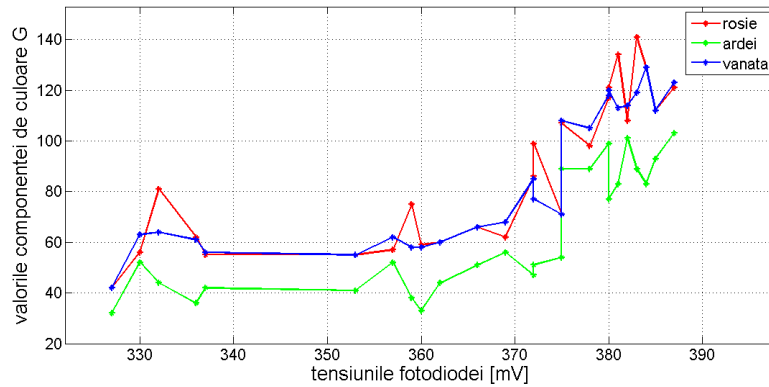


Fig. 4. 16. Analiza influenței luminii asupra componentei de culoare „G”

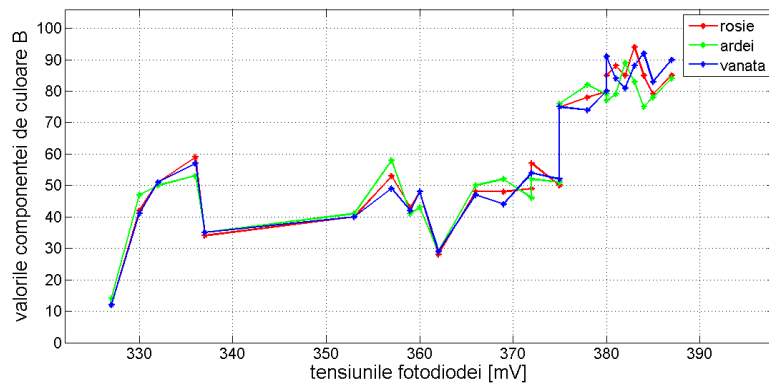


Fig. 4. 17. Analiza influenței luminii asupra componentei de culoare „B”

### Concluzii desprinse din experimentul 3:

Frunzele de ardei se disting de frunzele de roșie și vânăță prin componentele de culoare G și R.

### 4.4 Programul de procesare a culorilor

În mediul MATLAB a fost dezvoltat un program de procesare a culorilor din imagine. Acest program a fost utilizat în experimentele 2 și 3.

Lansarea programului este realizată printr-un fișier MATLAB denumit *CULOAREA\_FORMEI*. Imaginile sunt prelucrate cu ajutorul unei interfețe grafice, prin parcurgerea următoarelor etape:

- Importarea imaginii în programul MATLAB
- Vizualizarea imaginii
- Binarizarea imaginii utilizând un filtru de culoare creat pe baza modelului de culoare HSV
- Curățarea zgomotului din imagine
- Selectarea obiectului corespunzător frunzei care prezintă interes
- Calculul mediei componentelor de culoare R, G, B, respectiv H, S, V ale pixelilor corespunzători obiectului selectat
- Analiza statistică pentru determinarea valorilor maxime, medii și minime ale componentelor de culoare R, G, B, respectiv H, S, V corespunzătoare frunzelor unei anumite specii de plantă

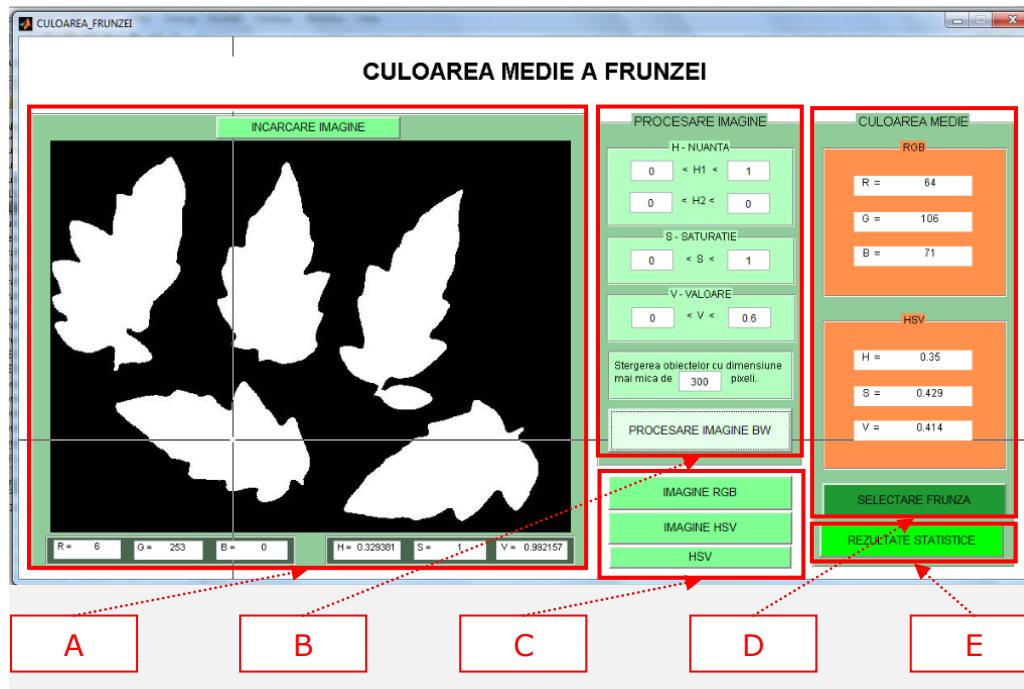


Fig. 4. 18. Interfața grafică a programului de procesare a culorilor

Interfața grafică este prezentată în Fig. 4. 18. Pentru o descriere explicită a modului de funcționare, interfața grafică este împărțită în cinci zone, notate cu A, B, C, D și E.

Zona "A" (Fig. 4. 19) conține un buton prin activarea căruia este posibilă selectarea imaginii din calculator și încărcarea acesteia în programul de procesare a imaginilor. Modulul "A" conține fereastra grafică în care sunt afișate imaginile. Sub fereastra grafică se află șase casete care au rolul de a afișa valorile componentelor de culoare RGB și HSV ale unui pixel selectat din imagine.

Zona "B" (Fig. 4. 20) conține casete de editare a parametrilor de procesare a imaginilor și butoane care permit activarea funcției de prelucrare a imaginilor. Segmentarea culorilor este realizată pe baza spațiului de culoare HSV. Pentru extragerea unei anumite nuanțe de culoare, sunt setate intervalele corespunzătoare componentelor H, S, și V. După setarea acestor intervale, în fereastra grafică apare imaginea binară. Programul are și o opțiune de curățare a zgomotului din imagine prin ștergerea automată a obiectelor de dimensiuni mici. Dimensiunea obiectelor considerate ca fiind mici, poate fi setată într-o casetă text aflată pe panoul numit *PROCESARE IMAGE*. Tot pe acest panou se află un buton care permite ștergerea din imagine a obiectelor care nu prezintă interes.

Zona "C" (Fig. 4. 21) conține butoane de afișare a imaginilor în spațiile de culoare RGB, respectiv HSV. Tot în acest modul există și un buton de afișare a unei imagini ce reprezintă spațiul de culoare HSV. Această imagine este deosebit de utilă, deoarece atunci când este selectată o anumită nuanță, în bara localizată sub imagine este afișată valoarea componentei H care reprezintă nuanța pixelului. Aceste informații sunt deosebit de utile în aplicații de filtrare a culorilor din imagine.

Zona "D" (Fig. 4. 22) permite selectarea unei anumite frunze din imagine. Pentru fiecare frunză selectată este calculată media aritmetică a fiecăreia dintre componentele RGB și HSV care corespund pixelilor localizați pe imaginea frunzei.

Zona "E" (Fig. 4. 23) permite efectuarea de calcule statistice și vizualizarea grafică a rezultatelor.

În anexa A este prezentată ordinograma programului și câteva secvențe din programul MATLAB.



Fig. 4. 19. Încărcarea imaginii



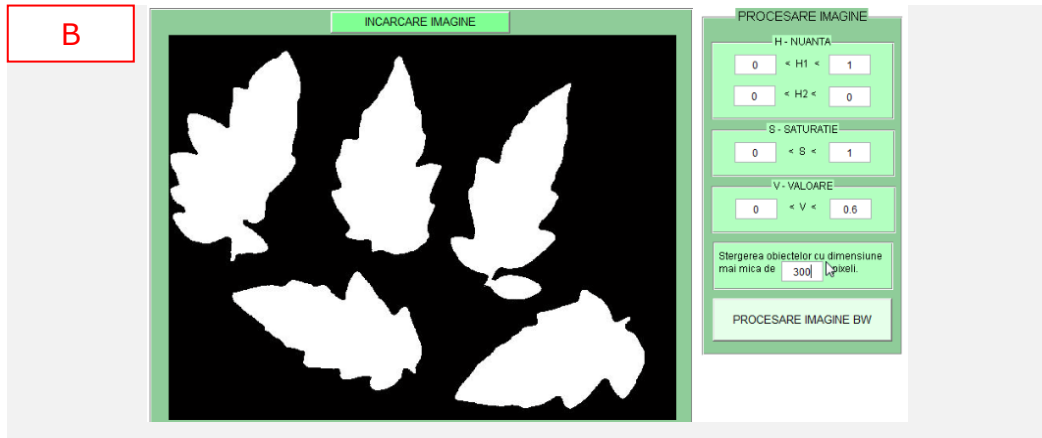


Fig. 4. 20. Segmentarea imaginii utilizând spațiul de culoare HSV

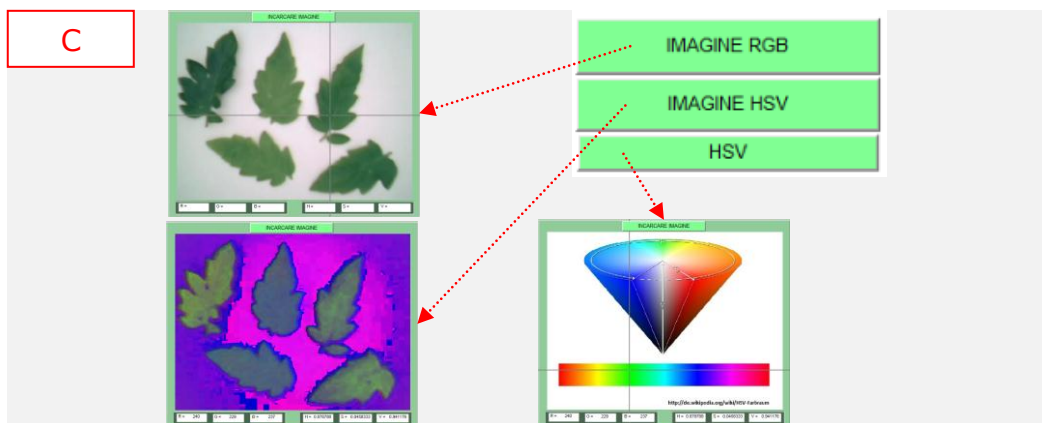


Fig. 4. 21. Afișarea imaginilor RGB și HSV; afișarea spațiului de culoare HSV

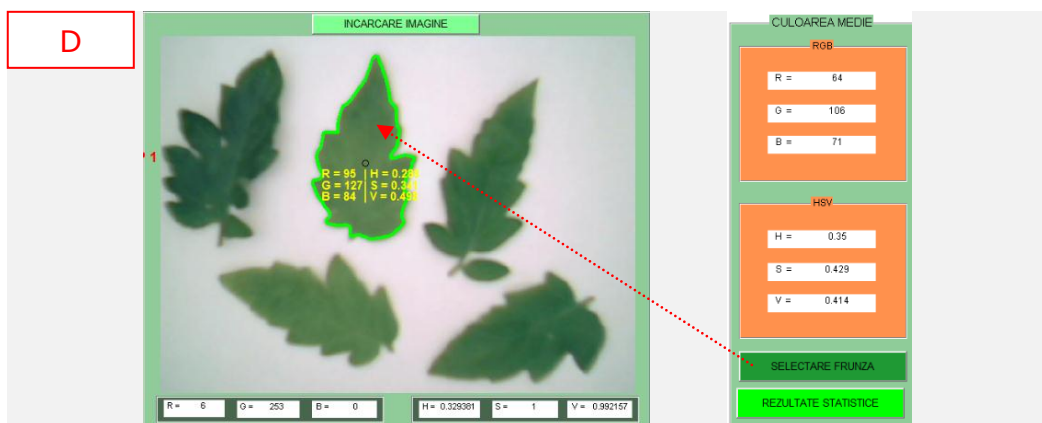


Fig. 4. 22. Selectarea frunzei și calculul valorii medii a componentelor de culoare a pixelilor corespunzători frunzei

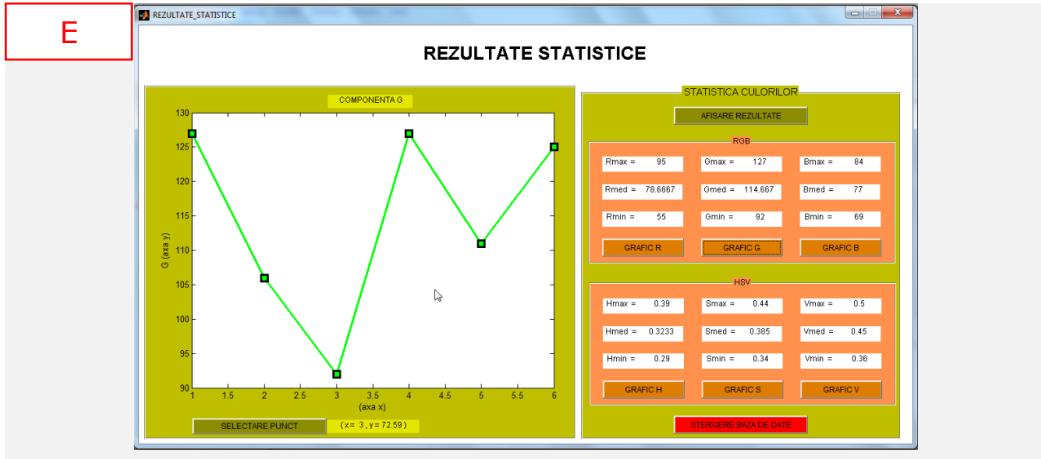


Fig. 4. 23. Interfață grafică de afișare a rezultatelor statistice

### 4.5 Concluzii

Prin analiza datelor rezultate din experimentele 1, 2 și 3 se ajunge la următoarea concluzie: *Recunoașterea speciilor de plante pe baza culorilor este posibilă în anumite condiții de iluminare.*

Din primele două experimente rezultă că eficiență maximă de diferențiere a frunzelor de roșie, ardei și vânăță apare atunci când tensiunea fotodiodei indică o valoare între 350-370 mV. Din păcate, dotarea tehnică a standului experimental nu a permis reglarea mai multor trepte de luminozitate. Graficele sunt neliniare. Pe alocuri apar tendințe crescătoare sau descrescătoare cauzate de instabilitățile sursei de lumină. Totuși, prin aceste aplicații a fost demonstrat faptul că diferențierea plantelor este posibilă în anumite condiții.

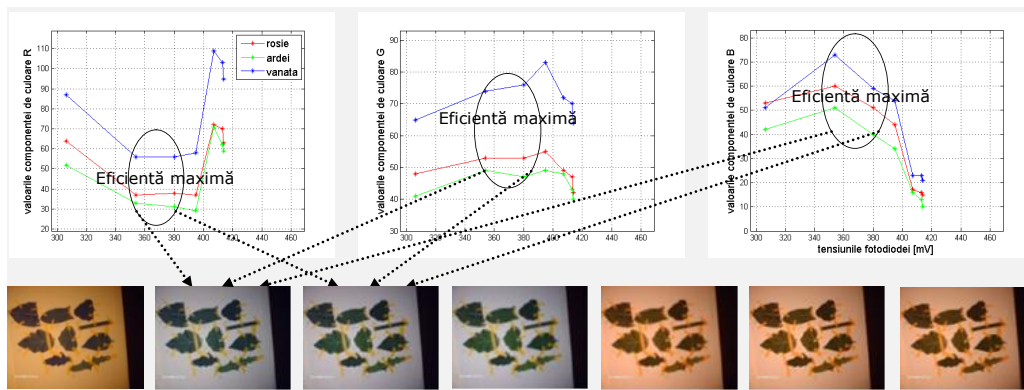


Fig. 4. 24. Experiment 1 – Studiul eficienței de recunoaștere a plantelor pe baza componentelor de culoare R, G, B

Din experimentul 2 rezultă că, la fel ca și în experimentul 1, eficiența maximă de diferențiere a platelor apare în jurul tensiunii de 360 mV.

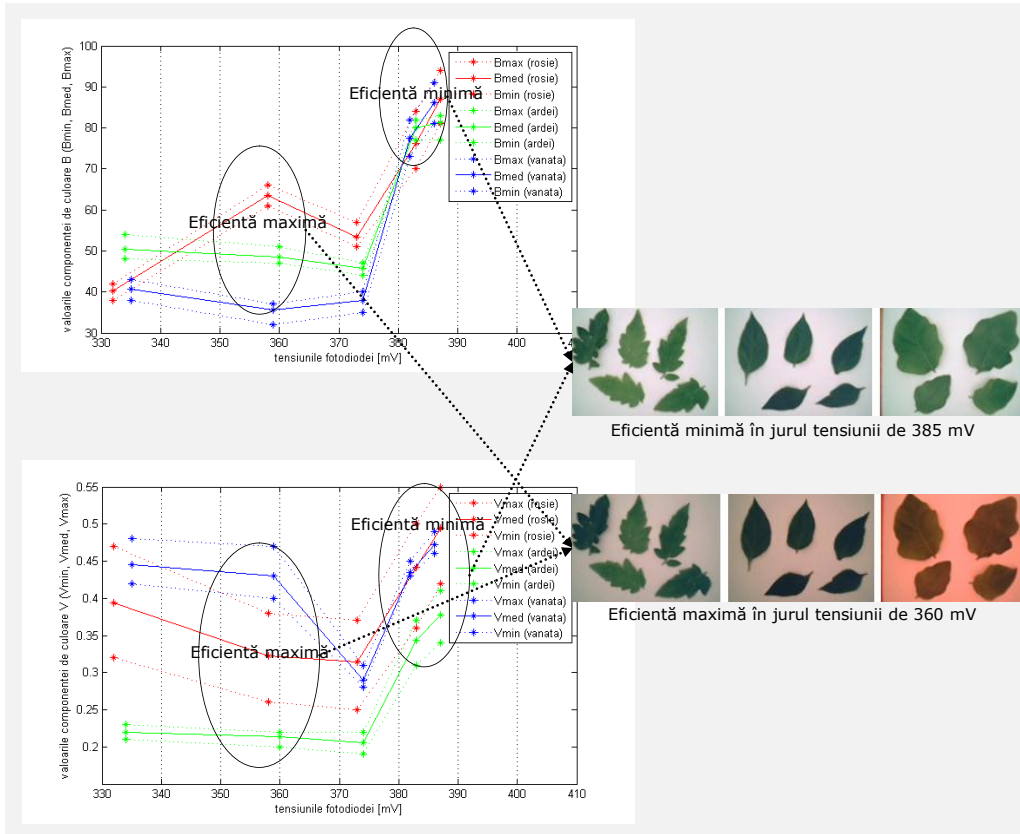


Fig. 4. 25. Experiment 2 – Studiul eficienței de recunoaștere a plantelor pe baza componentelor de culoare R, G, B sau H, S, V

Din experimentul 3 rezultă că frunzele de ardei se deosebesc puternic de frunzele de roșie și vânăță prin valoarea componente de culoare „G”.

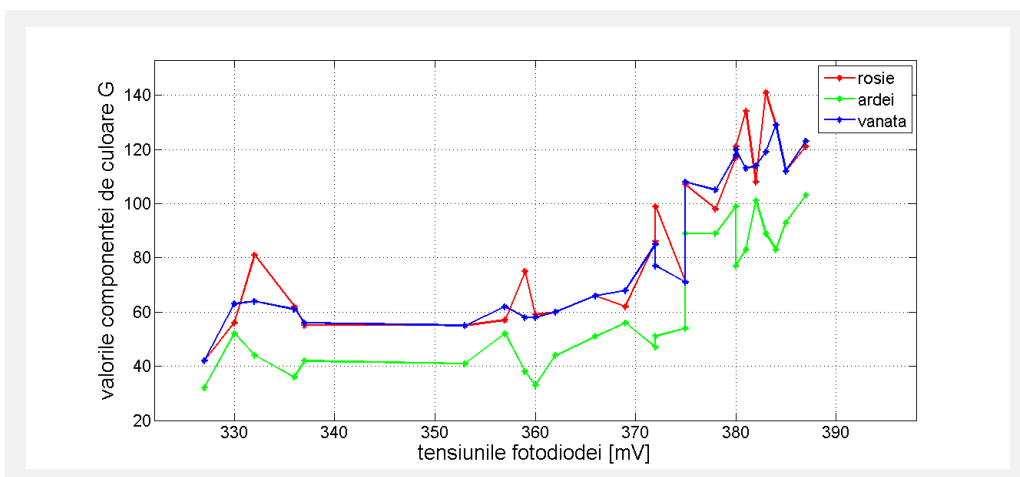


Fig. 4. 26. Experiment 3 – Recunoașterea frunzei de ardei pe baza componente „G”

## 52 Contribuții la recunoașterea speciilor de plante pe baza culorilor - 4

Tab. 4. 13 prezintă o analiză comparativă a concluziilor desprinse din experimentele 1, 2 și 3.

Experiment	Analiză comparativă
Experiment 1	<ul style="list-style-type: none"><li>▪ Din cauza numărului redus pe pixeli analizați (doar zece pixeli), rezultatele nu sunt foarte precise, dar au oferit indicii importante privind posibilitatea de recunoaștere a plantelor pe baza culorilor.</li><li>▪ Din experiment a fost constatat faptul că există diferențe ale valorilor componentelor R, G, B în funcție de specia de plantă.</li></ul>
Experiment 2	<ul style="list-style-type: none"><li>▪ Spre deosebire de experimentul 1, datorită faptului că a fost utilizat un program care permite analiza automată a culorilor tuturor pixelilor corespunzători imaginii frunzei, a fost posibilă determinarea unor intervale ale valorilor RGB și HSV specifice speciei de plantă. A fost constatat faptul că numărul de pixeli analizați nu influențează mult rezultatele experimentelor. Rezultatele obținute în experimentele 1 și 2 sunt similare.</li><li>▪ La fel ca și în experimentul 1, din experimentul 2 a rezultat faptul că există condiții de iluminare în care eficiența de recunoaștere a speciilor de plante pe baza culorilor este maximă. În experimentele 1 și 2, eficiența maximă a fost obținută atunci când tensiunea fotodiodei a indicat valori de aprox. 360 mV.</li></ul>
Experiment 3	<ul style="list-style-type: none"><li>▪ Rezultatele obținute în experimentul 3 contrazic concluziile desprinse din experimentele 1 și 2 potrivit cărora fiecare dintre speciile de plantă (roșie, ardei și vânăță) are domenii de culoare specifice. Din experiment a rezultat faptul că intervalele de culoare ale frunzelor de roșie și vânăță se suprapun.</li><li>▪ Prin experimentul 3 (la fel ca și prin experimentele 1 și 2) a fost demonstrat faptul că unele specii de plante pot fi recunoscute pe baza culorilor. De exemplu, frunza de ardei se deosebește de frunzele de roșie și vânăță prin valorile componentelor de culoare R și G.</li></ul>

Tab. 4. 13. Analiză comparativă a concluziilor desprinse din experimentele 1, 2 și 3

## 5 Contribuții la recunoașterea speciilor de plante pe baza formelor frunzelor

### 5.1 Introducere

În cadrul capitolului sunt prezentate rezultatele unor cercetări care vizează sinteza hardware și software a unui sistem de percepție vizuală pentru recunoașterea speciilor de plante. Cercetările au ca scop final detectarea buruienilor din culturile agricole. Realizarea obiectivului este considerată a fi posibilă în cadrul a două etape:

- Recunoașterea speciilor de plante în imaginile achiziționate.
- Separarea componentelor corespunzătoare plantelor cultivate de restul componentelor din imagine care se atribuie buruienilor.

Activitatea desfășurată a fost structurată în experimentele 1, 2, 3 și 4 ale acestui capitol.

### 5.2 Experiment 1 - Identificarea culorilor și a formelor prin procesare de imagini

Aplicația software are ca scop final recunoașterea și localizarea plantelor. Pentru a atinge acest obiectiv, în mediul de programare MATLAB, a fost dezvoltat un program de recunoaștere a obiectelor din imagine. În cadrul acestui subcapitol este prezentat un program software de recunoaștere a unor figuri geometrice simple (pătrat, triunghi și cerc). Programul a fost ulterior extins pentru aplicații de recunoaștere a frunzelor. Fig. 5. 1 prezintă structura aplicației MATLAB.

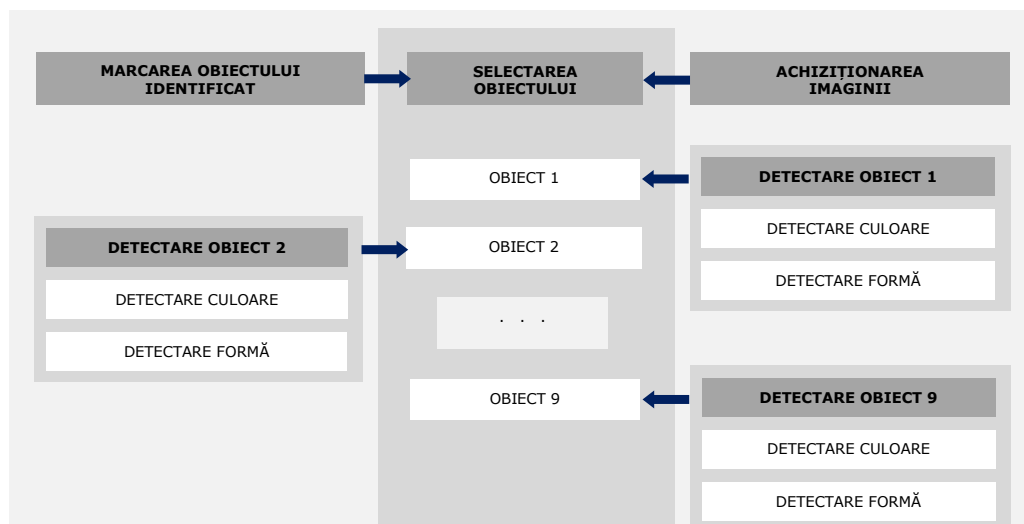


Fig. 5. 1. Structura aplicației MATLAB pentru recunoașterea obiectelor

OBIECT = [formă & culoare]		
[pătrat & roșu]	[pătrat & verde]	[pătrat & albastru]
[cerc & roșu]	[cerc & verde]	[cerc & albastru]
[triunghi & roșu]	[triunghi & verde]	[triunghi & albastru]

Fig. 5. 2. Obiectele utilizate în experiment

Aplicația a fost construită modular, pentru a separa diferite funcții în părți independente. Astfel, fiecare modul poate fi modificat fără ca restul programului să fie perturbat. Diagrama din Fig. 5. 1 descrie structura programului, ilustrând prin săgeți legătura dintre module.

Diferențierea figurilor geometrice (pătrat, cerc și triunghi) este realizată pe baza valorilor parametrilor de circularitate:

$$circularitate = \frac{4 * pi * area}{perimetru^2} \quad [\text{MATLAB – Image Processing Toolbox}] \quad (5. 1)$$

Valoare circularitate	Pătrat		Cerc		Triunghi	
	Min	Max	Min	Max	Min	Max
	0.72	0.83	0.89	1	0.59	0.67

Tab. 5. 1. Intervalele de circularitate corespunzătoare formelor: pătrat, cerc și triunghi

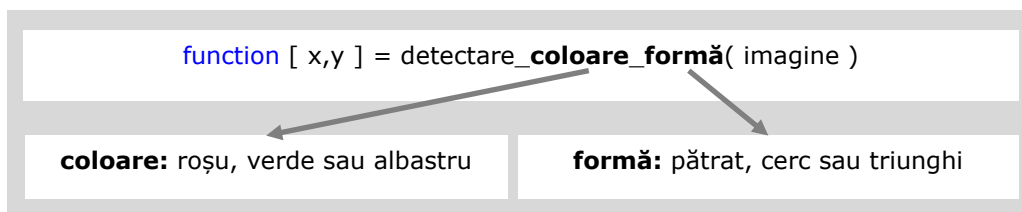


Fig. 5. 3. Explicarea funcției detectare\_culoare\_forma

Funcția MATLAB	Explicarea funcției
<code>function [ imagine ] = achizitionare_imagine()</code>	Lansează aplicația de prelevare a imaginilor
<code>function [ x, y ] = detectare_culoare_forma(imagine)</code>	Extrage din imagine culoarea ce prezintă interes. Returnează imaginea binarizată. Analizează imaginea binarizată pentru a detecta formele. Returnează coordonatele (x, y) ale centrului obiectului detectat.
<code>function imageModif = marcarea_obiect(imagine, x, y, t )</code>	Marchează în imagine centrul de greutate al obiectului detectat.

Tab. 5. 2. Explicarea funcțiilor programului

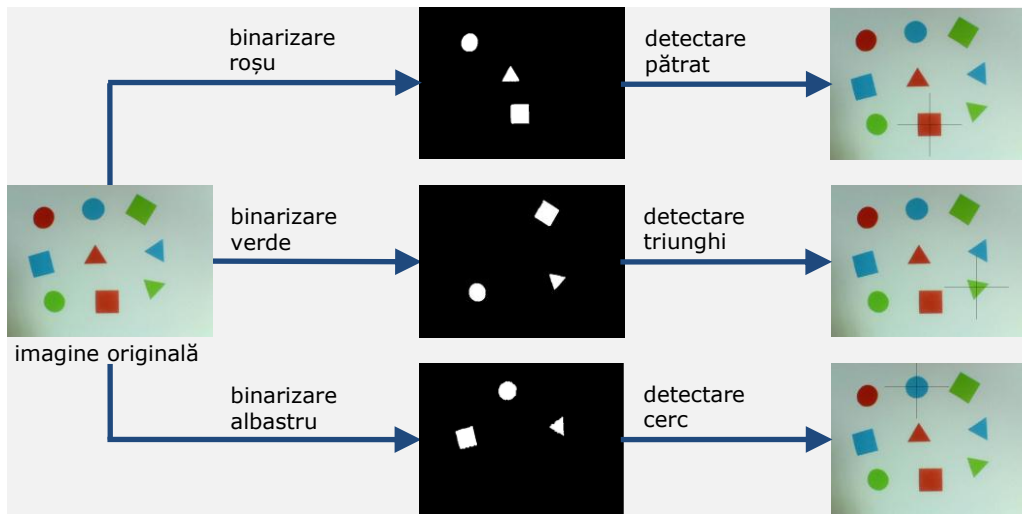


Fig. 5. 4. Procesarea imaginii pentru detectarea culorilor și a formelor

### Concluzii desprinse din experimentul 1:

- Obiectele din Fig. 5. 4 pot fi recunoscute eficient pe baza culorilor și a caracteristicilor geometrice.
- Unele lentile ale camerei video creează distorsiuni ale imaginii. Din această cauză, obiectele îndepărtate de punctul principal al imaginii (centrul imaginii) sunt deformate. Distorsiunile imaginii creează probleme de recunoaștere a obiectelor.

### 5.3 Experiment 2 - Recunoașterea semnăturilor frunzelor

Scopul prezentului studiu este realizarea unui sistem de vedere artificială pentru recunoașterea și localizarea frunzelor individuale. Pentru a atinge acest obiectiv, a fost dezvoltat un program de procesare a imaginilor în MATLAB. Sistemul este capabil să detecteze și să localizeze frunzele unor specii de plante (roșie, ardei și vânăță) pe baza caracteristicilor de semnătură [Gui, 1999], [Gonzalez, 2009].

Procesul de recunoaștere artificială a frunzelor pornește de la observația că oamenii recunosc frunzele, în cele mai multe cazuri, pe baza identificării marginilor. Câteva tipuri de margini ale frunzelor sunt prezentate în Fig. 5. 5. Pe acest considerent se poate preciza că marginile frunzelor de roșie sunt lobate, marginile frunzelor de ardei sunt netede, marginile frunzelor de vânăță sunt ondulate. În Fig. 5. 6 sunt prezentate câteva imagini de frunze ce aparțin celor trei specii de plante.



Fig. 5. 5. Tipuri de margini ale frunzelor [\*\*\* Img, Leaf]



Fig. 5. 6. Frunze de roșie, ardei și vânăță (de la stânga la dreapta)

Pornind de la această concluzie, una dintre cele mai relevante caracteristici de descriere ale marginilor frunzelor este realizată cu funcția *semnătura\_frunzei*, care calculează distanța de la centrul de greutate ( $x_{CG}$ ,  $y_{CG}$ ) al formei obiectului până la fiecare punct de pe conturul (frontiera) formei.

Fig. 5. 7 ilustrează procedeul de obținere a semnăturii frunzei. Primul punct din stânga (aflat pe conturul frunzei) este punctul de START. Pornind de la punctul de start, pixelii conturului sunt parcurși în sensul acelor de ceasornic. Este calculată distanța (exprimată în pixel) între centrul de greutate al frunzei și fiecare pixel de pe contur. Semnătura rezultă prin ilustrarea grafică a valorilor succesive ale distanțelor.

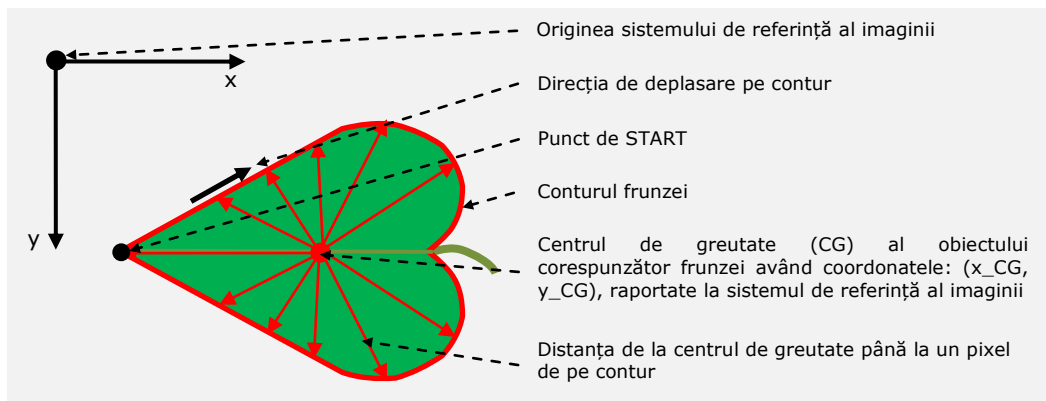


Fig. 5. 7. Distanțele de la centrul de greutate până la fiecare pixel care corespunde marginii frunzei în imagine. Detalii în anexa B.

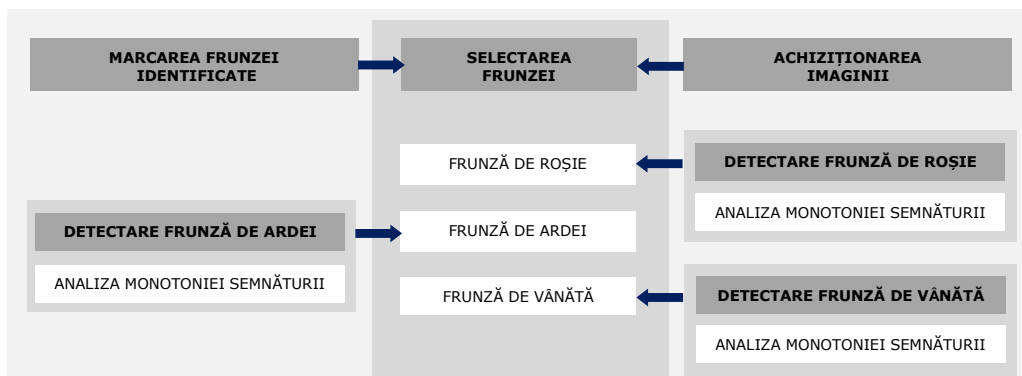


Fig. 5. 8. Structura aplicației MATLAB (concept aflat în fază de studiu)



Structura aplicației MATLAB este prezentată în Fig. 5. 8. Programul este construit modular pentru a crea posibilitatea de a adăuga module pentru detectarea altor specii de plante.

În Tab. 5. 3 sunt explicate principalele funcții ale programului. Ordinograma programului și unele secvențe din program sunt prezentate în anexa B1.

Funcția MATLAB	Explicarea funcției
<code>function [imagine] = achizitionare_imagine()</code>	Lansează aplicația de prelevare a imaginilor
<code>function [x, y] = detectare_frunza(imagine)</code>	Extrage din imagine conturul frunzei și returnează valorile (x, y) corespunzătoare poziției pixelilor localizați pe contur. Returnează coordonatele (x0, y0) ce corespund centrului frunzei detectate.
<code>function [semnatura]= caracteristici_frunza (contur)</code>	Returnează două caracteristici de formă ale frunzei (compactitate și semnătură).

Tab. 5. 3. Explicarea principalelor funcții ale programului

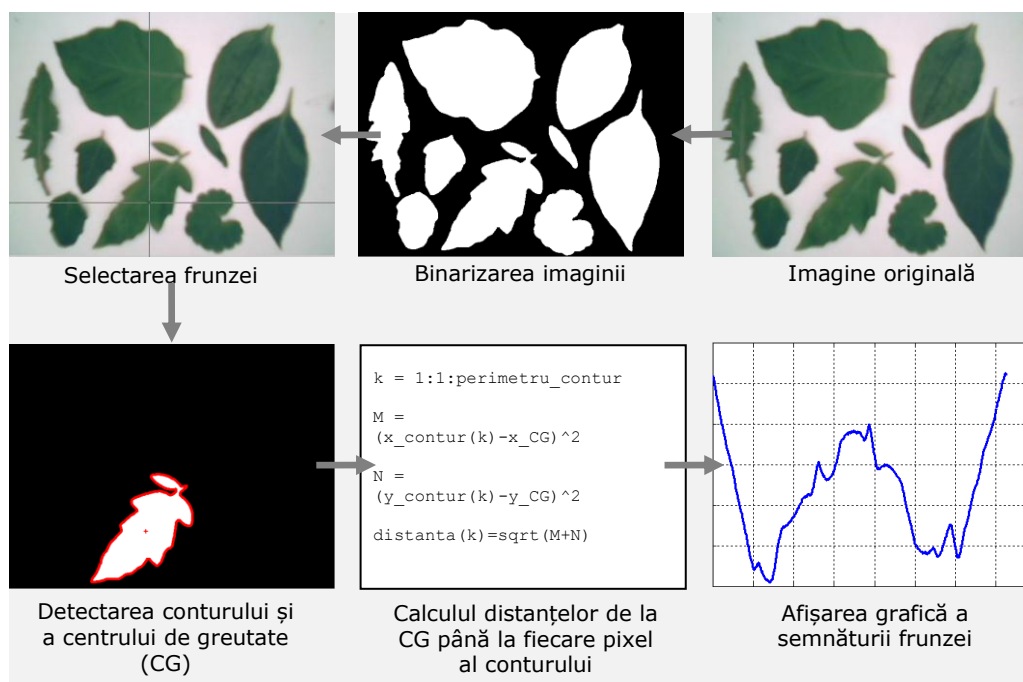


Fig. 5. 9. Etapele de determinare a semnăturii frunzei

Fig. 5. 9 prezintă succesiunea operațiilor efectuate pentru determinarea semnăturii unei frunze. Considerând  $k$  ca fiind un pixel oarecare de pe conturul imaginii frunzei, distanța (exprimată în pixel) de la CG până la pixelul  $k$  este determinată prin relația:

**58** Contribuții la recunoașterea speciilor de plante pe baza formelor frunzelor - 5

$$\begin{aligned} \text{distanța}(k) &= \\ &= \sqrt{(x_{\text{contur}}(k) - x_{\text{CG}})^2 + (y_{\text{contur}}(k) - y_{\text{CG}})^2} \end{aligned} \quad (5. 2)$$

În Fig. 5. 10, Fig. 5. 11 și Fig. 5. 12 sunt prezentate semnăturile corespunzătoare frunzelor de roșie, ardei respectiv vânăță din Fig. 5. 6. Prin analiza semnăturilor, plantele pot fi discriminate pe bază de amplitudine, rapoarte dimensionale, puncte de inflexiune, minime locale și maxime locale. Comparând graficele din Fig. 5. 10, Fig. 5. 11 și Fig. 5. 12, se poate observa că semnătura frunzei de ardei se deosebește de semnăturile frunzelor de roșie și vânăță prin numărul punctelor de inflexiune (Tab. 5. 4).

Număr puncte de inflexiune	Roșie		Ardei		Vânăță	
	Min	Max	Min	Max	Min	Max
	12	17	4	6	13	16

Tab. 5. 4. Numărul punctelor de inflexiune corespunzătoare semnăturilor frunzelor de roșie, ardei și vânăță

Din primele experimente a rezultat că un astfel de concept face posibilă diferențierea plantelor atunci când conturul plantei este complet vizibil, și dacă imaginile nu sunt prelevate din perspectivă.

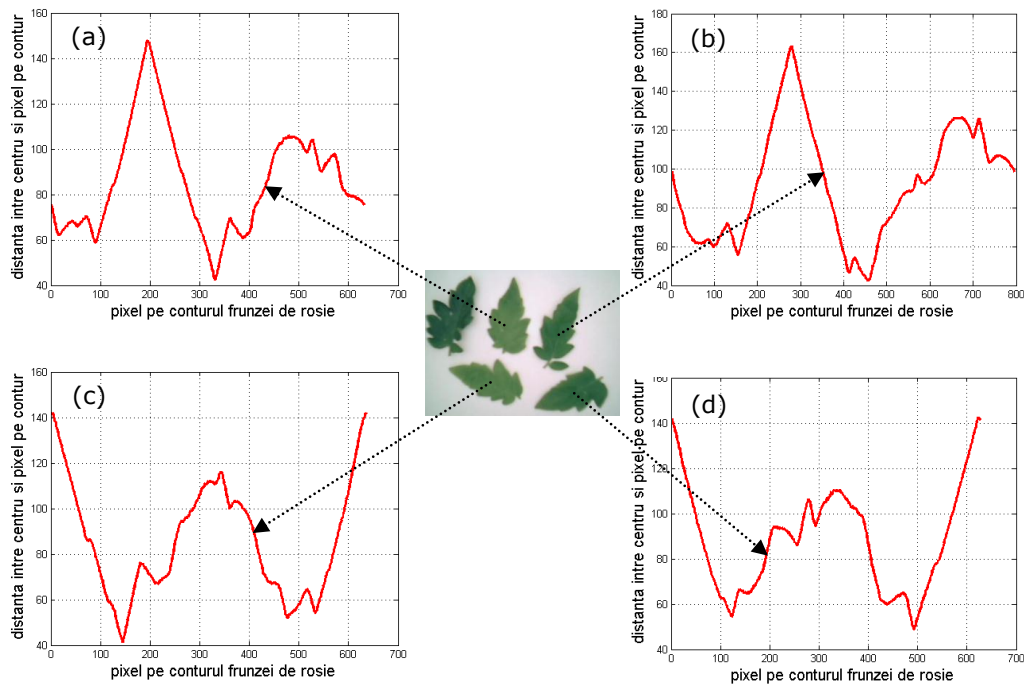


Fig. 5. 10. Semnături specifice conturului frunzelor de roșie

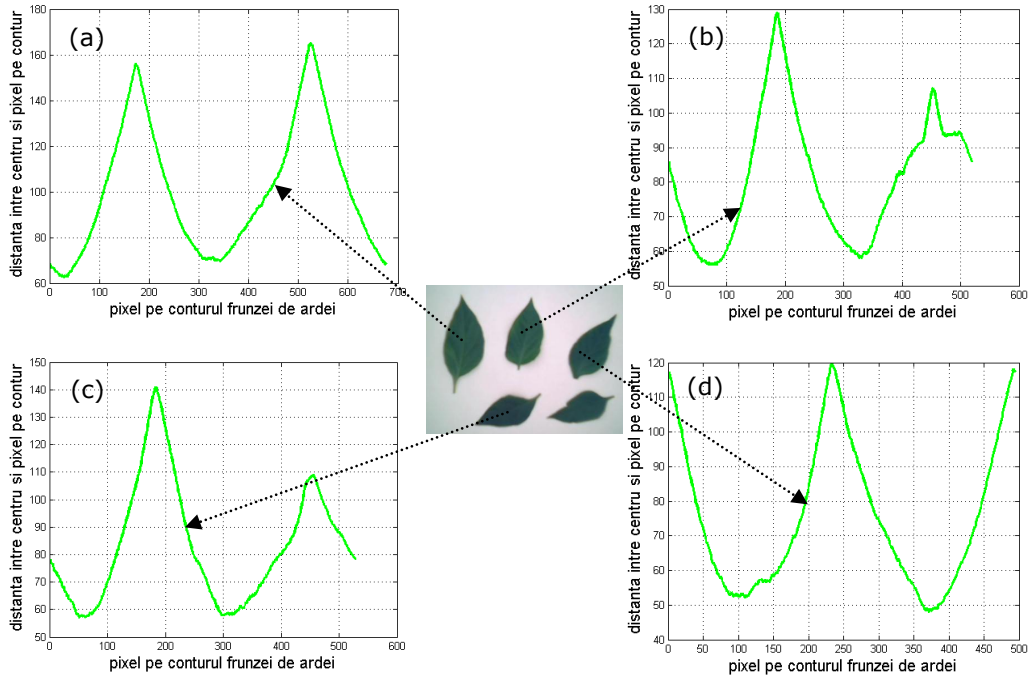


Fig. 5. 11. Semnături specifice conturului frunzelor de ardei

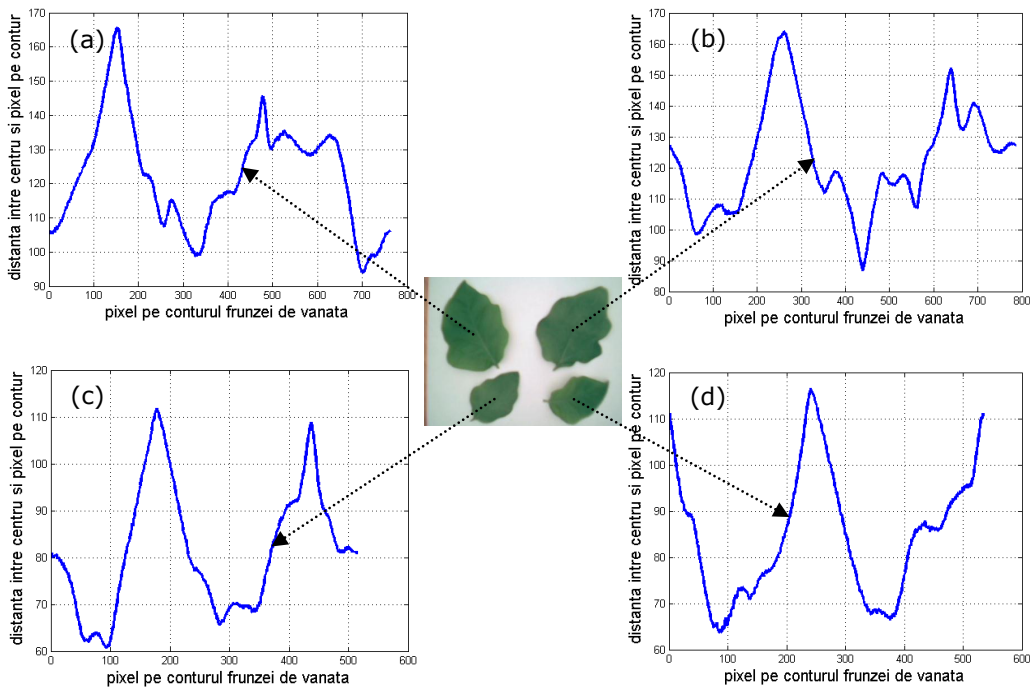


Fig. 5. 12. Semnături specifice conturului frunzelor de vânăță

### 5.4 Experiment 3 - Recunoașterea formelor frunzelor

Funcția *regionprops*, accesibilă în MATLAB, oferă posibilitatea de determinare rapidă a unor caracteristici geometrice corespunzătoare obiectelor din imaginea binară. Funcția principală *regionprops* înglobează un set de funcții care returnează valorile unor parametrii geometrici. Pentru recunoașterea frunzelor au fost utilizate combinații ale următoarelor funcții: 'Area', 'Perimeter', 'Centroid', 'ConvexHull', 'MajorAxisLength', 'MinorAxisLength', 'Eccentricity', 'Solidity'. Pentru a determina geometria frunzelor, au fost utilizați parametrii F1, F2, F3, F4 și F5.

$$F1 = \frac{4 * pi * 'Area'}{'Perimeter'^2} \quad (5. 3)$$

$$F2 = \frac{'Perimeter'}{perimetru\_ 'ConvexHull'} \quad (5. 4)$$

$$F3 = \frac{'MajorAxisLength'}{'MinorAxisLength'} \quad (5. 5)$$

$$F4 = 'Eccentricity' \quad (5. 6)$$

$$F5 = 'Solidity' \quad (5. 7)$$

Lansarea aplicației *DETECTARE\_PLANTE* se realizează în mod clasic prin apelarea programului *DETECTARE\_PLANTE.exe*, dezvoltat în mediul MATLAB. Pentru facilitarea modului de lucru, programul a fost completat interfețe grafice corespunzătoare (anexa B2). Într-o fereastră a interfeței grafice este inserată imaginea supusă analizei. Interfața grafică este prevăzută cu butoane de comandă a căror denumire este asociată speciei de plantă. În cazul plantelor de cultură, activarea unuia dintre butoane are ca efect marcarea în imagine a frunzei recunoscute printr-un contur de culoare galbenă și afișarea denumirii speciei de plantă (Fig. 5. 13, Fig. 5. 14 și Fig. 5. 15). Prin activarea butonului denumit *FRUNZA DE BURUJANA* frunzele recunoscute ca fiind buruieni sunt marcate printr-un contur de culoare roșie (Fig. 5. 16). Corectitudinea recunoașterii este confirmată de către operator.

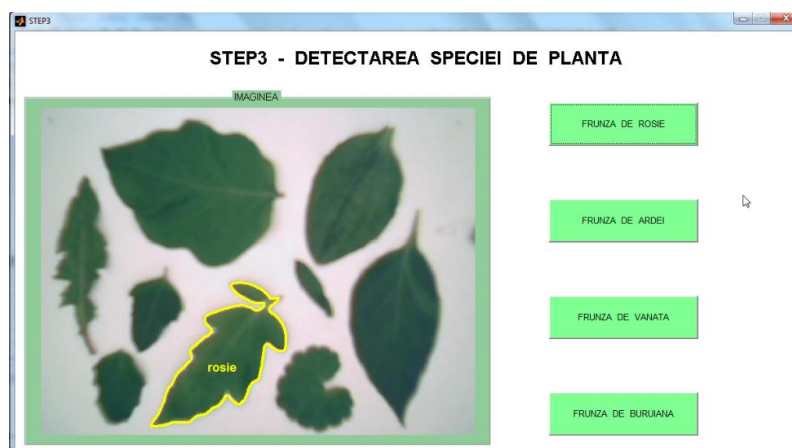


Fig. 5. 13. Recunoașterea frunzei de roșie

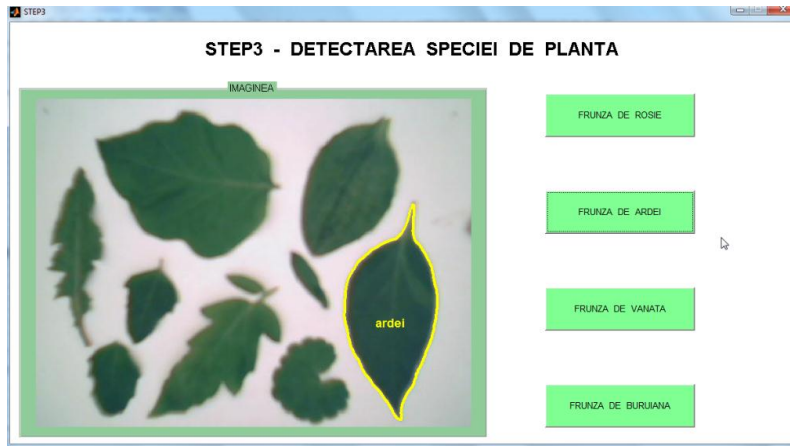


Fig. 5. 14. Recunoașterea frunzei de ardei

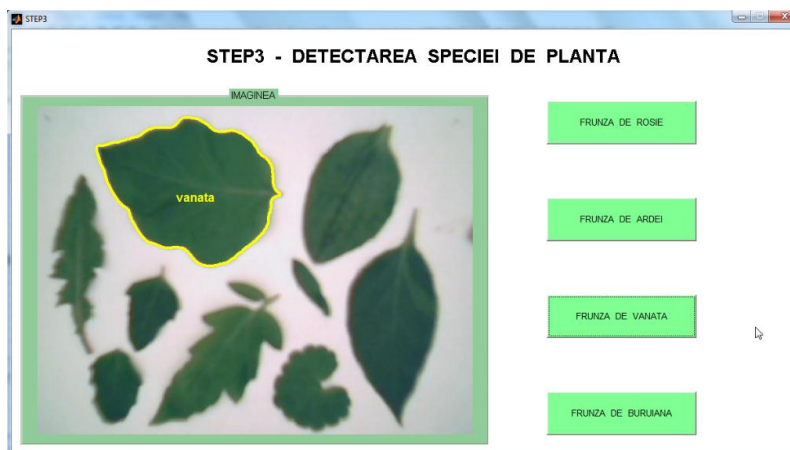


Fig. 5. 15. Recunoașterea frunzei de vânătă

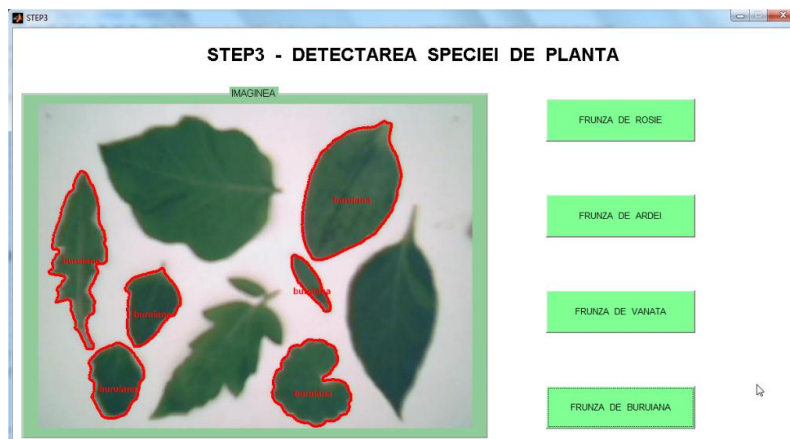


Fig. 5. 16. Detectarea buruienilor

### Concluzii desprinse din experimentul 3:

Prin acest procedeu, în aproximativ 80% din cazuri, frunzele plantelor pot fi identificate corect.

### 5.5 Selectarea frunzelor în cadrul unei aplicații robotizate

Utilizarea unui sistem robot în cadrul unei aplicații de recunoaștere a frunzelor de plantă a constituit tema acestui experiment. Experimentul este dezvoltat utilizând robotul ER V+ (SCORBOT din dotarea laboratorului CIM al Departamentului de Mecatronică). O cameră web (tip A4TECH) a fost montată pe dispozitivul de ghidare al robotului ER V+ în apropierea efectorului final având ca scop achiziționarea de imagini pentru un set de frunze aparținând unor specii diferite de plante (Fig. 5. 17). Informațiile de la camera video sunt transferate prin port USB spre unitatea centrală a unui calculator pe care rulează o aplicație de procesare a imaginilor. Informațiile de la calculator spre controlerul robotului sunt transmise prin port serial RS232.

Sarcina robotului ER V+ a constat în indicarea frunzei recunoscute cu un știft atașat efectorului final. Atingerea scopului urmărit a presupus prelucrarea imaginii achiziționate, extragerea unor parametri de formă și poziție, și comanda robotului în baza acestor date.

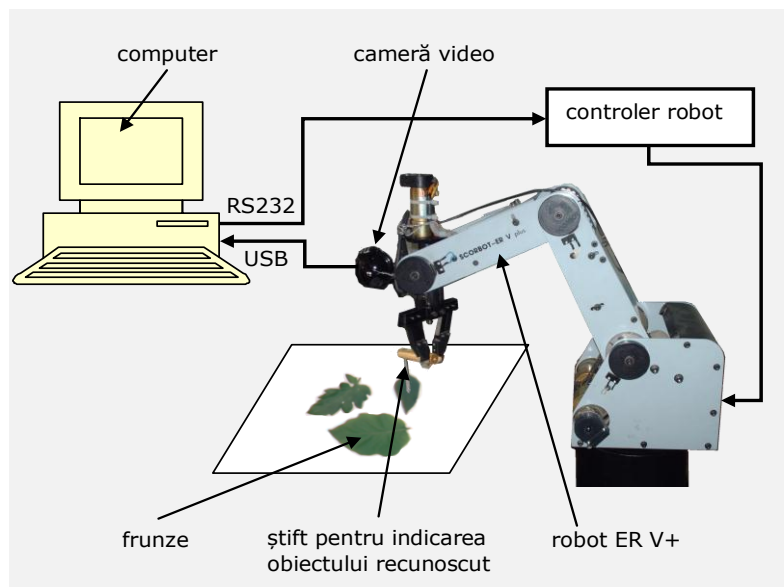


Fig. 5. 17. Schema standului experimental

Elementelor componente ale sistemului le sunt atașate sistemele de referință  $R$ ,  $S$  și  $C$ ; unde  $R$  este sistemul de referință atașat bazei robotului,  $S$  este sistemul de referință al scenei,  $C$  este sistemul de referință al camerei și  $F$  este factor de scalare. Parametrul  $F$ , exprimat în [mm/pixel], este utilizat pentru a face conversia din pixel în mm. Sistemele de referință  $R$ ,  $S$ ,  $C$  sunt prezentate în Fig. 5. 18.

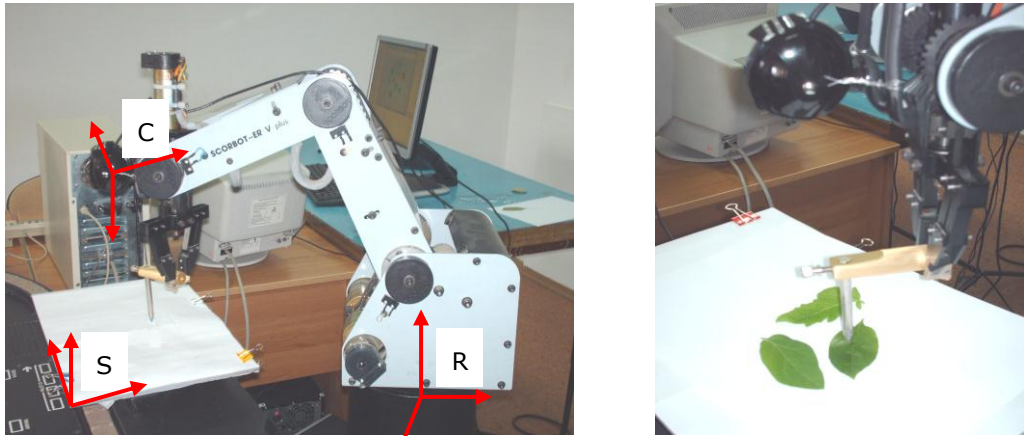


Fig. 5. 18. Aplicație robotizată de selectare a frunzelor

Funcțiile de comandă ale robotului SCORBOT ER V+ (prin programul MATLAB) sunt explicitate în Tab. 5. 5.

Funcția MATLAB	Explicarea funcției
<code>function</code> initialisationRobot (port serial)	Activează portul serial COM1.
<code>function</code> commandRobot (robot, comanda)	Returnează răspunsul instantaneu al robotului.
<code>function</code> movementRobot (robot, Cartesian, X, Y)	Declanșează mișcări ale robotului în coordonate carteziane conform cu transformarea (5. 8).

Tab. 5. 5. Explicarea funcțiilor utilizate la comanda robotului [Paléologue, 2009], [Grigorescu, 2010]

Imaginile prelevate de la sistemul senzorial optic sunt importate în programul MATLAB pentru a fi procesate digital. Speciile de plante sunt recunoscute în imagine pe baza metodelor prezentate în experimentele 2 și 3. Prin procesarea imaginilor sunt determinate coordonatele centrelor de greutate ale obiectelor din imagine. Coordonatele  $(x_C, y_C)$  centrului de greutate al obiectului detectat în imagine, transformate cu funcția *imageModif*, vor fi utilizate în programarea/comanda robotului în cadrul aplicației de indicare a frunzei recunoscute. Este necesară o transformare a coordonatelor din sistemul referință  $C$  în sistemul de referință  $R$ , atașat bazei robotului (Fig. 5. 18). Coordonatele  $(x_R, y_R)$  în sistemul de coordonate al robotului sunt determinate prin transformarea:

$$\underline{R}_{rP} = \underline{R}_{TS} \cdot \left[ \left( \underline{S}_{TC} \cdot \underline{C}_{rP} \right) \cdot F \right] \quad [\text{Grigorescu, 2010}] \quad (5. 8)$$

Înainte de lansarea aplicației, a fost efectuată o calibrare off-line a poziției și orientării camerei raportată la sistemul de referință  $R$  atașat bazei robotului.



## 5.6 Programe de recunoaștere a frunzelor

**Programul de învățare a formelor.** Acest program este utilizat pentru determinarea intervalelor în care se situează valorile parametrilor  $F_1, F_2, F_3, F_4$  și  $F_5$ . Printr-o analiză statistică sunt determinate intervalele parametrilor  $F_n$ , unde  $n = \{1, 2, 3, 4, 5\}$ . Valorile  $F_n$  sunt cuprinse în intervalul  $[F_n\_min; F_n\_max]$ .

$$F_n\_min < F_n < F_n\_max \quad (5.9)$$

Operatorul selectează, prin intermediul interfeței grafice, în mod succesiv câte un obiect din imaginea afișată. Fiecărui obiect  $i$  se vor asocia în acest mod parametrii  $F_n$  calculați. În anexa B2 este prezentată ordinograma programului și câteva secvențe din programul MATLAB. Fig. 5. 19 prezintă una din interfețele grafice ale programului.

Procedura de învățare a unei forme trebuie aplicată unui număr mare de obiecte aparținând unei clase. Rezultatele sunt prelucrate statistic și definesc inegalitățile (5. 9). Rezultă în acest mod o bază de date pe clase de frunze.

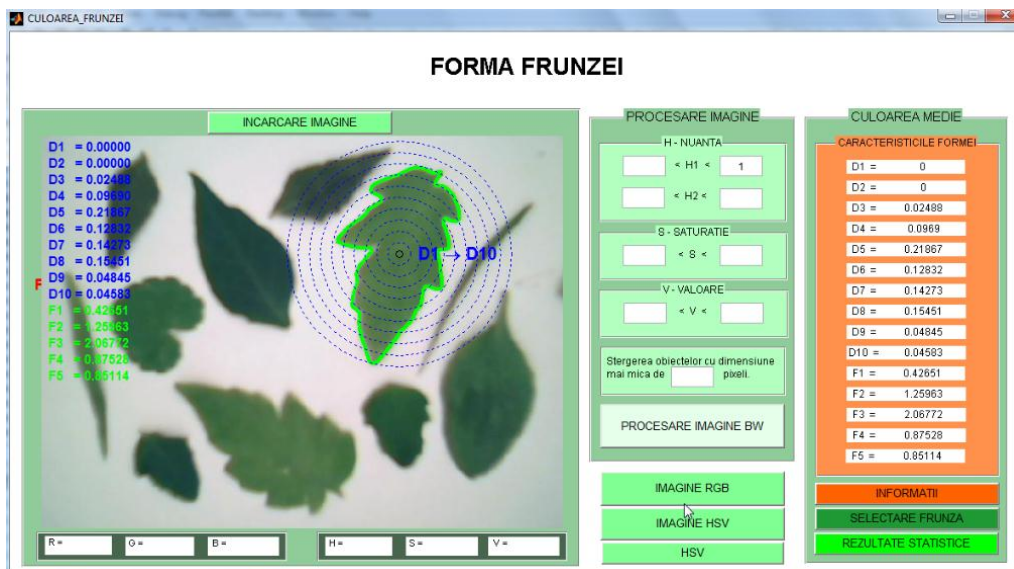
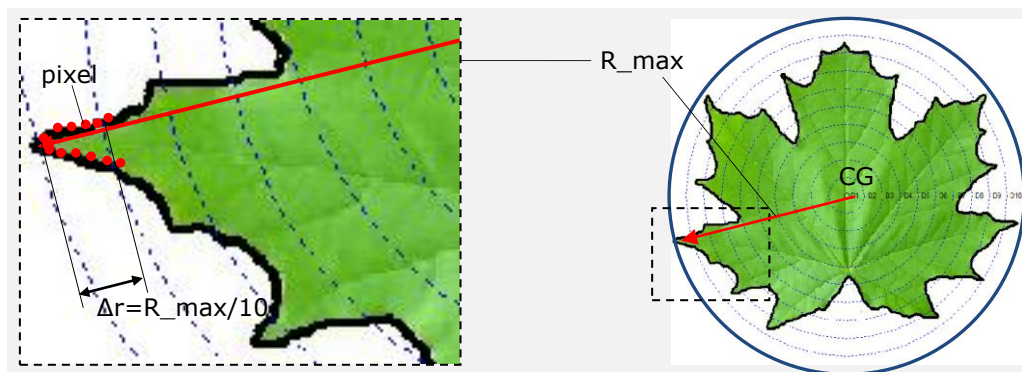


Fig. 5. 19. Interfața grafică a programului de învățare a formelor

Programul de învățare a formelor conține și varianta calculului unui parametru  $D_n$  care să caracterizeze obiectul din imagine (Fig. 5. 20). În acest concept se trasează un cerc virtual al cărui centru coincide cu centrul de greutate al obiectului ( $CG$ ) din imagine. Raza maximă a cercului ( $R\_max$ ) este egală cu distanța de la  $CG$  până la pixelul cel mai îndepărtat de pe contur. Urmează a fi trasate un număr suplimentar de cercuri virtuale concentrice cu cel anterior. În cadrul programului este determinat numărul pixelilor  $N\_pix\_n$  aflați în intervalul  $n$ . Perimetrul conturului obiectului selectat este notat cu  $P\_ob$ . Parametrul  $D_n$  este calculat prin relația:

$$D_n = \frac{N\_pix\_n}{P\_ob} \quad (5.10)$$



Fig. 5. 20. Procedul de determinare a parametrilor  $D_n$  ( $D_1 \div D_{10}$ )

Parametrii  $D_n$  vor fi asociați obiectului achiziționat și vor constitui un parametru suplimentar pentru recunoașterea frunzelor (Fig. 5. 20).

**Programul de recunoaștere a frunzelor.** Programul conține o interfață grafică pentru setarea intervalelor parametrilor  $F_1 \div F_5$ . Au fost dezvoltate module pentru recunoașterea frunzelor de roșie, ardei și vânăță.

Frunzele necunoscute din imagine sunt analizați în baza parametrilor specificați anterior. Valorile determinate prin calcul sunt comparate cu cele din baza de date asociată. În funcție de modul de încadrare în intervalele (5. 9), programul indică apartenența frunzei la o anumită clasă a plantelor de cultură sau clasă buruiiană.

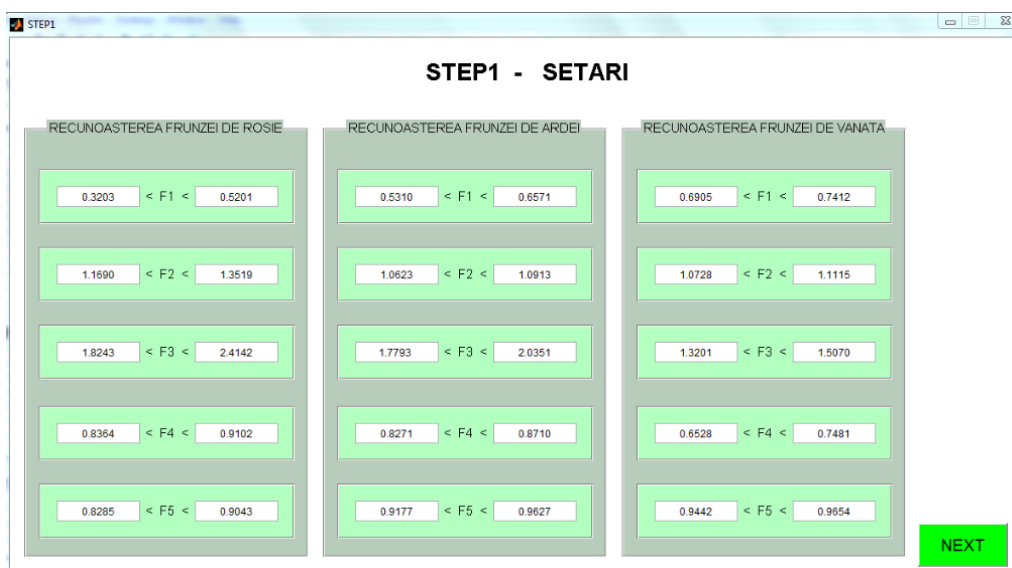
Fig. 5. 21. Setarea parametrilor  $F_1, F_2, F_3, F_4, F_5$  corespunzător frunzelor de roșie, ardei, respectiv vânăță



Fig. 5. 22. Procesarea imaginii

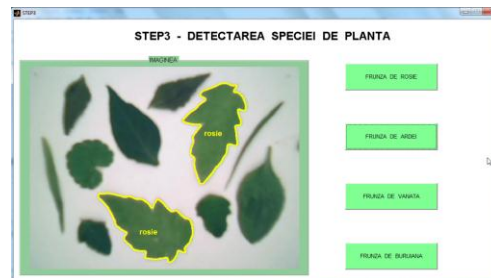


Fig. 5. 23. Recunoașterea frunzelor

În anexa B3 este prezentată ordinograma programului și câteva secvențe din programul MATLAB. Fig. 5. 21, Fig. 5. 22 și Fig. 5. 23 prezintă interfețele grafice ale programului.

## 5.7 Concluzii

Experimentul 3 demonstrează faptul că frunzele plantelor pot fi recunoscute pe baza unor proprietăți geometrice simple. În experimentele efectuate, procentul de recunoaștere corectă a fost de aproximativ 80%. Acesta este un rezultat bun având în vedere simplitatea algoritmilor utilizați. Cercetările urmează să fie extinse pentru recunoașterea frunzelor văzute din perspectivă, și a frunzelor parțial vizibile în imagine.

Aplicațiile robotizate demonstrează faptul că acest tip de aplicații ar putea fi utilizate în scopuri agricole. Orientarea camerei amplasată pe un braț robotizat poate fi modificată prin intermediul unui mecanism adecvat. Acest lucru permite prelevarea unei imagini optime, în care întreaga frunză a plantei poate fi vizibilă în imaginea camerei. O direcție viitoare de cercetare poate avea în vedere dezvoltarea tehnicilor de „visual servoing” în scopul orientării și deplasării cu precizie a camerei video, pentru o mai bună achiziționare de imagini.

## 6 Localizarea obiectelor în spațiul tridimensional

### 6.1 Introducere

Pentru localizarea optică a obiectelor din spațiul tridimensional, este necesar să fie cunoscută distanța de la focarul camerei până la obiectul din scenă. În acest capitol sunt prezentate două procedee de localizare a obiectului în raport cu sistemul de referință al camerei video. Primul procedeu abordează modul în care poate fi extrasă valoarea distanței dintr-o imagine prelevată de la o cameră monoculară, utilizând ca și obiect de lucru un cerc cu diametrul cunoscut. Cel de-al doilea procedeu abordează determinarea valorilor distanțelor prin prelevare de informații de la un sistem stereo binocular.

### 6.2 Măsurarea distanței prin utilizarea unui cerc

Prin acest procedeu, distanța este estimată dintr-o singură imagine. Pentru determinarea distanței este utilizat un cerc cu diametrul cunoscut. Evident, se poate pune întrebarea: "De ce trebuie utilizat un cerc?" Răspuns: „Datorită particularităților geometrice ale cercului în reprezentările vederilor din perspectivă.” Proiecția cercului pe planul imaginii este, de cele mai multe ori, o elipsă. Algoritmii de calcul au fost dezvoltați pornind de la observația că lungimea axei mari a elipsei (proiecției cercului) rămâne constantă pentru toate pozițiile și orientările, dacă centrul cercului ( $P_{world}$ ) este plasat în același plan care este perpendicular pe axa optică a camerei. Fig. 6. 1 ilustrează exemple de proiecții ale cercurilor pe planul imagine.

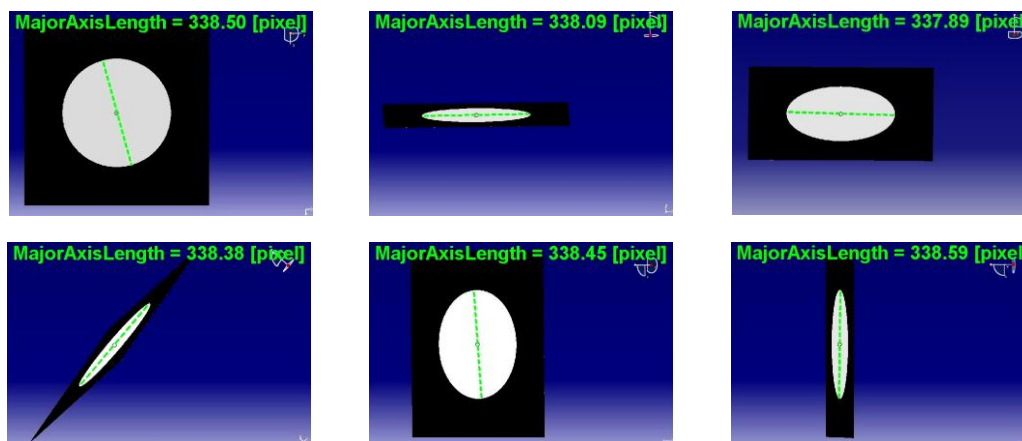


Fig. 6. 1. Exemple de proiecții ale cercului. Lungimea axei mari a elipsei rămâne constantă, dacă centrul cercului este deplasat în același plan, care este perpendicular pe axa optică a camerei video.

Lungimile axelor mari ale elipselor au fost determinate prin procesarea imaginilor în mediul MATLAB. Se poate trage concluzia că lungimea axei mari rămâne aproximativ constantă. Aceasta înseamnă că cercurile pot fi utilizate pentru a calcula factorii de scalare. Toate cercurile care au centrele în același plan, au factori de scalare identici. Centrele imaginilor amplasate în diferite plane, perpendiculare pe axa optică a camerei, au factori de scalare diferiți. Factorii de scalare sunt necesari pentru a face conversia din coordonatele 2D ale imaginii în coordonate 3D ale sistemului global. Fig. 6. 2 este o reprezentare geometrică simplificată, în care:

- $d\_cerc\_world$  reprezintă lungimea diametrului cercului [mm];
- $d\_elipsa\_im$  reprezintă lungimea [pixel] axei mari a elipsei corespunzătoare proiecției cercului în imaginii;
- $M\_im$  reprezintă punctul principal;
- $P\_im$  coordonatele punctului din imagine, raportate la sistemul de referință al imaginii ( $im$ );
- $P\_im\_cam$  reprezintă coordonatele punctului din imagine, raportat la sistemul de referință al camerei ( $cam$ );
- $s\_cam$  reprezintă factor de scalare [mm/pixel];
- $P\_world$  reprezintă un punct din scenă raportat la sistemul de referință al camerei;
- $f$  reprezintă distanța focală [pixel];
- $d\_FP\_world$  reprezintă distanța de la punctul focal până la centrul cercului [mm].

Algoritmii sunt construiți pe baza proporționalităților între mărimile obiectelor în imagine și mărimile reale ale obiectelor. Factorul de scalare, notat cu  $s\_cam$ , reprezintă raportul dintre lungimea diametrului cercului [mm] și lungimea axei mari a elipsei [pixel].

$$s\_cam = \frac{d\_cerc\_world}{d\_elipsa\_im} \quad [\text{mm/pixel}] \quad (6.1)$$

$$x\_P\_im\_cam = x\_P\_im - x\_M\_im \quad [\text{pixel}] \quad (6.2)$$

$$y\_P\_im\_cam = y\_P\_im - y\_M\_im \quad [\text{pixel}] \quad (6.3)$$

$$x\_P\_world = s\_cam * x\_P\_im\_cam \quad [\text{mm}] \quad (6.4)$$

$$y\_P\_world = s\_cam * y\_P\_im\_cam \quad [\text{mm}] \quad (6.5)$$

$$z\_P\_world = s\_cam * f \quad [\text{mm}] \quad (6.6)$$

$$d\_FP\_world = \sqrt{x\_P\_world^2 + y\_P\_world^2 + z\_P\_world^2} \quad [\text{mm}] \quad (6.7)$$

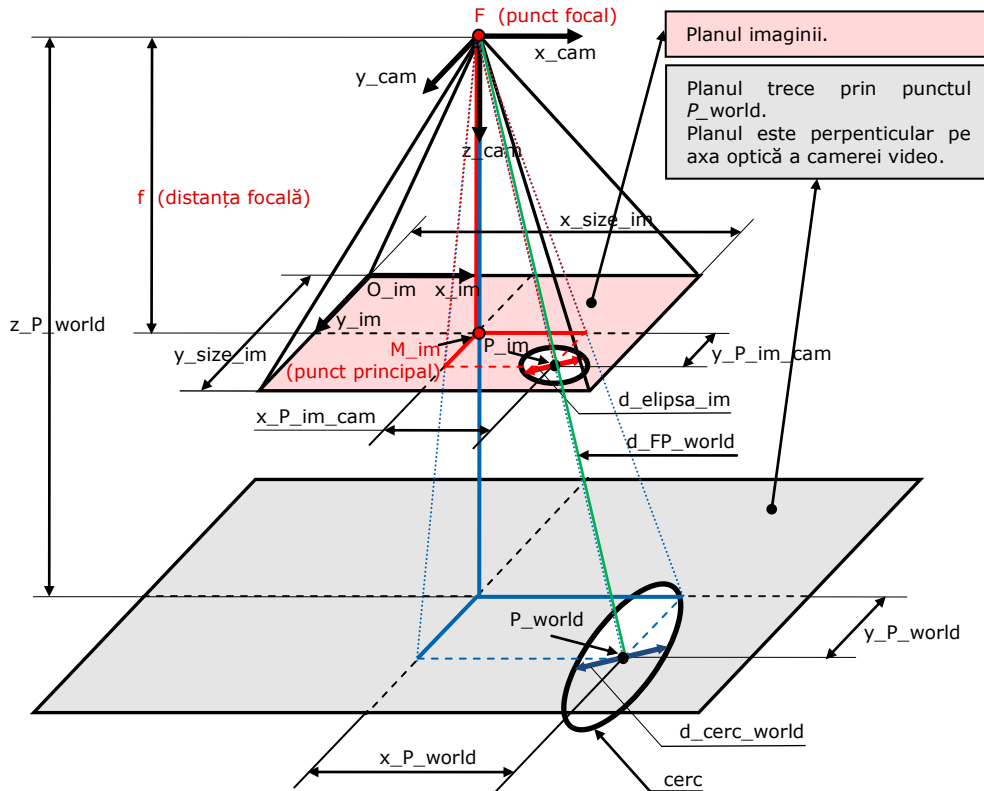


Fig. 6. 2. Reprezentare geometrică simplificată a relațiilor imagine-scenă

Relațiile (6. 1) ÷ (6. 7) sunt prezentate într-o formă simplificată, bazată pe modelul camerei *pinhole*. În realitate, algoritmi sunt mai complecși, din cauză că includ mai mulți parametri intrinseci. În majoritatea cazurilor există două distanțe focale; prima distanță focală corespunde câmpului vizual orizontal, iar a doua distanță focală corespunde câmpului vizual vertical. Pentru a înlătura efectele distorsiunilor cauzate de lentile, trebuie să fie făcută o corecție a imaginii. Valorile parametrilor intrinseci rezultă din procesul de calibrare intrinsecă a camerelor video.

### 6.3 Măsurarea distanțelor utilizând vederea stereo

Această tehnică permite determinarea valorilor distanțelor pe baza informațiilor obținute de la un sistem stereo binocular [Ciocarlie, 2011], [Dodoc, 1989]. Această tehnică poate fi aplicată doar în cazul în care sunt cunoscute valorile parametrilor intrinseci și extrinseci ai camerelor video. Calibrarea intrinsecă și extrinsecă a camerelor video este primul pas care trebuie efectuat pentru utilizarea acestui procedeu de determinare a distanțelor. Prin procesul de calibrare extrinsecă sunt determinate valorile matricei de rotație ( $M_{\text{Rotatie}}$ ), exprimat prin relația (6. 19), și a vectorului de translație ( $M_{\text{Translatie}}$ ), exprimat prin relația (6. 20). Presupunând faptul că distanța focală corespunzătoare câmpului vizual orizontal este egală cu distanța focală corespunzătoare câmpului vizual vertical, factorii de

scalare  $s_{cam1}$  (raportat la camera 1) și  $s_{cam2}$  (raportat la camera 2) pot fi determinați din relația (6. 18). Relația (6. 31) este utilizată pentru a calcula distanța de la punctul focal  $F1$  al camerei 1 până la punctul  $P_{world}$ . Relația (6. 33) este utilizată pentru a calcula distanța de la punctul focal  $F2$  al camerei 2 până la punctul  $P_{world}$ . Parametrii pot fi vizualizați în reprezentarea geometrică din Fig. 6. 3.

Se pornește de la ipoteza că valorile următorilor parametri de intrare sunt cunoscute:

- Distanțele focale ale camerelor 1 și 2. Notăția:  $f1, f2$
- Dimensiunea imaginii 1 pe axele x și y. Notăția:  $x\_size\_im1, y\_size\_im1$
- Dimensiunea imaginii 2 pe axele x și y. Notăția:  $x\_size\_im2, y\_size\_im2$
- Coordonatele proiecției punctului  $P_{world}$  în imaginea 1. Notăția:  $P_{im1}$
- Coordonatele proiecției punctului  $P_{world}$  în imaginea 2. Notăția:  $P_{im2}$
- Matricea de rotație și vectorul de translație utilizate pentru a realiza transformarea din sistemul de referință 1 al camerei 1 în sistemul de referință al camerei 2. Notăția:  $M\_Rotatie, M\_Translatie$

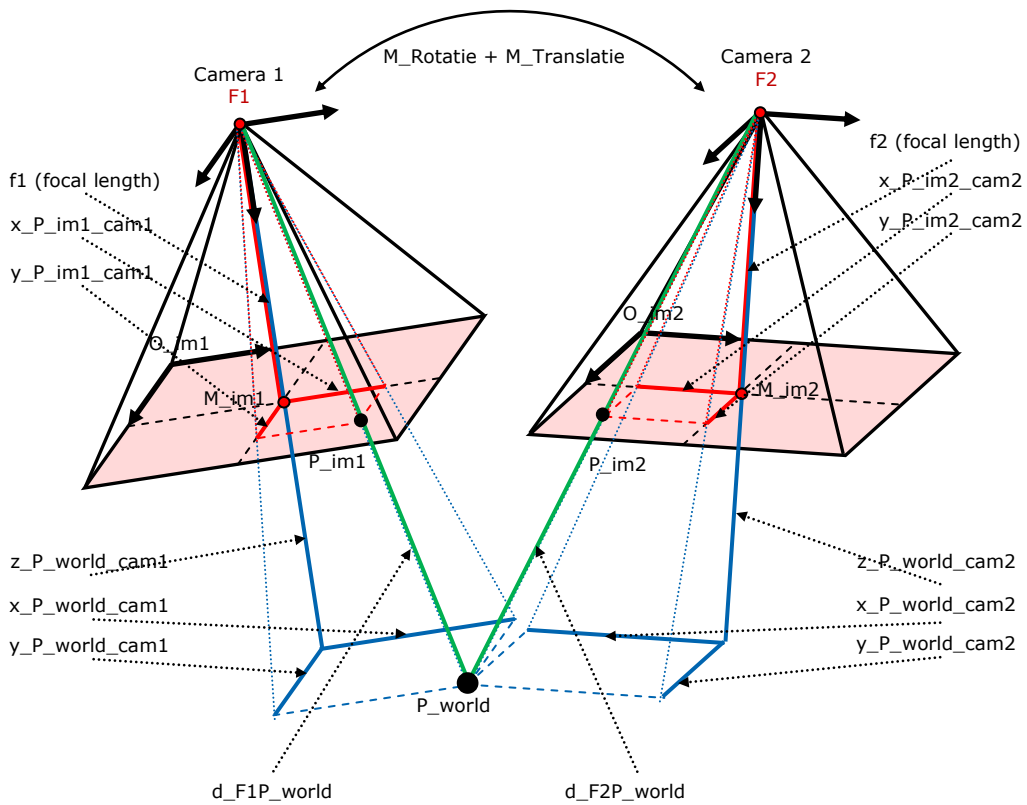


Fig. 6. 3. Reprezentare geometrică simplificată a unui sistem stereo binocular

### Modelul matematic

**Etapa1-camera1:** Coordonatele proiecției punctului  $P_{world}$  în imaginea 1 sunt specificate în relațiile (6. 8) și (6. 9).

$$P_{im1} = [x_{P_{im1}} \ y_{P_{im1}} \ z_{P_{im1}}]^T \quad (6. 8)$$

$$z_{P_{im1}} = 0 \quad (6. 9)$$

**Etapa1-camera2:** Coordonatele proiecției punctului  $P_{world}$  în imaginea 2 sunt specificate în relațiile (6. 10), și (6. 11).

$$P_{im2} = [x_{P_{im2}} \ y_{P_{im2}} \ z_{P_{im2}}]^T \quad (6. 10)$$

$$z_{P_{im2}} = 0 \quad (6. 11)$$

**Etapa2-camera1:** Inițial coordonatele proiecțiilor punctului  $P$  în planul imagine este raportat la sistemul de referință al  $im1$ . Etapa2-camera1 constă în transformarea coordonatelor punctului  $P_{im1}$  raportate la sistemul de referință  $im1$ , în coordonate raportate la sistemul de referință  $cam1$ . În acest scop sunt utilizate relațiile (6. 12), (6. 13) și (6. 14).

$$x_{P_{im1}_{cam1}} = x_{P_{im1}} - x_{M_{im1}} \quad (6. 12)$$

$$y_{P_{im1}_{cam1}} = y_{P_{im1}} - y_{M_{im1}} \quad (6. 13)$$

$$z_{P_{im1}_{cam1}} = (f1_x + f1_y) / 2 \quad (6. 14)$$

**Etapa2-camera2:** Inițial coordonatele proiecțiilor punctului  $P$  în planul imagine este raportat la sistemul de referință al  $im2$ . Etapa2\_camera2 constă în transformarea coordonatelor punctului  $P_{im2}$  raportate la sistemul de referință  $im2$ , în coordonate raportate la sistemul de referință  $cam2$ . În acest scop sunt utilizate relațiile (6. 15), (6. 16) și (6. 17).

$$x_{P_{im2}_{cam2}} = x_{P_{im2}} - x_{M_{im2}} \quad (6. 15)$$

$$y_{P_{im2}_{cam2}} = y_{P_{im2}} - y_{M_{im2}} \quad (6. 16)$$

$$z_{P_{im2}_{cam2}} = (f2_x + f2_y) / 2 \quad (6. 17)$$

**Etapa3:** Conversia din pixel în milimetru este realizată prin utilizarea unor factori de scalare. Factorii de scalare corespunzători punctului  $P$  în imaginea 1 respectiv în imaginea 2, pot fi determinați pe baza relațiilor (6. 18), (6. 19) și (6. 20).

$$\begin{bmatrix} x_{P_{im2}_{cam2}} * s_{cam2} \\ y_{P_{im2}_{cam2}} * s_{cam2} \\ z_{P_{im2}_{cam2}} * s_{cam2} \end{bmatrix} = \begin{bmatrix} r11 & r12 & r13 \\ r21 & r22 & r23 \\ r31 & r32 & r33 \end{bmatrix} * \begin{bmatrix} x_{P_{im1}_{cam1}} * s_{cam1} \\ y_{P_{im1}_{cam1}} * s_{cam1} \\ z_{P_{im1}_{cam1}} * s_{cam1} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad (6. 18)$$

$$M_{\_Rotatie} = \begin{bmatrix} r11 & r12 & r13 \\ r21 & r22 & r23 \\ r31 & r32 & r33 \end{bmatrix} \quad (6. 19)$$

$$M_{\_Translatie} = \begin{bmatrix} t\_x \\ t\_y \\ t\_z \end{bmatrix} \quad (6. 20)$$

Se presupune că parametrii extrinseci ai camerei video sunt cunoscuți. Pornind de la această ipoteză, urmează a fi determinați factorii de scalare  $s\_cam1$  și  $s\_cam2$ . Valorile factorilor de scalare sunt determinați din sistemul de ecuații (6. 21).

$$\begin{cases} s\_cam2 * x\_P\_im2\_cam2 = s\_cam1 * \begin{pmatrix} r11 * x\_P\_im1\_cam1 + \dots \\ \dots + r12 * y\_P\_im1\_cam1 + \dots \\ \dots + r13 * z\_P\_im1\_cam1 \end{pmatrix} + t\_x \\ s\_cam2 * y\_P\_im2\_cam2 = s\_cam1 * \begin{pmatrix} r21 * x\_P\_im1\_cam1 + \dots \\ \dots + r22 * y\_P\_im1\_cam1 + \dots \\ \dots + r23 * z\_P\_im1\_cam1 \end{pmatrix} + t\_y \end{cases} \quad (6. 21)$$

Pentru a simplifica calculele, au fost făcute prescurtările indicate în relațiile (6. 22) și (6. 23).

$$v1 = r11 * x\_P\_im1\_cam1 + r12 * y\_P\_im1\_cam1 + r13 * z\_P\_im1\_cam1 \quad (6. 22)$$

$$v2 = r21 * x\_P\_im1\_cam1 + r22 * y\_P\_im1\_cam1 + r23 * z\_P\_im1\_cam1 \quad (6. 23)$$

Din sistemul (6. 21) și relațiile (6. 22) și (6. 23) rezultă sistemul (6. 24).

$$\begin{cases} -s\_cam1 * v1 + s\_cam2 * x\_P\_im2\_cam2 = t\_x \\ -s\_cam1 * v2 + s\_cam2 * y\_P\_im2\_cam2 = t\_y \end{cases} \quad (6. 24)$$

Sistemul de ecuații (6. 24) este rezolvat prin regula lui Cramer.

$$A = \begin{vmatrix} -v1 & x\_P\_im2\_cam2 \\ -v2 & y\_P\_im2\_cam2 \end{vmatrix} \quad (6. 25)$$

$$A1 = \begin{vmatrix} t\_x & x\_P\_im2\_cam2 \\ t\_y & y\_P\_im2\_cam2 \end{vmatrix} \quad (6. 26)$$



$$A2 = \begin{vmatrix} -v1 & t\_x \\ -v2 & t\_y \end{vmatrix} \quad (6.27)$$

$$s\_cam1 = \frac{A1}{A} \quad (6.28)$$

$$s\_cam2 = \frac{A2}{A} \quad (6.29)$$

**Etapa4-camera1:** Etapa constă în determinarea distanței exprimate în milimetri dintre camera1 și punctul  $P\_world$ . Sistemul (6. 30) reprezintă coordonatele punctului  $P\_world$  raportate la sistemul de referință  $cam1$ .

$$\begin{cases} x\_P\_world\_cam1 = s\_cam1 * x\_P\_im1\_cam1 \\ y\_P\_world\_cam1 = s\_cam1 * y\_P\_im1\_cam1 \\ z\_P\_world\_cam1 = s\_cam1 * z\_P\_im1\_cam1 \end{cases} \quad (6.30)$$

Distanța dintre punctul focal  $F1$  și punctul  $P\_world$ , exprimată în milimetri, este determinată prin relația (6. 31).

$$\begin{aligned} d\_F1P\_world = \\ = \sqrt{(x\_P\_world\_cam1)^2 + (y\_P\_world\_cam1)^2 + (z\_P\_world\_cam1)^2} \end{aligned} \quad (6.31)$$

**Etapa4-camera2:** Etapa constă în determinarea distanței exprimate în milimetri dintre camera2 și punctul  $P\_world$ . Sistemul (6. 32) reprezintă coordonatele punctului  $P\_world$  raportate la sistemul de referință  $cam2$ .

$$\begin{cases} x\_P\_world\_cam2 = s\_cam2 * x\_P\_im2\_cam2 \\ y\_P\_world\_cam2 = s\_cam2 * y\_P\_im2\_cam2 \\ z\_P\_world\_cam2 = s\_cam2 * z\_P\_im2\_cam2 \end{cases} \quad (6.32)$$

Distanța dintre punctul focal  $F2$  și punctul  $P\_world$ , exprimată în milimetri, este determinată prin relația (6. 33).

$$\begin{aligned} d\_F2P\_world = \\ = \sqrt{(x\_P\_world\_cam2)^2 + (y\_P\_world\_cam2)^2 + (z\_P\_world\_cam2)^2} \end{aligned} \quad (6.33)$$

Algoritmii (6. 8)÷(6. 33) au fost implementați într-un program de măsurare a distanțelor pe bază de vedere stereo.

## 6.4 Programul de măsurare a distanțelor

În mediul MATLAB a fost dezvoltat un program de măsurare a distanțelor pe bază de vedere stereo. Lansarea programului este realizată printr-un fișier MATLAB denumit *DETERMINAREA\_DISTANTEI* (Fig. 6. 4). Imaginile prelevate de la un sistem stereoscopic format din două camere video monoculare sunt prelucrate în scopul determinării distanțelor de la punctul din scenă până la focarele celor două camere video. Sunt parcurse următoarele etape:

- setarea parametrilor intrinseci și extrinseci ai camerelor video;
- importarea imaginilor în programul MATLAB;
- vizualizarea imaginilor;
- selectarea manuală a punctelor omoloage în cele două imagini.



Fig. 6. 4. Interfața grafică a programului utilizat la măsurarea distanțelor prin vedere stereo

În anexa G este prezentată ordinograma programului MATLAB și câteva secvențe de cod.

## 7 Calibrarea camerelor video

### 7.1 Calibrare intrinsecă

#### 7.1.1 Introducere

În acest capitol este prezentată o metodă simplă de determinare a distanței focale a camerelor video. Determinarea distanței focale poate fi realizată utilizând doar două obiecte de calibrare de formă circulară al căror diametru este cunoscut. Procedeul poartă denumirea de calibrare intrinsecă a camerei.

Ca obiecte de calibrare, sunt utilizate două cercuri cu diametre cunoscute, notate  $d\_cerc1\_world$ , respectiv  $d\_cerc2\_world$ . De asemenea, distanța dintre centrele cercurilor, notată cu  $d\_P1P2\_world$ , este cunoscută. În Fig. 7. 1 este prezentată o schemă a obiectelor de calibrare.

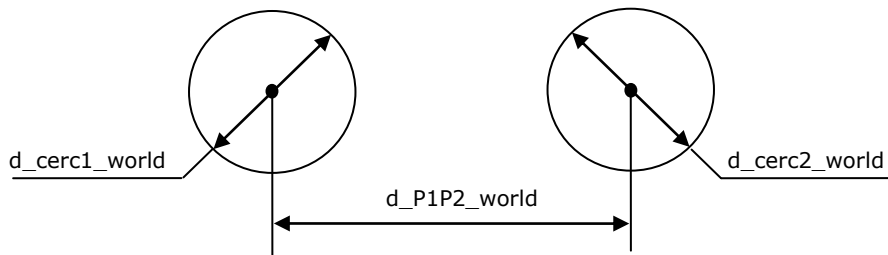


Fig. 7. 1. Obiecte geometrice utilizate pentru calibrarea intrinsecă

#### 7.1.2 Algoritmi matematici

Modelul matematic a fost realizat pe baza modelului geometric prezentat în Fig. 7. 2. Obiectivul este determinarea distanței focale. În Fig. 7. 2, distanța focală este notată cu  $f$  de culoare roșie. Inițial sunt cunoscute: dimensiunile celor două cercuri exprimate în milimetri, distanța dintre centrele celor două cercuri exprimată în milimetri și dimensiunea imaginii exprimată în pixeli. Prin analizarea imaginii celor două cercuri, sunt determinate coordonatele proiecțiilor centrelor cercurilor în imagine. De asemenea, prin procesarea imaginii sunt determinate valorile dimensiunilor axelor mari ale elipselor ce reprezintă proiecțiile cercurilor în imagine. Cunoscând dimensiunile diametrelor reale (exprimat în milimetri) ale cercurilor și dimensiunile axelor mari (exprimate în pixeli) ale elipselor ce reprezintă proiecțiile cercurilor în imagine, pot fi calculați factorii de scalare  $s1$  și  $s2$ . Factorii de scalare sunt utilizați la conversia din pixel în milimetru. Distanța focală este determinată prin relația (7. 14 d).

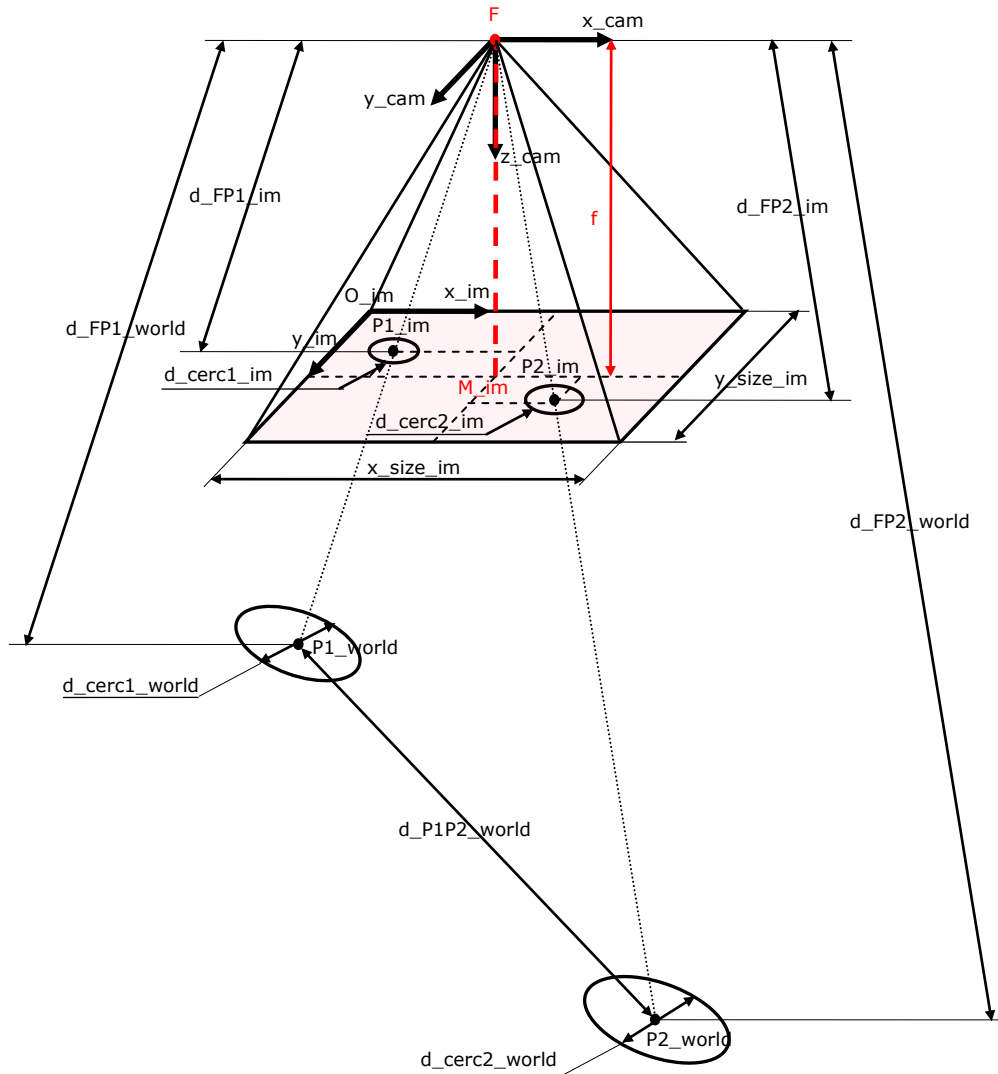


Fig. 7. 2. Modelul geometric pentru determinarea distanței focale

Coordonatele proiecției centrului cercului 1 în imagine sunt  $x_{P1\_im}$  și  $y_{P1\_im}$ . Coordonatele proiecției centrului cercului 2 în imagine sunt  $x_{P2\_im}$  și  $y_{P2\_im}$ . Coordonatele sunt raportate la sistemul de referință al imaginii, notat  $im$ . Originea sistemului de referință  $im$  a fost translatată în focarul camerei. Astfel a rezultat sistemul de referință  $cam$  cu originea în focarul  $F$ . Axa  $z$  a sistemului de referință  $cam$  coincide cu axa optică a camerei. Prin relațiile (7. 1), (7. 2) și (7. 3) sunt determinate coordonatele proiecției centrului cercului 1 în imagine, raportat la sistemul de referință  $cam$ . Prin relațiile (7. 4), (7. 5) și (7. 6) sunt determinate coordonatele proiecției centrului cercului 2 în imagine, raportat la sistemul de referință  $cam$ .

$$x\_P1\_im\_cam = |x\_M\_im - x\_P1\_im| \quad (7.1)$$

$$y\_P1\_im\_cam = |y\_M\_im - y\_P1\_im| \quad (7.2)$$

$$z\_P1\_im\_cam = f \quad (7.3)$$

$$x\_P2\_im\_cam = |x\_M\_im - x\_P2\_im| \quad (7.4)$$

$$y\_P2\_im\_cam = |y\_M\_im - y\_P2\_im| \quad (7.5)$$

$$z\_P2\_im\_cam = f \quad (7.6)$$

Factorii de scalare corespunzători celor două obiecte sunt determinați prin raportul dintre valoarea diametrului real și valoarea axei mari a proiecției cercului pe planul imagine. Valorile factorilor de scalare ale obiectelor 1 și 2, sunt determinate utilizând relațiile (7. 7), respectiv (7. 8).

$$s1 = \frac{d\_cerc1\_world}{d\_cerc1\_im} \quad [\text{mm/pixel}] \quad (7.7)$$

$$s2 = \frac{d\_cerc2\_world}{d\_cerc2\_im} \quad [\text{mm/pixel}] \quad (7.8)$$

Modelul matematic a fost construit pe baza proporționalității scenă/imagine. Factorii de scalare  $s1$  și  $s2$  sunt utilizați pentru a reduce numărul de necunoscute. În relația (7. 9) sunt indicate fracțiile ce pot fi înlocuite cu factorul de scalare  $s1$ .

$$\left\{ \begin{array}{l} s1 = \frac{x\_P1\_world\_cam}{x\_P1\_im\_cam} \\ s1 = \frac{y\_P1\_world\_cam}{y\_P1\_im\_cam} \\ s1 = \frac{z\_P1\_world\_cam}{f} \end{array} \right. \quad (7.9)$$

În relația (7. 10) sunt indicate fracțiile ce pot fi înlocuite cu factorul de scalare  $s2$ .

$$\left\{ \begin{array}{l} s_2 = \frac{x_{P2\_world\_cam}}{x_{P2\_im\_cam}} \\ s_2 = \frac{y_{P2\_world\_cam}}{y_{P2\_im\_cam}} \\ s_2 = \frac{z_{P2\_world\_cam}}{f} \end{array} \right. \quad (7.10)$$

Considerând sistemul de referință al camerei, distanța dintre punctul P1\_world și punctul P2\_world poate fi determinată utilizând relația (7.11).

$$\begin{aligned} d_{P1P2\_world} &= \\ &= \left( (x_{P1\_world\_cam} - x_{P2\_world\_cam})^2 + \right. \\ &\quad \left. + (y_{P1\_world\_cam} - y_{P2\_world\_cam})^2 + \right. \\ &\quad \left. + (z_{P1\_world\_cam} - z_{P2\_world\_cam})^2 \right)^{\frac{1}{2}} \end{aligned} \quad (7.11)$$

Din relațiile (7.9) rezultă:

$$\left\{ \begin{array}{l} x_{P1\_world\_cam} = s_1 * x_{P1\_im\_cam} \\ y_{P1\_world\_cam} = s_1 * y_{P1\_im\_cam} \\ z_{P1\_world\_cam} = s_1 * f \end{array} \right. \quad (7.12)$$

Din relațiile (7.13) rezultă:

$$\left\{ \begin{array}{l} x_{P2\_world\_cam} = s_2 * x_{P2\_im\_cam} \\ y_{P2\_world\_cam} = s_2 * y_{P2\_im\_cam} \\ z_{P2\_world\_cam} = s_2 * f \end{array} \right. \quad (7.13)$$

Din relațiile (7.11), (7.12), (7.13) rezultă ecuația (7.14 a):

$$\begin{aligned}
 d_{P1P2\_world} &= \\
 &= \left( \begin{aligned} &(s1 * x_{P1\_im\_cam} - s2 * x_{P2\_im\_cam})^2 + \\ &+ (s1 * y_{P1\_im\_cam} - s2 * y_{P2\_im\_cam})^2 + \\ &+ (s1 * f - s2 * f)^2 \end{aligned} \right)^{\frac{1}{2}} \quad (7.14 a)
 \end{aligned}$$

Din ecuația (7. 14 a) rezultă ecuația (7. 14 b):

$$\begin{aligned}
 -(s1 * f - s2 * f)^2 &= \\
 &= \left( \begin{aligned} &(s1 * x_{P1\_im\_cam} - s2 * x_{P2\_im\_cam})^2 + \\ &+ (s1 * y_{P1\_im\_cam} - s2 * y_{P2\_im\_cam})^2 - \\ &- d_{P1P2\_world}^2 \end{aligned} \right) \quad (7.14 b)
 \end{aligned}$$

Din relația (7. 14 b) rezultă relația (7. 14 c):

$$f^2 = \frac{\left( \begin{aligned} &(s1 * x_{P1\_im\_cam} - s2 * x_{P2\_im\_cam})^2 + \\ &+ (s1 * y_{P1\_im\_cam} - s2 * y_{P2\_im\_cam})^2 - \\ &- d_{P1P2\_world}^2 \end{aligned} \right)}{(s1 - s2)^2} \quad (7.14 c)$$

Valoarea distanței focale  $f$  este singura necunoscută din relația (7. 14 c). Astfel, valoarea distanței focale poate fi determinată utilizând relația (7. 14 d).

$$f = \left( \frac{\left( \begin{aligned} &(s1 * x_{P1\_im\_cam} - s2 * x_{P2\_im\_cam})^2 + \\ &+ (s1 * y_{P1\_im\_cam} - s2 * y_{P2\_im\_cam})^2 - \\ &- d_{P1P2\_world}^2 \end{aligned} \right)}{(s1 - s2)^2} \right)^{\frac{1}{2}} \quad (7.14 d)$$

### 7.1.3 Descrierea experimentului

În experiment au fost utilizate două obiecte de calibrare de formă circulară. Distanțele dintre centrele celor două cercuri sunt cunoscute. În experimentul realizat, pentru fiecare dintre cercuri dimensiunea diametrului este 176 mm, iar distanța dintre centrele celor două cercuri este 581 mm. Algoritmii de calibrare au fost implementați în mediul de programare MATLAB. Programul MATLAB este recomandat a fi utilizat, datorită faptului că posedă funcții speciale pentru procesarea imaginilor. După procesarea imaginilor și a datelor de intrare, rulând programul `START.m`, este obținută valoarea parametrului de ieșire  $f$ , unde  $f$  reprezintă valoarea distanței focale exprimată în pixel. Programul `START.m` conține subprogramele `algoritmi.m`, `procesare_imagine.m` și `filtru_culoare.m`, acestea fiind prezentate în anexa C1. În experiment au fost utilizate un număr de cinci camere. De la fiecare dintre camere au fost prelevate câte zece imagini ale obiectelor de calibrare. În Fig. 7. 3 este prezentată interfața grafică a programului de calibrare intrinsecă. Parametrii de intrare sunt introduși în rubrica *SETARI*. Diametrul cercului 1 este notat cu  $d\_cerc1\_world$ . Diametrul cercului 2 este notat cu  $d\_cerc2\_world$ . Distanța dintre centrele cercurilor este notată cu  $d\_P1P2\_world$ . În rubrica *PROCESAREA IMAGINII* sunt setate valorile componentelor spațiului de culoare HSV. Atunci când valoarea componentei  $V$  este mai mare de 0.8, sunt extrase din imagine toate elementele care au culoarea albă. Pentru afișarea rezultatului trebuie să fie apăsat butonul *DISTANTA FOCALA*.

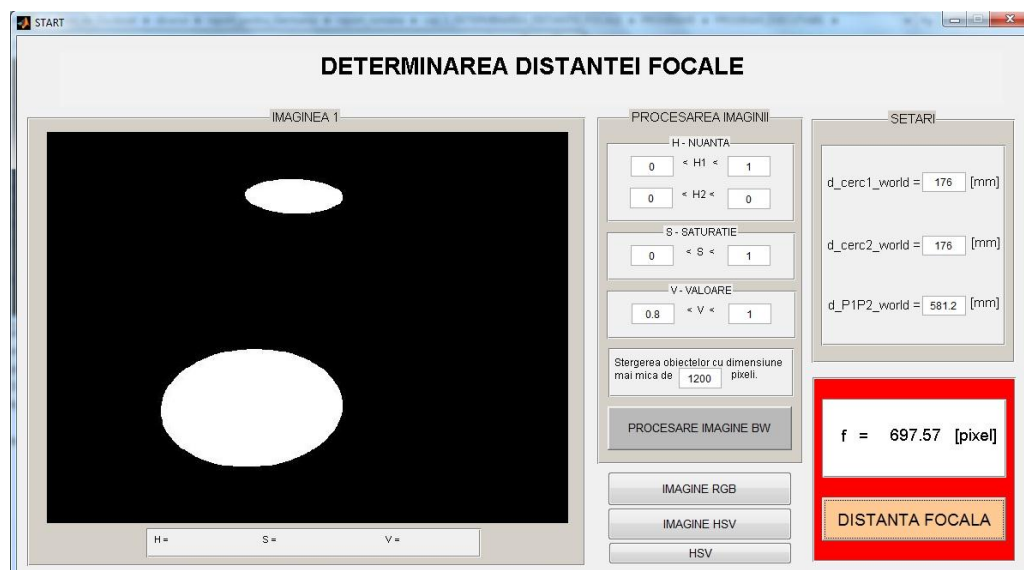


Fig. 7. 3. Interfața grafică

Tipurile de camere utilizate în experiment, respectiv distanțele focale corespunzătoare, sunt prezentate în Tab. 7. 1. Distanța focală de referință notată cu  $f\_ref$ , a fost determinată utilizând programul „Camera Calibration Toolbox for MATLAB”. În Tab. 7. 1 este menționată dimensiunea imaginii, aceasta fiind notată cu  $size\_im$ .









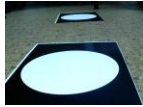














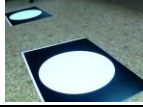


















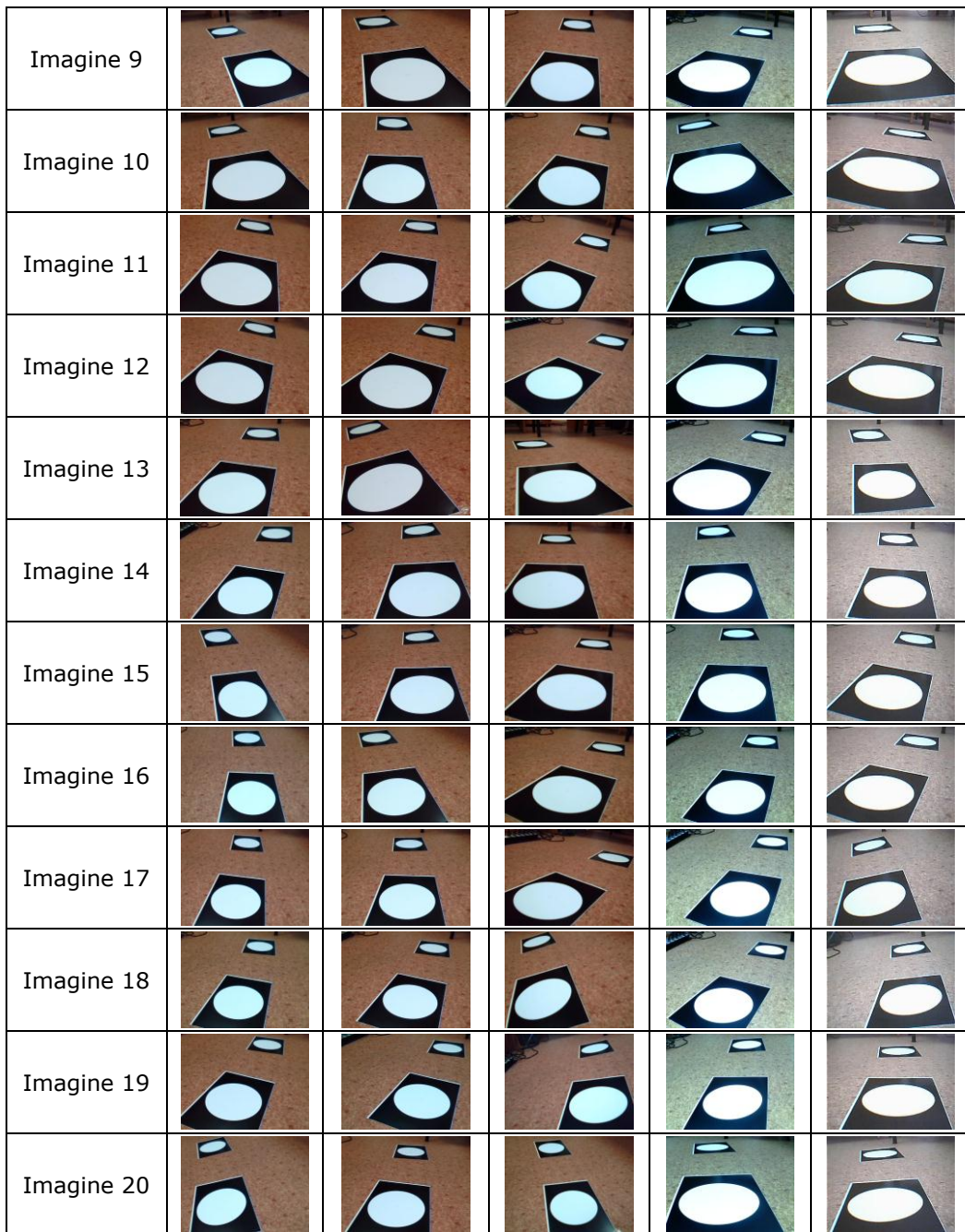
	Tipul camerei	size_im [pixel] x [pixel]	f_ref [pixel]
Camera 1	Logitech C160	640 x 480	739,0605
Camera 2	Logitech C160	640 x 480	764,2329
Camera 3	Logitech C200	640 x 480	694,3049
Camera 4	A4Tech G	640 x 480	825,4410
Camera 5	A4Tech V	320 x 240	478.4197

Tab. 7. 1. Tipurile de camere și caracteristicile principale

### 7.1.4 Rezultatele experimentului

Imaginile utilizate în experiment, sunt prezentate în Tab. 7. 2.

Imagini prelevate					
	Camera 1	Camera 2	Camera 3	Camera 4	Camera 5
Imagine 1					
Imagine 2					
Imagine 3					
Imagine 4					
Imagine 5					
Imagine 6					
Imagine 7					
Imagine 8					



Tab. 7. 2. Imaginile utilizate în experiment

Precizia metodei este analizată prin compararea distanței focale rezultate  $f$ , cu distanța focală de referință notată cu  $f_{ref}$ . Eroarea este determinată prin diferența dintre valorile celor două distanțe focale ( $f$ , respectiv  $f_{ref}$ ).

Camera 1	f_ref [pixel]	f [pixel]	Eroare [pixel]	Eroare [%]
Imagine 1	739,0605	721,55	17,5105	2,3693
Imagine 2	739,0605	715,22	23,8405	3,2258
Imagine 3	739,0605	708,14	30,9205	4,1838
Imagine 4	739,0605	692,85	46,2105	6,2526
Imagine 5	739,0605	697,04	42,0205	5,6857
Imagine 6	739,0605	693,18	45,8805	6,2079
Imagine 7	739,0605	721,66	17,4005	2,3544
Imagine 8	739,0605	698,08	40,9805	5,5449
Imagine 9	739,0605	687,35	51,7105	6,9968
Imagine 10	739,0605	676,25	62,8105	8,4987
Imagine 11	739,0605	687,61	51,4505	6,9616
Imagine 12	739,0605	690,62	48,4405	6,5543
Imagine 13	739,0605	705,50	33,5605	4,5410
Imagine 14	739,0605	753,91	14,8495	2,0092
Imagine 15	739,0605	732,27	6,7905	0,9188
Imagine 16	739,0605	690,19	48,8705	6,6125
Imagine 17	739,0605	708,88	30,1805	4,0836
Imagine 18	739,0605	721,33	17,7305	2,3991
Imagine 19	739,0605	715,15	23,9105	3,2353
Imagine 20	739,0605	707,05	32,0105	4,3312
Media	739,0605	711,25	27,8095	3,7628

Tab. 7. 3. Camera 1 – Erorile distanțelor focale.

Camera 2	f_ref [pixel]	f [pixel]	Eroare [pixel]	Eroare [%]
Imagine 1	764,2329	784,08	19,8471	2,5970
Imagine 2	764,2329	720,60	43,6329	5,7094
Imagine 3	764,2329	790,58	26,3471	3,4475
Imagine 4	764,2329	744,67	19,5629	2,5598
Imagine 5	764,2329	699,84	64,3929	8,4258
Imagine 6	764,2329	702,82	61,4129	8,0359
Imagine 7	764,2329	714,60	49,6329	6,4945
Imagine 8	764,2329	753,25	10,9829	1,4371
Imagine 9	764,2329	738,70	25,5329	3,3410
Imagine 10	764,2329	727,41	36,8229	4,8183
Imagine 11	764,2329	720,78	43,4529	5,6858
Imagine 12	764,2329	747,00	17,2329	2,2549
Imagine 13	764,2329	729,83	34,4029	4,5016
Imagine 14	764,2329	712,26	51,9729	6,8007
Imagine 15	764,2329	710,14	54,0929	7,0781
Imagine 16	764,2329	741,47	22,7629	2,9785
Imagine 17	764,2329	712,73	51,5029	6,7392
Imagine 18	764,2329	738,82	25,4129	3,3253
Imagine 19	764,2329	786,63	22,3971	2,9307
Imagine 20	764,2329	707,55	56,6829	7,4170
Media	764,2329	730,72	33,5119	4,3850

Tab. 7. 4. Camera 2 – Erorile distanțelor focale.

Camera 3	f_ref [pixel]	f [pixel]	Eroare [pixel]	Eroare [%]
Imagine 1	694,3049	691,03	3,2749	0,4717
Imagine 2	694,3049	675,55	18,7549	2,7012
Imagine 3	694,3049	695,87	1,5651	0,2254
Imagine 4	694,3049	714,18	19,8751	2,8626
Imagine 5	694,3049	662,25	32,0549	4,6168
Imagine 6	694,3049	681,51	12,7949	1,8428
Imagine 7	694,3049	675,75	18,5549	2,6724
Imagine 8	694,3049	669,17	25,1349	3,6202
Imagine 9	694,3049	677,08	17,2249	2,4809
Imagine 10	694,3049	685,94	8,3649	1,2048
Imagine 11	694,3049	690,90	3,4049	0,4904
Imagine 12	694,3049	727,06	32,7551	4,7177
Imagine 13	694,3049	721,49	27,1851	3,9154
Imagine 14	694,3049	677,50	16,8049	2,4204
Imagine 15	694,3049	698,02	3,7151	0,5351
Imagine 16	694,3049	703,82	9,5151	1,3704
Imagine 17	694,3049	726,51	32,2051	4,6385
Imagine 18	694,3049	682,41	11,8949	1,7132
Imagine 19	694,3049	718,06	23,7551	3,4214
Imagine 20	694,3049	697,95	3,6451	0,5250
Media	694,3049	704,37	10,0671	1,4500

Tab. 7. 5. Camera 3 – Eroările distanțelor focale.

Camera 4	f_ref [pixel]	f [pixel]	Eroare [pixel]	Eroare [%]
Imagine 1	825,4410	706,23	119,2110	14,4421
Imagine 2	825,4410	788,42	37,0210	4,4850
Imagine 3	825,4410	743,59	81,8510	9,9160
Imagine 4	825,4410	727,17	98,2710	11,9053
Imagine 5	825,4410	683,91	141,5310	17,1461
Imagine 6	825,4410	742,42	83,0210	10,0578
Imagine 7	825,4410	768,92	56,5210	6,8474
Imagine 8	825,4410	789,93	35,5110	4,3021
Imagine 9	825,4410	789,60	35,8410	4,3420
Imagine 10	825,4410	727,19	98,2510	11,9028
Imagine 11	825,4410	781,8	43,6410	5,2870
Imagine 12	825,4410	767,52	57,9210	7,0170
Imagine 13	825,4410	759,96	65,4810	7,9329
Imagine 14	825,4410	777,91	47,5310	5,7583
Imagine 15	825,4410	778,32	47,1210	5,7086
Imagine 16	825,4410	753,73	71,7110	8,6876
Imagine 17	825,4410	713,12	112,3210	13,6074
Imagine 18	825,4410	739,60	85,8410	10,3994
Imagine 19	825,4410	766,03	59,4110	7,1975
Imagine 20	825,4410	769,02	56,4210	6,8353
Media	825,4410	760,70	64,7400	7,8431

Tab. 7. 6. Camera 4 – Eroările distanțelor focale.

Camera 5	f_ref [pixel]	f [pixel]	Eroare [pixel]	Eroare [%]
Imagine 1	478,4197	471,78	6,6397	1,3878
Imagine 2	478,4197	448,38	30,0397	6,2789
Imagine 3	478,4197	481,02	2,6003	0,5435
Imagine 4	478,4197	473,89	4,5297	0,9468
Imagine 5	478,4197	467,14	11,2797	2,3577
Imagine 6	478,4197	463,08	15,3397	3,2063
Imagine 7	478,4197	467,17	11,2497	2,3514
Imagine 8	478,4197	470,73	7,6897	1,6073
Imagine 9	478,4197	464,63	13,7897	2,8823
Imagine 10	478,4197	465,11	13,3097	2,7820
Imagine 11	478,4197	466,56	11,8597	2,4789
Imagine 12	478,4197	463,33	15,0897	3,1541
Imagine 13	478,4197	470,67	7,7497	1,6199
Imagine 14	478,4197	470,34	8,0797	1,6888
Imagine 15	478,4197	453,42	24,9997	5,2255
Imagine 16	478,4197	457,07	21,3497	4,4625
Imagine 17	478,4197	458,43	19,9897	4,1783
Imagine 18	478,4197	456,54	21,8797	4,5733
Imagine 19	478,4197	464,00	14,4197	3,0140
Imagine 20	478,4197	467,83	10,5897	2,2135
Media	478,4197	462,82	15,6007	3,2609

Tab. 7. 7. Camera 5 – Erorile distanțelor focale.

### 7.1.5 Concluzii

Metoda este simplă și flexibilă, dar precizia este relativ scăzută. Eroarea poate depăși uneori procentul de 10%. Pentru o precizie mai ridicată este recomandat ca procedeul de calibrare să fie repetat de mai multe ori; la sfârșit fiind făcută media dintre rezultatele intermediare obținute. Metoda trebuie utilizată cu prudență, deoarece există riscul ca, în anumite condiții, factorii de scalare  $s_1$  și  $s_2$  să aibă valori egale sau aproape egale. Egalitatea sau diferența mică dintre valorile factorilor de scalare  $s_1$  și  $s_2$  conduc la rezultate extrem de eronate. Din acest motiv imaginile trebuie prelevate din unghiuri, astfel încât distanța dintre focarul camerei și centrul primului cerc să fie mult diferită de distanța dintre focarul camerei și centrul celui de-al doilea cerc.

Avantaje	Dezavantaje
<ul style="list-style-type: none"> <li>▪ Metoda este simplă și flexibilă.</li> <li>▪ Sunt utilizate doar două obiecte de calibrare.</li> <li>▪ Este necesară prelevarea unei singure imagini.</li> </ul>	<ul style="list-style-type: none"> <li>▪ Precizia este scăzută.</li> <li>▪ Nu este permis ca axa optică a camerei să fie perpendiculară pe planul obiectelor, deoarece este necesar ca factorii de scalare <math>s_1</math> și <math>s_2</math> să aibă valori diferite.</li> </ul>

Tab. 7. 8. Avantajele și dezavantajele metodei de calibrare extrinsecă



## 7.2 Calibrare extrinsecă prin metode analitice

### 7.2.1 Introducere

În acest capitol este prezentată o metodă simplă și rapidă de determinare a parametrilor extrinseci ai camerelor. Prin această metodă pot fi determinate matricea de rotație și vectorul de translație utilizate pentru a face trecerea din sistemul de referință al camerei 1 în sistemul de referință al camerei 2 și invers. Metoda este avantajoasă prin faptul că nu este necesară o instalație de calibrare sofisticată, calibrarea putând fi realizată utilizând doar patru obiecte de formă circulară al căror diametru este cunoscut. Obiectele de calibrare pot fi amplasate arbitrar în spațiu, iar distanțele relative dintre centrele cercurilor pot fi necunoscute.

Ca și obiecte de calibrare, sunt utilizate patru cercuri cu diametre cunoscute, notate  $d\_cerc1\_world$ ,  $d\_cerc2\_world$ ,  $d\_cerc3\_world$ , respectiv  $d\_cerc4\_world$ . Cercurile sunt amplasate arbitrar, distanțele relative dintre centre fiind necunoscute.

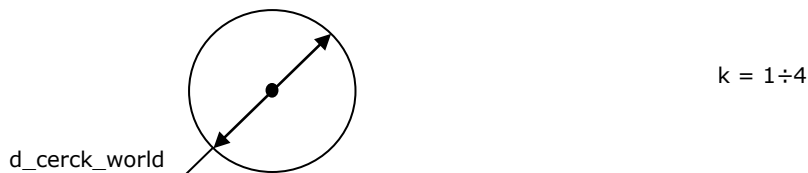


Fig. 7. 4. Obiecte geometrice utilizate pentru calibrare

### 7.2.2 Modelul matematic

Obiectele de calibrare, utilizate în această metodă, sunt de formă circulară. Patru cercuri, având dimensiuni cunoscute, sunt amplasate aleator în spațiu. Centrele celor patru cercuri sunt notate cu  $P1\_world$ ,  $P2\_world$ ,  $P3\_world$ , respectiv  $P4\_world$ . Obiectele sunt vizibile în imaginile a două camere video, notate cu  $cam1$ , respectiv  $cam2$ . Punctele  $P1\_im1$ ,  $P2\_im1$ ,  $P3\_im1$ , respectiv  $P4\_im1$  reprezintă proiecțiile punctelor  $P1\_world$ ,  $P2\_world$ ,  $P3\_world$ , respectiv  $P4\_world$  în imaginea corespunzătoare camerei 1. Punctele  $P1\_im2$ ,  $P2\_im2$ ,  $P3\_im2$ , respectiv  $P4\_im2$  reprezintă proiecțiile punctelor  $P1\_world$ ,  $P2\_world$ ,  $P3\_world$ , respectiv  $P4\_world$  în imaginea corespunzătoare camerei 2. Sunt utilizate șase sisteme de referință:  $im1$ ,  $cam1$ ,  $offset\_cam1$ , respectiv  $im2$ ,  $cam2$ ,  $offset\_cam2$ . Sistemul de referință  $im1$  este sistemul de referință al imaginii 1. Sistemul de referință  $im2$  este sistemul de referință al imaginii 2. Sistemul de referință  $cam1$  este sistemul de referință al camerei 1. Originea sistemului de referință  $cam1$  coincide cu focarul  $F1$ , iar axele  $x\_cam1$  și  $y\_cam1$  sunt paralele cu axele  $x\_im1$  și  $y\_im1$  corespunzătoare imaginii 1. Sistemul de referință  $cam2$  este sistemul de referință al camerei 2. Originea sistemului de referință  $cam2$  coincide cu focarul  $F2$ , iar axele  $x\_cam2$  și  $y\_cam2$  sunt paralele cu axele  $x\_im2$  și  $y\_im2$  corespunzătoare imaginii 2. Sistemul de coordonate  $offset\_cam1$  a rezultat în urma translației sistemului de coordonate  $cam1$  în punctul  $P4\_world$ . Sistemul de coordonate  $offset\_cam2$  a rezultat în urma translației sistemului de coordonate  $cam2$  în punctul  $P4\_world$ . Pozițiile relative dintre sistemele de referință  $cam1$  și  $offset\_cam1$  pot fi determinate utilizând factori de scalare obținuți din raportul scenă imagine. Astfel, sunt cunoscute relațiile dintre sistemele de referință  $im1$ ,  $cam1$  și  $offset\_cam1$ , respectiv  $im2$ ,  $cam2$  și

*offset\_cam2*. Utilizând modelul matematic poate fi determinată relația dintre sistemele de referință *offset\_cam1* și *offset\_cam2*. Algoritmii de determinare a matricei de rotație sunt relativ simpli datorită faptului că originea sistemului de referință *offset\_cam1* coincide cu originea sistemului de referință *offset\_cam2*. După determinarea valorilor matricei de rotație urmează a fi determinate valorile vectorului de translație. Modelul geometric este prezentat în Fig. 7. 5.

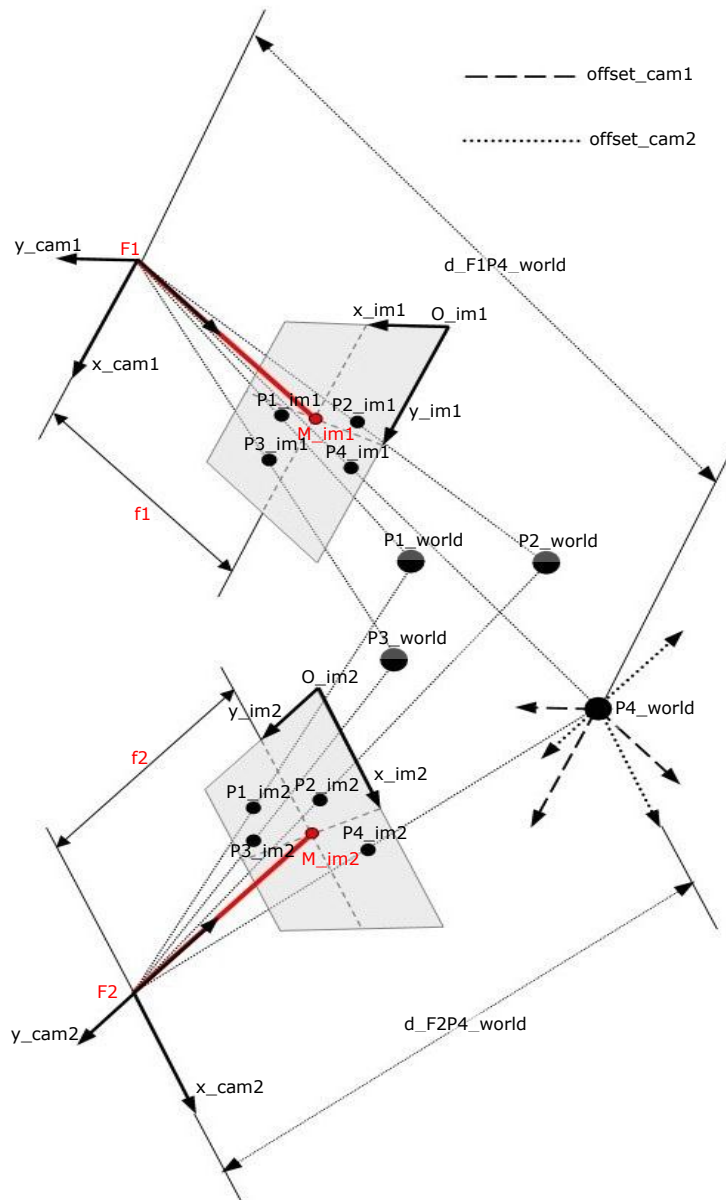


Fig. 7. 5. Modelul geometric pentru calibrarea extrinsecă [Tilneac, 2011b]

**Etapile de elaborare a modelului matematic:**

Tab. 7. 9 reflectă structura modelului matematic, indicând etapele prin care modelul matematic a fost elaborat. Modelul matematic a fost elaborat în opt etape. Primele șase etape sunt elaborate în paralel pentru camera 1 și camera 2.

<p><b>Etapa1-camera1</b></p> <p>Determinarea coordonatelor proiecțiilor centrelor cercurilor în imaginea 1, raportate la sistemul de referință al imaginii 1.</p>	<p><b>Etapa1-camera2</b></p> <p>Determinarea coordonatelor proiecțiilor centrelor cercurilor în imaginea 2, raportate la sistemul de referință al imaginii 2.</p>
<p><b>Etapa2-camera1</b></p> <p>Calculul distanțelor de la punctul focal <math>F1_x</math> respectiv <math>F1_y</math> până la fiecare dintre proiecțiile centrelor cercurilor în imaginea 1.</p>	<p><b>Etapa2-camera2</b></p> <p>Calculul distanțelor de la punctul focal <math>F2_x</math> respectiv <math>F2_y</math> până la fiecare dintre proiecțiile centrelor cercurilor în imaginea 2.</p>
<p><b>Etapa3-camera1</b></p> <p>Determinarea factorilor de scalare corespunzători fiecărui punct. Factorii de scalare sunt utilizați la conversia din pixel în mm, iar valorile acestora sunt distincte pe fiecare dintre axele x și y.</p>	<p><b>Etapa3-camera2</b></p> <p>Determinarea factorilor de scalare corespunzători fiecărui punct. Factorii de scalare sunt utilizați la conversia din pixel în mm, iar valorile acestora sunt distincte pe fiecare dintre axele x și y.</p>
<p><b>Etapa4-camera1</b></p> <p>Determinarea coordonatelor punctelor <math>P1_{world}</math>, <math>P2_{world}</math>, <math>P3_{world}</math> și <math>P4_{world}</math>, raportate la sistemul de referință cam1.</p>	<p><b>Etapa4-camera2</b></p> <p>Determinarea coordonatelor punctelor <math>P1_{world}</math>, <math>P2_{world}</math>, <math>P3_{world}</math> și <math>P4_{world}</math>, raportate la sistemul de referință cam2.</p>
<p><b>Etapa5-camera1</b></p> <p>Vectorul de poziție corespunzător punctelor <math>P1_{world}</math>, <math>P2_{world}</math>, <math>P3_{world}</math> și <math>P4_{world}</math>, raportat la sistemul de referință cam1.</p>	<p><b>Etapa5-camera2</b></p> <p>Vectorul de poziție corespunzător punctelor <math>P1_{world}</math>, <math>P2_{world}</math>, <math>P3_{world}</math> și <math>P4_{world}</math>, raportat la sistemul de referință cam2.</p>
<p><b>Etapa6-camera1</b></p> <p>Determinarea coordonatelor punctelor <math>P1_{world}</math>, <math>P2_{world}</math>, <math>P3_{world}</math> și <math>P4_{world}</math>, raportate la sistemul de referință <math>offset\_cam1</math>.</p>	<p><b>Etapa6-camera2</b></p> <p>Determinarea coordonatelor punctelor <math>P1_{world}</math>, <math>P2_{world}</math>, <math>P3_{world}</math> și <math>P4_{world}</math>, raportate la sistemul de referință <math>offset\_cam2</math>.</p>
<p><b>Etapa7</b></p> <p>Determinarea matricei de rotație utilizate pentru transformarea valorilor coordonatelor raportate la sistemul de referință <math>offset\_cam1</math>, în valori raportate la sistemul de referință <math>offset\_cam2</math>.</p>	
<p><b>Etapa8</b></p> <p>Determinarea vectorului de translație utilizat pentru transformarea valorilor coordonatelor raportate la sistemul de referință cam1, în valori raportate la sistemul de referință cam2.</p>	

Tab. 7. 9. Etapele de elaborare a modelului matematic



Modelul matematic a fost realizat pe baza figurii geometrice prezentate în Fig. 7. 5. Urmează o descriere detaliată a modelului matematic realizat prin parcurgerea celor opt etape prezentate în Tab. 7. 9.

### Etapa1\_camera1

Pentru imaginea 1, coordonatele proiecțiilor centrelor celor patru cercuri sunt specificate în relațiile (7. 15), (7. 17), (7. 19) și (7. 21).

$$P1\_im1 = [x\_P1\_im1 \ y\_P1\_im1 \ z\_P1\_im1]^T \quad (7. 15)$$

$$z\_P1\_im1 = 0 \quad (7. 16)$$

$$P2\_im1 = [x\_P2\_im1 \ y\_P2\_im1 \ z\_P2\_im1]^T \quad (7. 17)$$

$$z\_P2\_im1 = 0 \quad (7. 18)$$

$$P3\_im1 = [x\_P3\_im1 \ y\_P3\_im1 \ z\_P3\_im1]^T \quad (7. 19)$$

$$z\_P3\_im1 = 0 \quad (7. 20)$$

$$P4\_im1 = [x\_P4\_im1 \ y\_P4\_im1 \ z\_P4\_im1]^T \quad (7. 21)$$

$$z\_P4\_im1 = 0 \quad (7. 22)$$

### Etapa1\_camera2

Pentru imaginea 2, coordonatele proiecțiilor centrelor celor patru cercuri sunt specificate în relațiile (7. 23), (7. 25), (7. 27) și (7. 29).

$$P1\_im2 = [x\_P1\_im2 \ y\_P1\_im2 \ z\_P1\_im2]^T \quad (7. 23)$$

$$z\_P1\_im2 = 0 \quad (7. 24)$$

$$P2\_im2 = [x\_P2\_im2 \ y\_P2\_im2 \ z\_P2\_im2]^T \quad (7. 25)$$

$$z\_P2\_im2 = 0 \quad (7. 26)$$

$$P3\_im2 = [x\_P3\_im2 \ y\_P3\_im2 \ z\_P3\_im2]^T \quad (7. 27)$$

$$z\_P3\_im2 = 0 \quad (7. 28)$$

$$P4\_im2 = [x\_P4\_im2 \ y\_P4\_im2 \ z\_P4\_im2]^T \quad (7. 29)$$

$$z_{P4\_im2} = 0 \quad (7.30)$$

### Etapa2\_camera1

Din cauza faptului că valorile factorilor de scalare diferă pe axa x și axa y, în algoritmi sunt utilizate două distanțe focale  $f1_x$  respectiv  $f1_y$ . Utilizând ecuațiile (7.31) ÷ (7.38) sunt determinate valorile distanțelor dintre focarul F1 și fiecare dintre punctele P1\_im1, P2\_im1, P3\_im1, respectiv P4\_im1 din imaginea 1. Pentru fiecare distanță există două valori distincte; diferența fiind dată de distanțele focale utilizate,  $f1_x$  sau  $f1_y$ . Atunci când este utilizată distanța focală  $f1_x$ , trebuiesc modificate coordonatele pe axa y, pentru a avea același factor de scalare pe ambele axe. Modificarea este realizată prin înmulțirea cu raportul  $f1_x/f1_y$ . Similar se procedează și în cazul utilizării distanței focale  $f1_y$ , coordonatele pe axa x fiind modificate prin înmulțirea cu raportul  $f1_y/f1_x$ .

$$\begin{aligned} d_{F1P1\_im1\_x} &= \\ &= \sqrt{(x_{P1\_im1} - x_{M\_im1})^2 + \left( (y_{P1\_im1} - y_{M\_im1}) * \frac{f1\_x}{f1\_y} \right)^2 + (f1\_x)^2} \quad (7.31) \end{aligned}$$

$$\begin{aligned} d_{F1P1\_im1\_y} &= \\ &= \sqrt{\left( (x_{P1\_im1} - x_{M\_im1}) * \frac{f1\_y}{f1\_x} \right)^2 + (y_{P1\_im1} - y_{M\_im1})^2 + (f1\_y)^2} \quad (7.32) \end{aligned}$$

$$\begin{aligned} d_{F1P2\_im1\_x} &= \\ &= \sqrt{(x_{P2\_im1} - x_{M\_im1})^2 + \left( (y_{P2\_im1} - y_{M\_im1}) * \frac{f1\_x}{f1\_y} \right)^2 + (f1\_x)^2} \quad (7.33) \end{aligned}$$

$$\begin{aligned} d_{F1P2\_im1\_y} &= \\ &= \sqrt{\left( (x_{P2\_im1} - x_{M\_im1}) * \frac{f1\_y}{f1\_x} \right)^2 + (y_{P2\_im1} - y_{M\_im1})^2 + (f1\_y)^2} \quad (7.34) \end{aligned}$$

$$\begin{aligned} d_{F1P3\_im1\_x} &= \\ &= \sqrt{(x_{P3\_im1} - x_{M\_im1})^2 + \left( (y_{P3\_im1} - y_{M\_im1}) * \frac{f1\_x}{f1\_y} \right)^2 + (f1\_x)^2} \quad (7.35) \end{aligned}$$

$$\begin{aligned}
 d_{F1P3\_im1\_y} &= \\
 &= \sqrt{\left( (x_{P3\_im1} - x_{M\_im1}) * \frac{f1\_y}{f1\_x} \right)^2 + (y_{P3\_im1} - y_{M\_im1})^2 + (f1\_y)^2} \quad (7.36)
 \end{aligned}$$

$$\begin{aligned}
 d_{F1P4\_im1\_x} &= \\
 &= \sqrt{(x_{P4\_im1} - x_{M\_im1})^2 + \left( (y_{P4\_im1} - y_{M\_im1}) * \frac{f1\_x}{f1\_y} \right)^2 + (f1\_x)^2} \quad (7.37)
 \end{aligned}$$

$$\begin{aligned}
 d_{F1P4\_im1\_y} &= \\
 &= \sqrt{\left( (x_{P4\_im1} - x_{M\_im1}) * \frac{f1\_y}{f1\_x} \right)^2 + (y_{P4\_im1} - y_{M\_im1})^2 + (f1\_y)^2} \quad (7.38)
 \end{aligned}$$

### Etapa2\_camera2

Aceeași metodă este aplicată în cazul imaginii 2. Din cauza faptului că valoarea factorilor de scalare diferă pe axele x și y, în algoritmi sunt utilizate două distanțe focale  $f2\_x$  respectiv  $f2\_y$ . Utilizând ecuațiile (7.39) ÷ (7.46) sunt determinate valorile distanțelor dintre focarul F2 și punctele P1\_im2, P2\_im2, P3\_im2, respectiv P4\_im2 din imaginea 2.

$$\begin{aligned}
 d_{F2P1\_im2\_x} &= \\
 &= \sqrt{(x_{P1\_im2} - x_{M\_im2})^2 + \left( (y_{P1\_im2} - y_{M\_im2}) * \frac{f2\_x}{f2\_y} \right)^2 + (f2\_x)^2} \quad (7.39)
 \end{aligned}$$

$$\begin{aligned}
 d_{F2P1\_im2\_y} &= \\
 &= \sqrt{\left( (x_{P1\_im2} - x_{M\_im2}) * \frac{f2\_y}{f2\_x} \right)^2 + (y_{P1\_im2} - y_{M\_im2})^2 + (f2\_y)^2} \quad (7.40)
 \end{aligned}$$

$$\begin{aligned}
 d_{F2P2\_im2\_x} &= \\
 &= \sqrt{(x_{P2\_im2} - x_{M\_im2})^2 + \left( (y_{P2\_im2} - y_{M\_im2}) * \frac{f2\_x}{f2\_y} \right)^2 + (f2\_x)^2} \quad (7.41)
 \end{aligned}$$

$$\begin{aligned}
 d_{F2P2\_im2\_y} &= \\
 &= \sqrt{\left( (x_{P2\_im2} - x_{M\_im2}) * \frac{f2\_y}{f2\_x} \right)^2 + (y_{P2\_im2} - y_{M\_im2})^2 + (f2\_y)^2} \quad (7.42)
 \end{aligned}$$

$$d_{F2P3\_im2\_x} = \sqrt{(x_{P3\_im2} - x_{M\_im2})^2 + \left( (y_{P3\_im2} - y_{M\_im2}) * \frac{f2\_x}{f2\_y} \right)^2 + (f2\_x)^2} \quad (7.43)$$

$$d_{F2P3\_im2\_y} = \sqrt{\left( (x_{P3\_im2} - x_{M\_im2}) * \frac{f2\_y}{f2\_x} \right)^2 + (y_{P3\_im2} - y_{M\_im2})^2 + (f2\_y)^2} \quad (7.44)$$

$$d_{F2P4\_im2\_x} = \sqrt{(x_{P4\_im2} - x_{M\_im2})^2 + \left( (y_{P4\_im2} - y_{M\_im2}) * \frac{f2\_x}{f2\_y} \right)^2 + (f2\_x)^2} \quad (7.45)$$

$$d_{F2P4\_im2\_y} = \sqrt{\left( (x_{P4\_im2} - x_{M\_im2}) * \frac{f2\_y}{f2\_x} \right)^2 + (y_{P4\_im2} - y_{M\_im2})^2 + (f2\_y)^2} \quad (7.46)$$

### Etapa3\_camera1

În cazul utilizării camerei 1, factorii de scalare pe axele x și y, reprezintă raportul dintre distanța  $d_{F1P\_world}$  și distanța  $d_{F1P\_im1}$ , raportul fiind utilizat pentru transformări din pixel în mm. Factorii de scalare corespunzători punctelor din imaginea 1 sunt calculați prin ecuațiile (7. 47) ÷ (7. 54).

$$s1\_cam1\_x = \frac{d_{F1P1\_world}}{d_{F1P1\_im1\_x}} \quad [\text{mm/pixel}] \quad (7.47)$$

$$s1\_cam1\_y = \frac{d_{F1P1\_world}}{d_{F1P1\_im1\_y}} \quad [\text{mm/pixel}] \quad (7.48)$$

$$s2\_cam1\_x = \frac{d_{F1P2\_world}}{d_{F1P2\_im1\_x}} \quad [\text{mm/pixel}] \quad (7.49)$$

$$s2\_cam1\_y = \frac{d_{F1P2\_world}}{d_{F1P2\_im1\_y}} \quad [\text{mm/pixel}] \quad (7.50)$$

$$s3\_cam1\_x = \frac{d_{F1P3\_world}}{d_{F1P3\_im1\_x}} \quad [\text{mm/pixel}] \quad (7.51)$$

$$s3\_cam1\_y = \frac{d\_F1P3\_world}{d\_F1P3\_im1\_y} \quad [\text{mm/pixel}] \quad (7.52)$$

$$s4\_cam1\_x = \frac{d\_F1P4\_world}{d\_F1P4\_im1\_x} \quad [\text{mm/pixel}] \quad (7.53)$$

$$s4\_cam1\_y = \frac{d\_F1P4\_world}{d\_F1P4\_im1\_y} \quad [\text{mm/pixel}] \quad (7.54)$$

### Etapa3\_camera2

În cazul utilizării camerei 2, factorii de scalare pe axele x și y, reprezintă raportul dintre distanța  $d_{F2P\_world}$  și distanța  $d_{F2P\_im2}$ , raportul fiind utilizat pentru transformări din pixel în mm. Factorii de scalare corespunzători punctelor din imaginea 2 sunt calculați prin ecuațiile (7. 55) ÷ (7. 62).

$$s1\_cam2\_x = \frac{d\_F2P1\_world}{d\_F2P1\_im2\_x} \quad [\text{mm/pixel}] \quad (7.55)$$

$$s1\_cam2\_y = \frac{d\_F2P1\_world}{d\_F2P1\_im2\_y} \quad [\text{mm/pixel}] \quad (7.56)$$

$$s2\_cam2\_x = \frac{d\_F2P2\_world}{d\_F2P2\_im2\_x} \quad [\text{mm/pixel}] \quad (7.57)$$

$$s2\_cam2\_y = \frac{d\_F2P2\_world}{d\_F2P2\_im2\_y} \quad [\text{mm/pixel}] \quad (7.58)$$

$$s3\_cam2\_x = \frac{d\_F2P3\_world}{d\_F2P3\_im2\_x} \quad [\text{mm/pixel}] \quad (7.59)$$

$$s3\_cam2\_y = \frac{d\_F2P3\_world}{d\_F2P3\_im2\_y} \quad [\text{mm/pixel}] \quad (7.60)$$

$$s4\_cam2\_x = \frac{d\_F2P4\_world}{d\_F2P4\_im2\_x} \quad [\text{mm/pixel}] \quad (7.61)$$

$$s4\_cam2\_y = \frac{d\_F2P4\_world}{d\_F2P4\_im2\_y} \quad [\text{mm/pixel}] \quad (7.62)$$

### Etapa4\_camera1

Coordonatele punctelor  $P1\_world$ ,  $P2\_world$ ,  $P3\_world$  și  $P4\_world$ , exprimate în milimetri și raportate la sistemul de referință  $cam1$ , sunt determinate utilizând

sistemele de ecuații (7. 63), (7. 64), (7. 65), respectiv (7. 66).

$$\left\{ \begin{array}{l} x_{P1\_world\_cam1} = s1\_cam1\_x * (x_{P1\_im1} - x_{M\_im1}) \\ y_{P1\_world\_cam1} = s1\_cam1\_y * (y_{P1\_im1} - y_{M\_im1}) \end{array} \right. \quad (7. 63)$$

$$z_{P1\_world\_cam1} = \sqrt{(d_{F1P1\_world})^2 - (x_{P1\_world\_cam1})^2 - (y_{P1\_world\_cam1})^2}$$

$$\left\{ \begin{array}{l} x_{P2\_world\_cam1} = s2\_cam1\_x * (x_{P2\_im1} - x_{M\_im1}) \\ y_{P2\_world\_cam1} = s2\_cam1\_y * (y_{P2\_im1} - y_{M\_im1}) \end{array} \right. \quad (7. 64)$$

$$z_{P2\_world\_cam1} = \sqrt{(d_{F1P2\_world})^2 - (x_{P2\_world\_cam1})^2 - (y_{P2\_world\_cam1})^2}$$

$$\left\{ \begin{array}{l} x_{P3\_world\_cam1} = s3\_cam1\_x * (x_{P3\_im1} - x_{M\_im1}) \\ y_{P3\_world\_cam1} = s3\_cam1\_y * (y_{P3\_im1} - y_{M\_im1}) \end{array} \right. \quad (7. 65)$$

$$z_{P3\_world\_cam1} = \sqrt{(d_{F1P3\_world})^2 - (x_{P3\_world\_cam1})^2 - (y_{P3\_world\_cam1})^2}$$

$$\left\{ \begin{array}{l} x_{P4\_world\_cam1} = s4\_cam1\_x * (x_{P4\_im1} - x_{M\_im1}) \\ y_{P4\_world\_cam1} = s4\_cam1\_y * (y_{P4\_im1} - y_{M\_im1}) \end{array} \right. \quad (7. 66)$$

$$z_{P4\_world\_cam1} = \sqrt{(d_{F1P4\_world})^2 - (x_{P4\_world\_cam1})^2 - (y_{P4\_world\_cam1})^2}$$

#### **Etapa4\_camera2**

Coordonatele punctelor P1\_world, P2\_world, P3\_world și P4\_world, exprimate în milimetri și raportate la sistemul de referință cam2, sunt determinate utilizând sistemele de ecuații (7. 67), (7. 68), (7. 69), respectiv (7. 70).

$$\left\{ \begin{array}{l} x_{P1\_world\_cam2} = s1\_cam2\_x * (x_{P1\_im2} - x_{M\_im2}) \\ y_{P1\_world\_cam2} = s1\_cam2\_y * (y_{P1\_im2} - y_{M\_im2}) \end{array} \right. \quad (7.67)$$

$$z_{P1\_world\_cam2} = \sqrt{(d_{F2P1\_world})^2 - (x_{P1\_world\_cam2})^2 - (y_{P1\_world\_cam2})^2}$$

$$\left\{ \begin{array}{l} x_{P2\_world\_cam2} = s2\_cam2\_x * (x_{P2\_im2} - x_{M\_im2}) \\ y_{P2\_world\_cam2} = s2\_cam2\_y * (y_{P2\_im2} - y_{M\_im2}) \end{array} \right. \quad (7.68)$$

$$z_{P2\_world\_cam2} = \sqrt{(d_{F2P2\_world})^2 - (x_{P2\_world\_cam2})^2 - (y_{P2\_world\_cam2})^2}$$

$$\left\{ \begin{array}{l} x_{P3\_world\_cam2} = s3\_cam2\_x * (x_{P3\_im2} - x_{M\_im2}) \\ y_{P3\_world\_cam2} = s3\_cam2\_y * (y_{P3\_im2} - y_{M\_im2}) \end{array} \right. \quad (7.69)$$

$$z_{P3\_world\_cam2} = \sqrt{(d_{F2P3\_world})^2 - (x_{P3\_world\_cam2})^2 - (y_{P3\_world\_cam2})^2}$$

$$\left\{ \begin{array}{l} x_{P4\_world\_cam2} = s4\_cam2\_x * (x_{P4\_im2} - x_{M\_im2}) \\ y_{P4\_world\_cam2} = s4\_cam2\_y * (y_{P4\_im2} - y_{M\_im2}) \end{array} \right. \quad (7.70)$$

$$z_{P4\_world\_cam2} = \sqrt{(d_{F2P4\_world})^2 - (x_{P4\_world\_cam2})^2 - (y_{P4\_world\_cam2})^2}$$

### Etapa5\_camera1

Vectorii de poziție raportați la cam1 corespunzători punctelor P1\_world, P2\_world, P3\_world și P4\_world sunt exprimați prin relațiile (7. 71), (7. 72), (7. 73), respectiv (7. 74).

$$P1\_world\_cam1 = \begin{bmatrix} x\_P1\_world\_cam1 \\ y\_P1\_world\_cam1 \\ z\_P1\_world\_cam1 \end{bmatrix} \quad (7.71)$$

$$P2\_world\_cam1 = \begin{bmatrix} x\_P2\_world\_cam1 \\ y\_P2\_world\_cam1 \\ z\_P2\_world\_cam1 \end{bmatrix} \quad (7.72)$$

$$P3\_world\_cam1 = \begin{bmatrix} x\_P3\_world\_cam1 \\ y\_P3\_world\_cam1 \\ z\_P3\_world\_cam1 \end{bmatrix} \quad (7.73)$$

$$P4\_world\_cam1 = \begin{bmatrix} x\_P4\_world\_cam1 \\ y\_P4\_world\_cam1 \\ z\_P4\_world\_cam1 \end{bmatrix} \quad (7.74)$$

### **Etapa5\_camera2**

Vectorii de poziție raportați la cam2 corespunzători punctelor P1\_world, P2\_world, P3\_world și P4\_world sunt exprimați prin relațiile (7.75), (7.76), (7.77), respectiv (7.78).

$$P1\_world\_cam2 = \begin{bmatrix} x\_P1\_world\_cam2 \\ y\_P1\_world\_cam2 \\ z\_P1\_world\_cam2 \end{bmatrix} \quad (7.75)$$

$$P2\_world\_cam2 = \begin{bmatrix} x\_P2\_world\_cam2 \\ y\_P2\_world\_cam2 \\ z\_P2\_world\_cam2 \end{bmatrix} \quad (7.76)$$

$$P3\_world\_cam2 = \begin{bmatrix} x\_P3\_world\_cam2 \\ y\_P3\_world\_cam2 \\ z\_P3\_world\_cam2 \end{bmatrix} \quad (7.77)$$

$$P4\_world\_cam2 = \begin{bmatrix} x\_P4\_world\_cam2 \\ y\_P4\_world\_cam2 \\ z\_P4\_world\_cam2 \end{bmatrix} \quad (7.78)$$



**Etapa6\_camera1**

Coordonatele punctelor P1\_world, P2\_world și P3\_world raportate la sistemul de referință *offset\_cam1* sunt determinate prin relațiile (7. 79), (7. 80) și (7. 81). Coordonatele punctului P4\_world raportate la sistemul de referință *offset\_cam1* au valorile egale cu zero, deoarece originea sistemului de coordonate *offset\_cam1* coincide cu punctul P4\_world.

$$\begin{bmatrix} x\_P1\_world\_offset\_cam1 \\ y\_P1\_world\_offset\_cam1 \\ z\_P1\_world\_offset\_cam1 \end{bmatrix} = \begin{bmatrix} x\_P1\_world\_cam1 \\ y\_P1\_world\_cam1 \\ z\_P1\_world\_cam1 \end{bmatrix} - \begin{bmatrix} x\_P4\_world\_cam1 \\ y\_P4\_world\_cam1 \\ z\_P4\_world\_cam1 \end{bmatrix} \quad (7. 79)$$

$$\begin{bmatrix} x\_P2\_world\_offset\_cam1 \\ y\_P2\_world\_offset\_cam1 \\ z\_P2\_world\_offset\_cam1 \end{bmatrix} = \begin{bmatrix} x\_P2\_world\_cam1 \\ y\_P2\_world\_cam1 \\ z\_P2\_world\_cam1 \end{bmatrix} - \begin{bmatrix} x\_P4\_world\_cam1 \\ y\_P4\_world\_cam1 \\ z\_P4\_world\_cam1 \end{bmatrix} \quad (7. 80)$$

$$\begin{bmatrix} x\_P3\_world\_offset\_cam1 \\ y\_P3\_world\_offset\_cam1 \\ z\_P3\_world\_offset\_cam1 \end{bmatrix} = \begin{bmatrix} x\_P3\_world\_cam1 \\ y\_P3\_world\_cam1 \\ z\_P3\_world\_cam1 \end{bmatrix} - \begin{bmatrix} x\_P4\_world\_cam1 \\ y\_P4\_world\_cam1 \\ z\_P4\_world\_cam1 \end{bmatrix} \quad (7. 81)$$

**Etapa6\_camera2**

Coordonatele punctelor P1\_world, P2\_world și P3\_world raportate la sistemul de referință *offset\_cam2* sunt determinate prin relațiile (7. 82), (7. 83) și (7. 84). Coordonatele punctului P4\_world raportate la sistemul de referință *offset\_cam2* au valorile egale cu zero, deoarece originea sistemului de coordonate *offset\_cam2* coincide cu punctul P4\_world.

$$\begin{bmatrix} x\_P1\_world\_offset\_cam2 \\ y\_P1\_world\_offset\_cam2 \\ z\_P1\_world\_offset\_cam2 \end{bmatrix} = \begin{bmatrix} x\_P1\_world\_cam2 \\ y\_P1\_world\_cam2 \\ z\_P1\_world\_cam2 \end{bmatrix} - \begin{bmatrix} x\_P4\_world\_cam2 \\ y\_P4\_world\_cam2 \\ z\_P4\_world\_cam2 \end{bmatrix} \quad (7. 82)$$

$$\begin{bmatrix} x\_P2\_world\_offset\_cam2 \\ y\_P2\_world\_offset\_cam2 \\ z\_P2\_world\_offset\_cam2 \end{bmatrix} = \begin{bmatrix} x\_P2\_world\_cam2 \\ y\_P2\_world\_cam2 \\ z\_P2\_world\_cam2 \end{bmatrix} - \begin{bmatrix} x\_P4\_world\_cam2 \\ y\_P4\_world\_cam2 \\ z\_P4\_world\_cam2 \end{bmatrix} \quad (7. 83)$$

$$\begin{bmatrix} x_{P3\_world\_offset\_cam2} \\ y_{P3\_world\_offset\_cam2} \\ z_{P3\_world\_offset\_cam2} \end{bmatrix} = \begin{bmatrix} x_{P3\_world\_cam2} \\ y_{P3\_world\_cam2} \\ z_{P3\_world\_cam2} \end{bmatrix} - \begin{bmatrix} x_{P4\_world\_cam2} \\ y_{P4\_world\_cam2} \\ z_{P4\_world\_cam2} \end{bmatrix} \quad (7.84)$$

### Etapa7

Transformarea dintr-un sistem de referință 1 într-un sistem de referință 2 poate fi realizată utilizând o matrice de rotație și un vector de translație. Pentru a realiza transformarea din sistemul de referință offset\_cam1 în sistemul de referință offset\_cam2 nu este necesar un vector de translație, deoarece originea sistemului offset\_cam1 coincide cu originea sistemului offset\_cam2. Astfel, este utilizată doar matricea de rotație notată cu  $M\_Rotatie$ . Inițial, valorile matricei de rotație sunt necunoscute. Etapa7 constă în determinarea valorilor matricei de rotație. Algoritmii de calcul au fost construiți pe baza relației (7. 85).

$$\begin{bmatrix} x_{P\_world\_offset\_cam2} \\ y_{P\_world\_offset\_cam2} \\ x_{P\_world\_offset\_cam2} \end{bmatrix} = \begin{bmatrix} r11 & r12 & r13 \\ r21 & r22 & r23 \\ r31 & r32 & r33 \end{bmatrix} * \begin{bmatrix} x_{P\_world\_offset\_cam1} \\ y_{P\_world\_offset\_cam1} \\ x_{P\_world\_offset\_cam1} \end{bmatrix} \quad (7.85)$$

$$M\_Rotatie = \begin{bmatrix} r11 & r12 & r13 \\ r21 & r22 & r23 \\ r31 & r32 & r33 \end{bmatrix} \quad (7.86)$$

Matricea de rotație conține parametrii  $r11, r12, r13, r21, r22, r23, r31, r32$  și  $r33$ . Valorile acestor parametri vor fi determinate din relațiile (7. 87), (7. 88) și (7. 89).

$$\begin{bmatrix} x_{P1\_world\_offset\_cam2} \\ y_{P1\_world\_offset\_cam2} \\ z_{P1\_world\_offset\_cam2} \end{bmatrix} = \begin{bmatrix} r11 & r12 & r13 \\ r21 & r22 & r23 \\ r31 & r32 & r33 \end{bmatrix} * \begin{bmatrix} x_{P1\_world\_offset\_cam1} \\ y_{P1\_world\_offset\_cam1} \\ z_{P1\_world\_offset\_cam1} \end{bmatrix} \quad (7.87)$$

$$\begin{bmatrix} x_{P2\_world\_offset\_cam2} \\ y_{P2\_world\_offset\_cam2} \\ z_{P2\_world\_offset\_cam2} \end{bmatrix} = \begin{bmatrix} r11 & r12 & r13 \\ r21 & r22 & r23 \\ r31 & r32 & r33 \end{bmatrix} * \begin{bmatrix} x_{P2\_world\_offset\_cam1} \\ y_{P2\_world\_offset\_cam1} \\ z_{P2\_world\_offset\_cam1} \end{bmatrix} \quad (7.88)$$

$$\begin{bmatrix} x_{P3\_world\_offset\_cam2} \\ y_{P3\_world\_offset\_cam2} \\ z_{P3\_world\_offset\_cam2} \end{bmatrix} = \begin{bmatrix} r11 & r12 & r13 \\ r21 & r22 & r23 \\ r31 & r32 & r33 \end{bmatrix} * \begin{bmatrix} x_{P3\_world\_offset\_cam1} \\ y_{P3\_world\_offset\_cam1} \\ z_{P3\_world\_offset\_cam1} \end{bmatrix} \quad (7.89)$$

Valorile parametrilor  $r11, r12$  și  $r13$  sunt determinate din sistemul de ecuații (7. 90).

$$\left\{ \begin{array}{l} x_{P1\_world\_offset\_cam2} = \begin{pmatrix} r11 * x_{P1\_world\_offset\_cam1} + \\ + r12 * y_{P1\_world\_offset\_cam1} + \\ + r13 * z_{P1\_world\_offset\_cam1} \end{pmatrix} \\ x_{P2\_world\_offset\_cam2} = \begin{pmatrix} r11 * x_{P2\_world\_offset\_cam1} + \\ + r12 * y_{P2\_world\_offset\_cam1} + \\ + r13 * z_{P2\_world\_offset\_cam1} \end{pmatrix} \\ x_{P3\_world\_offset\_cam2} = \begin{pmatrix} r11 * x_{P3\_world\_offset\_cam1} + \\ + r12 * y_{P3\_world\_offset\_cam1} + \\ + r13 * z_{P3\_world\_offset\_cam1} \end{pmatrix} \end{array} \right. \quad (7.90)$$

Valorile parametrilor  $r21$ ,  $r22$  și  $r23$  sunt determinate din sistemul de ecuații (7.91).

$$\left\{ \begin{array}{l} y_{P1\_world\_offset\_cam2} = \begin{pmatrix} r21 * x_{P1\_world\_offset\_cam1} + \\ + r22 * y_{P1\_world\_offset\_cam1} + \\ + r23 * z_{P1\_world\_offset\_cam1} \end{pmatrix} \\ y_{P2\_world\_offset\_cam2} = \begin{pmatrix} r21 * x_{P2\_world\_offset\_cam1} + \\ + r22 * y_{P2\_world\_offset\_cam1} + \\ + r23 * z_{P2\_world\_offset\_cam1} \end{pmatrix} \\ y_{P3\_world\_offset\_cam2} = \begin{pmatrix} r21 * x_{P3\_world\_offset\_cam1} + \\ + r22 * y_{P3\_world\_offset\_cam1} + \\ + r23 * z_{P3\_world\_offset\_cam1} \end{pmatrix} \end{array} \right. \quad (7.91)$$

Valorile parametrilor  $r31$ ,  $r32$  și  $r33$  sunt determinate din sistemul de ecuații (7.92).

$$\left\{ \begin{array}{l} z_{P1\_world\_offset\_cam2} = \begin{pmatrix} r31 * x_{P1\_world\_offset\_cam1} + \\ + r32 * y_{P1\_world\_offset\_cam1} + \\ + r33 * z_{P1\_world\_offset\_cam1} \end{pmatrix} \\ z_{P2\_world\_offset\_cam2} = \begin{pmatrix} r31 * x_{P2\_world\_offset\_cam1} + \\ + r32 * y_{P2\_world\_offset\_cam1} + \\ + r33 * z_{P2\_world\_offset\_cam1} \end{pmatrix} \\ z_{P3\_world\_offset\_cam2} = \begin{pmatrix} r31 * x_{P3\_world\_offset\_cam1} + \\ + r32 * y_{P3\_world\_offset\_cam1} + \\ + r33 * z_{P3\_world\_offset\_cam1} \end{pmatrix} \end{array} \right. \quad (7.92)$$

Sistemele sunt rezolvate după regula lui Cramer.

$$A = \begin{bmatrix} x_{P1\_world\_offset\_cam1} & y_{P1\_world\_offset\_cam1} & z_{P1\_world\_offset\_cam1} \\ x_{P2\_world\_offset\_cam1} & y_{P2\_world\_offset\_cam1} & z_{P2\_world\_offset\_cam1} \\ x_{P3\_world\_offset\_cam1} & y_{P3\_world\_offset\_cam1} & z_{P3\_world\_offset\_cam1} \end{bmatrix} \quad (7.93)$$

$$A1 = \begin{bmatrix} x_{P1\_world\_offset\_cam2} & y_{P1\_world\_offset\_cam1} & z_{P1\_world\_offset\_cam1} \\ x_{P2\_world\_offset\_cam2} & y_{P2\_world\_offset\_cam1} & z_{P2\_world\_offset\_cam1} \\ x_{P3\_world\_offset\_cam2} & y_{P3\_world\_offset\_cam1} & z_{P3\_world\_offset\_cam1} \end{bmatrix} \quad (7.94)$$

$$A2 = \begin{bmatrix} x_{P1\_world\_offset\_cam1} & x_{P1\_world\_offset\_cam2} & z_{P1\_world\_offset\_cam1} \\ x_{P2\_world\_offset\_cam1} & x_{P2\_world\_offset\_cam2} & z_{P2\_world\_offset\_cam1} \\ x_{P3\_world\_offset\_cam1} & x_{P3\_world\_offset\_cam2} & z_{P3\_world\_offset\_cam1} \end{bmatrix} \quad (7.95)$$

$$A3 = \begin{bmatrix} x_{P1\_world\_offset\_cam1} & y_{P1\_world\_offset\_cam1} & x_{P1\_world\_offset\_cam2} \\ x_{P2\_world\_offset\_cam1} & y_{P2\_world\_offset\_cam1} & x_{P2\_world\_offset\_cam2} \\ x_{P3\_world\_offset\_cam1} & y_{P3\_world\_offset\_cam1} & x_{P3\_world\_offset\_cam2} \end{bmatrix} \quad (7.96)$$

$$A4 = \begin{bmatrix} y_{P1\_world\_offset\_cam2} & y_{P1\_world\_offset\_cam1} & z_{P1\_world\_offset\_cam1} \\ y_{P2\_world\_offset\_cam2} & y_{P2\_world\_offset\_cam1} & z_{P2\_world\_offset\_cam1} \\ y_{P3\_world\_offset\_cam2} & y_{P3\_world\_offset\_cam1} & z_{P3\_world\_offset\_cam1} \end{bmatrix} \quad (7.97)$$

$$A5 = \begin{bmatrix} x_{P1\_world\_offset\_cam1} & y_{P1\_world\_offset\_cam2} & z_{P1\_world\_offset\_cam1} \\ x_{P2\_world\_offset\_cam1} & y_{P2\_world\_offset\_cam2} & z_{P2\_world\_offset\_cam1} \\ x_{P3\_world\_offset\_cam1} & y_{P3\_world\_offset\_cam2} & z_{P3\_world\_offset\_cam1} \end{bmatrix} \quad (7.98)$$

$$A6 = \begin{bmatrix} x_{P1\_world\_offset\_cam1} & y_{P1\_world\_offset\_cam1} & y_{P1\_world\_offset\_cam2} \\ x_{P2\_world\_offset\_cam1} & y_{P2\_world\_offset\_cam1} & y_{P2\_world\_offset\_cam2} \\ x_{P3\_world\_offset\_cam1} & y_{P3\_world\_offset\_cam1} & y_{P3\_world\_offset\_cam2} \end{bmatrix} \quad (7.99)$$

$$A7 = \begin{bmatrix} z_{P1\_world\_offset\_cam2} & y_{P1\_world\_offset\_cam1} & z_{P1\_world\_offset\_cam1} \\ z_{P2\_world\_offset\_cam2} & y_{P2\_world\_offset\_cam1} & z_{P2\_world\_offset\_cam1} \\ z_{P3\_world\_offset\_cam2} & y_{P3\_world\_offset\_cam1} & z_{P3\_world\_offset\_cam1} \end{bmatrix} \quad (7.100)$$

$$A8 = \begin{bmatrix} x_{P1\_world\_offset\_cam1} & z_{P1\_world\_offset\_cam2} & z_{P1\_world\_offset\_cam1} \\ x_{P2\_world\_offset\_cam1} & z_{P2\_world\_offset\_cam2} & z_{P2\_world\_offset\_cam1} \\ x_{P3\_world\_offset\_cam1} & z_{P3\_world\_offset\_cam2} & z_{P3\_world\_offset\_cam1} \end{bmatrix} \quad (7.101)$$

$$A9 = \begin{bmatrix} x\_P1\_world\_offset\_cam1 & y\_P1\_world\_offset\_cam1 & z\_P1\_world\_offset\_cam2 \\ x\_P2\_world\_offset\_cam1 & y\_P2\_world\_offset\_cam1 & z\_P2\_world\_offset\_cam2 \\ x\_P3\_world\_offset\_cam1 & y\_P3\_world\_offset\_cam1 & z\_P3\_world\_offset\_cam2 \end{bmatrix} \quad (7. 102)$$

Valorile elementelor, ce constituie matricea de rotație, sunt determinate prin relațiile (7. 103) ÷ (7. 111). Matricea de rotație rezultată este exprimată prin relația (7. 112).

$$r11 = \frac{\det A1}{\det A} \quad (7. 103) \quad r12 = \frac{\det A2}{\det A} \quad (7. 104) \quad r13 = \frac{\det A3}{\det A} \quad (7. 105)$$

$$r21 = \frac{\det A4}{\det A} \quad (7. 106) \quad r22 = \frac{\det A5}{\det A} \quad (7. 107) \quad r23 = \frac{\det A6}{\det A} \quad (7. 108)$$

$$r31 = \frac{\det A7}{\det A} \quad (7. 109) \quad r32 = \frac{\det A8}{\det A} \quad (7. 110) \quad r33 = \frac{\det A9}{\det A} \quad (7. 111)$$

$$M\_Rotatie = \begin{bmatrix} r11 & r12 & r13 \\ r21 & r22 & r23 \\ r31 & r32 & r33 \end{bmatrix} \quad (7. 112)$$

### Etapa8

Relația (7. 113) este utilizată pentru determinarea valorilor vectorului de translație. Vectorul de translație este exprimat prin relația (7. 114).

$$\begin{bmatrix} t\_x \\ t\_y \\ t\_z \end{bmatrix} = \begin{bmatrix} x\_P1\_world\_cam2 \\ y\_P1\_world\_cam2 \\ z\_P1\_world\_cam2 \end{bmatrix} - \begin{bmatrix} r11 & r12 & r13 \\ r21 & r22 & r23 \\ r31 & r32 & r33 \end{bmatrix} * \begin{bmatrix} x\_P1\_world\_cam1 \\ y\_P1\_world\_cam1 \\ z\_P1\_world\_cam1 \end{bmatrix} \quad (7. 113)$$

$$M\_Translatie = \begin{bmatrix} t\_x \\ t\_y \\ t\_z \end{bmatrix} \quad (7. 114)$$

Calibrarea extrinsecă constă în determinarea matricei de rotație și a vectorului de translație. Acest obiectiv a fost atins prin parcurgerea celor opt etape prezentate anterior.

### 7.2.3 Descrierea experimentului

În experiment au fost utilizate patru obiecte de calibrare de formă circulară. Centrele celor patru cercuri trebuie să fie necoplanare. Distanțele dintre centrele cercurilor sunt necunoscute. În experimentul realizat, pentru fiecare dintre cercuri dimensiunea diametrului este 176 mm. Algoritmii de calibrare au fost implementați în mediul de programare MATLAB. Sunt prelevate imagini de la cele două camere video.



Fig. 7. 6. Imagine prelevată de la camera 1 (a); imagine prelevată de la camera 2 (b)

În Fig. 7. 7 este prezentată prima pagină a interfeței grafice, pagină ce corespunde programului START.fig.

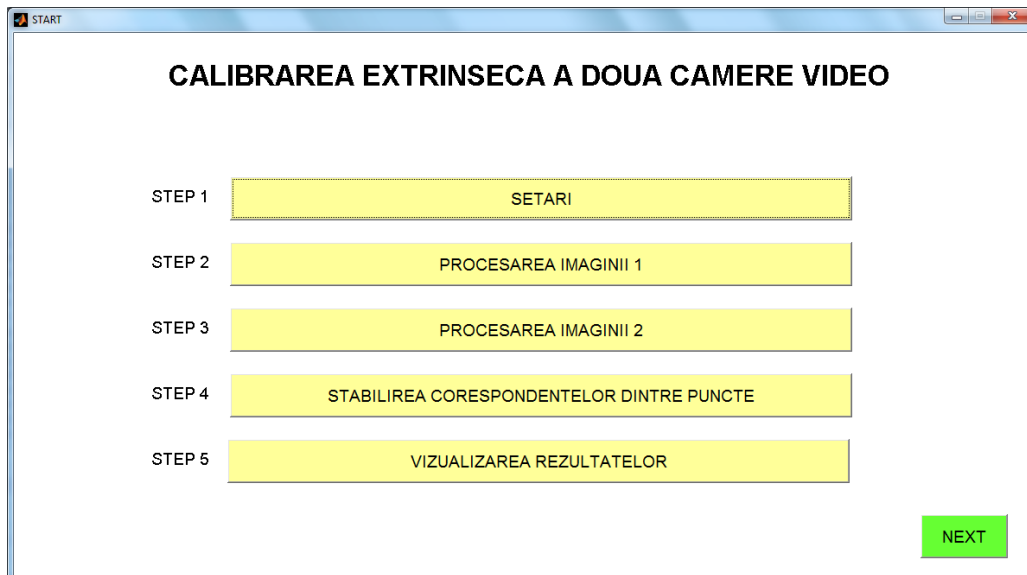


Fig. 7. 7. Interfață grafică - START

În Fig. 7. 8 este prezentată a doua pagină a interfeței grafice, pagină ce corespunde programului STEP1.fig. În această interfață grafică sunt setate valorile distanțelor focale, valorile diametrelor cercurilor și tipul camerei.

Fig. 7. 8. Interfața grafică – STEP1

În Fig. 7. 9 este prezentată a treia pagină a interfeței grafice, pagină ce corespunde programului STEP2.fig. În această interfață grafică este procesată imaginea color a camerei 1 de tip 2D, sau sunt selectate centrele cercurilor în imaginea camerei 1 de tip 3D-ToF.

Fig. 7. 9. Interfața grafică – STEP2

În Fig. 7. 10 este prezentată a patra pagină a interfeței grafice, pagină ce corespunde programului STEP3.fig. În această interfață grafică este procesată imaginea color a camerei 2 de tip 2D, sau sunt selectate centrele cercurilor în imaginea camerei 2 de tip 3D-ToF.

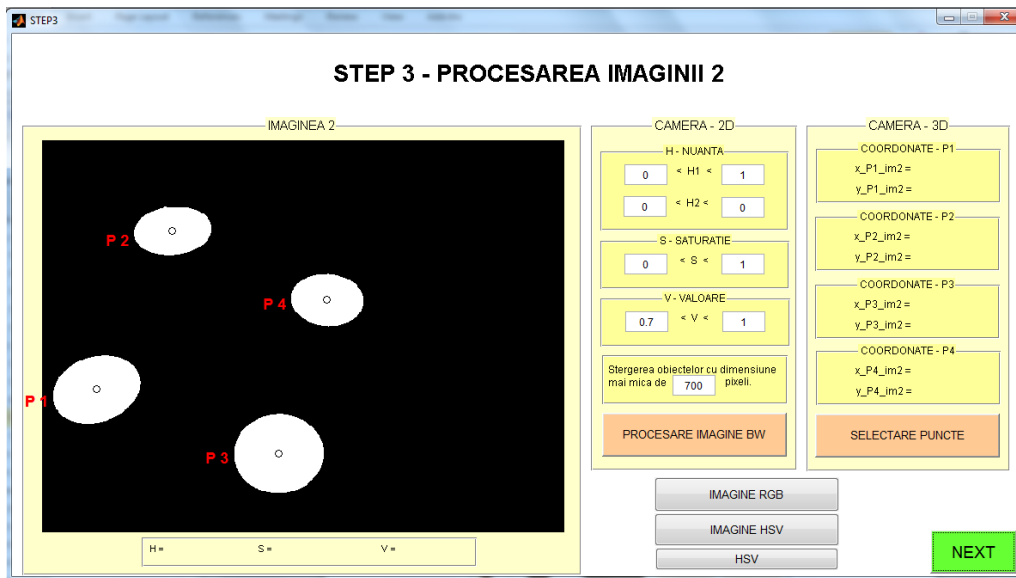


Fig. 7. 10. Interfața grafică – STEP3

În Fig. 7. 11 și Fig. 7. 12 este prezentată a cincea pagină a interfeței grafice, pagină ce corespunde programului STEP4.fig. În această interfață grafică sunt stabilite corespondențele dintre punctele din imaginea 1 și punctele din imaginea 2.



Fig. 7. 11. Corespondență puncte - STEP4 (camera 1)





Fig. 7. 12. Corespondență puncte - STEP4 (camera 2)

În Fig. 7. 13 este prezentată a șasea pagină a interfeței grafice, pagină ce corespunde programului STEP5.fig. În această interfață grafică sunt afișate rezultatele calibrării extrinseci; matricea de rotație și vectorul de translație.

În anexa C2 este prezentată ordinograma programului MATLAB și câteva secvențe de cod.

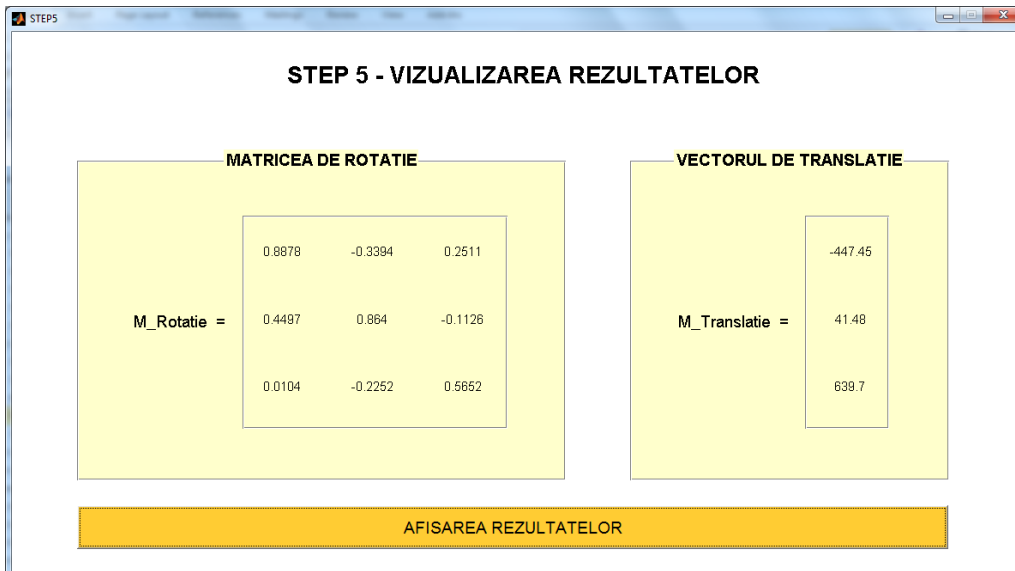


Fig. 7. 13. Rezultatele calibrării extrinseci

### 7.2.4 Rezultatele experimentului

În Tab. 7. 10 sunt prezentate tipurile de camere utilizate în experiment și caracteristicile principale ale acestora. În dreptul fiecărui tip de cameră video este menționată dimensiunea imaginii (notată cu  $size\_im$ ), distanța focală pe axa x (notată cu  $f\_x$ ) și distanța focală pe axa y (notată cu  $f\_y$ ).

Tipul camerei	size_im [pixel] x [pixel]	f_x [pixel]	f_y [pixel]
Logitech C160	640 x 480	695,67270	695,19737
Logitech C160	640 x 480	682,18133	682,65164
3D-ToF pmd-ifm	50 x 64	86,20690	76,19048

Tab. 7. 10. Tipurile de camere și caracteristicile principale

Precizia rezultatelor calibrării extrinseci este verificată prin utilizarea unor algoritmi de corelare a imaginilor (*camera-matching*). În Fig. 7. 14 este prezentată o pagină din interfața grafică a programului prin care este verificată precizia calibrării extrinseci. În prima coloană a panoului *REZULTATE* sunt afișate valorile coordonatelor centrelor cercurilor în imaginea 1. În a doua coloană sunt afișate valorile coordonatelor cercurilor în imaginea 2. În a treia coloană sunt afișate valorile coordonatelor punctelor în imaginea 2, obținute prin algoritmi de corelare a imaginilor (*camera-matching*). În urma procedurii de calibrare extrinsecă sunt obținute matricea de rotație și vectorul de translație, așa cum este prezentat în Fig. 7. 13. Matricea de rotație și vectorul de translație sunt parametri de intrare pentru algoritmi de *camera-matching*. Ideal este ca valorile din coloana a treia să coincidă cu valorile din coloana a doua. În coloana a patra sunt afișate erorile.



Fig. 7. 14. Verificarea preciziei de calibrare extrinsec

### 7.3 Calibrare extrinsecă utilizând modele CAD

Având două camere web și prelevând două imagini din unghiuri diferite, se obțin imaginile webcam1.jpg și webcam2.jpg.

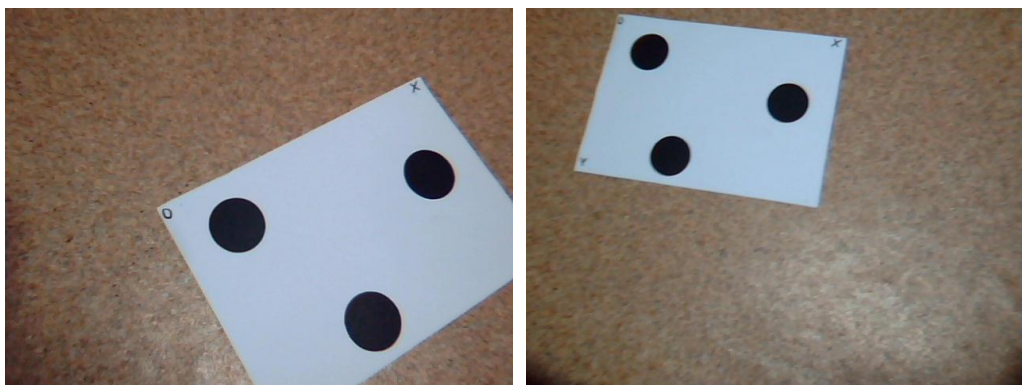


Fig. 7. 15. Imagini prelevate de la camerele video 1 și 2

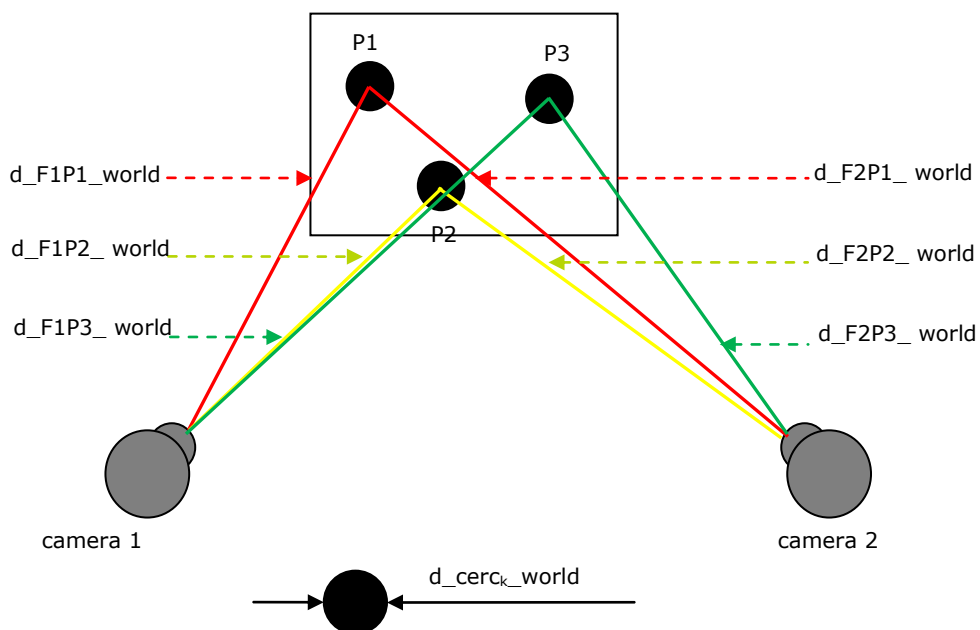


Fig. 7. 16. Calibrarea extrinsecă a camerelor video utilizând trei cercuri de dimensiune cunoscută

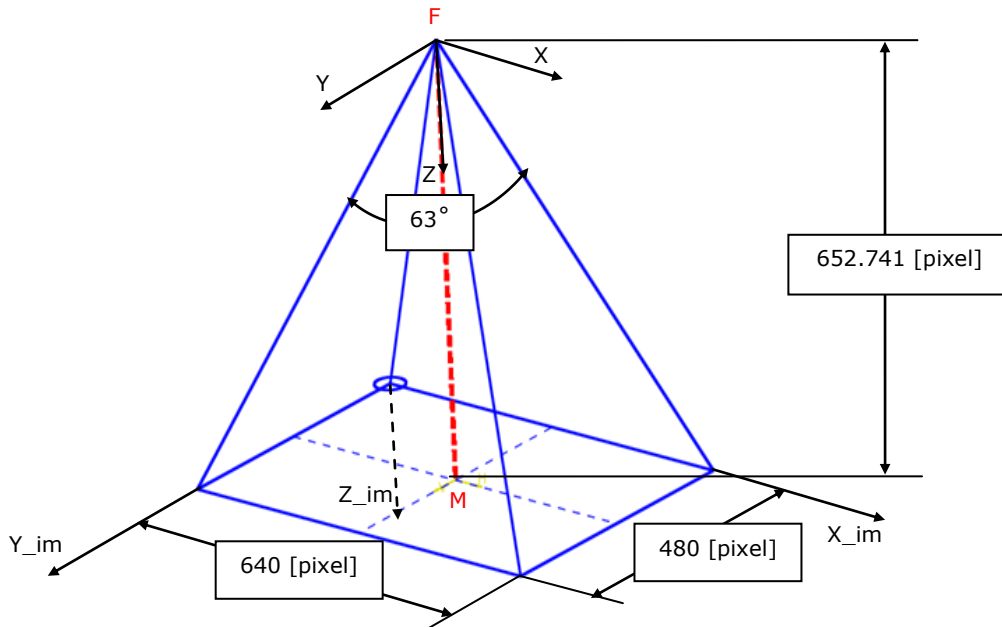


Fig. 7. 17. Câmpul vizual al camerei Logitech C200

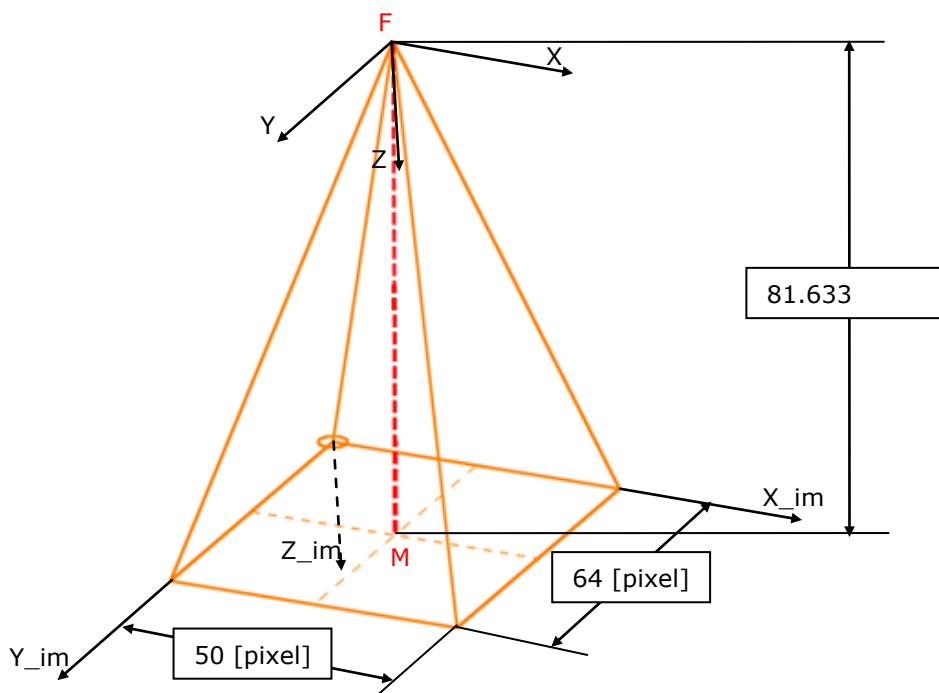


Fig. 7. 18. Câmpul vizual al camerei 3D-ToF pmd-ifm

### 7.3 - Calibrare extrinsecă utilizând modele CAD109

Metoda se bazează pe principiul cunoașterii distanțelor dintre focarul camerei web și centrul fiecărui disc din imagine. Fiecare din dreptele corespunzătoare distanțelor dintre focarul camerei1 și punct, este concurrentă cu dreapta corespunzătoare distanței dintre focarul camerei2 și punct. Prin procesarea imaginilor webcam1.jpg și webcam2.jpg, utilizând programul MATLAB, se obțin coordonatele centrelor cercurilor, și distanțele în mm de la punctul focal F până la centrele cercurilor.

$$k = \{1,2,3\} \quad (7.115)$$

$$Pk = \{P1, P2, P3\} \quad (7.116)$$

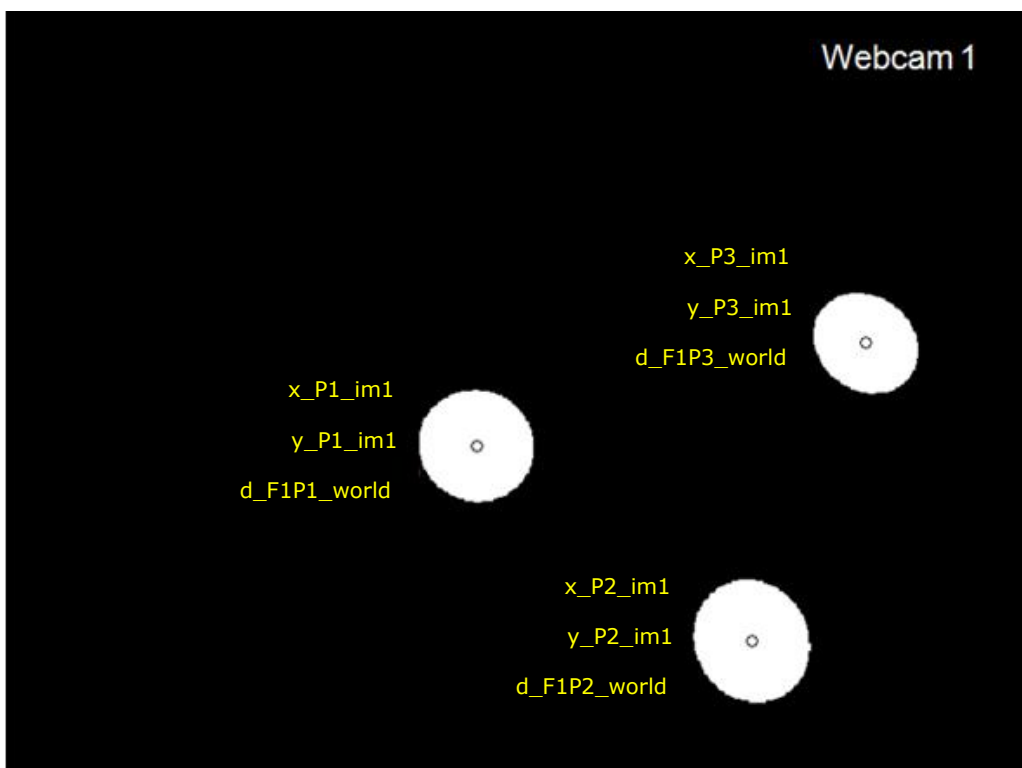


Fig. 7. 19. Procesarea imaginii 1

$$Pk\_im1 = [x\_Pk\_im1 \ y\_Pk\_im1 \ 0]^T \quad (7.117)$$

$$d\_F1Pk\_im1 = \sqrt{(x\_Pk\_im1 - x\_M\_im1)^2 + (y\_Pk\_im1 - y\_M\_im1)^2 + (f1)^2} \quad (7.118)$$

$$x\_Pk\_im1\_cam1 = x\_Pk\_im1 - x\_M\_im1 \quad (7.119)$$

### 110 Calibrarea camerelor video - 7

$$y\_Pk\_im1\_cam1 = y\_Pk\_im1 - y\_M\_im1 \quad (7.120)$$

$$z\_Pk\_im1\_cam1 = f1 \quad (7.121)$$

În cazul utilizării camerei 2D:

$$sk\_cam1 = \frac{d\_cerck\_world}{d\_elipsak\_im1} \quad (7.122)$$

În cazul utilizării camerei 3D-ToF:

$$sk\_cam1 = \frac{d\_F1Pk\_world}{d\_F1Pk\_im1} \quad (7.123)$$

$$x\_Pk\_world\_cam1 = sk\_cam1 * x\_Pk\_im1\_cam1 \quad (7.124)$$

$$y\_Pk\_world\_cam1 = sk\_cam1 * y\_Pk\_im1\_cam1 \quad (7.125)$$

$$z\_Pk\_world\_cam1 = sk\_cam1 * z\_Pk\_im1\_cam1 \quad (7.126)$$



Fig. 7. 20. Procesarea imaginii 2

$$Pk\_im2 = [x\_Pk\_im2 \ y\_Pk\_im2 \ 0]^T \quad (7.127)$$

$$d\_F2Pk\_im2 = \sqrt{(x\_Pk\_im2 - x\_M\_im2)^2 + (y\_Pk\_im2 - y\_M\_im2)^2 + (f2)^2} \quad (7.128)$$

$$x\_Pk\_im2\_cam2 = x\_Pk\_im2 - x\_M\_im2 \quad (7.129)$$

$$y\_Pk\_im2\_cam2 = y\_Pk\_im2 - y\_M\_im2 \quad (7.130)$$

$$z\_Pk\_im2\_cam2 = f2 \quad (7.131)$$

În cazul utilizării camerei 2D:

$$sk\_cam2 = \frac{d\_cerck\_world}{d\_elipsak\_im2} \quad (7.132)$$

În cazul utilizării camerei 3D-ToF:

$$sk\_cam2 = \frac{d\_F2Pk\_world}{d\_F2Pk\_im2} \quad (7.133)$$

$$x\_Pk\_world\_cam2 = sk\_cam2 * x\_Pk\_im2\_cam2 \quad (7.134)$$

$$y\_Pk\_world\_cam2 = sk\_cam2 * y\_Pk\_im2\_cam2 \quad (7.135)$$

$$z\_Pk\_world\_cam2 = sk\_cam2 * z\_Pk\_im2\_cam2 \quad (7.136)$$

Câmpurile vizuale ale camerelor video (Fig. 7. 17 și Fig. 7. 18), precum și distanțele de la focarul fiecărei camere până la fiecare din centrele cercurilor, sunt modelate într-un program CAD. În această aplicație a fost utilizat programul CATIA. Într-un fișier de ansamblu se pot crea constrângeri prin fixarea modelelor distanțelor în punctele corespunzătoare modelelor centrelor cercurilor. Prin acest procedeu, valorile parametrilor extrinseci pot fi măsurate în CATIA (Fig. 7. 21).

$\alpha = ?$  - valoare măsurată în CATIA

$\beta = ?$  - valoare măsurată în CATIA

$\gamma = ?$  - valoare măsurată în CATIA

$T_x = ?$  - valoare măsurată în CATIA

$T_y = ?$  - valoare măsurată în CATIA

$T_z = ?$  - valoare măsurată în CATIA

## 112 Calibrarea camerelor video - 7

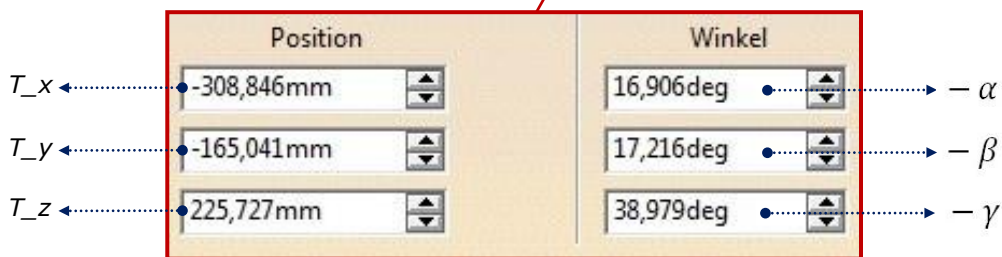
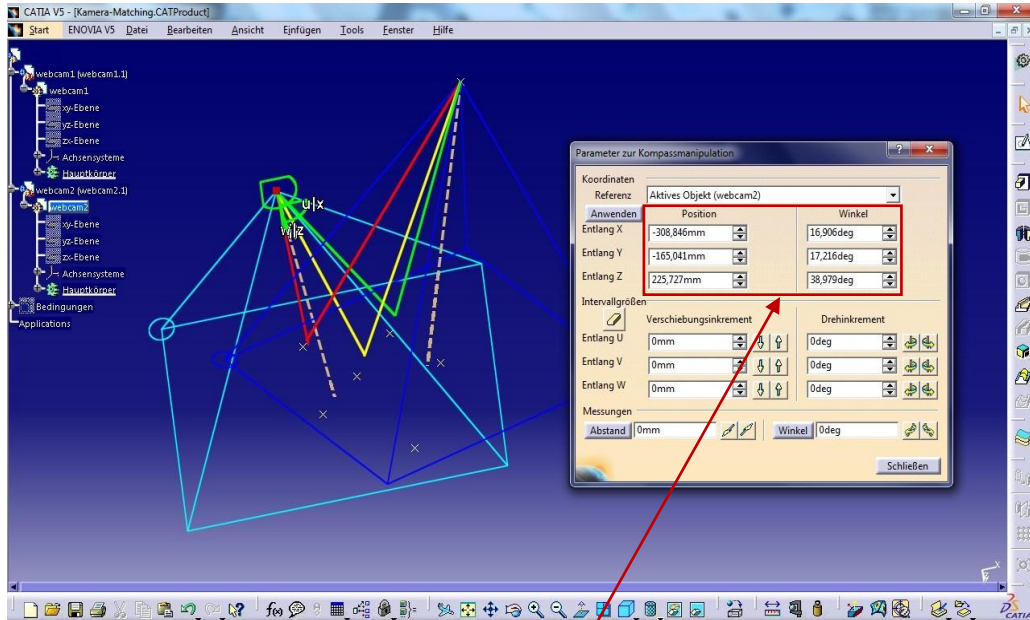


Fig. 7. 21. Măsurarea parametrilor extrinseci în programul CATIA [Tilneac, 2011a]

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \quad (7.137)$$

$$R_y = \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix} \quad (7.138)$$

$$R_z = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (7.139)$$



$$M\_Rotatie = (R\_x * R\_y * R\_z)^{-1} \quad (7.140)$$

$$M\_Translatie = [T\_x \ T\_y \ T\_z]^T \quad (7.141)$$

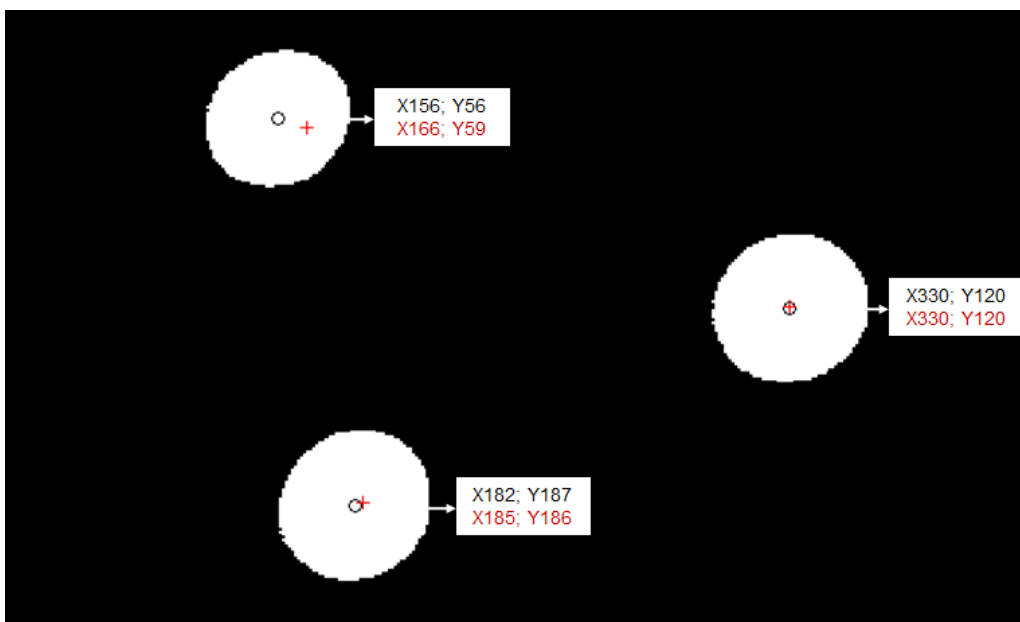


Fig. 7. 22. Verificarea preciziei de calibrare

## 7.4 Concluzii

Fiecare dintre metodele de calibrare a camerelor video, prezentate în acest capitol, utilizează obiecte de calibrare de formă circulară.

Metoda de calibrare	Numărul de cercuri
Calibrare intrinsecă	2 cercuri
Calibrare extrinsecă cu modele CAD	3 cercuri
Calibrare extrinsecă prin metodă analitică	4 cercuri

Metodele de calibrare sunt simple și ușor de aplicat. Datorită faptului că nu necesită o instalație de calibrare, prin utilizarea metodelor analitice calibrarea poate fi realizată cu succes direct pe terenurile agricole. Precizia de calibrare este relativ scăzută pentru a fi utilizată în aplicații industriale, dar este suficient de ridicată pentru aplicații în agricultură.

Metoda de calibrare prin modele CAD poate fi utilizată pentru verificarea rezultatelor obținute prin metodele analitice. Fig. 7. 23÷Fig. 7. 25 prezintă modelele

CAD ale camerelor 3D-ToF și Logitech C200 utilizate în procesul de calibrare extrinsecă [Tilneac, 2011a].

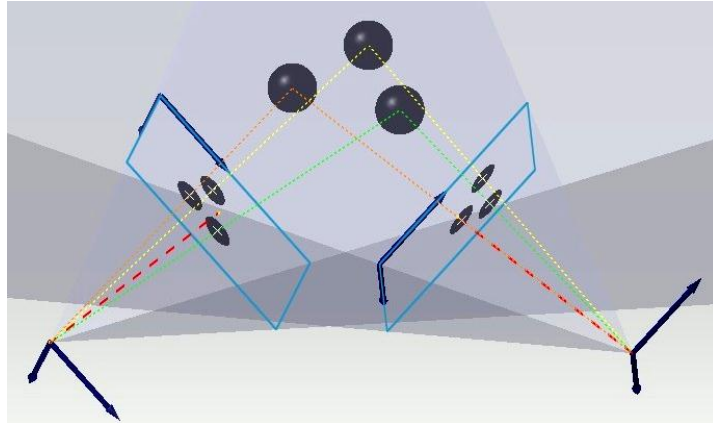


Fig. 7. 23. Calibrarea extrinsecă a camerelor video Logitech C200 în mediul CATIA V5

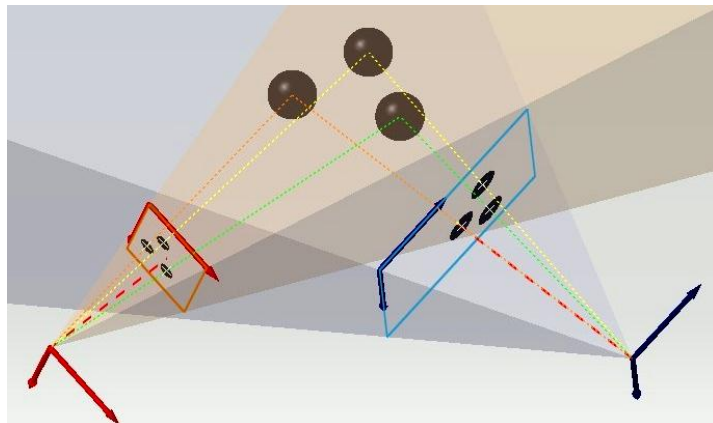


Fig. 7. 24. Calibrarea extrinsecă a camerelor 3D-ToF și Logitech C200 în mediul CATIA V5

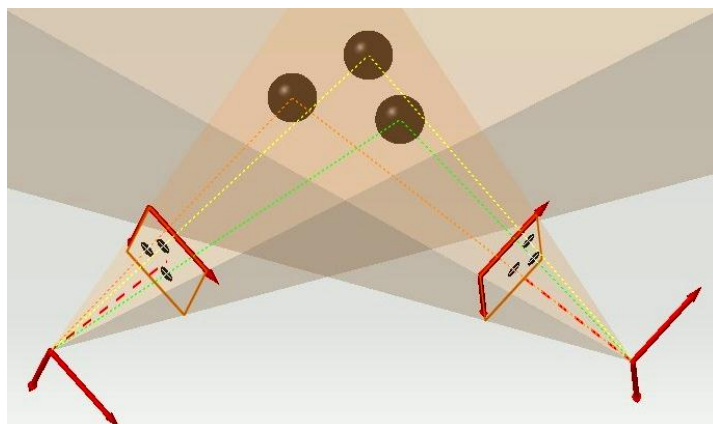


Fig. 7. 25. Calibrarea extrinsecă a camerelor video 3D-ToF în mediul CATIA V5

## 8 Diferențierea buruiană/plantă pe bază de măsurări optice tridimensionale

### 8.1 Introducere

Acest capitol prezintă rezultatele unor cercetări în domeniul vederii artificiale cu aplicabilitate în domeniul roboților pentru agricultură. Sistemul vizual conține două camere web care formează un sistem stereoscopic binocular. Diferențierea buruiană/plantă este realizată pe baza diferențelor de înălțime între plantă și buruiană. Înălțimile plantelor sunt determinate prin procesarea perechilor de imagini achiziționate de la cele două camere. Plantele sunt selectate prin utilizarea unui filtru de înălțime. În experimentul realizat, plantele care au înălțime mai mică de 85 mm sunt considerate ca fiind buruieni, iar plantele cu înălțime mai mare de 85 mm sunt considerate ca fiind plante de cultură. Vederea stereo permite obținerea simultană de informații despre culoare și distanță. Fig.8.1 prezintă standul experimental utilizat în experimente de măsurare tridimensională prin vedere stereo.

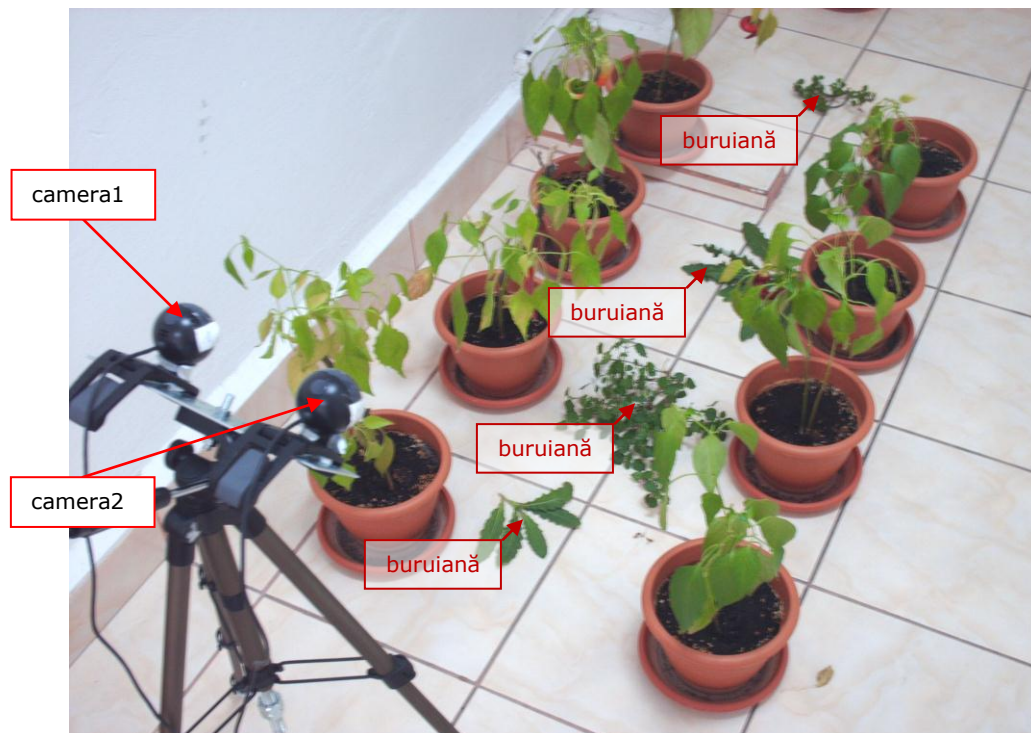


Fig. 8. 1. Stand de măsurare a înălțimilor prin vedere stereo

## 8.2 Localizarea planului solului

Înălțimea plantei este considerată a fi distanța maximă între punctele, identificate ca fiind pe suprafață plantei, și planul solului. În acest scop, trebuie să fie determinată ecuația planului solului raportat la sistemul de coordonate al camerei 1. Planul poate fi definit prin trei puncte necolineare  $P1(x1, y1, z1)$ ,  $P2(x2, y2, z2)$ , și  $P3(x3, y3, z3)$ . Ecuația planului determinat de cele trei puncte necolineare este:

$$\begin{vmatrix} x & y & z & 1 \\ x1 & y1 & z1 & 1 \\ x2 & y2 & z2 & 1 \\ x3 & y3 & z3 & 1 \end{vmatrix} = 0 \quad (8. 1)$$

Prin dezvoltarea ecuației (8.1) după prima linie a determinantului rezultă ecuația planului:

$$a * x + b * y + c * z + d = 0 \quad (8. 2)$$

unde valorile coeficienților sunt determinate prin relațiile (8.3), (8.4), (8.5), (8.6).

$$a = \begin{vmatrix} y1 & z1 & 1 \\ y2 & z2 & 1 \\ y3 & z3 & 1 \end{vmatrix} \quad (8. 3)$$

$$b = - \begin{vmatrix} x1 & z1 & 1 \\ x2 & z2 & 1 \\ x3 & z3 & 1 \end{vmatrix} \quad (8. 4)$$

$$c = \begin{vmatrix} x1 & y1 & 1 \\ x2 & y2 & 1 \\ x3 & y3 & 1 \end{vmatrix} \quad (8. 5)$$

$$d = - \begin{vmatrix} x1 & y1 & z1 \\ x2 & y2 & z2 \\ x3 & y3 & z3 \end{vmatrix} \quad (8. 6)$$

Pentru a determina coordonatele a trei puncte de pe planul solului, au fost utilizate trei cercuri cu diametrul cunoscut. Cele trei cercuri sunt amplasate pe planul solului. Punctele  $P1, P2, P3$  corespund centrelor cercurilor.

Fig.8.2 prezintă o modalitate de amplasare a cercurilor pe planul solului. Se presupun cunoscute diametrele cercurilor  $d\_cerc_k\_world$ . Din imaginea achiziționată ale celor trei cercuri se determină dimensiunea axei mari a elipselor corespunzătoare proiecțiilor cercurilor în imagine. Pe baza valorilor determinate se poate calcula factorul de scalare:

$$k = 1,2,3 \quad (8. 7)$$

$$s_k\_cam = \frac{d\_cerc_k\_world}{d\_elipsa\_im} \quad [\text{mm/pixel}] \quad (8. 8)$$

Coordonatele unui punct  $P_k$  din imagine față de sistemul de referință al camerei este determinat prin relațiile:

$$x_{P_k\_im\_cam} = (x_{P_k\_im}) - (x_{M\_im}) [\text{pixe}] \quad (8. 9)$$

$$y_{P_k\_im\_cam} = (y_{P_k\_im}) - (y_{M\_im}) [\text{pixel}] \quad (8. 10)$$

Valorile coordonatelor  $x_k, y_k, z_k$  ( $k=1,2,3$ ) din ecuația planului (8.1) sunt egale cu:

$$x_k = x_{P_k\_world} [\text{mm}] \quad (8. 11)$$

$$y_k = y_{P_k\_world} [\text{mm}] \quad (8. 12)$$

$$z_k = z_{P_k\_world} [\text{mm}] \quad (8. 13)$$

$$x_{P_k\_world} = s_{k\_cam} * (x_{P_k\_im\_cam}) [\text{mm}] \quad (8. 14)$$

$$y_{P_k\_world} = s_{k\_cam} * (y_{P_k\_im\_cam}) [\text{mm}] \quad (8. 15)$$

$$z_{P_k\_world} = s_{k\_cam} * f [\text{mm}] \quad (8. 16)$$

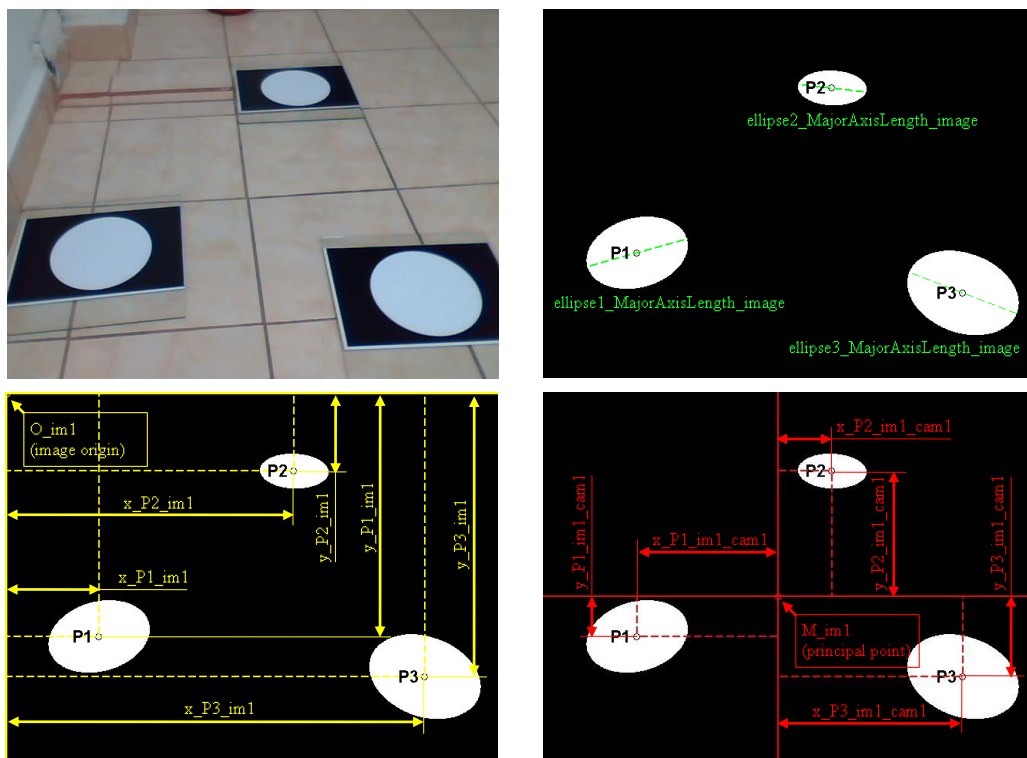


Fig. 8. 2. Procesarea imaginii în mediul MATLAB

## 118 Diferențierea buruiană/plantă pe bază de măsurări optice tridimensionale - 8

Pentru verificarea metodologiei și evaluarea corectitudinii ipotezelor, am considerat trei cercuri cu diametrul 176 mm. Pentru realizarea calculului am dezvoltat în mediul MATLAB un program denumit *ECUATIA\_PLANULUI*. Interfețele grafice ale programului sunt prezentate în Fig. 8. 3 și Fig. 8. 4.

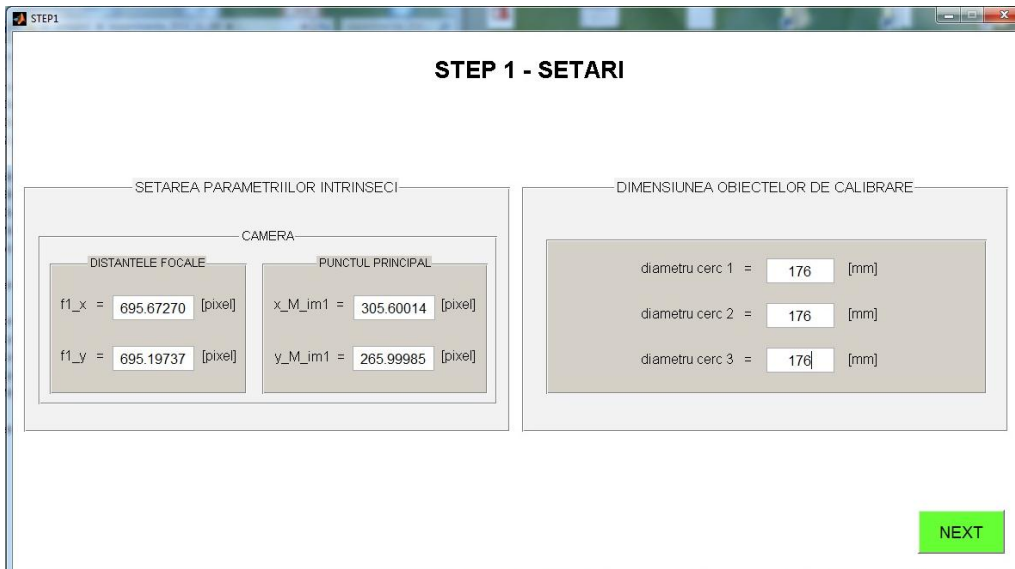


Fig. 8. 3. Interfața grafică de setări a programului *ECUATIA\_PLANULUI*

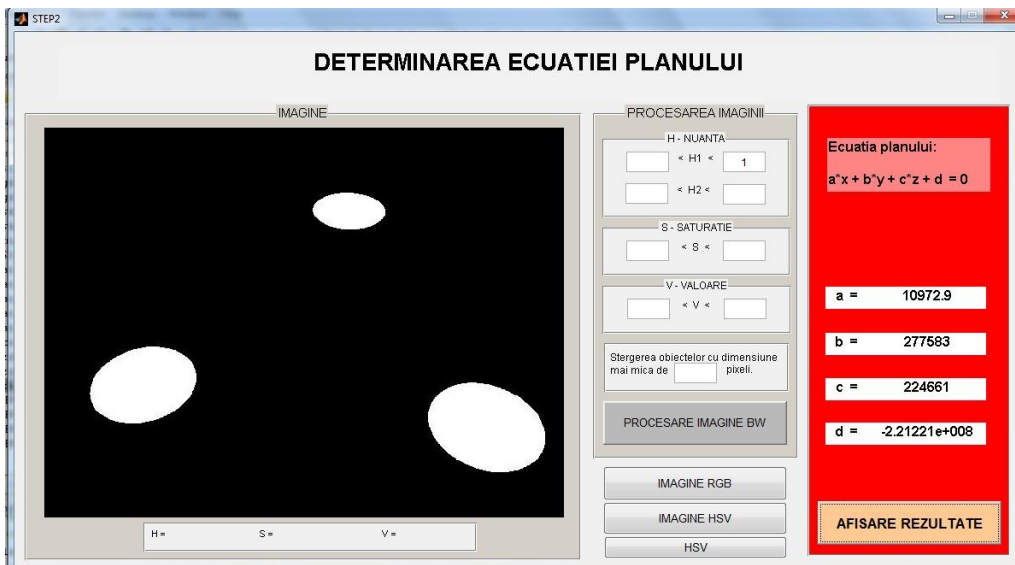


Fig. 8. 4. Interfața grafică pentru procesarea imaginii și afișarea valorilor coeficienților corespunzători ecuației planului

Imaginea celor trei cercuri este importată în folder-ul programului numit „image”. Există două modalități de lansare a programului: lansare din mediul MATLAB a

fișierului STEP1.m sau lansarea unei aplicații independente în format .exe care are denumirea *ECUATIA\_PLANULUI*.





După lansarea programului este afișată pe monitor interfața grafică prezentată în Fig. 8. 3. În fereastra grafică sunt introduse valorile parametrilor de intrare:

- distanțele focale corespunzătoare câmpului vizual orizontal, respectiv vertical ( $f_x = 695,67$  respectiv  $f_y = 695,19$ );
- coordonatele punctului principal al imaginii ( $x_{M\_im} = 305,60$  respectiv  $y_{M\_im} = 265,99$ );
- dimensiunile cercurilor exprimate în mm ( $d_{cerc_k\_world} = 176$  mm unde  $k = \{1,2,3\}$ )

Setarea filtrelor de culoare este realizată prin interfața grafică prezentată în Fig. 8. 4. În programul *ECUATIA\_PLANULUI* sunt implementați algoritmi (8. 1)÷(8. 16).

### 8.3 Determinarea ecuației planului pe baza mai multor imagini

Precizia de determinare a ecuației planului poate fi ridicată prin utilizarea mai multor imagini (Tab. 8. 1) ale celor trei cercuri amplasate în locuri diferite. În acest subcapitol este prezentat un procedeu de calcul bazat pe media aritmetică a valorilor absolute ale coeficienților omologi corespunzători ecuațiilor planelor rezultate prin procesarea fiecărei imagini. Urmează a fi prezentat un exemplu în care este utilizat acest procedeu. În experiment au fost utilizate patru imagini a câte trei cercuri amplasate pe planul solului.

Imagine 1	Imagine2	Imagine3	Imagine4
a1 = -3.0802e+003	a2 = 1.0574e+004	a3 = 1.7997e+004	a4 = -3.3669e+003
b1 = -2.4170e+005	b2 = 2.7200e+005	b3 = 1.6436e+005	b4 = -2.4932e+005
c1 = -2.3068e+005	c2 = 2.2202e+005	c3 = 1.4349e+005	c4 = -2.2781e+005
d1 = 2.3249e+008	d2 = -2.1736e+008	d3 = -1.4322e+008	d4 = 2.2960e+008
			

Tab. 8. 1. Determinarea ecuației planului pe baza mai multor imagini

$$|a\_mediu| = \frac{|a1| + |a2| + |a3| + |a4|}{4} \quad (8. 17)$$

$$|b\_mediu| = \frac{|b1| + |b2| + |b3| + |b4|}{4} \quad (8. 18)$$

$$|c\_mediu| = \frac{|c1| + |c2| + |c3| + |c4|}{4} \quad (8. 19)$$

$$|d\_mediu| = \frac{|d1| + |d2| + |d3| + |d4|}{4} \quad (8. 20)$$

Rezultate	
<p><i>Varianta 1</i></p> <p><math>a\_mediu = \text{sign}(a1) *  a\_mediu </math></p> <p><math>b\_mediu = \text{sign}(b1) *  b\_mediu </math></p> <p><math>c\_mediu = \text{sign}(c1) *  c\_mediu </math></p> <p><math>d\_mediu = \text{sign}(d1) *  d\_mediu </math></p>	<p><math>a\_mediu = -8.7545e+003</math></p> <p><math>b\_mediu = -231845</math></p> <p><math>c\_mediu = -206000</math></p> <p><math>d\_mediu = 205667500</math></p>
<p><i>Varianta 2</i></p> <p><math>a\_mediu = -\text{sign}(a1) *  a\_mediu </math></p> <p><math>b\_mediu = -\text{sign}(b1) *  b\_mediu </math></p> <p><math>c\_mediu = -\text{sign}(c1) *  c\_mediu </math></p> <p><math>d\_mediu = -\text{sign}(d1) *  d\_mediu </math></p>	<p><math>a\_mediu = 8.7545e+003</math></p> <p><math>b\_mediu = 231845</math></p> <p><math>c\_mediu = 206000</math></p> <p><math>d\_mediu = -205667500</math></p>

Tab. 8. 2. Variante de stabilire a semnelor coeficienților

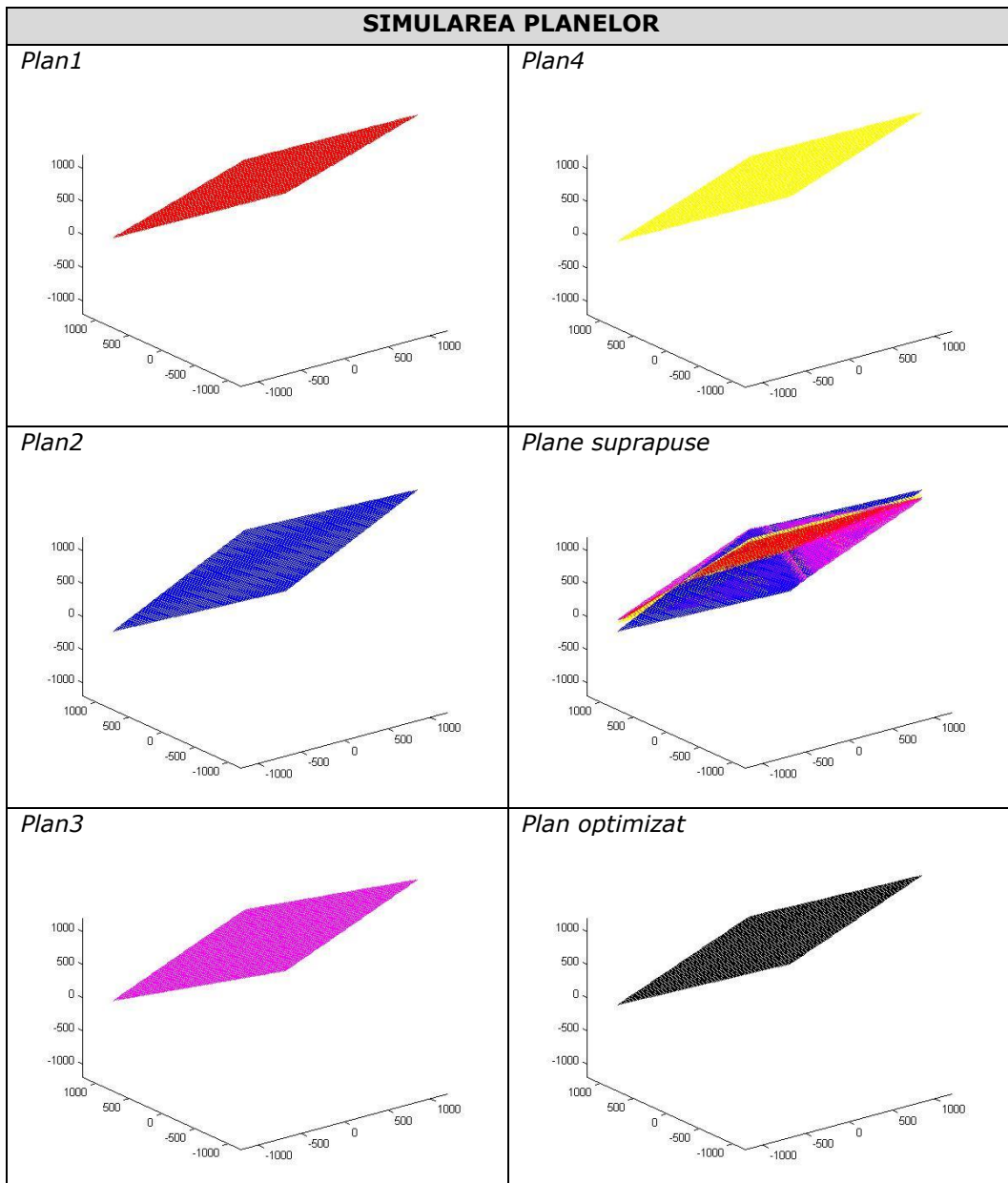
Observație!!! - Rezultatele sunt identice indiferent de varianta utilizată (varianta1 sau varianta2).

VERIFICAREA PRECIZIEI							
Punct	Eroare	Punct	Eroare	Punct	Eroare	Punct	Eroare
P11	6.6341	P21	16.4874	P31	4.2700	P41	0.5395
P12	15.0052	P22	7.8112	P32	6.4764	P42	11.1721
P13	2.5921	P23	21.8630	P33	19.4665	P43	3.1986

Tab. 8. 3. Verificarea preciziei

Eroarea maximă este 21.8630 mm





#### 8.4 Diferențierea buruiană/plantă

Plantele sunt detectate pe baza caracteristicilor de culoare și înălțime. Distanța dintre un punct (localizat pe suprafața plantei) și planul solului este determinată prin relația (8.21).  $P$  este un punct pe suprafața plantei, care are coordonatele  $(x_p, y_p, z_p)$  raportate la sistemul de referință al camerei.

$$Dist = \frac{|a * x_p + b * y_p + c * z_p + d|}{\sqrt{a^2 + b^2 + c^2}} \quad (8. 21)$$

$$Inaltime\_Planta = \max(Dist) \quad (8. 22)$$

Programul MATLAB de procesare a imaginilor are un modul principal prin care sunt apelate diferite funcții. O parte dintre funcțiile programului sunt originale, iar o parte sunt funcții *open source* disponibile pe internet. Adresa site-ului de internet, de la care au fost importante o parte din funcțiile de corelare a imaginilor, este indicată în [\*\*\* Kovesi]. Cele mai importante funcții sunt apelate din program în următoarea ordine:

1. harris.m (funcție *open source*)
2. matchbycorrelation.m (funcție *open source*)
3. ransacfitfundmatrix.m (funcție *open source*)
4. distanta\_camera\_punct.m (funcție originală)
5. distanta\_punct\_plan.m (funcție originală)
6. detectare\_planta.m (funcție originală)

Prin funcția **harris.m** sunt detectate colțurile obiectelor din imagine. Această funcție a fost dezvoltată pe baza metodei publicate de către Harris și Stephens în anul 1988 [Harris, 1988]. Funcțiile **matchbycorrelation.m** și **ransacfitfundmatrix.m** sunt utilizate la corelarea punctelor omoloage din cele două imagini. Funcția **distanta\_camera\_punct.m** calculează distanța de la camera1 până la un punct am amplasat în spațiul 3D al scenei. Funcția **distanta\_punct\_plan.m** calculează distanța de la un punct, amplasat în planul 3D al scenei, până la planul solului. Funcția **detectare\_planta.m** detectează obiectele verzi din imagine, utilizând un filtru de culoare.

Detectarea plantelor (plante cultivate și buruieni) este realizată prin segmentarea imaginii și filtrarea obiectelor de culoare verde. Diferențierea buruiană-plantă este realizată prin utilizarea unui filtru de înălțime. Principalele etape de procesare a imaginilor sunt prezentate în Fig. 8. 5. Rezultatele experimentului de diferențiere a buruienilor de plantele cultivate sunt prezentate în Fig. 8. 6.

În anexa D este prezentată ordinograma programului MATLAB și câteva secvențe de cod.

## 8.5 Concluzii

Rezultatele experimentelor executate în laborator arată că poate fi discriminată planta de buruiană prin utilizarea unui prag de separare a înălțimilor. Metoda poate fi utilă doar atunci când există o diferență semnificativă între înălțimile plantelor de cultură și buruieni. Este necesar ca în viitor să fie investigate tehnici de scădere a timpului de procesare a imaginilor, pentru a permite sistemului să funcționeze în timp real. De asemenea, programul de corelare a imaginilor sistemului binocular necesită îmbunătățiri, pentru găsirea unui număr mai ridicat de puncte omoloage.

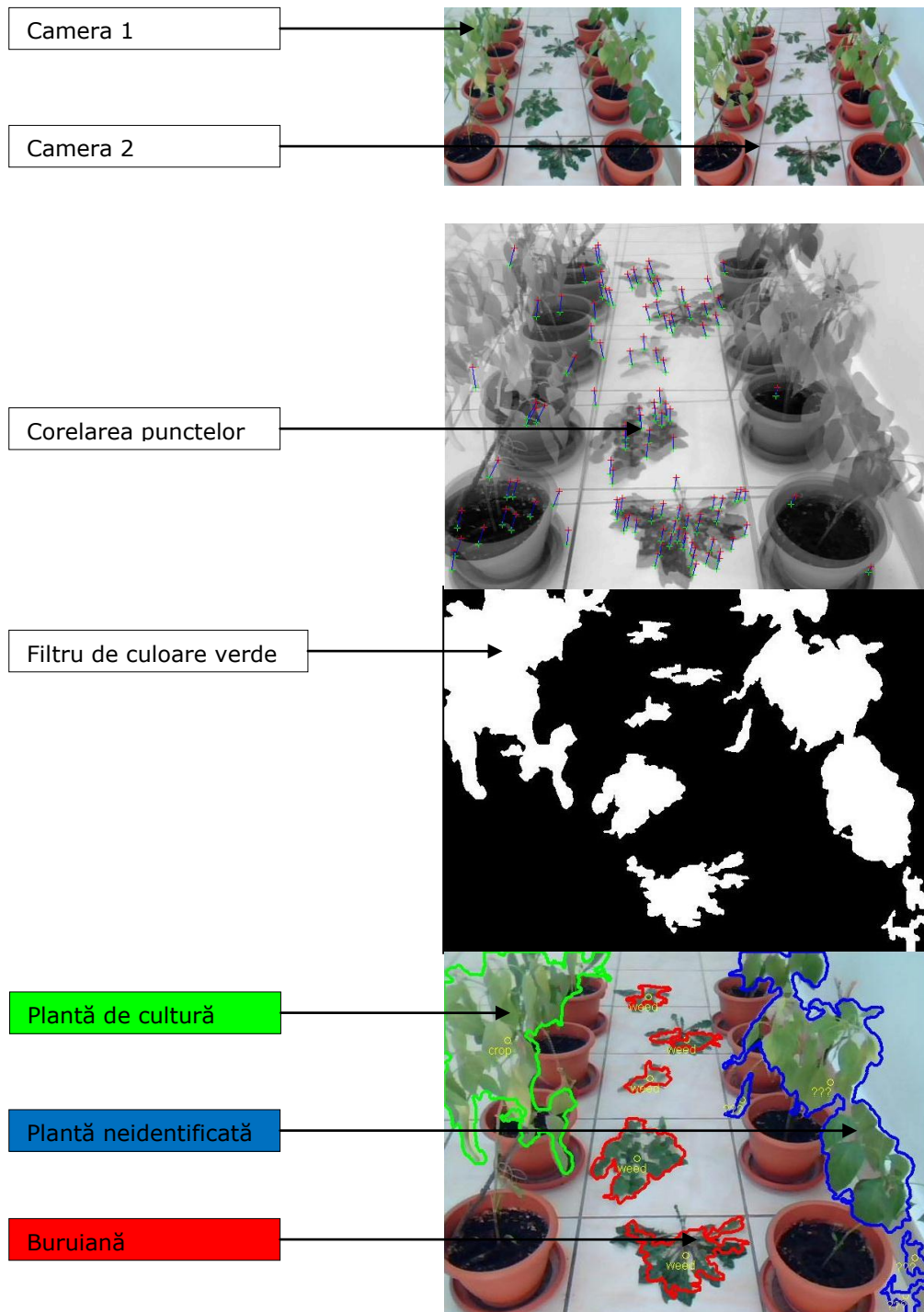


Fig. 8. 5. Procesarea imaginilor în procesul de diferențiere buruiană/plantă

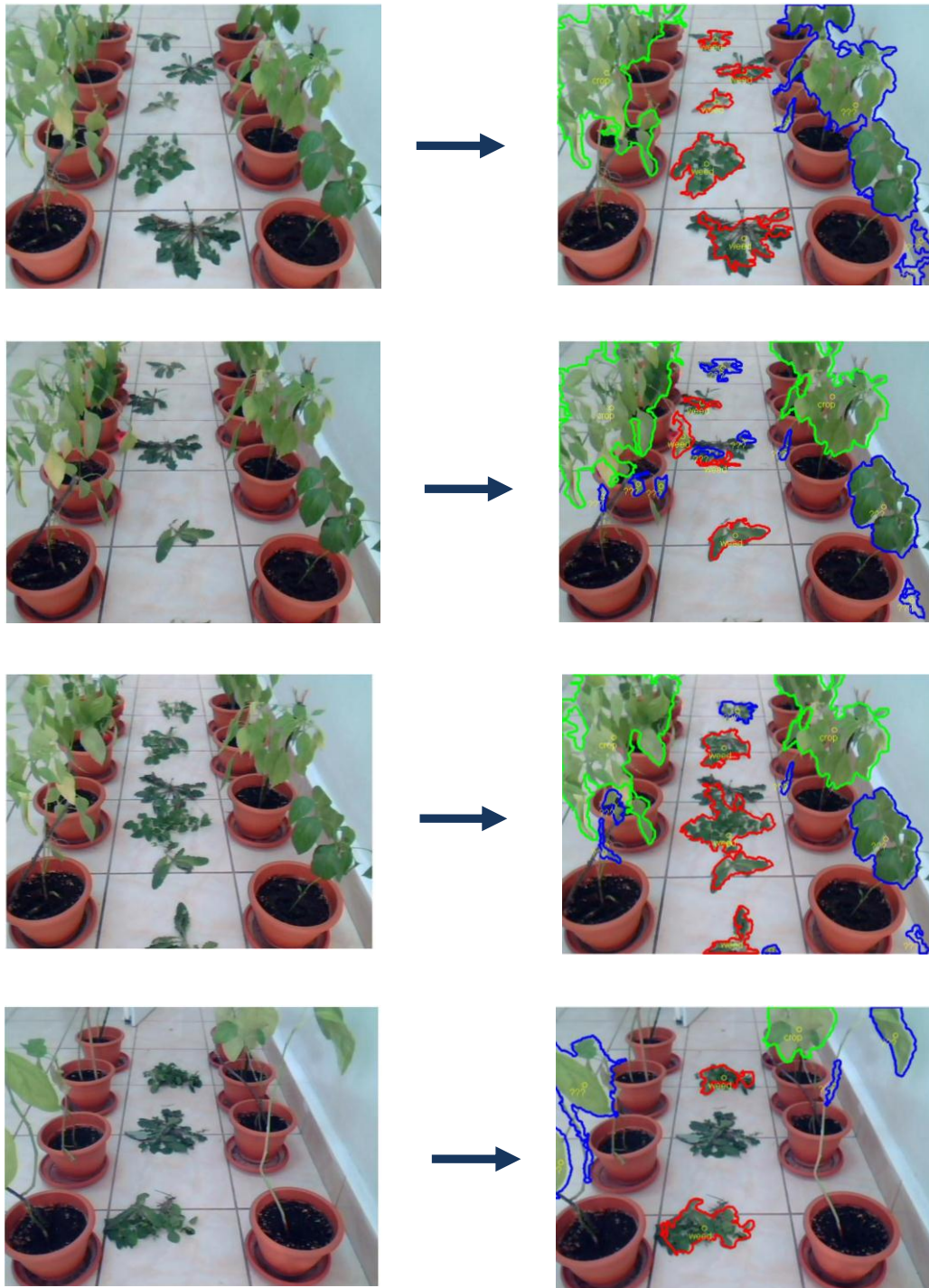


Fig. 8. 6. Rezultatele experimentului de diferențiere buruiană/plantă cultivată.

## 9 Utilizarea camerei 3D-ToF (Time of Flight) pentru aplicații în agricultură

### 9.1 Introducere

Pe parcursul stagiului la „Hochschule Osnabrück” am făcut parte din echipa de studenți care au participat la concursul internațional de roboți pentru aplicații în agricultură „Field Robot Event”.

Activitatea desfășurată în această perioadă a vizat:

- Analiza posibilităților de integrare a camerelor 3D-ToF în structura unui robot pentru aplicații în agricultură;
- Analiza posibilităților de corelare a camerelor 3D-ToF cu 2D-color pentru a obține simultan informații despre culoarea unui obiect și distanța acestuia față de focarele camerelor video.

Robotul vizat pentru integrarea camerelor 3D-ToF în structura sa a fost robotul „Optimize Prime”

### 9.2 Analiza plantelor utilizând un sistem stereo format dintr-o cameră 3D-ToF (Time of Flight) și o cameră 2D-color

Procesul de realizare a corespondenței dintre imagini prelevate de la camere video distincte, poartă denumirea de stereo-corelație [Sutton, 2009]. În literatura de specialitate stereo-corelația este întâlnită des sub denumirea de *camera-matching* sau *stereo-matching* [Sutton, 2009].

Considerând două imagini - imaginea 1 prelevată de la camera 1, și imaginea 2 prelevată de la camera 2 - stereo-corelația constă în punerea în corespondență a unui punct ( $P_{im1}$ ) din imaginea 1 cu corespondentul său din imaginea 2 ( $P_{im2}$ ).  $P_{im1}$  și  $P_{im2}$  sunt proiecțiile imaginii ale aceluiași punct din lumea reală  $P_{world}$  în cadrul celor două imagini.

Semnificația noțiunilor și principiul de lucru sunt prezentate în Fig. 9. 1 [Tilneac, 2010].

Activitatea a avut ca principal obiectiv realizarea de *camera-matching* între o cameră 3D-ToF (*Time of Flight*) a firmelor pmd-ifm și o cameră 2D-color Logitech C200. Avantajul obținut prin corelarea imaginilor provenite de la cele două camere este obținerea simultană a două tipuri de informații; distanță respectiv culoare. De la camera 3D-ToF se obțin informații despre distanțe, iar de la camera 2D-color, se obțin informații despre culoare. Pentru a atinge acest obiectiv, a fost necesară conceperea unor algoritmi de calcul, având ca scop corelarea imaginii prelevate de la camera 3D-ToF, cu imaginea prelevată de la camera 2D-color.

Imagini achiziționate cu camera 3D-ToF și camera 2D-color sunt prezentate în Fig. 9. 2. Imaginile sunt prelevate pe terenuri agricole cultivate cu porumb în condiții specifice mediului natural. Pe astfel de imagini ar urma să fie desfășurată activitatea de corelare a punctelor omoloage.



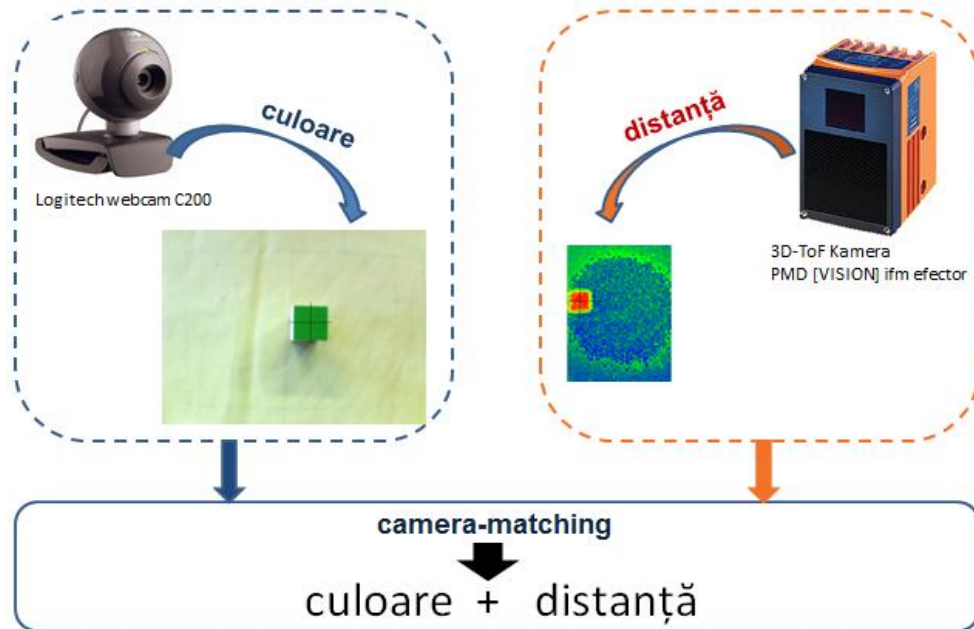


Fig. 9. 1. Corelarea imaginilor prelevate de la camera 2D-color cu imagini prelevate de la camera 3D-ToF

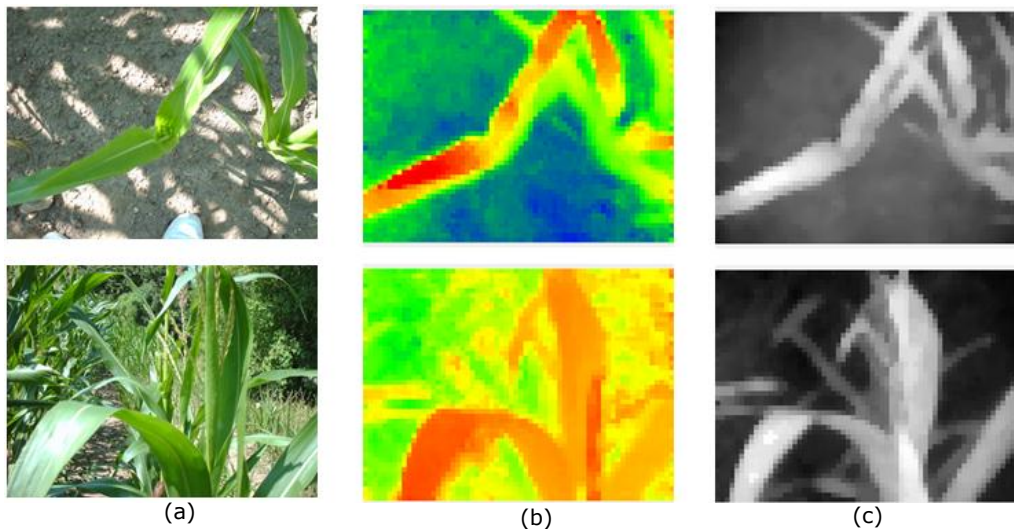


Fig. 9. 2. Imagini prelevate de la camera 2D color (a) și imagini prelevate de la camera 3D-ToF (b) și (c)

Programul de lucru, CORELAREA\_IMAGINILOR.exe a fost dezvoltat în mediul MATLAB. Programul poate fi lansat din mediul MATLAB prin lansarea aplicației START.m. Lansarea în lucru se realizează și în mod direct din fișierul \*.exe și are ca efect deschiderea interfeței grafice din Fig. 9. 3. Utilizatorul are două opțiuni pe care le poate selecta pentru activitatea în continuare. La alegerea opțiunii 3D-2D se va

deschide fereastra de setări din Fig. 9. 4. O fereastră similară se va deschide și la selectarea butonului 3D-3D.

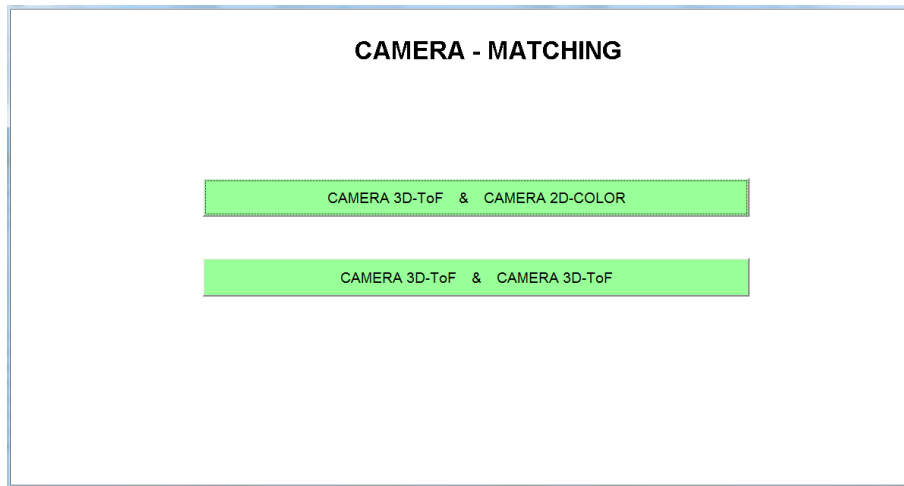


Fig. 9. 3. Interfața grafică pentru lansarea programului

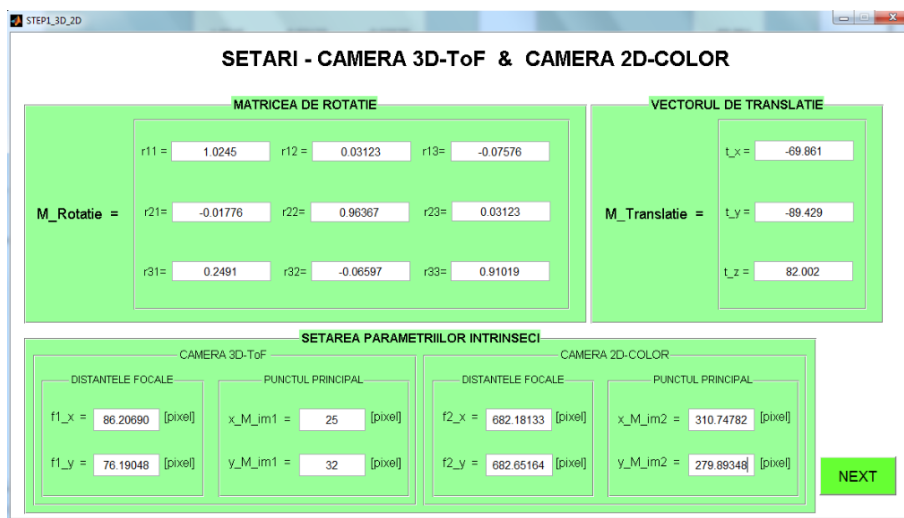


Fig. 9. 4. Fereastra de lucru pentru setarea parametrilor de intrare

Sistemul de camere 3D și 2D a fost în prealabil calibrat. Parametrii obținuți în urma calibrării (matricele de rotație, vectorii de translație, distanțele focale și punctele principale) sunt introduși ca parametri de intrare în fereastra de lucru pentru setarea parametrilor (Fig. 9. 4).

Imaginile propuse pentru analiză sunt încărcate într-un folder *imagini\_3D-2D*. În Fig. 9. 5, Fig. 9. 6 și Fig. 9. 7 sunt prezentate trei cazuri de pe parcursul analizelor efectuate. Operatorul poate selecta un punct de pe imagine obținând în schimb setul de parametri dorit:

- distanța de la focarul camerei 3D până la punctul din scenă selectat în imagine;

## 128 Utilizarea camerei 3D-ToF (Time of Flight) pentru aplicații în agricultură - 9

- coordonatele punctului in imaginea 3D
- distanța de la focarul camerei 2D până la punctul din scenă selectat în imagine;
- coordonatele punctului in imaginea 2D

În urma analizei efectuate vor fi afișate valorile componentelor de culoare RGB și HSV ale pixelului omolog din imaginea color.

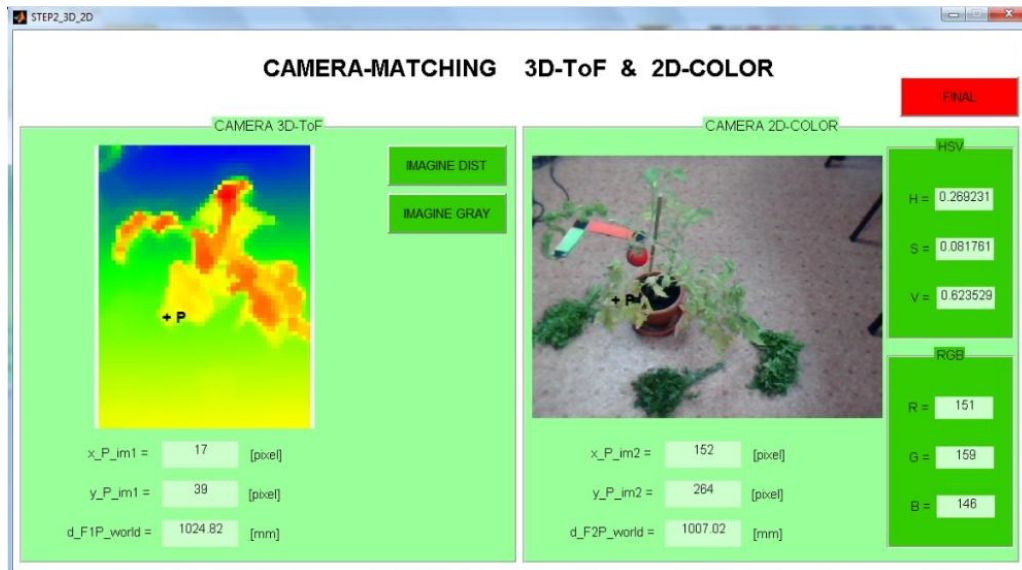


Fig. 9. 5. Analiza unei plante de roșie

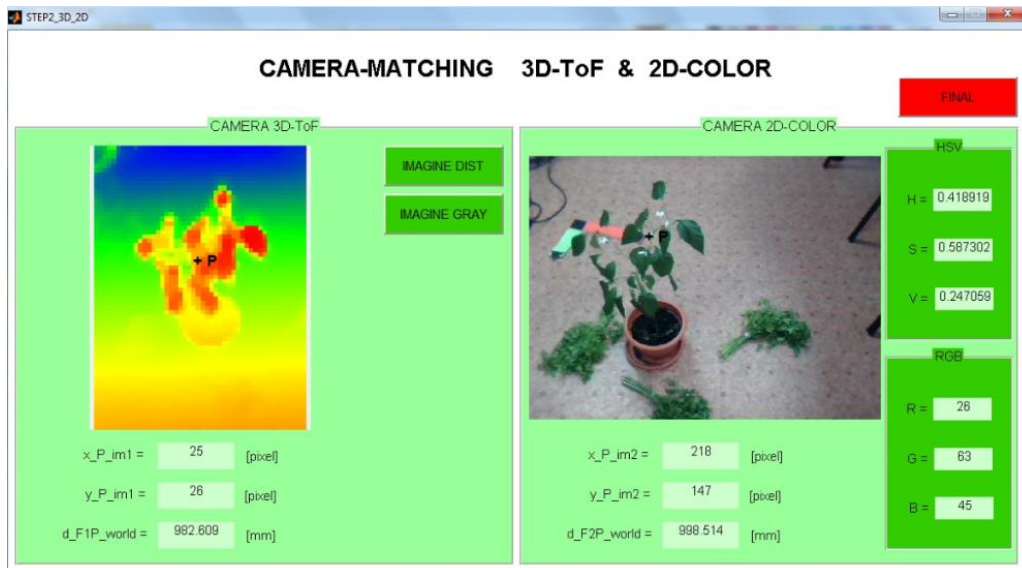


Fig. 9. 6. Analiza unei plante de ardei



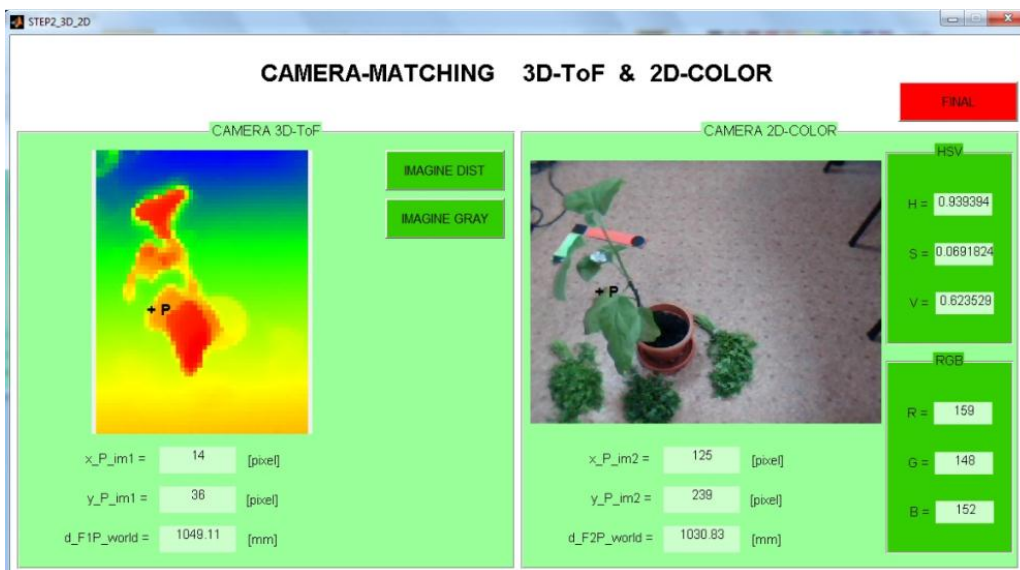


Fig. 9. 7. Analiza unei plante de vânăță

Integrarea celor două camere 3D și 2D pe o structură de robot mobil este prezentată în Fig. 9. 8. Cele două camere sunt poziționate pe șasiul robotului mobil. Pe același șasiu este fixată structura serială a unui robot care are ca scop realizarea efectivă a aplicației agricole. În figură sunt modelate câmpurile vizuale ale celor două camere. Proiectul prezentat în figură este realizat în mediul CATIA V5 [Tilneac, 2011].

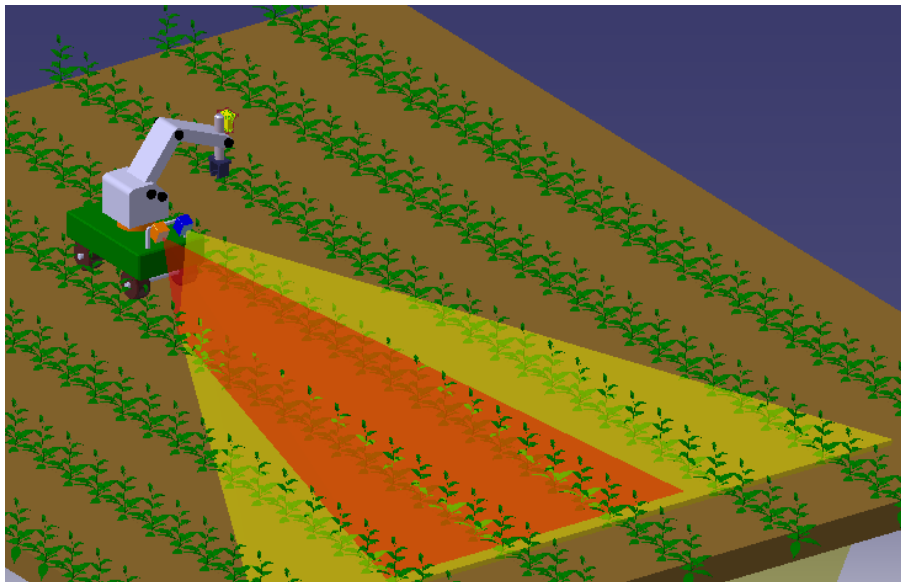


Fig. 9. 8. Concept de robot pentru agricultură dotat cu un sistem de camere video format din o cameră 2D-color și o cameră 3D-ToF

## 130 Utilizarea camerei 3D-ToF (Time of Flight) pentru aplicații în agricultură - 9

Cele două sisteme pot fi utilizate, pe baza parametrilor menționați anterior și a unor procesări de imagini în ansamblul plantei, la detectarea plantelor individuale sau a rândului de culturi.

În anexa E este prezentată ordinograma programului MATLAB și câteva secvențe de cod.

### 9.3 Utilizarea camerelor 3D-ToF (Time of Flight) la navigarea robotului mobil

Sistemul mecatronic autonom mobil avut în vedere pentru partea experimentală și respectiv concursul menționat este prezentat în Fig. 9. 9. Se remarcă în figură poziționarea camerelor 3D pe un suport care să le permită vizualizarea de la o înălțime corespunzătoare a spațiului de lucru.

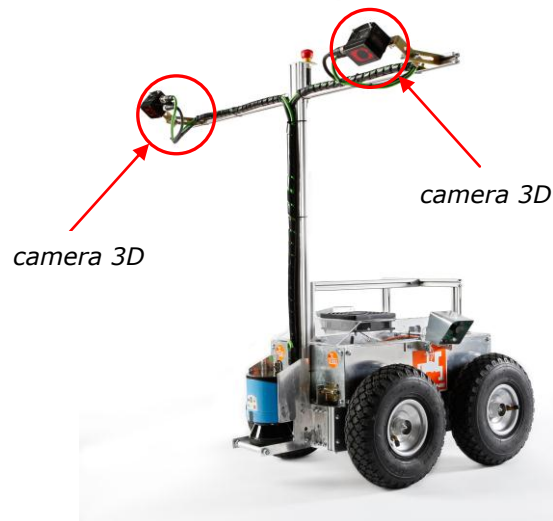


Fig. 9. 9. Camerele 3D amplasate pe robotul „Optimaize Prime” (Hochschule Osnabrück)

Pentru realizarea funcției de navigare a robotului subsistemul specific al acestuia a fost prevăzut cu două camere 3D Time-of-flight (ToF). Avantajul utilizării camerelor 3D este capacitatea acestora de a genera în timp real imagini pe toate cele trei dimensiuni. Camera 3D este de tipul IFM PMD[vision]® O3 cu frecvența: 25 Hz; domeniul de măsurare: 7.5 m; rezoluția: 64x48 pixeli; unghiurile de câmp ale imaginii: 40° / 30°; tensiunea: 24 V; curentul electric: < 1.0 A.

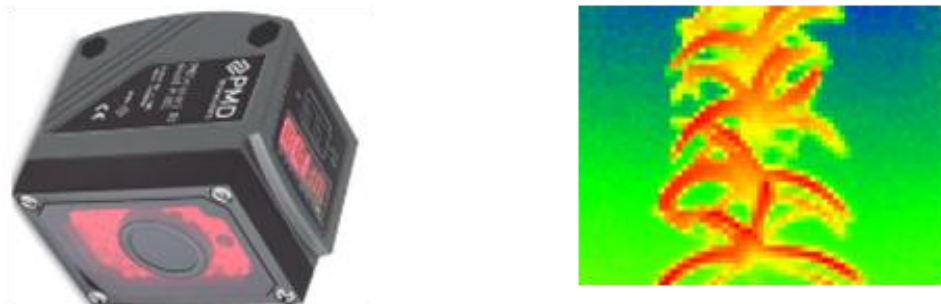


Fig. 9. 10. Camera 3D-ToF (dreapta) și o imagine prelevată de la această cameră

Camera 3D (Fig. 9. 10) generează o matrice ce reprezintă distanțele corespunzătoare fiecărui pixel din imagine. Identificarea rândurilor este realizată prin regresie lineară utilizând coordonatele (x,y) ce corespund distanțelor cuprinse în intervalul [Hmin; Hmax] . Robotul urmărește rândul prin menținerea dreptei rezultate prin metoda regresiei lineare în mijlocul imaginii (paralel cu axa x). În cazul în care nu există deviații (offset sau unghiuri), robotul va merge drept înainte. Atunci când apar deviații, robotul virează la dreapta sau la stânga cu scopul de a corecta erorile (Tab. 9. 1).

Coefficienții  $a_1$  și  $a_0$  (corespunzători ecuației drepte) sunt determinați din relațiile (9. 2), respectiv (9. 3).

$$y = a_1 * x + a_0 \tag{9. 1}$$

$$a_1 = \frac{n \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \sum_{i=1}^n y_i}{n \sum_{i=1}^n x_i^2 - \left( \sum_{i=1}^n x_i \right)^2} \tag{9. 2}$$

$$a_0 = \frac{\sum_{i=1}^n x_i^2 \sum_{i=1}^n y_i - \sum_{i=1}^n x_i \sum_{i=1}^n x_i y_i}{n \sum_{i=1}^n x_i^2 - \left( \sum_{i=1}^n x_i \right)^2} \tag{9. 3}$$

Parametrul  $H$  [m] reprezintă distanța dintre focarul camerei 3D-ToF și un punct din scenă. Intervalul [Hmin; Hmax] este stabilit în funcție de înălțimea plantelor de cultură. Pe baza acestui interval a fost realizat un filtru de înălțime.

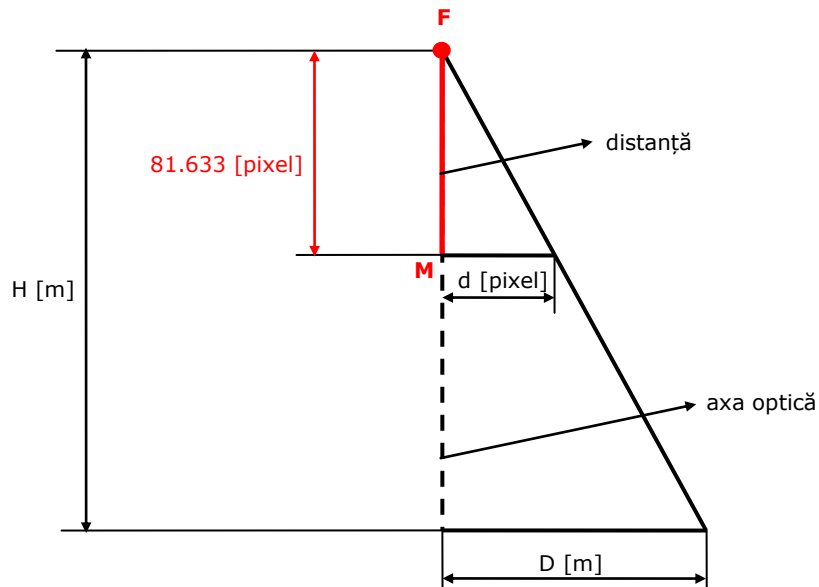


Fig. 9. 11. Geometria raportului imagine-scenă corespunzător camerei 3D-ToF

**132** Utilizarea camerei 3D-ToF (Time of Flight) pentru aplicații în agricultură - 9

Parametrul  $d$  [pixel] reprezintă deviația față de axa care trece prin centrul imaginii și este paralelă cu axa  $x$  a imaginii. Se notează cu  $M$  punctul principal al imaginii, iar cu  $A$  este notat punctul de intersecție între dreapta determinată prin regresie liniară și dreapta paralelă cu axa  $y$  care trece prin punctul principal al imaginii. Valoarea deviației în imagine determinată prin diferența valorilor coordonatelor pe axa  $y$  a punctelor  $M$  și  $A$ .

$$d = y_M - y_A \quad [\text{pixel}] \quad (9. 4)$$

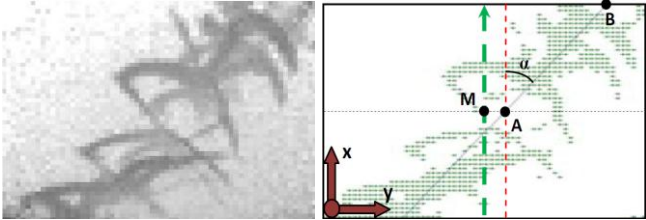
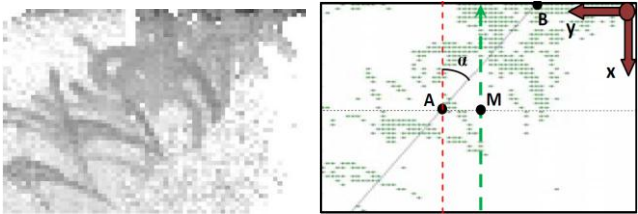
Raportul imagine-scenă a fost determinat geometric (Fig. 9. 11). Parametrul  $D$  reprezintă deviația exprimată în metri și este determinat prin relația:

$$D = \frac{H * d}{81.633} \quad [\text{m}] \quad (9. 5)$$

Deviația unghiulară  $\alpha$  [°] exprimată în grade este determinată în relația:

$$\alpha = \arctg\left(\frac{y_B - y_A}{x_B - x_A}\right) \quad (9. 6)$$

Corecția devierii vehiculului de la traiectorie este realizată cu ajutorul parametrilor  $d$  și  $\alpha$  a căror valori rezultă din relațiile (9. 4) respectiv (9. 6).

Camera-3D stânga	
deviațiile traiectoriei robotului	corecția traiectoriei
	$\alpha < 0$ - virare dreapta $\alpha > 0$ - virare stânga $d > 0$ - offset dreapta $d < 0$ - offset stânga
Camera-3D dreapta	
deviațiile traiectoriei robotului	corecția traiectoriei
	$\alpha > 0$ - virare dreapta $\alpha < 0$ - virare stânga $d < 0$ - offset dreapta $d > 0$ - offset stânga

Tab. 9. 1. Detectarea și urmărirea rândului de culturi

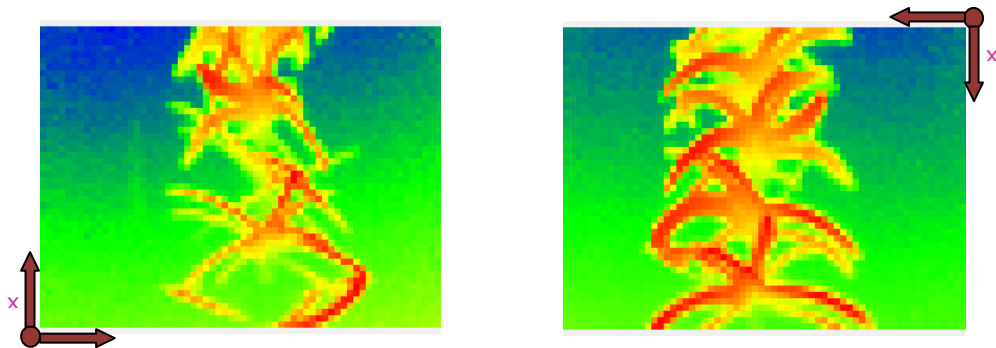


Fig. 9. 12. Imagini prelevate de la sistemul de camere 3D-ToF



Fig. 9. 13. Imagini prelevate în timpul experimentului efectuat la „Hochschule Osnabrück”

## 134 Utilizarea camerei 3D-ToF (Time of Flight) pentru aplicații în agricultură - 9

---

În anexa F este prezentată schema logică a programului C++ de detectare a rândului de culturi prin metoda regresiei liniare.

### 9.4 Concluzii

Utilizarea camerelor 3D-ToF aduce avantaje majore în cazul aplicațiilor de procesare a datelor în timp real. În agricultură, camera 3D-ToF este utilizată în aplicații de detectare a rândurilor de culturi și în aplicații de recunoaștere a plantelor individuale prin corelarea imaginilor camerei 3D-ToF cu imagini ale camerei 2D-color.

<b>Avantaje</b>	<b>Dezavantaje</b>
<ul style="list-style-type: none"><li>▪ Permite obținerea în timp real a informațiilor despre distanțe.</li><li>▪ Pentru fiecare pixel din imagine este returnată valoarea distanței exprimată în metri de la focarul camerei până la punctul din scenă ce corespunde pixelului din imagine.</li></ul>	<ul style="list-style-type: none"><li>▪ Rezoluția imaginii este foarte scăzută (50x64 pixeli).</li><li>▪ Imaginile nu sunt color.</li></ul>

## 10 Proiectarea și realizarea robotului *GardenerRob* pentru aplicații în agricultură

### 10.1 Introducere

În acest capitol, autoarea prezintă un concept de robot pentru agricultură de dimensiuni mici. Structura acestui robot este una hibridă: un robot serial este fixat pe șasiul unui robot mobil. Acest tip de robot a fost proiectat pentru a fi utilizat pe terenuri agricole cu suprafețe restrânse (grădini și sere). Robotul *GardenerRob* este un sistem mecatronic destinat pentru combaterea buruienilor.

Structura robotului serial se compune din dispozitivul de ghidare (DG) și efectorul final (EF).

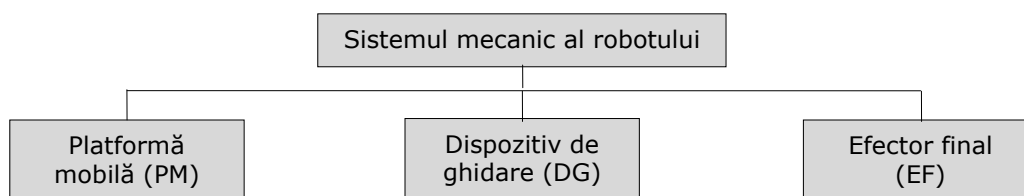


Fig. 10. 1. Structura sistemului mecanic a robotului [Kovacs, 2001]

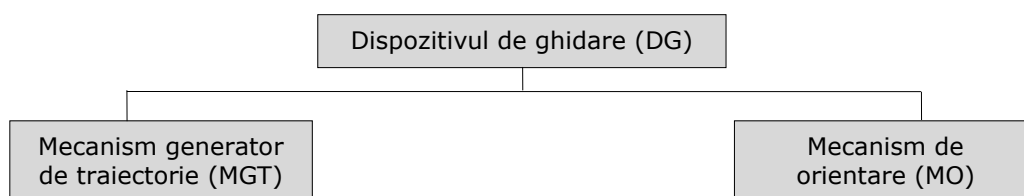


Fig. 10. 2. Structura dispozitivului de ghidare a robotului [Kovacs, 2001]

### 10.2 Structura dispozitivului de ghidare (DG) și a efectorului final (EF)

Dispozitivul de ghidare al robotului proiectat este compus din mecanismul generator de traiectorie (MGT) și mecanismul de orientare (MO).

Schema cinematică a MGT (mecanism generator de traiectorie) cu topologie serială și spațiul de lucru asigurat de acesta sunt reprezentate în Fig. 10. 3 [\*\*\*Robotica]. Robotul lucrează în coordonate cilindrice (RTT). Mecanismul generator de traiectorie al robotului are trei grade de mobilitate și este prevăzut cu două axe de translație. Spațiul de lucru (conform Fig. 10. 3) al robotului *GardenerRob* este  $O'O'' = 520$  mm, respectiv  $O''O''' = 520$  mm.

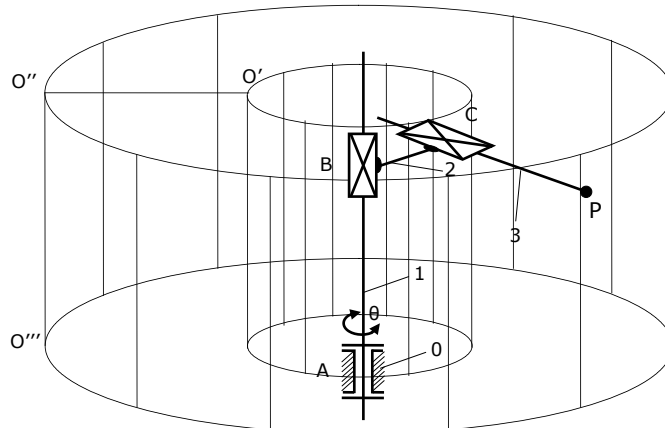


Fig. 10. 3. Schema cinematică și spațiul de lucru al robotului

Mecanismul de orientare al robotului are un singur grad de mobilitate și are schema cinematică din Fig. 10. 4. Efectorul final al robotului efectuează mișcarea de supinație-pronație denumită și „Roll” (rotație în jurul axei z). Punctul caracteristic „P” al robotului se identifică cu vârful sculei care materializează efectorul final.

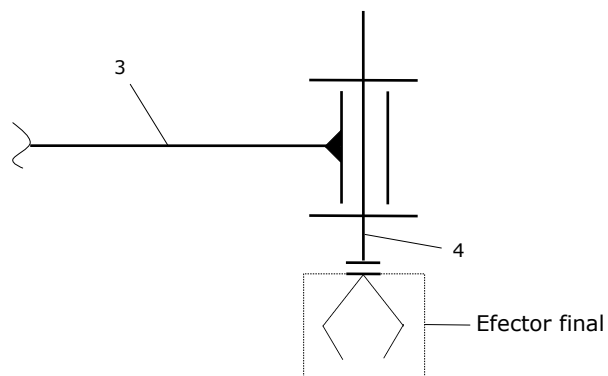


Fig. 10. 4. Schema cinematică a mecanismului de orientare

### 10.3 Modelul geometric direct și invers al robotului serial

Aplicația presupune localizarea punctului caracteristic a efectorului final de combatere a buruienilor. Zona este recunoscută pe baza sistemului senzorial optic. În acest scop este necesară trecerea coordonatelor din sistemul de referință al camerei în sistemul de referință atașat bazei robotului serial R-S. Conform cu schema cinematică analizată și procedurile de calcul geometric, fiecărui element din cadrul DG și EF i se va atașa un sistem de referință propriu [\*\*\*Robotica].



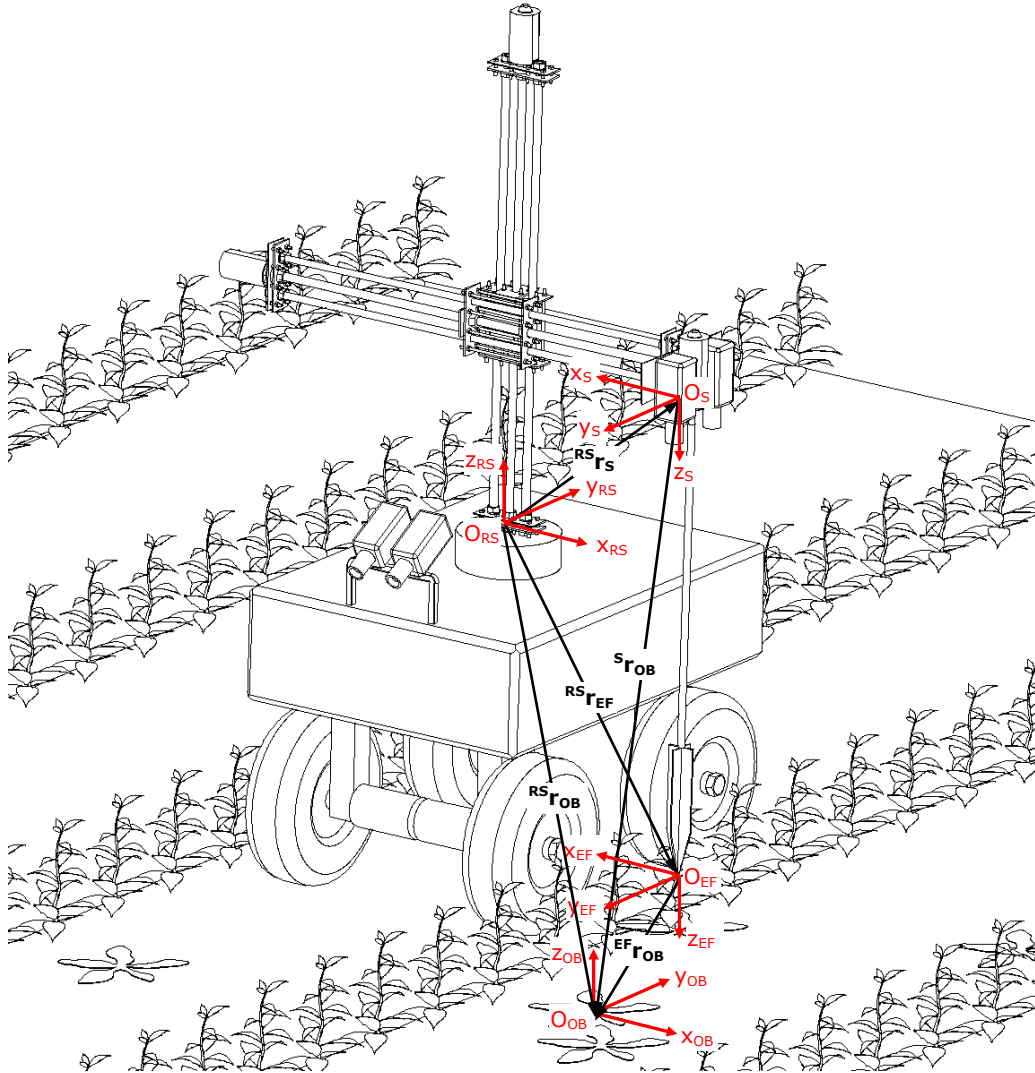


Fig. 10. 5. Situația robotului în spațiul de lucru

În Fig. 10. 5 se prezintă situația robotului *GardenerRob* în spațiul de lucru agricol. Fiecărei componente a sistemului i se va atașa un sistem de referință propriu:

- robotul serial RS – sistemul de coordonate  $O_{RS}X_{RS}Y_{RS}Z_{RS}$
- efectorul final – sistemul de coordonate  $O_{EF}X_{EF}Y_{EF}Z_{EF}$
- sistemul senzorial optic – sistemul de coordonate  $O_SX_SY_SZ_S$
- obiectul (buruiiana) – sistemul de coordonate  $O_{OB}X_{OB}Y_{OB}Z_{OB}$

Trecerea coordonatelor punctelor de pe plantă din sistemul de referință ( $O_{OB}X_{OB}Y_{OB}Z_{OB}$ ) în sistemul de referință atașat bazei robotului R-S ( $O_{RS}X_{RS}Y_{RS}Z_{RS}$ ) are la bază transformarea:

$${}^{RS}r_{OB} = {}^{RS}T_S * S_{r_{OB}} \quad (10. 1)$$

unde valorile matricei de transformare  ${}^{RS}T_S$  sunt determinate prin calibrare, iar valorile componentelor vectorului  $S_{r_{OB}}$  sunt determinate prin procesarea imaginilor prelevate de la sistemul stereoscopic amplasat în apropierea efectorului final.

Trecerea coordonatelor punctului caracteristic M din sistemul de referință al efectorului final ( $O_{EF}X_{EF}Y_{EF}Z_{EF}$ ) în sistemul de referință atașat bazei robotului R-S ( $O_{RS}X_{RS}Y_{RS}Z_{RS}$ ) are la bază transformarea:

$${}^{RS}T_{EF} = \begin{bmatrix} \cos(\theta_1) & \sin(\theta_1) & 0 & \sin(\theta_1) * d_3 + \cos(\theta_1) * a_2 \\ \sin(\theta_1) & -\cos(\theta_1) & 0 & -\cos(\theta_1) * d_3 + \sin(\theta_1) * a_2 \\ 0 & 0 & -1 & -d_4 + d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10. 2)$$

Calculule detaliate sunt prezentate în relațiile (10. 7) ÷ (10. 16).

Trecerea coordonatelor punctelor de pe plantă din sistemul de referință ( $O_{OB}X_{OB}Y_{OB}Z_{OB}$ ) în sistemul de referință atașat efectorului final EF( $O_{EF}X_{EF}Y_{EF}Z_{EF}$ ) are la bază transformarea:

$${}^{EF}r_{OB} = {}^{EF}T_{RS} * {}^{RS}r_{OB} \quad (10. 3)$$

Traietoria punctului caracteristic va fi aleasă astfel încât să nu dăuneze plantei cultivate.

Schema cinematică pentru întregul lanț cinematic al robotului *GardenerRob* este prezentată în Fig. 10. 6. Sunt evidențiate și sistemele de referință precizate anterior.

Axele (axa verticală și axa orizontală) mecanismului generator de traiectorie sunt acționate prin intermediul unor transmisii cu cuple elicoidale de rostogolire și anume: axa de ridicare a brațului (axa verticală) și axa de extensie a brațului (axa orizontală). Lungimea perpendiculararei comune a celor două axe încrucișate este denumită „offset” și este notată cu „a”.

Modelul geometric direct al robotului serial *GardenerRob* (structură RTT) este calculat pe baza metodei Denavit – Hartenberg.

Matricele de transformare omogene elementare sunt: matricea de rotație în jurul axei x notată  $Rot(x_i, \theta_i)$ ; matricea de rotație în jurul axei y notată  $Rot(y_i, \theta_i)$ ; matricea de rotație în jurul axei z notată  $Rot(z_i, \theta_i)$  [Rădulescu, 2008].

$$Rot(x_i, \theta_i) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta_i & -\sin \theta_i & 0 \\ 0 & \sin \theta_i & \cos \theta_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10. 4)$$

$$Rot(y_i, \theta_i) = \begin{bmatrix} \cos \theta_i & 0 & \sin \theta_i & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta_i & 0 & \cos \theta_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10.5)$$

$$Rot(z_i, \theta_i) = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & 0 \\ \sin \theta_i & \cos \theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10.6)$$

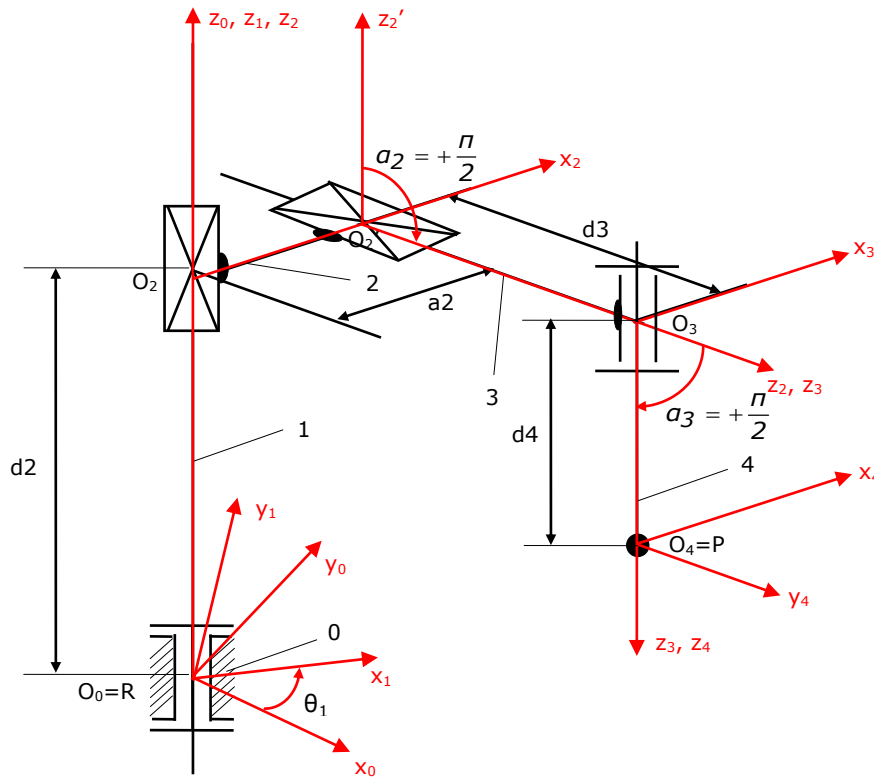


Fig. 10. 6. Sistemele de coordonate atașate axelor robotului

Cupla	Parametrul			
	$\theta$	$d$	$a$	$\alpha$
0-1	$\theta_1$	0	0	0
1-2	0	$d_2$	$a_2$	$\alpha_2 = +\pi/2$
2-3	0	$d_3$	0	$\alpha_3 = +\pi/2$
3-4	0	$d_4$	0	0

Tab. 10. 1. Tabelul Denavit-Hartenberg

$${}^0T_1 = \text{Rot}(z_1, \theta_1) \quad (10.7)$$

$${}^0T_1 = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10.8)$$

$${}^1T_2 = \text{Trans}(z_2, d_2) * \text{Trans}(x_2, a_2) * \text{Rot}(x_2, +\frac{\pi}{2}) \quad (10.9)$$

$${}^1T_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & a_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & a_2 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10.10)$$

$${}^2T_3 = \text{Trans}(z_3, d_3) * \text{Rot}(x_3, +\frac{\pi}{2}) \quad (10.11)$$

$${}^2T_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10.12)$$

$${}^3T_4 = \text{Trans}(z_4, d_4) \quad (10.13)$$

$${}^3T_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10.14)$$

$${}^0T_4 = {}^0T_1 * {}^1T_2 * {}^2T_3 * {}^3T_4 \quad (10.15)$$

$${}^0T_4 = \begin{bmatrix} \cos(\theta_1) & \sin(\theta_1) & 0 & \sin(\theta_1) * d_3 + \cos(\theta_1) * a_2 \\ \sin(\theta_1) & -\cos(\theta_1) & 0 & -\cos(\theta_1) * d_3 + \sin(\theta_1) * a_2 \\ 0 & 0 & -1 & -d_4 + d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10.16)$$

$${}^{RS}T_{EF} = {}^0T_4 \quad (10.17)$$

Pe baza transformărilor se obține modelul geometric direct. După transformări se poate obține modelul geometric invers. Pornind de la expresia matricii de transformare omogenă  ${}^0T_4$  aferentă mecanismului generator de traiectorie rezultă relația:

$${}^{RS}P = \left[ {}^{RS}P_x \quad {}^{RS}P_y \quad {}^{RS}P_z \right]^T \quad (10.18)$$

unde  ${}^{RS}P$  reprezintă coordonatele punctului caracteristic al efectorului final raportat la sistemul de referință atașat bazei robotului serial (R-S).

$$\begin{cases} {}^{RS}P_x = a_2 * \cos(\theta_1) + d_3 * \sin(\theta_1) \\ {}^{RS}P_y = a_2 * \sin(\theta_1) - d_3 * \cos(\theta_1) \\ {}^{RS}P_z = d_2 - d_4 \end{cases} \quad (10.19)$$

1. *Determinarea valorii parametrului  $d_2$*

Valoarea parametrului  $d_4$  este constantă. Valoarea parametrului  $d_2$  este determinată prin relația:

$$d_2 = {}^{RS}P_z + d_4 \quad (10.20)$$

2. *Determinarea valorii parametrului  $d_3$*

Valoarea parametrului  $a_2$  este constantă. Primele două ecuații ale sistemului (10.20) conduc la obținerea lui  $d_3$  prin relația [Rădulescu, 2008]:

$$d_3 = \sqrt{{}^{RS}P_x^2 + {}^{RS}P_y^2 - a_2^2} \quad (10.21)$$

3. *Determinarea valorii parametrului  $\theta_1$*

Valoarea parametrului  $\theta_1$  este determinată din primele două ecuații ale sistemului (10.20):

$$\sin(\theta_1) = \frac{{}^{RS}P_x - a_2 * \cos(\theta_1)}{d_3} \quad (10.22)$$

$${}^{RS}P_y = a_2 * \frac{{}^{RS}P_x - a_2 * \cos(\theta_1)}{d_3} - d_3 * \cos(\theta_1) \quad (10.23)$$

$$\theta_1 = \arccos\left(\frac{a_2 * {}^{RS}P_x - d_3 * {}^{RS}P_y}{a_2^2 + d_3^2}\right) \quad (10.24)$$

#### 10.4 Modul de funcționare al robotului autonom *GardenerRob*

Pe baza experienței cumulate în timpul stagiului de pregătire la Universitatea din Osnabrück a fost posibilă elaborarea conceptuală a unui sistem mecatronic destinat combaterii buruienilor.

Structura generală a robotului *GardenerRob* este prezentată în Fig. 10. 7. Sistemul se compune dintr-un robot mobil și un robot serial fixat pe șasiul robotului mobil. Robotul mobil are un subsistem de locomoție bazat pe două roți de tracțiune și alte două roți pentru asigurarea stabilității sistemului. Cotele de gabarit ale robotului sunt prezentate în Fig. 10. 8.

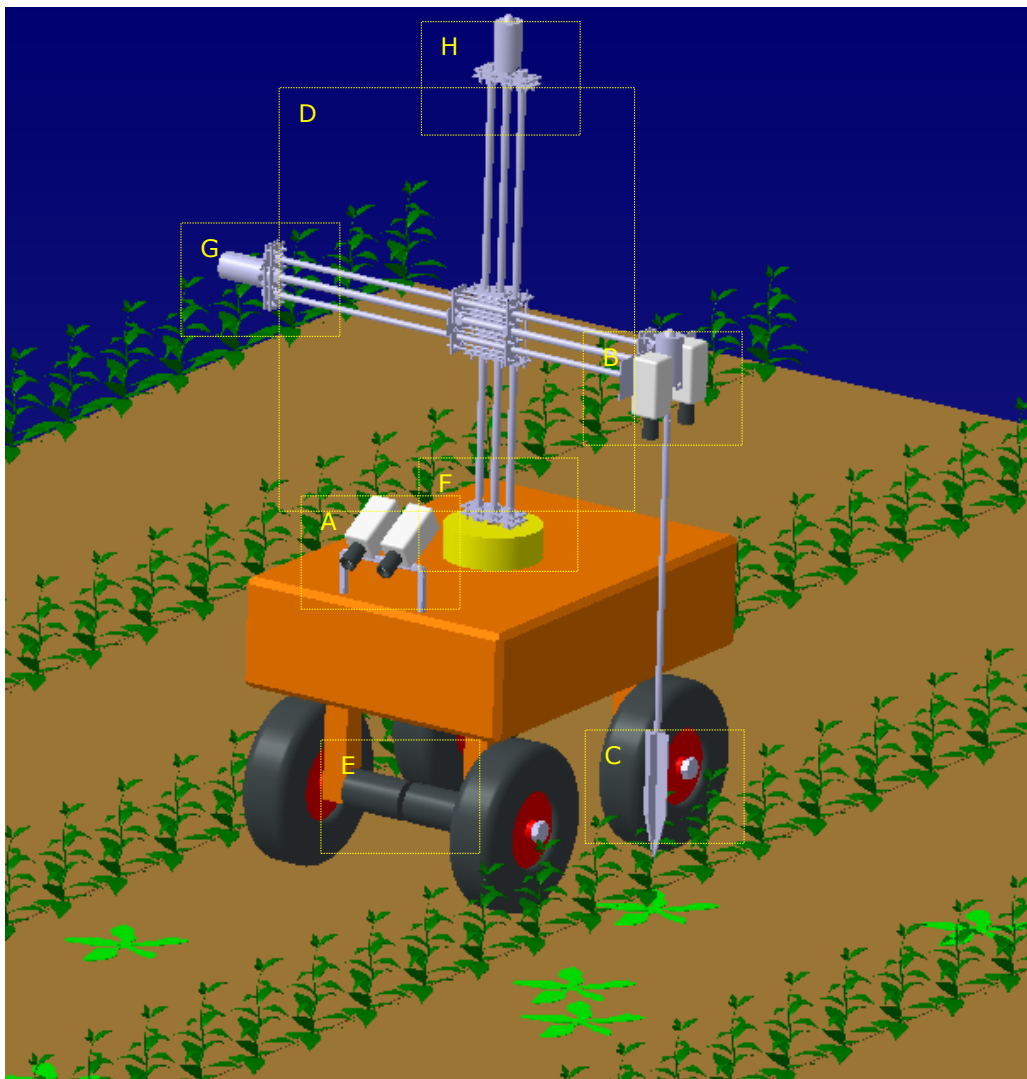


Fig. 10. 7. Robotul *GardenerRob* modelat în mediul CATIA V5

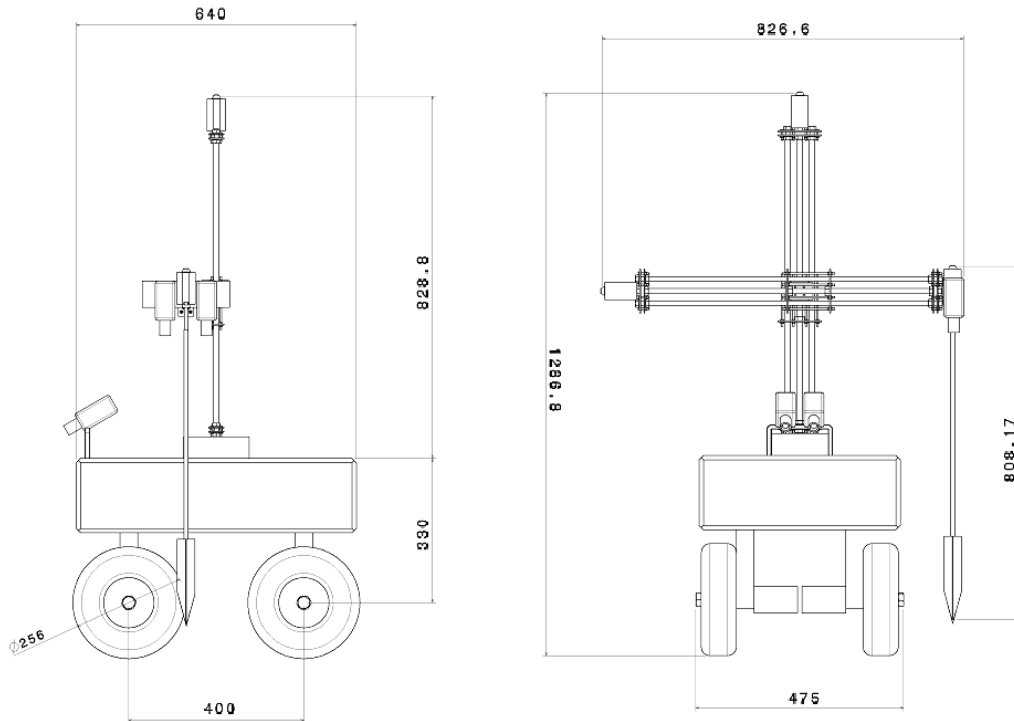


Fig. 10. 8. Dimensiunile de gabarit ale robotului *GardenerRob*

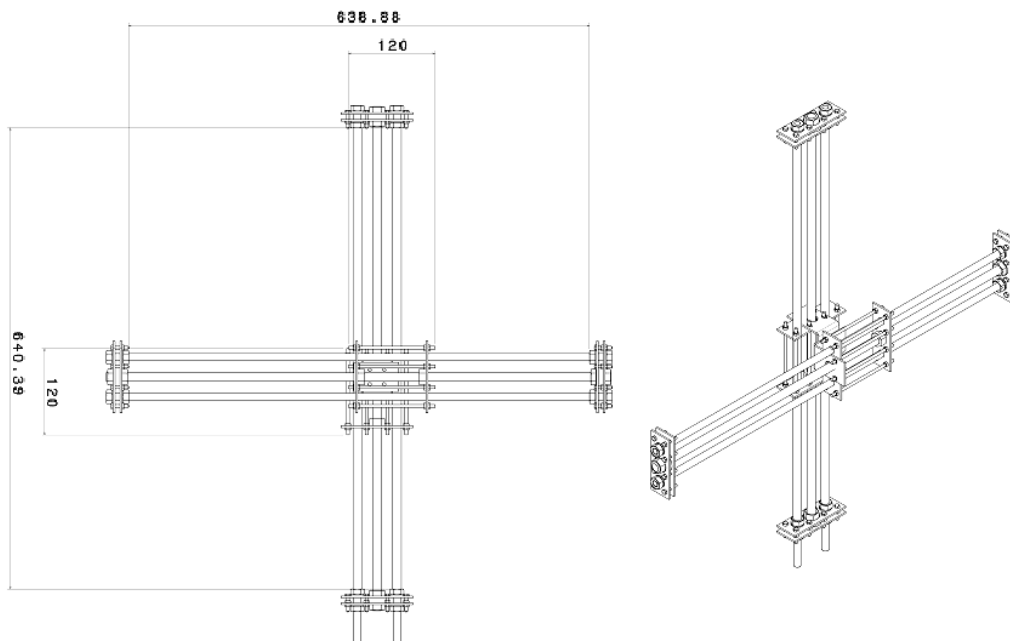


Fig. 10. 9. Cursele dispozitivul de translație pe verticală și orizontală

Ghidajele de translație pe verticală și orizontală (Fig. 10. 7, zona D) sunt realizate pe bază de cuple cinematice elicoidale de alunecare. Aceste transmisii au ca elemente active o tijă filetată și o piuliță prin intermediul cărora mișcarea de rotație a motorului este transformată în mișcare de translație. Ghidajele permit robotului să realizeze curse de translație de aproximativ 520 mm pe verticală și orizontală (Fig. 10. 9).

Sistemul de navigare al robotului *GardenerRob* este bazat pe recunoașterea mediului prin procesarea perechilor de imagini prelevate de la un sistem stereoscopic format din două camere video monocular (Fig. 10. 7, zona A). Prin acest sistem rândurile de culturi sunt detectate și localizate.

Sistemul de localizare a plantelor individuale (buruieni) este format din sistemul de vedere stereo din Fig. 10. 7 în zona B. Sistemul senzorial optic este amplasat în apropierea efectorului final, având axele optice paralele cu direcția acestuia. Prin procesarea perechilor de imagini sunt determinate valorile vectorului  ${}^S r_{OB}$  utilizat pentru a localiza centrul de greutate a buruienii detectate în imagine.

Efectorul final (Fig. 10. 7, zona C) distruge rădăcina buruienii prin introducerea unui dispozitiv în sol și realizarea unor mișcări de rotație în jurul axei verticale.

Sistemul de acționare al robotului este alcătuit din șase motoare de curent continuu alimentate de un acumulator care furnizează tensiune de 12V. Motoarele 1 și 2 (Fig. 10. 7, zona E) acționează roțile de tracțiune ale robotului mobil. Motorul 3 (Fig. 10. 7, zona F) este un servomotor prin intermediul căruia este realizată rotația mecanismului generator de traiectorie a robotului serial de tip RTT. Motorul 4 (Fig. 10. 7, zona G) acționează mișcarea de extensie braț pe axa orizontală de translație. Motorul 5 (Fig. 10. 7, zona H) acționează mișcarea de ridicare/coborâre a brațului pe axa verticală de translație. Motorul 6 (Fig. 10. 7, zona B) acționează mișcarea de rotație a efectorului final.

Sistemul de orientare al robotului mobil are o configurație de tip diferențial. Robotul virează la dreapta sau la stânga pe baza diferențelor de viteze corespunzătoare celor două roți de tracțiune.

Sistemul de comandă al robotului este dezvoltat în mediul MATLAB. Sistemul decizional al robotului funcționează pe baza informațiilor obținute de la sistemul de vedere artificială. Servomotoarele sunt comandate din programul MATLAB prin intermediul unui microcontroler de tip BASIC Stamp.

## 10.5 Concluzii

Pentru atingerea scopului urmărit am avut în vedere integrarea de module mecatronice într-o structură unitară. Prețul relativ ridicat al acestor module și timpul limitat au împiedicat finalizarea proiectului.



# 11 Concluzii finale, contribuții originale și direcții viitoare de cercetare

## 11.1 Concluzii finale

Problema recunoașterii plantelor din culturile agricole și detectarea buruienilor constituie o problemă de interes și cu aplicabilitate largă. Prin problematica abordată și prin modul de integrare *hardware* și *software* tema poate fi inclusă într-o aplicație de tip mecatronic.

Au fost desfășurate o serie de studii referitoare la stadiul actual din domeniu, în scopul stabilirii parametrilor de lucru pentru elaborarea tezei. În urma acestor studii s-a putut concluziona că utilizarea sistemelor senzoriale optice oferă facilități deosebite. În concordanță cu filozofia mecatronică, funcția de recunoaștere a plantelor și a buruienilor realizată de un sistem biologic-operator uman poate fi înlocuită cu un sistem senzorial artificial. Se răspunde astfel uneia din cerințele realizării unui sistem mecatronic.

Teza de doctorat include în conținutul său referiri la un număr de trei specii de plante (roșie, ardei și vânăță). Limitarea a fost impusă practic de limitările în timp a cercetărilor. Cercetările efectuate au confirmat ipotezele avute în vedere la realizarea diverselor modele, au indicat viitoare direcții de cercetare și în același timp au atras atenția asupra necesității utilizării unor echipamente profesionale și a unor sisteme de calcul puternice.

Obiectivul principal al tezei a constat în **detectarea automată a buruienilor de pe terenurile agricole în vederea combaterii acestora prin sisteme mecanice**. Studiile efectuate și rezultatele obținute răspund atât obiectivului principal cât și obiectivelor secundare specificate în capitolul 3 al tezei.

## 11.2 Contribuții originale

Studiile efectuate și rezultatele obținute au condus, în urma unei raportări la realizări similare din domeniul abordat, la posibilitatea enunțării unui set de contribuții personale:

- Analiza influenței luminozității asupra componentelor de culoare RGB și HSV ale frunzelor de roșie, ardei și vânăță.
- Dezvoltarea în mediul MATLAB a unui program de procesare a culorilor unei frunze selectate în imagine și analiza statistică a culorilor specifice anumitor specii de plante.
- Analiza semnăturilor frunzelor de roșie, ardei și vânăță în scopul identificării acestora pe baza numărului de puncte de inflexiune.
- Dezvoltarea unui program de învățare a speciilor de plante pe baza culorilor și a formelor frunzelor. Formele frunzelor au fost determinate pe baza unor proprietăți geometrice: arie, perimetru, diagonale.
- Dezvoltarea unui program de recunoaștere a speciilor de plante prin analiza formei frunzei.

- Aplicații robotizate de „visual servoing” pentru indicarea frunzelor recunoscute în imagine.
- Dezvoltarea unei metode originale de determinare a distanței focale a camerei video. Pentru această aplicație a fost dezvoltat un program MATLAB prevăzut cu interfață grafică.
- Dezvoltarea unei metode originale de calibrare extrinsecă camerelor video prin utilizarea a patru obiecte de calibrare de formă circulară. Programul MATLAB este prevăzut cu interfață grafică pentru utilizator.
- Dezvoltarea unei metode originale de calibrare extrinsecă a camerelor video prin utilizare de modele CAD. Metoda se bazează pe modelarea sistemului optic format din două camere video. Constrângerile geometrice sunt realizate cu ajutorul a trei puncte de control care corespund centrelor a trei obiecte de calibrare de formă circulară.
- Corelarea punctelor omoloage ale imaginilor prelevate de la un sistem de vedere stereo format dintr-o cameră 3D-ToF (Time of Flight) și o cameră 2D color. Pentru această aplicație a fost dezvoltat un program MATLAB prevăzut cu interfață grafică.
- Dezvoltarea unui sistem de navigare a robotului printre rândurile de culturi utilizând două camere 3D-ToF (Time of Flight).
- Diferențiere buruienă/plantă pe baza diferențelor de înălțime măsurate printr-un sistem stereo binocular.
- Elaborarea conceptuală a unui sistem mecatronic destinat combaterii buruienilor.

### **11.3 Direcții viitoare de cercetare**

Pe baza cercetărilor efectuate se pot estima câteva direcții pentru cercetări viitoare:

- Continuarea studiilor privind diferențierea plantelor prin analiza monotoniei semnăturilor corespunzătoare speciei de plantă;
- Crearea unor baze de date pentru un număr mare de plante de cultură care să permită integrarea lor într-un sistem mecatronic dezvoltat pentru aplicații practice;
- Dezvoltarea unor modele de recunoaștere a frunzelor în timp real într-un mediu care poate fi perturbat de o serie de factori de mediu;
- Intensificarea experimentelor în aer liber și dezvoltarea de sisteme adaptabile la schimbările de luminozitate;
- Rezolvarea problemelor legate de perspectiva imaginilor și suprapunerea plantelor;
- Integrarea hardware a unor sisteme mecanice pentru combaterea buruienilor pe un robot mobil din dotarea Departamentului de Mecatronică și continuarea cercetărilor din domeniul abordat.
- Analiza posibilităților de utilizare a imaginilor termice ale plantelor pentru recunoașterea lor.

## Lista publicațiilor personale

- **Tilneac M.**, Dolga V., 2010 – „Individual Plant Recognition Using the RGB Color Model”, Proc. IEEE Mediterranean Electrotechnical Conference, IEEE MELECON 2010, Valletta, Malta, ISBN 978-1-4244-5794-6, Aprilie 2010, Pag.. 1147-1152. Lucrare indexată ISI - Web of Knowledge.
- **Tilneac M.**, Dolga V. – „Extrinsic Calibration of a Multi-Camera Network Used for Individual Plant Phenotyping”, Proceedings of Intelligent Computer Communication and Processing (ICCP), 2011 IEEE International Conference, Cluj-Napoca, Romania, ISBN 978-1-4577-1479-5, August 2011, Pag. 353 – 359. Lucrare indexată BDI.
- **Tilneac M.**, Dolga V., 2009 – „Agricultural Robots: Individual Plant Recognition”, Proc. Danube Adria Association for Automation & Manufacturing, DAAAM 2009, Viena, Austria, ISSN 1726-9679, Noiembrie 2009, Pag. 969-970. Lucrare indexată ISI - Web of Knowledge.
- **Tilneac M.**, Grigorescu S., Paléologue V., Dolga V., 2010 – „Wrist Mounted Camera for Colour and Shape Recognition”, Proceedings of the 5th International Conference on Optimization of the Robots and Manipulators, OPTIROB 2010, Călimănești, Romania, ISBN 978-981-08-5840-7, Mai 2010, Pag. 200-204. Lucrare indexată BDI.
- **Tilneac M.**, Dolga V., 2011 – „Extrinsic Camera Calibration and Camera-Matching for Plant Phenotyping Using CAD (Computer-Aided Design) Models”, The 10-th International Conference on Mechatronics and Precision Engineering, COMEFIM 10, București, Romania, Mai 2011, Proceedings of the International Journal Romanian Review Precision Mechanics, Optics & Mechatronics, edited by INCDMTM, București, România, ISSN 1584-5982, Volume 39, 2011, Pag.. 45-50. Lucrare indexată BDI
- Grigorescu S., **Tilneac M.**, Dolga V., Paléologue V., 2010 – „Wrist Mounted Camera for Leaves Shape Recognition”, Proceedings of the International Journal ROBOTICA & MANAGEMENT, Timișoara, România, ISSN 1453-2069, Volume 15, Issue 1, Pag. 47-50.
- Bussemeyer L., Klose R., Linz A., Thiel M., Wunder E., Ruckelshausen A., **Tilneac M.**, 2010 - „Agro-sensor systems for outdoor plant phenotyping platforms in low and high density crop field plots”, Proceedings of the 68th International Conference Land-Technik 2010, Braunschweig, Germania, ISBN 978-3-18-092111-2, Pag. 213 - 218. Lucrare indexată BDI.

**148** Lista publicațiilor personale

---

- Feldkämper N., Hufendiek H., Jahn H., Kampmeyer C., Kerksen S., Lemke C., Mosler D., Rave P., Simon S., **Tilneac M.**, Westerhoff J., Wunder E., 2010 - „Team Optimaizer”, Proceedings of the 8th Field Robot Event 2010, 11-13 Iunie 2010, Braunschweig, Germania, Pag. 185-202.
- **Tilneac M.**, Dolga V., Grigorescu S., Bîtea M. A., 2012 - „3D stereo vision measurements for weed-crop discrimination”, ELECTRONICS AND ELECTRICAL ENGINEERING, ISSN 1392 – 1215. (În curs de publicare.)

## Bibliografie

### A

[Agrawal, 2003] – Agrawal M., Davis L. S., 2003 – „Camera calibration using spheres: A semi-definite programming approach“, IEEE International Conference on Computer Vision 2003, ISBN 0-7695-1950-4, Volume 2, Pag. 782-789.  
[http://ljk.imag.fr/membres/Bill.Triggs/events/iccv03/cdrom/iccv03/0782\\_agrawal.pdf](http://ljk.imag.fr/membres/Bill.Triggs/events/iccv03/cdrom/iccv03/0782_agrawal.pdf) (13.12.2010)

[Alenya, 2011] – Alenya G., Dellen B., Torras C., 2011 – „3D modelling of leaves from color and ToF data for robotized plant measuring“, Robotics and Automation (ICRA), IEEE International Conference, ISSN 1050-4729, Pag. 3408-3414.  
[http://www.iri.upc.edu/research/webprojects/pau/IMATGES/figspapers/Alenya\\_icra11.pdf](http://www.iri.upc.edu/research/webprojects/pau/IMATGES/figspapers/Alenya_icra11.pdf) (02.09.2011)

[Andersen, 2002] – Andersen H. J., 2002 – „Outdoor Computer Vision and Weed Control“, Ph.D. Thesis, Aalborg University, Denmark  
<http://www.cvmt.dk/~hja/publications/all.pdf> (12.12.2008)

[Andersen, 2005a] – Andersen H. J., Reng L., Kirk K., 2005 – „Geometric plant properties by relaxed stereo vision using simulated annealing“, Computers and Electronics in Agriculture, Volume 49, Issue 2, Pag. 219-232.  
[www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Andersen, 2005b] – Andersen H. J., Kirk K., Dideriksen T.L., Madsen C., Holte M.B., Bak T., 2005 – „Obstacle detection by stereo vision, introducing the PQ method“, Proceedings of the Second International Conference on Informatics in Control, Automation and Robotics, ICINCO 2005, Pag. 250 – 257.  
<http://www.thomasdideriksen.dk/misc/pqmethod.pdf> (19.12.2011)

[Astrand, 2002] – Astrand B., Baerveldt A. J., 2002 – „An Agricultural Mobile Robot with Vision-Based Perception for Mechanical Weed Control“, Autonomous Robots 13, Pag. 21–35.  
<http://www.springerlink.com/content/04u07q23cwmxe3p5/> (22.06.2011)

[Astrand, 2005] – Astrand B., Baerveldt A. J., 2005 – „A vision based row-following system for agricultural field machinery“, Mechatronics, Volume 15, Issue 2, Pag. 251-269. [www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

### B

[Backman, 2008] – Backman J., Hyyti H., Kalmari J., Kinnari J., Hakala A., Poutiainen V., Tamminen P., Vaatainen H., Oksanen T., Kostamo J., Tiisanen J.,

## 150 Bibliografie

---

2008 - "4M - Mean Maize Maze Machine", Proceedings of the 6th Field Robot Event, Pag. 9-39.

[http://www.fieldrobot.nl/downloads/Proceedings\\_FRE2008.pdf](http://www.fieldrobot.nl/downloads/Proceedings_FRE2008.pdf) (08.01.2012)

[Bakker, 2007] - Bakker T., Wouters H., Asselt K., Bontsema J., Tang L., Müller J., van Straten G., 2008 - „A vision based row detection system for sugar beet“, Computers and Electronics in Agriculture, Volume 60, Issue 1, Pag. 87-95, [www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Bakker, 2009a] - Bakker T., van Asselt K., Bontsema J., Müller J., van Straten G., 2010 - „Systematic design of an autonomous platform for robotic weeding“, Journal of Terramechanics, Volume 47, Issue 2, Pag. 63-73. [www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Bakker, 2009b] - Bakker T., 2009 - „An Autonomous Robot for Weed Control - Design, Navigation and Control“, Ph.D. Thesis, Wageningen University, The Netherlands. <http://edepot.wur.nl/1099> (24.06.2011)

[Bakker, 2011] - Bakker T., van Asselt K., Bontsema J., Müller J., van Straten G., 2011 - „Autonomous navigation using a robot platform in a sugar beet field“, Biosystems Engineering, [www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Beardsley, 1992] - Beardsley P., Murray D., Zisserman A., 1992 - „Camera Calibration Using Multiple Images“, Proceedings of the 2nd European Conference on Computer Vision, Pag. 312-320.

[http://www.robots.ox.ac.uk/ActiveVision/Publications/beardsley\\_etal\\_eccv1992/beardsley\\_etal\\_eccv1992.pdf](http://www.robots.ox.ac.uk/ActiveVision/Publications/beardsley_etal_eccv1992/beardsley_etal_eccv1992.pdf) (13.12.2010)

[Beghin, 2010] - Beghin T., Cope J. S., Remagnino P., Barman S., 2010 - „Shape and texture based plant leaf classification“, International Conference on Advanced Concepts for Intelligent Vision Systems (ACVIS), 2010, Pag. 345-353.

[http://www.computing.surrey.ac.uk/morphidas/Papers/acivs2010\\_Beghin.pdf](http://www.computing.surrey.ac.uk/morphidas/Papers/acivs2010_Beghin.pdf) (19.12.2011)

[Benson, 2003] - Benson E. R., Reid J. F., Zhang Q., 2003 - „Machine vision-based guidance system for an agricultural small-grain harvester“, Trans. ASAE 46(4), Pag. 1255-1264.

<http://age-web.age.uiuc.edu/faculty/qzhang/Publications/2003TransASAE46%284%29Benson.pdf> (22.12.2011)

[Billingsley, 2008] - Billingsley J., Visala A., Dunn M., 2008 - „Robotics in Agriculture and Forestry“, Handbook of Robotics, ISBN: 978-3-540-23957-4, Springer

[Biskup, 2009] - Biskup B., 2009 - „Determination of structural parameters of single plants and canopies using 3D techniques“, Ph.D Thesis, Heinrich-Heine-University Dusseldorf, Germany.

<http://docserv.uni-duesseldorf.de/servlets/DerivateServlet/Derivate-12536/Dissertation%20Biskup.pdf> (22.12.2011)

[Blas, 2010] - Blas M. R., 2010 - „Fault-Tolerant Vision for Vehicle Guidance in Agriculture”, Ph.D Thesis, Technical University of Denmark  
[http://www.google.ro/#q=stereo+vision+agriculture&hl=de&prmd=imvns&ei=FIrvTtL\\_CzsOdTE3KwI&start=40&sa=N&bav=on.2.or.r\\_gc.r\\_pw..cf.osb&fp=f0cba6f953ff7bfb&biw=1390&bih=549](http://www.google.ro/#q=stereo+vision+agriculture&hl=de&prmd=imvns&ei=FIrvTtL_CzsOdTE3KwI&start=40&sa=N&bav=on.2.or.r_gc.r_pw..cf.osb&fp=f0cba6f953ff7bfb&biw=1390&bih=549) (19.12.2011)

[Bossu, 2008] - Bossu J., Gée Ch., Jones G., Truchetet F., 2009 - „Wavelet transform to discriminate between crop and weed in perspective agronomic images”, Computers and Electronics in Agriculture, Volume 65, Issue 1, Pag. 133-143.  
[www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Bouguet] - Bouguet J. Y. - “Camera Calibration Toolbox for Matlab”, Online.  
[http://www.vision.caltech.edu/bouguetj/calib\\_doc/](http://www.vision.caltech.edu/bouguetj/calib_doc/) (09.01.2012)

[Burgos, 2008] - Burgos-Artizzu X., Ribeiro A., Tellaache A., Pajares G., Fernández-Quintanilla C., 2009 - „Improving weed pressure assessment using digital images from an experience-based reasoning approach” - Computers and Electronics in Agriculture, Volume 65, Issue 2, March 2009, Pag. 176-185. [www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Burgos, 2009] - Burgos-Artizzu X., Ribeiro A., Tellaache A., Pajares G., Fernández-Quintanilla C., 2010 - „Analysis of natural images processing for the extraction of agricultural elements”, Image and Vision Computing, Volume 28, Issue 1, Pag. 138-149. [www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Burgos, 2010] - Burgos-Artizzu X., Ribeiro A., Guijarro M., Pajares G., 2011, - „Real-time image processing for crop/weed discrimination in maize fields”, Computers and Electronics in Agriculture, Volume 75, Issue 2, Pag. 337-346. [www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Busemeyer, 2010] - Busemeyer L., Klose R., Linz A., Thiel M., Wunder E., Ruckelshausen A., Tilneac M., 2010 - „Agro-sensor systems for outdoor plant phenotyping platforms in low and high density crop field plots”, Proceedings of the 68th International Conference Land-Technik 2010, Braunschweig, Germania, ISBN 978-3-18-092111-2, Pag. 213 - 218.

## C

[Cao, 2005] - Cao X., Shah M., 2005 - „Camera Calibration and Light Source Estimation from Images with Shadows”, IEEE Conference, Computer Vision and Şablon Recognition, ISBN: 0-7695-2372-2, Volume 2, Pag. 918-923.  
[http://server.cs.ucf.edu/~vision/papers/Extracting\\_CVPR.pdf](http://server.cs.ucf.edu/~vision/papers/Extracting_CVPR.pdf) (13.12.2010)

[Chaki, 2011] - Chaki J., Parekh R., 2011 - „Plant Leaf Recognition using Shape based Features and Neural Network classifiers”, International Journal of Advanced Computer Sciences and Applications, Volume 2, Issue 10, Pag. 41-47.  
<http://www.thesai.org/Downloads/Volume2No10/Paper%207-Plant%20Leaf%20Recognition%20using%20Shape%20based%20Features%20and%20Neural%20Network%20classifiers.pdf> (19.12.2011)

## 152 Bibliografie

---

[Chen, 2003] - Chen X., Yang J., Waibel A., 2003 - „Calibration of a Hybrid Camera Network“, IEEE International Conference on Computer Vision, ISBN 0-7695-1950-4, Volume 1, Pag. 150-155.  
[http://lear.inrialpes.fr/people/triggs/events/iccv03/cdrom/iccv03/0150\\_chen.pdf](http://lear.inrialpes.fr/people/triggs/events/iccv03/cdrom/iccv03/0150_chen.pdf)  
(13.12.2010)

[Chen, 2004] - Chen Q., Wu H., Wada T., 2004 - „Camera Calibration with Two Arbitrary Coplanar Circles“, European Conference on Computer Vision, Volume 3, Pag. 521-532.  
<http://www.wakayama-u.ac.jp/~wuhy/Eccv2004Final.pdf> (22.12.2011)

[Ciocarlie, 2011] - Ciocarlie M., Pantofaru C., Hsiao K., Bradski G., Brook P., Dreyfuss E., 2011 - „A Side of Data with My Robot“, IEEE ROBOTICS&AUTOMATION MAGAZINE, ISSN 1070-9932, Volume 18, Issue 2, Pag. 44-57.

[Ciolac, 2003] - Ciolac V., Popescu C., Bîrlă G. A., Ciotlăuș A., Rață G., 2003 - „Tehnologia GPS în Agricultură“, Editura Mirton, ISBN 973-661-026-8.

## D

[Datta, 2009] - Datta A., Kim J. S., Kanade T., 2009 - „Accurate Camera Calibration using Iterative Refinement of Control Points“, IEEE Conference, Computer Vision, ISBN 978-1-4244-4442-7, Pag. 1201-1208.  
[http://www.ri.cmu.edu/pub\\_files/2009/10/VS.ICCV.2009.pdf](http://www.ri.cmu.edu/pub_files/2009/10/VS.ICCV.2009.pdf) (13.12.2010)

[Dedousis, 2007] - Dedousis A., 2007 - „An Investigation into the Design of Precision Weeding Mechanisms for Inter and Intra-Row Weed Control“, Ph.D Thesis, Cranfield University, United Kingdom.  
<https://dspace.lib.cranfield.ac.uk/bitstream/1826/2009/1/Dedousis%20PhD%20The%20sis.pdf> (12.12.2008)

[Deutscher, 2002] - Deutscher J., Isard M., MacCormick J., 2002 - „Automatic Camera Calibration from a Single Manhattan Image“, The 7th European Conference on Computer Vision Copenhagen, Volume 4, Pag. 175-188.  
<http://users.dickinson.edu/~jmac/publications/calibrate.pdf> (13.12.2010)

[Do, 1999] - Do Y., 1999 - „Application of Neural Networks for Stereo-Camera Calibration“, International Joint Conference on Neural Networks  
[http://www.ri.cmu.edu/pub\\_files/pub2/do\\_yongtae\\_1999\\_1/do\\_yongtae\\_1999\\_1.pdf](http://www.ri.cmu.edu/pub_files/pub2/do_yongtae_1999_1/do_yongtae_1999_1.pdf)  
(13.12.2010)

[Dodoc, 1989] - Dodoc P., 1989 - „Teoria și construcția aparatelor optice“, Editura Tehnică, ISBN 973-31-0030-7.

[Drummond, 2002] - Drummond T., Cipolla R., 2002 - „Real-Time Tracking of Complex Structures with on-line Camera Calibration“, Image and Vision Computing, Volume 20, Issues 5-6, Pag. 427-433.  
<http://www.sciencedirect.com> (13.12.2010)

[Du, 2006] - Du J. X., Huang D. S., Wang X. F., Gu X., 2006 - „Computer-Aided Plant Species Identification (CAPSI) Based on Leaf Shape Matching Technique“,



Transactions of the Institute of Measurement and Control, Volume 28, Issue 3, Pag. 275-285.

[http://scholar.google.ro/scholar?start=40&q=leaf+recognition&hl=de&as\\_sdt=0](http://scholar.google.ro/scholar?start=40&q=leaf+recognition&hl=de&as_sdt=0)  
(08.10.2011)

## E

[Ericson, 2008] – Ericson E., Astrand B., 2008 – „Visual Odometry System for Agricultural Field Robots”, Proceedings of the World Congress on Engineering and Computer Science 2008, San Francisco, USA.  
[http://scholar.google.ro/scholar?start=40&q=weed+control+robot&hl=de&as\\_sdt=0&as\\_ylo=2008](http://scholar.google.ro/scholar?start=40&q=weed+control+robot&hl=de&as_sdt=0&as_ylo=2008) (14.06.2011)

[Ericson, 2010] - Ericson E., Astrand B., 2010 – „Row-detection on an agricultural field using omnidirectional camera”, The 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, ISBN: 978-1-4244-6674-0, Pag. 4982 – 4987.  
<http://ieeexplore.ieee.org/Xplore/questhome.jsp> (20.09.2011)

## F

[Faugeras, 1992] – Faugeras O. D., Luong Q. T., Maybank S. J., 1992 – „Camera self-calibration: Theory and experiments”, Second European Conference on Computer Vision, Santa Margherita Ligure, Italy.  
[http://reference.kfupm.edu.sa/content/c/a/camera\\_self\\_calibration\\_theory\\_and\\_exe\\_73144.pdf](http://reference.kfupm.edu.sa/content/c/a/camera_self_calibration_theory_and_exe_73144.pdf) (13.12.2010)

[Feldkämper, 2010] - Feldkämper N., Hufendiek H., Jahn H., Kampmeyer C., Kerksen S., Lemke C., Mosler D., Rave P., Simon S., Tilneac M., Westerhoff J., Wunder E., 2010 – „Team Optimaizer”, Proceedings 8th Field Robot Event 2010, 11-13 June 2010, Braunschweig, Germany, Pag. 185-202.  
[http://rzbl04.biblio.etc.tu-bs.de:8080/docportal/servlets/MCRFileNodeServlet/DocPortal\\_derivate\\_00021118/Proceedings\\_8th\\_FieldRobotEvent\\_2010.pdf;jsessionid=AF38FADB9C02737474D80F2263B9EAA7](http://rzbl04.biblio.etc.tu-bs.de:8080/docportal/servlets/MCRFileNodeServlet/DocPortal_derivate_00021118/Proceedings_8th_FieldRobotEvent_2010.pdf;jsessionid=AF38FADB9C02737474D80F2263B9EAA7) (08.01.2012)

[Fender, 2006] – Fender F., Hanneken M., In der Stroh S., Kielhorn A., Linz A., Ruckelshausen A., 2006 – „Sensor fusion meets GPS – Individual plant detection”, Proceedings of CIRG EurAgEng/VDI-MEG.

[Fiala, 2010] - Fiala M., Shu C., 2010 – „Fully Automatic Camera Calibration Using Self-Identifying Calibration Targets”, National Research Council of Canada,  
<http://nparc.cisti-icist.nrc-cnrc.gc.ca/npsi/ctrl?action=shwart&index=an&req=8913774&lang=en> (13.12.2010)

[Fung, 2003] - Fung G., Yung N., Pang G., 2003 - „Camera calibration from road lane markings”, Optical Engineering, Volume 42, Issue 10, Pag. 2967-2977.  
<http://hub.hku.hk/bitstream/10722/42976/1/91626.pdf> (13.12.2010)

**G**

[Gobor, 2007] – Gobor Z., 2007 – “Development of a novel mechatronic system for mechanical weed control of the intra-row area in row crops based on detection of single plants and adequate controlling of the hoeing tool in real-time”, Ph.D Thesis, Bonn University, Germany.

[http://www.landtechnik.uni-bonn.de/ifl\\_research/pp\\_12/gobor\\_dis\\_final\\_2008\\_01\\_17.pdf](http://www.landtechnik.uni-bonn.de/ifl_research/pp_12/gobor_dis_final_2008_01_17.pdf) (12.12.2008)

[Ghazali, 2008] – Ghazali K. H., Mustafa M. M., Hussain A., 2008 – „Machine Vision System for Automatic Weeding Strategy using Image Processing Technique”, American-Eurasian J. Agric. & Environ. Sci., 3 (3), Pag. 451-458. [http://www.idosi.org/aejaes/jaes3\(3\)/23.pdf](http://www.idosi.org/aejaes/jaes3(3)/23.pdf) (14.01.2012)

[Gonzalez, 2009] - Gonzalez R., Woods R., Eddins S., 2009 – „Digital Image Processing Using MATLAB”, ISBN 9780982085400.

[Griepentrog, 2006] - Griepentrog H. W., Nørreremark M., Nielsen J., 2006 – “Autonomous intra-row rotor weeding based on GPS”, Proceedings CIGR World Congress - Agricultural Engineering for a Better World, Bonn, Germany <http://www.staff.kvl.dk/~hwg/pdf/papers/Griepentrog2006b.pdf> (12.11.2011)

[Griepentrog, 2007] - Griepentrog H. W., Blackmore B. S., 2007 – „Autonomous Crop Establishment and Control System”, Proceedings Land-Technik Engineering the Future 2007 - Engineering Solutions for Energy and Food Production, Hanover, Germany, VDI-Verlag, Duesseldorf, Germany, Pag. 175-181. <http://www.staff.kvl.dk/~hwg/pdf/papers/Griepentrog2007AgEng.pdf> (12.11.2011)

[Griepentrog, 2010] – Griepentrog H.W., Ruckelshausen A., Jørgensen R.N., Lund I., 2010 – „Autonomous systems for plant protection”, Precision Crop Protection, The Challenge and Use of Heterogeneity, Springer, Pag. 323-334.

[Grigorescu, 2010] - Grigorescu S., Tilneac M., Dolga V., Paléologue V., 2010 – „Wrist Mounted Camera for Leaves Shape Recognition”, Proceedings of the International Journal ROBOTICA & MANAGEMENT, Timișoara, România, ISSN 1453-2069, Volume 15, Issue 1, Pag. 47-50.

[Gu, 2005] - Gu X., Du J. X., Wang X. F., 2005 - „Leaf Recognition Based on the Combination of Wavelet Transform and Gaussian Interpolation”, International Conference on Intelligent Computing, ICIC 2005, Hefei, China. [http://www.intelengine.cn/doc/2006/LNCS\\_2005/Gu%20X.pdf](http://www.intelengine.cn/doc/2006/LNCS_2005/Gu%20X.pdf) (08.10.2011)

[Gui, 1999] – Gui V., 1999 – „Prelucrarea Imaginilor”, Editura Politehnica Timișoara, ISBN 973-9389-26-0.

[Guijarro, 2010] – Guijarro M., Pajares G., Riomoros I., Herrera P.J., Burgos-Artizxu X.P., Ribeiro A., 2011 – „Automatic segmentation of relevant textures in agricultural images”, Computers and Electronics in Agriculture, Volume 75, Issue 1, Pag. 75-83, [www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Guyer, 1986] - Guyer D. E., Miles G. E., Schreiber M. M., Mitchell O. R., Vanderbilt V.C., 1986 - „Machine vision and image processing for plant identification”, Trans. ASAE 29(6), Pag. 1500-1507.  
<http://www.ask-force.org/web/Precision-Biotechnology/Guyer-Machine-Vision-1986.pdf> (19.12.2011)

## H

[Hague, 2000] - Hague T., Marchant J. A., Tillett N. D., 2000 - „Ground based sensing systems for autonomous agricultural vehicles”, Computers and Electronics in Agriculture 25, Pag. 11-28.  
<http://www8.cs.umu.se/research/ifor/dl/LOCALIZATION-NAVIGATION/Ground%20based%20sensing%20systems.pdf> (22.12.2011)

[Harris, 1988] - Harris C. G., Stephens M. J., 1988 - „A combined corner and edge detector”, Proceedings Fourth Alvey Vision Conference, Pag. 147-151.  
<http://walle.sdsmt.edu/faculty/rhoover/teaching/F11/692/Papers/HarrisStephens.pdf> (12.12.2011)

[Hâruta, 2011] - Hâruta O., 2011 - „Elliptic Fourier analysis of crown shapes in Quercus petraea trees”, Annals of Forest Research, Volume 54, Issue 1, Pag.. 99-117. <http://www.editurasilvica.ro/afr/54/1/Hâruta.pdf> (19.12.2011)

[Heikkila, 2000] - Heikkila J., 2000 - „Geometric Camera Calibration Using Circular Control Points”, IEEE Şablon Analysis and Machine Intelligence, ISSN: 0162-8828, Volume 22, Issue 10, Pag. 1066-1077.  
<http://wcours.gel.ulaval.ca/2006/a/19263/default/7references/heikkila2000PAMI.pdf> (13.12.2010)

[Hemming, 2001] - Hemming J., Rath T., 2001 - „Computer-vision-based weed identification under field conditions using controlled lighting”, J. Agric. Eng. Res. 78 (3), Pag. 233-243, <http://www.hemming.nl/jh/bib/jaer2000.pdf> (15.11.2011)

[Hemming, 2002] - Hemming J., Rath T., 2002 - „Image Processing for Plant Determination Using the Hough Transform and Clustering Methods”, Gartenbauwissenschaft, 67 (1), Pag. 1-10.  
<http://www.hemming.nl/jh/bib/Gawi2002.pdf> (12.12.2011)

[Henten, 2002] - van Henten E.J., Hemming J., van Tuijl B.A.J., Kornet J.G., Meuleman J., Bontsema J., van Os E.A., 2002 - „An Autonomous Robot for Harvesting Cucumbers in Greenhouses”, Autonomous Robots, Volume 13, Issue 3, Pag. 241-258. [www.springerlink.com](http://www.springerlink.com) (27.06.2011)

[Home, 2002] - Home M. C. W., Tillett N. D., Hague T., Godwin R. J., 2002 - „An experimental study of lateral positional accuracy achieved during inter-row cultivation”, 5th EWRS Workshop on Physical and Cultural Weed Control, Pag. 101-110. [http://www.ewrs.org/pwc/doc/2002\\_Pisa.pdf](http://www.ewrs.org/pwc/doc/2002_Pisa.pdf) (12.12.2008)

[Huang, 2009] - Huang Y. J., Lee F. F., 2010 - „An automatic machine vision-guided grasping system for Phalaenopsis tissue culture plantlets”, Computers and Electronics in Agriculture, Volume 70, Issue 1, January 2010, Pag. 42-51,

[www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

## J

[Jafari, 2004] – Jafari A., Mohtasebi S. S., Jahromi H. E., Omid M., 2004 – „Color Feature Extraction by Means of Discriminant Analysis for Weed Segmentation“, ASAE/CSAE Annual International Meeting  
<http://utcan.ut.ac.ir/members/omid/Webconf/conf04.pdf> (12.12.2008)

[Jelacic, 2006] – Jelacic M., 2006 – „Unsupervised Learning for Plant Recognition“, Master Thesis, Computer and Electrical Engineering, Halmstad University, Sweden.  
<http://dSPACE.hh.se/dSPACE/bitstream/2082/542/1/0620%20MJ.pdf> (12.12.2008)

[Jeon, 2009] – Jeon H., Tian L., 2009 - „Direct application end effector for a precise weed control robot“, Biosystems Engineering, Volume 104, Issue 4, Pag. 458-464.  
<http://www.sciencedirect.com/science/article/pii/S1537511009002724>  
(27.06.2011)

[Jeon, 2011] – Jeon H., Tian L., Zhu H., 2011 - „Robust Crop and Weed Segmentation under Uncontrolled Outdoor Illumination“,  
<http://www.mdpi.com/1424-8220/11/6/6270/pdf> (14.06.2011)

[Ji, 2001] – Ji Q., Zhang Y., 2001 – „Camera Calibration with Genetic Algorithms“, IEEE Systems, Man and Cybernetics, ISSN: 1083-4427, Volume 31, Issue 2, Pag. 120-130.  
[http://www.ecse.rpi.edu/homepag./qji/Papers/genetic\\_calib\\_smc.pdf](http://www.ecse.rpi.edu/homepag./qji/Papers/genetic_calib_smc.pdf) (13.12.2010)

[Jiang, 2010] - Jian G. Q., ZHAO C. J., Si Y. S., 2010 - „A Machine Vision Based Crop Rows Detection For Agricultural Robots“, Proceedings of the 2010 International Conference on Wavelet Analysis and Şablon Recognition, Qingdao, 11-14 July 2010,  
<http://ieeexplore.ieee.org/Xplore/questhome.jsp> (20.09.2011)

[Joochim, 2010] – Joochim C. and Roth H., 2010 - „Mobile Robot Exploration Based On Three Dimension Cameras Acquisition“, presented at the 2nd IFAC Symposium on Telematics Applications - TA2010, Timisoara, Romania

[Jørgensen, 2007] – Jørgensen R.N., Sørensen C.G., Maagaard J., Havn I., Jensen K., Søgaard H.T., Sørensen L.B., 2007 – „HortiBot: A System Design of a Robotic Tool Carrier for High-tech Plant Nursing“, Agricultural Engineering International: the CIGR E-Journal.  
<http://ecommons.cornell.edu/bitstream/1813/10601/1/ATOE%2007%20006%20Jørgensen%2011July2007.pdf> (20.01.2012)

## K

[Kadir, 2011a] - Kadir A., Nugroho L. E., Susanto A., Santosa P. I., 2011 – „A Comparative Experiment of Several Shape Methods in Recognizing Plants“. International Journal of Computer Science & Information Technology (IJSIT) , 3 (3), Pag. 256-263.  
<http://ijcttjournal.org/volume-1/Issue-3/IJCTT-V1I3P130.pdf> (22.12.2011)

- [Kadir, 2011b] - Kadir A., Nugroho L. E., Susanto A., Santosa P. I., 2011 - „Foliage Plant Retrieval Using Polar Fourier Transform, Color Moments and Vein Features”, Signal & Image Processing : An International Journal (SIPIJ) Vol.2, No.3, September 2011, <http://arxiv.org/ftp/arxiv/papers/1110/1110.1513.pdf> (22.12.2011)
- [Kebapci, 2010] - Kebapci H., Yanikoglu B., Unal G., 2010 - „Plant Image Retrieval Using Color, Shape and Texture Features”, The Computer Journal , 53 (1). <https://research.sabanciuniv.edu/14086/1/compj-plants.pdf> (15.11.2011)
- [Kim, 2002] - Kim J. S., Kim H. W., Kweon I. S., 2002 - „A Camera Calibration Method using Concentric Circles for Vision Applications”, The 5th Asian Conference on Computer Vision. [http://www.aprs.org.au/accv2002/accv2002\\_proceedings/Kim515.pdf](http://www.aprs.org.au/accv2002/accv2002_proceedings/Kim515.pdf) (14.09.2011)
- [Kise, 2005] - Kise M., Zhang Q., Rovira Mas F., 2005 - „A stereovision-based crop row detection method for tractor-automated guidance”, Biosyst. Eng. 90 (4), Pag. 357-367. <http://age-web.age.uiuc.edu/faculty/qzhang/Publications/2005BE90%284%29Kise.pdf> (15.11.2011)
- [Klose, 2008] - Klose R., Ruckelshausen A., Thiel M., Marquering J., 2008 - „Weedy - a Sensor Fusion Based Autonomous Field Robot for Selective Weed Control”, International Conference Agricultural Engineering/AgEng, Pag. 167-172.
- [Klose, 2009] - Klose R., Penlington J. and Ruckelshausen A., 2009 - „Usability Study of 3D Time-of-Flight Cameras for Automatic Plant Phenotyping”, The 1st International Workshop on Computer Image Analysis in Agriculture, Potsdam, Germany, 2009, Available : [http://www.atb-potsdam.de/CIGR-ImageAnalysis/images/09\\_110\\_Klose.pdf](http://www.atb-potsdam.de/CIGR-ImageAnalysis/images/09_110_Klose.pdf) (13.09.2011)
- [Klose, 2010] - Klose R., Möller K., Vielstädte C. and Ruckelshausen A., 2010 - „Modular System Architecture for Individual Plant Phenotyping with an Autonomous Field Robot”, presented at the 2nd International Conference on Machine Control & Guidance, Bonn, Germany, 2010, Available : [http://www.mcg.uni-bonn.de/proceedings/34\\_klose.pdf](http://www.mcg.uni-bonn.de/proceedings/34_klose.pdf) (13.09.2011)
- [Koterba, 2005] - Koterba S., Baker S., Matthews I., Hu C., Xiao J., Cohn J., Kanade T., 2005 - „Multi-View AAM Fitting and Camera Calibration”, Tenth IEEE International Conference on Computer Vision, ISBN: 0-7695-2334-X, Volume 1, Pag. 1550-5499. [http://www.ri.cmu.edu/pub\\_files/pub4/koterba\\_seth\\_c\\_2005\\_1/koterba\\_seth\\_c\\_2005\\_1.pdf](http://www.ri.cmu.edu/pub_files/pub4/koterba_seth_c_2005_1/koterba_seth_c_2005_1.pdf) (13.12.2010)
- [Kovacs, 2001] - Kovacs F., Rădulescu C., Varga Ş., Grigorescu S., Diaconu A., 2001 - „Sisteme de Fabricație Flexibilă – Robotică”, Editura Orizonturi Universitare, ISBN 973-8109-61-2.
- [Kroulik, 2010] - Kroulík M., Roubíček J., Roubíček T., Dlouhý M., 2010 - "Eduro Maxi HD – Navigation in the maize", Proceedings of the 8th Field Robot Event, Pag. 82-87.

## L

[Lee, 1998] - Lee W. S., 1998 - „Robotic weed control system for tomatoes”, Ph.D Thesis, University of California, Department of Biological and Agricultural Engineering, USA.  
<http://www.abe.ufl.edu/wlee/Dissertation/RoboticWeedControlSystem-WonSukLee.pdf> (15.11.2011)

[Lee, 1999] - Lee W. S., Slaughter D. C., Giles D. K., 1999 - „Robotic Weed Control System for Tomatoes”, [www.springerlink.com](http://www.springerlink.com) (27.06.2011)

[Leemans, 2006] - Leemans V., Destain M. F., 2006 - „Line cluster detection using a variant of the Hough transform for culture row localisation”, Image and Vision Computing, Volume 24, Issue 5, 1 May 2006, Pag. 541-550.  
[www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Lin, 2008] - Lin T. T., Hsiung Y. K., Hong G. L., Chang H. K., Lu F. M, 2008 - „Development of a virtual reality GIS using stereo vision”, Computer and Electronics in Agriculture, Volume 63, Pag. 38-48.  
[http://140.112.94.185/ttlin/uploads/tadnews/file/nsn\\_9\\_1.pdf](http://140.112.94.185/ttlin/uploads/tadnews/file/nsn_9_1.pdf) (19.12.2011)

[Lv, 2006] - Lv F., Zhao T., Nevatia R., 2006 - „Camera Calibration from Video of a Walking Human”, IEEE Şablon Analysis and Machine Intelligence, ISSN: 0162-8828, Volume 28, Issue 9, Pag. 1513-1518.  
[http://vipbase.net/homepage/calibration\\_pami.pdf](http://vipbase.net/homepage/calibration_pami.pdf) (13.12.2010)

## M

[Manea, 2002] - Manea D., 2002 - „Agrotehnică şi Herbologie”, Editura Orizonturi Universitare, ISBN 973-8391-74-1, Pag. 65-84.

[Marita, 2006] - Marita T., Oniga F., Nedevschi S., Graf T. and Schmidt R., 2006 - „Camera Calibration Method for Far Range Stereovision Sensors Used in Vehicles”, presented at the Intelligent Vehicles Symposium, Tokyo, Japan.  
[http://users.utcluj.ro/~vision/index\\_files/Publications/IVS2006-2.pdf](http://users.utcluj.ro/~vision/index_files/Publications/IVS2006-2.pdf) (13.09.2011)

[Memony, 2001] - Memony Q., Khan S., 2001 - "Camera calibration and three-dimensional world reconstruction of stereo-vision using neural networks", International Journal of Systems Science  
[http://www.cs.ucf.edu/~khan/p1155\\_s.pdf](http://www.cs.ucf.edu/~khan/p1155_s.pdf) (13.12.2010)

[Meyer, 2011] - Meyer G. E., 2011 - „Machine Vision Identification of Plants”, Recent Trends for Enhancing the Diversity and Quality of Soybean Products, ISBN 978-953-307-533-4.  
[http://www.intechopen.com/source/pdfs/22613/InTech-Machine\\_vision\\_identification\\_of\\_plants.pdf](http://www.intechopen.com/source/pdfs/22613/InTech-Machine_vision_identification_of_plants.pdf) (19.12.2011)

[Möller, 2010] - Möller J., 2010 - „Computer vision - A versatile technology in automation of agriculture machinery”, Journal of Agricultural Engineering, Volume 47, Issue 4, Pag. 28-36.

[http://www.clubofbologna.org/ew/documents/KNR\\_Moeller.pdf](http://www.clubofbologna.org/ew/documents/KNR_Moeller.pdf) (19.12.2011)

[Monta, 2005] - Monta M., Kondo N., Nakatsuka K., 2005 - „Human Sensing System for Safety Agricultural Robot”, Department of Agricultural Engineering, Okayama University, Japan.

[http://mama.agr.okayama-u.ac.jp/kenkyu/96/kenji/En\\_ken.html](http://mama.agr.okayama-u.ac.jp/kenkyu/96/kenji/En_ken.html) (12.12.2008)

## N

[Netaji, 2008] - Netaji H., Azimifar Z., Zamani M., 2008 - „Using Fast Fourier Transform for Weed Detection in Corn Fields”.

<http://ieeexplore.ieee.org/Xplore/questhome.jsp> (20.09.2011)

[Neto, 2006] - Neto J. C., Meyer G. E., Jones D. D., Samal A. K., 2006 - „Plant species identification using Elliptic Fourier leaf shape analysis”, Computer and Electronics in Agriculture, Issue 2, Pag. 121-134.

[http://trinity.engr.uconn.edu/~vamsik/dr\\_birge/elliptic\\_fourier\\_analysis.pdf](http://trinity.engr.uconn.edu/~vamsik/dr_birge/elliptic_fourier_analysis.pdf)  
(15.11.2011)

[Nielsen, 2005] - Nielsen J. D., Nielsen K. M., Bendtzen J., 2005 - „Design of Embedded System and Data Communication for an Agricultural Autonomous Vehicle”. <http://vbn.aau.dk/files/641398/62.pdf> (24.06.2011)

[Nieuwenhuizen, 2009] - Nieuwenhuizen A. T., 2009 - „Automated detection and control of volunteer potato plants”, Ph.D Thesis, Wageningen University, 2009, <http://edepot.wur.nl/11952#page=61> (02.09.2011)

[Noguchi, 1998] - Noguchi N., Reid J.F., Zhang Q., Tian L. F., 1998 - “Vision Intelligence for Precision Farming Using FUZZY Logic Optimized Genetic Algorithm and Artificial Neural Network”, An ASAE Meeting Presentation, UILU-ENG-98-7020.

<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.15.3302&rep=rep1&type=pdf> (14.01.2012)

## O

[Oberndorfer, 2006] - Oberndorfer T., 2006 - „Embedded vision system for intra-row Weeding”, Master Thesis, Computer and Electrical Engineering, Halmstad University, Sweden.

<http://dspace.hh.se/dspace/bitstream/2082/533/1/0623%20TO.pdf> (12.12.2008)

[Ortiz, 2006] - Ortiz J. M., Olivares M., 2006 - „A Vision Based Navigation System for an Agricultural Field Robot”.

<http://www.natalnet.br/anais/LARS2006/PDFFiles/83.pdf> (22.06.2011)

## P

[Paléologue, 2009] - Paléologue V., 2009 - „Reconnaissance d'image et manipulation robotique, Mode d'emploi et rapport de synthèse du projet à l'Université Polytechnique de Timisoara”.

[Pedersen, 2006] – Pedersen S. M., Fountas S., Have H., Blackmore B. S., 2006 – „Agricultural robots—system analysis and economic feasibility”, <http://www.unibots.com/Papers/PRAGAgRobotsEconomics.pdf> (24.06.2011)

[Pérez, 2000] - Pérez A. J., López F., Benlloch J. V., Christensen S, 2000 - „Colour and shape analysis techniques for weed detection in cereal fields”, *Computers and Electronics in Agriculture* 25(3): Pag. 197–212.  
[http://www.disca.upv.es/articulos/docs/revistas/AR001\\_00.PDF](http://www.disca.upv.es/articulos/docs/revistas/AR001_00.PDF) (20.11.2011)

[Persson, 2008] - Persson M., Astrand B., 2008 – „Classification of crops and weeds extracted by active shape models”, *Biosystems Engineering*, Volume 100, Issue 4, August 2008, Pag. 484-497. [www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Piron, 2008] - Piron A., Leemans V., Kleynen O., Destain M. F., 2008 – „Determination of plant height for weed detection in stereoscopic images”, *Agricultural Engineering Conference 2008*, Hersonissos, Greece.  
<http://orbi.ulg.ac.be/bitstream/2268/17767/1/Ageng2008.pdf> (19.12.2011)

[Piron, 2009] - Piron A., Leemans V., Lebeau F., Destain M. F., 2009 – „Improving in-row weed detection in multispectral stereoscopic image”, *Computers and Electronics in Agriculture*, Issue 69, Pag. 73-79.  
<http://orbi.ulg.ac.be/bitstream/2268/27435/1/Piron%20A,%20Compag%202.pdf> (19.12.2011)

[Piron, 2010] - Piron A., van der Heijden F, Destain M. F., 2011 – „Weed detection in 3D images” *Precision Agriculture Journal*, 12(5), Pag. 607-622.  
<http://orbi.ulg.ac.be/bitstream/2268/79753/1/weedddetection.pdf> (21.12.2011)

## R

[Rădulescu, 2008] – Rădulescu C., 2008 – „Curs de robotică avansată”

[Rădulescu, 2000] – Rădulescu C., 2000 – „Robocare și sisteme de robocare”, Editura Mirton, Timișoara, ISBN 973-585-258-6.

[Ramalingam, 2005] - Ramalingam S., Sturm P., Lodha S., 2005 – „Towards Complete Generic Camera Calibration”, *IEEE Computer Vision and Şablon Recognition*, ISBN 0-7695-2372-2, Volume 1, Pag. 1093-1098.  
<http://perception.inrialpes.fr/Publications/2005/RSL05a/RamalingamSturmLodha-cvpr05.pdf> (13.12.2010)

[Rath, 2002] - Rath T., Hemming J., Kawollek M., van Henten E., 2002 – „Maschinen lernen sehen – Autonome Robotersysteme im Gartenbau”, <http://www.hemming.nl/jh/bib/vz2002autonomesysteme.pdf> (22.06.2011)

[Rekleitis, 2005] - Rekleitis I., Dudek G., 2005 – „Automated Calibration of a Camera Sensor Network”, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, ISBN 0-7803-8912-3, Pag. 3384-3389.  
<http://www.cim.mcgill.ca/~yiannis/Publications/iros05-0987.pdf> (13.12.2010)



[Rovira, 2005] - Rovira-Mas F., Zhang Q., Reid J. F., 2005- „Creation of Three-dimensional Crop Maps based on Aerial Stereoimages“, Biosystems Engineering (2005) 90 (3), Pag. 251-259. <http://age-web.age.uiuc.edu/faculty/qzhang/Publications/2005BE90%283%29Rovira.pdf> (02.09.2011)

[Rovira, 2006] - Rovira-Más F., Zhang Q., Kise M., Reid J., 2006 – „Agricultural 3D Maps with Stereovision“, Position, Location, And Navigation Symposium, 2006 IEEE/ION, ISBN: 0-7803-9454-2, Pag. 1045-1053. <http://ieeexplore.ieee.org/Xplore/questhome.jsp> (20.09.2011)

[Rovira, 2007] - Rovira-Más, F., Zhang Q., Reid J. F., 2008 – „Stereo vision three-dimensional terrain maps for precision agriculture“, Computers and Electronics in Agriculture, Volume. 60, Pag. 133-143. <http://kjen.dk/bib/kj/2008%20Stereo%20vision%20three-dimensional%20terrain%20maps%20for%20precision%20agriculture.pdf> (19.12.2011)

[Rovira, 2011] - Rovira-Más F., 2011 – „Global 3D Terrain Maps for Agricultural Applications, Advances in Theory and Applications of Stereo Vision“, ISBN: 978-953-307-516-7, InTech [http://www.intechopen.com/source/pdfs/12975/InTech-Global\\_3d\\_terrain\\_maps\\_for\\_agricultural\\_applications.pdf](http://www.intechopen.com/source/pdfs/12975/InTech-Global_3d_terrain_maps_for_agricultural_applications.pdf) (19.12.2011)

[Ruckelshausen, 2006] – Ruckelshausen A., Klose R., Linz A., Marquering J., Thiel M., Tolke S., 2006 – „Autonome Roboter zur Unkrautbekämpfung“, Zeitschrift für Pflanzenkrankheiten und Pflanzenschutz, ISSN 1861-4051. <http://www.unibots.com/Papers/2006-Unkrauttagung.pdf> (12.12.2008)

[Ruckelshausen, 2007] – Ruckelshausen A., 2007 – „Autonomous robots in agricultural field trials“, Proceedings of the International Symposium Agricultural Field Trials – Today and Tomorrow, Stuttgart-Hohenheim, Pag. 190-198. <https://www.uni-hohenheim.de/bioinformatik/symposium/symposium%20englisch/vortraege/Ruckelshausen.pdf> (12.12.2008)

[Ruckelshausen, 2009] – Ruckelshausen A., Biber P., Dorna M., Gremmes H., Klose R., Linz A., Rahe R., Resch R., Thiel M., Trautz D. and Weiss U., 2009 – „BoniRob: an autonomous field robot platform for individual plant phenotyping“, European Conference Precision Agriculture (ECPA), Pag. 841-847.

## S

[Savii, 2004] - Savii G. G., 2004 – „Camera Calibration Using Compound Genetic-Simplex Algorithm“, Journal of Optoelectronics and Advanced Materials Vol. 6, No. 4, December 2004, Pag. 1255 – 1261. [http://www.inoe.ro/JOAM/pdf6\\_4/Savii.pdf](http://www.inoe.ro/JOAM/pdf6_4/Savii.pdf) (20.01.2012)

[Shah, 1996] – Shah S., Aggarwal J. K., 1996 – „Intrinsic Parameter Calibration Procedure for a (High-Distortion) Fish-Eye Lens Camera with Distortion Model and Accuracy Estimation“, Şablon Recognition, Volume 11, Pag. 1775-1788.

<http://cvrc.ece.utexas.edu/aggarwaljk/Publications/S.%20Shah,%20Intrinsic%20Parameter%20Calibration%20Procedure%20for%20fish%20eye%20lens.pdf>

(13.12.2010)

[Shih, 1993] – Shih S.W., Hung Y.P., Lin W.S., 1993 – „Accurate Linear Technique for Camera Calibration Considering Lens Distortion by Solving an Eigenvalue Problem”, *Optical Engineering*, Volume 32, Issue 1, Pag. 138-149.

<http://ntur.lib.ntu.edu.tw/bitstream/246246/141336/1/14.pdf> (13.12.2010)

[Shrestha, 2002] - Shrestha D.S., Steward B.L., Kaspar T.C., 2002 – „Determination of early stage corn plant height using stereo-vision”, *Proceedings of the 6th International Conference on Precision Agriculture*.

[http://www.webpag.uidaho.edu/~devs/Research/Publications/Peer%20Reviewed/16%20CP\\_02%20DETERMINATION%20OF%20EARLY%20STAGE%20CORN%20PLANT%20HEIGHT%20USING%20STEREO-VISION.pdf](http://www.webpag.uidaho.edu/~devs/Research/Publications/Peer%20Reviewed/16%20CP_02%20DETERMINATION%20OF%20EARLY%20STAGE%20CORN%20PLANT%20HEIGHT%20USING%20STEREO-VISION.pdf) (19.12.2011)

[Shrestha, 2005] - Shrestha D.S., Steward B.L., 2005 – „Shape and size analysis of corn plant canopies for plant population and spacing sensing”, *Applied Engineering in Agriculture*. 21(2): Pag. 295-303.

<http://www.public.iastate.edu/~bsteward/appliedengineercorncanopysshape.pdf> (22.12.2011)

[Singh, 2010] - Singh K., Gupta I., Gupta S., 2010 – „SVM-BDT PNN and Fourier Moment Technique for Classification of Leaf Shape”, *International Journal of Signal Processing, Image Processing and Şablon Recognition*, 3 (4), Pag.. 67-78.

[http://www.sersc.org/journals/IJSIP/vol3\\_no4/6.pdf](http://www.sersc.org/journals/IJSIP/vol3_no4/6.pdf) (20.12.2011)

[Slaughter, 1999] – Slaughter D.C., Chen P., Curley R. G., 1999 – „Vision guided precision cultivation”, *Precision Agriculture* 1 (2), Pag. 199–216.

<http://www.springerlink.com/content/h507n3784361654k/> (15.11.2011)

[Slaughter, 2000] - Slaughter D.C., Giles D.K., Lamm R.D., Lee, W.S., 2000 - „Robotic Weed Control Systems for California Row Crops”, *AGENG 2000*

<http://baesil.engineering.ucdavis.edu/files/ageng2000.pdf> (12.12.2008)

[Slaughter, 2008] – Slaughter D.C., Giles D.K., Downey D., 2008 - „Autonomous robotic weed control systems: A review”, *Computers and Electronics in Agriculture*, Volume 61, Issue 1, April 2008, Pag. 63-78, [www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Søgaard, 2003] - Søgaard H. T., Olsen H. J., 2003 – „Determination of crop rows by image analysis without segmentation” *Computers and Electronics in Agriculture*, 38, Pag. 141–158.

<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.107.7941> (15.11.2011)

[Søgaard, 2005] – Søgaard H.T., 2005 - „Weed Classification by Active Shape Models”, *Biosystems Engineering*, Volume 91, Issue 3, July 2005, Pag. 271-281, [www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Song, 2011] - Song Y., Glasbey C., Heijden G., Polder G., Dieleman A. – „Combining stereo and Time-of-Flight images with application to automatic plant phenotyping”, <http://www.bioss.ac.uk/staff/chris/scia11.pdf> (11.09.2011)

[Sørensen, 2005] - Sørensen M.J., 2005 - „Feedback Control of a Class of Nonholonomic Hamiltonian Systems”, Ph.D. Thesis, Aalborg University, Denmark.  
[http://vbn.aau.dk/files/1501472/mjs\\_thesis\\_final\\_31\\_10\\_05.pdf](http://vbn.aau.dk/files/1501472/mjs_thesis_final_31_10_05.pdf) (24.06.2011)

[Sørensen, 2009] - Sørensen C.G., Jørgensen R.N., Maagaard J., Bertelsen K.K., Dalgaard L., Nørremark M., 2009 - „Conceptual and user-centric design guidelines for a plant nursing robot”, Biosystems Engineering, Volume 105, Issue 1, January 2010, Pag. 119-129, [www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Sutton, 2009] - Sutton M., Ortu J.J., Schreier H., 2009 - „Image Correlation for Shape, Motion and Deformation Measurements”, Springer, ISBN 978-0-387-78746-6

## T

[Tănăsioiu, 1995] - Tănăsioiu H. S., 1995 - „Roboți de servicii în legumicultură”, Master Thesis, POLITEHNICA University of Timisoara, Romania.

[Tang, 2003] - Tang L., Tian L., Steward B. L., 2003 - „Classification of broadleaf and grass weeds using Gabor wavelets and an artificial neural network”, Transactions of the ASAE 46, Pag. 1247-1254.  
<http://www.public.iastate.edu/~bsteward/gaborwaveletpaper.pdf> (22.12.2011)

[Tellaèche, 2007] - Tellaèche A., Burgos-Artizzu X., Pajares G., Ribeiro A., 2007 - „A vision-based method for weeds identification through the Bayesian decision theory”, Şablon Recognition, Volume 41, Issue 2, February 2008, Pag. 521-530.  
[www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Tellaèche, 2008] - Tellaèche A., Burgos-Artizzu X., Pajares G., Ribeiro A., Fernandez-Quintanilla C. - „A new vision-based approach to differential spraying in precision agriculture”, Computers and electronics in agriculture 60 ( 2 0 0 8 ) Pag. 144-155. <http://digital.csic.es/bitstream/10261/7227/1/Comp-Elec-Agric-144-2008.pdf> (12.12.2008)

[Tellaèche, 2010] - Tellaèche A., Pajares G., Burgos-Artizzu X., Ribeiro A., 2010 - „A computer vision approach for weeds identification through Support Vector Machines”, Applied Soft Computing, Volume 11, Issue 1, January 2011, Pag. 908-915. [www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Teramoto, 2002] - Teramoto H., Xu G., 2002 - „Camera Calibration by a Single Image of Balls: From Conics to the Absolute Conic”, The 5th Asian Conference on Computer Vision.  
[http://www.aprs.org.au/accv2002/accv2002\\_proceedings/Teramoto499.pdf](http://www.aprs.org.au/accv2002/accv2002_proceedings/Teramoto499.pdf) (13.12.2010)

[Tian, 1998] - Tian L. F. Slaughter D. C., 1998 - „Environmentally adaptive segmentation algorithm for outdoor image segmentation”, Computers and Electronics in Agriculture, Issue 21, Pag. 153-168.  
<http://age-web.age.uiuc.edu/faculty/lft/papers/1391.pdf> (15.11.2011)

[Tillett, 2007] – Tillett N.D., Hague T., Grundy A.C., Dedousis A.P., 2007 – „Mechanical within-row weed control for transplanted crops using computer vision”, *Biosystems Engineering*, Volume 99, Issue 2, February 2008, Pag. 171-178, [www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Tilneac, 2009] - Tilneac M., Dolga V., 2009 – „Agricultural Robots: Individual Plant Recognition”, *Proc. Danube Adria Association for Automation & Manufacturing, DAAAM 2009*, Viena, Austria, ISSN 1726-9679, Noiembrie 2009, Pag. 969-970.

[Tilneac, 2010a] - Tilneac M., Dolga V., 2010 – „Individual Plant Recognition Using the RGB Color Model”, *Proc. IEEE Mediterranean Electrotechnical Conference, IEEE MELECON 2010*, Valletta, Malta, ISBN 978-1-4244-5794-6, Aprilie 2010, Pag.. 1147-1152.  
[http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=5476365](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5476365) (09.01.2012)

[Tilneac, 2010b] - Tilneac M., Grigorescu S., Paléologue V., Dolga V., 2010 – „Wrist Mounted Camera for Colour and Shape Recognition”, *Proceedings of the 5th International Conference on Optimization of the Robots and Manipulators, OPTIROB 2010*, Călimănești, Romania, ISBN 978-981-08-5840-7, Mai 2010, Pag. 200-204.

[Tilneac, 2011a] - Tilneac M., Dolga V., 2011 – „Extrinsic Camera Calibration and Camera-Matching for Plant Phenotyping Using CAD (Computer-Aided Design) Models”, *The 10-th International Conference on Mechatronics and Precision Engineering, COMEFIM 10*, București, Romania, Mai 2011, *Proceedings of the International Journal Romanian Review Precision Mechanics, Optics & Mechatronics*, edited by INCDMTM, București, România, ISSN 1584-5982, Volume 39, 2011, Pag.. 45-50.

[Tilneac, 2011b] - Tilneac M., Dolga V. – „Extrinsic Calibration of a Multi-Camera Network Used for Individual Plant Phenotyping”, *Proceedings of Intelligent Computer Communication and Processing (ICCP), 2011 IEEE International Conference*, Cluj-Napoca, Romania, ISBN 978-1-4577-1479-5, August 2011, Pag. 353 – 359.  
[http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=6047896](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=6047896) (09.01.2012)

[Toma, 2005] - Toma L., 2005 – „Contribution on Stereo Vision 3D Accurate Measurements”, *Ph.D Thesis, Universitatea POLITEHNICA Timișoara*, Romania.

[Tsai, 1987] – Tsai R., 1987 – „A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses”, *IEEE Journal of Robotics and Automation*, ISSN: 0882-4967, Volume 3, Issue 4, Pag. 323-344.  
<http://vigir.missouri.edu/~qdesouza/Research/Calibration/Calibration%20-%20Tsai.pdf> (13.12.2010)

## U

[Ulissi, 2011] - Ulissi V., Antonucci F., Costa C., Benincasa P., Menesatti P., 2011 – „Morphological variation on tomato leaves due to different nitrogen content”, *Agricultural Engineering, International: CIGR JOURNAL*, 1731(13-2): Pag. 1-10.  
<http://cigrjournal.org/index.php/Ejournal/article/viewFile/1732/1404> (19.12.2011)

**V**

[Vinstrup, 2005] - Vinstrup A., Sørensen C., Bak T., Jørgensen R., Christensen S., Griepentrog W., Ibarra J., Norremark M., Have H., 2005 - "Field Robots in Agriculture".

<http://www.scribd.com/doc/35156493/Field-Robots-in-Agriculture> (14.01.2012)

[Vioix, 2002] - Vioix J. B., Douzals J. P., Truchetet F., Assemat L., Guillemain J. P., 2002 - „Spatial and spectral method for weeds detection and localization”, EURASIP JASP 7, Pag. 679–685.

<http://dl.acm.org/citation.cfm?id=1283172> (22.12.2011)

[Vrindts, 2002] - Vrindts E., De Baerdemaeker J., Ramon H., 2002 - „Weed detection using canopy reflection”, Precision Agriculture 3: Pag. 63–80.

<http://www.springerlink.com/content/x258884t86753124/> (22.12.2011)

**W**

[Wang, 1991] - Wang L.L., Tsai W.H., 1991 - „Camera Calibration by Vanishing Lines for 3-D Computer Vision”, IEEE Transactions, Şablon Analysis and Machine Intelligence, ISSN 0162-8828, Volume 13, Issue 4, Pag. 370-376.

[http://cgkit.nutn.edu.tw:8080/cgit/PaperDL/WWD\\_090127103743.pdf](http://cgkit.nutn.edu.tw:8080/cgit/PaperDL/WWD_090127103743.pdf) (13.12.2010)

[Wang, 2006] - Wang D., Zhang X., Liu Y., 2006 - „Recognition system of leaf images based on neuronal network”, Journal of Forestry Research, 17(3): Pag. 243–246. <http://www.springerlink.com/content/evlv71n5tv603125/> (08.10.2011)

[Wang, 2009] - Wang H., Zhang W., Zhou G., Yan G., Clinton N., 2009 - „Image-based 3D corn reconstruction for retrieval of geometrical structural parameters”, International Journal of Remote Sensing, Volume 30, Issue 20, Pag. 5505-5513.

[http://www.slrss.cn/download/09\\_SCI/%E7%8E%8B%E6%B5%A9%E6%98%9F.pdf](http://www.slrss.cn/download/09_SCI/%E7%8E%8B%E6%B5%A9%E6%98%9F.pdf) (22.12.2011)

[Watchareeruetai, 2007] - Watchareeruetai U., Takeuchi Y., Matsumoto T., Kudo H., Ohnishi N. 2007 - „A lawn weed detection in winter season based on color information”, MVA2007 IAPR Conference on Machine Vision Applications, May 16-18, 2007, Tokyo, Japan

<http://b2.cvl.iis.u-tokyo.ac.jp/mva/proceedings/2007CD/papers/13-23.pdf>

(12.12.2008)

[Weiss, 2010a] - Weiss U., Biber P., 2010 - „Semantic Place Classification and Mapping for Autonomous Agricultural Robots”, [http://www.ais.uni-bonn.de/~holz/iros10wssm/proceedings/03\\_weiss\\_biber\\_final.pdf](http://www.ais.uni-bonn.de/~holz/iros10wssm/proceedings/03_weiss_biber_final.pdf) (02.09.2011)

[Weiss, 2010b] - Weiss U., Biber P., Laible S., Bohlmann K., Zell A., 2010 - „Plant Species Classification using a 3D LIDAR Sensor and Machine Learning”, <http://www.ra.cs.uni-tuebingen.de/publikationen/2010/weiss2010icmla.pdf>

(02.09.2011)

[Weiss, 2011] - Weiss U., Biber P., 2011 - „Plant detection and mapping for agricultural robots using a 3D LIDAR sensor”, Robotics and Autonomous Systems,

Volume 59, Issue 5, May 2011, Pag. 265-273, [www.sciencedirect.com](http://www.sciencedirect.com) (27.06.2011)

[Weng, 1992] – Weng J., Cohen P., Herniou M., 1992 – „Camera Calibration with Distortion Models and Accuracy Evaluation”, IEEE Transactions, Şablon Analysis and Machine Intelligence, ISSN 0162-8828, Volume 14, Issue 10, Pag. 965 – 980.  
[http://homes.dsi.unimi.it/~borgnese/Teaching/IntelligentSystems/Old/IntelligentSystems\\_2003\\_2004/Documents/Vision/92\\_WengCohenHerniou\\_Camcal.pdf](http://homes.dsi.unimi.it/~borgnese/Teaching/IntelligentSystems/Old/IntelligentSystems_2003_2004/Documents/Vision/92_WengCohenHerniou_Camcal.pdf) (13.12.2010)

[Wilczkowiak, 2001] – Wilczkowiak M., Boyer E., Sturm P., 2001 – „Camera Calibration and 3D Reconstruction from Single Images Using Parallelepipeds”, IEEE International Conference, Computer Vision, ISBN 0-7695-1143-0, Volume 1, Pag. 142-148.  
[http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=937510](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=937510) (13.12.2010)

[Wong, 2003] – Wong K. Y., Mendonca P., Cipolla R., 2003 – „Camera Calibration from Surfaces of Revolution”, IEEE Transactions, Şablon Analysis and Machine Intelligence, ISSN 0162-8828, Volume 25, Issue 2, Pag. 147 – 161.  
<http://www.computer.org/portal/web/csdl/transactions/tpami#4> (13.12.2010)

[Wu, 2007] - Wu S. G., Bao F. S., Xu E. Y., Wang Y. X., Chang Y. F., Xiang Q. L., 2007 – „A Leaf Recognition Algorithm for Plant Classification Using Probabilistic Neural Network”, [http://arxiv.org/PS\\_cache/arxiv/pdf/0707/0707.4289v1.pdf](http://arxiv.org/PS_cache/arxiv/pdf/0707/0707.4289v1.pdf) (08.10.2011)

## X

[Xie, 2005] – Xie J., 2005, - „Neuro-Fuzzy Modelling and Model-Based Fault Detection on an Autonomous Vehicle”, Master Thesis, Aalborg University, Denmark.  
<http://www.control.auc.dk/~jxie03/preamble.pdf> (23.06.2011)

[Xu, 2000] - Xu G., Terai J. I., Shum H. Y., „A Linear Algorithm for Camera Self-Calibration, Motion and Structure Recovery for Multi-Planar Scenes from Two Perspective Images”, IEEE Conference on Computer Vision and Şablon Recognition, ISBN: 0-7695-0662-3, Volume 2, Pag. 474-479.  
[http://reference.kfupm.edu.sa/content/l/i/a\\_linear\\_algorithm\\_for\\_camera\\_self\\_calib\\_66318.pdf](http://reference.kfupm.edu.sa/content/l/i/a_linear_algorithm_for_camera_self_calib_66318.pdf) (13.12.2010)

## Z

[Zhang, 1999] – Zhang Z., 1999 – „Flexible Camera Calibration by Viewing a Plane from Unknown Orientations”, Seventh International Conference on Computer Vision.  
<http://www.cse.iitk.ac.in/users/vision/dipakmj/papers/zhan99.pdf> (13.12.2010)

[Zhang, 2004] – Zhang Z., 2004 – „Camera Calibration With One-Dimensional Objects”, IEEE Transactions, Şablon Analysis and Machine Intelligence, ISSN 0162-8828, Volume 26, Issue 7, Pag. 892-899.  
<http://research.microsoft.com/en-us/um/people/zhang/papers/zhangpami-04-07-calib-1d.pdf> (13.12.2010)

[Zhang, 2007] - Zhang H., Wong K., Y., Zhang G., 2007 - „Camera Calibration from Images of Spheres“, IEEE Transactions, Şablon Analysis and Machine Intelligence, ISSN 0162-8828, Volume 29 , Issue 3, Pag. 499-502.  
<http://www.computer.org/portal/web/csdl/doi/10.1109/TPAMI.2007.45>  
 (13.12.2010)

[Zhang, 2008] - Zhang Z., 2008 - „A Flexible New Technique for Camera Calibration“, IEEE Transactions, Şablon Analysis and Machine Intelligence, ISSN 0162-8828, Volume 22, Issue 11, Pag. 1330 - 1334.  
<http://www.johnloomis.org/ece595b/notes/CameraCalibration/readings/TR98-71.pdf> (13.12.2010)

[Zhang, 2008b] - Zhang W., Wang H., Zhou G., Yan G., 2008 - „Corn 3D Reconstruction with Photogrammetry“, International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences.  
[http://www.isprs.org/proceedings/XXXVII/congress/8\\_pdf/10\\_WG-VIII-10/09.pdf](http://www.isprs.org/proceedings/XXXVII/congress/8_pdf/10_WG-VIII-10/09.pdf)  
 (19.12.2011)

[Zhao, 2008] - Zhao J., 2008 - „Stereo Imaging and Obstacle Detection Methods for Vehicle Guidance“, Ph.D Thesis, University of New South Wales, Australia.  
[http://scholar.googleusercontent.com/scholar?q=cache:5Z1JD7bGO08J:scholar.google.com/&hl=de&as\\_sdt=0&as\\_vis=1](http://scholar.googleusercontent.com/scholar?q=cache:5Z1JD7bGO08J:scholar.google.com/&hl=de&as_sdt=0&as_vis=1) (23.12.2011)

\*\*\*

[\*\*\* FRE, 2004] - \*\*\* - Proceedings of the 2th Field Robot Event 2004,  
[http://www.fieldrobot.nl/downloads/Proceedings\\_FRE2004.pdf](http://www.fieldrobot.nl/downloads/Proceedings_FRE2004.pdf) (08.01.2012)

[\*\*\* FRE, 2005] - \*\*\* - Proceedings of the 3th Field Robot Event 2005,  
[http://www.fieldrobot.nl/downloads/Proceedings\\_FRE2005.pdf](http://www.fieldrobot.nl/downloads/Proceedings_FRE2005.pdf) (08.01.2012)

[\*\*\* FRE, 2006] - \*\*\* - Proceedings of the 4th Field Robot Event 2006,  
[http://www.fieldrobot.nl/downloads/Proceedings\\_FRE2006.pdf](http://www.fieldrobot.nl/downloads/Proceedings_FRE2006.pdf) (08.01.2012)

[\*\*\* FRE, 2007] - \*\*\* - Proceedings of the 5th Field Robot Event 2007,  
[http://www.fieldrobot.nl/downloads/Proceedings\\_FRE2007.pdf](http://www.fieldrobot.nl/downloads/Proceedings_FRE2007.pdf) (08.01.2012)

[\*\*\* FRE, 2008] - \*\*\* - Proceedings of the 6th Field Robot Event 2008,  
[http://www.fieldrobot.nl/downloads/Proceedings\\_FRE2008.pdf](http://www.fieldrobot.nl/downloads/Proceedings_FRE2008.pdf) (08.01.2012)

[\*\*\* FRE, 2009] - \*\*\* - Proceedings of the 7th Field Robot Event 2009,  
[http://www.fieldrobot.nl/downloads/Proceedings\\_FRE2009.pdf](http://www.fieldrobot.nl/downloads/Proceedings_FRE2009.pdf) (08.01.2012)

[\*\*\* FRE, 2010] - \*\*\* - Proceedings of the 8th Field Robot Event 2010,  
[http://rzbl04.biblio.etc.tu-bs.de:8080/docportal/servlets/MCRFileNodeServlet/DocPortal\\_derivate\\_00021118/Proceedings\\_8th\\_FieldRobotEvent\\_2010.pdf;jsessionid=AF38FADB9C02737474D80F2263B9EAA7](http://rzbl04.biblio.etc.tu-bs.de:8080/docportal/servlets/MCRFileNodeServlet/DocPortal_derivate_00021118/Proceedings_8th_FieldRobotEvent_2010.pdf;jsessionid=AF38FADB9C02737474D80F2263B9EAA7) (08.01.2012)

## 168 Bibliografie

---

- [\*\*\* Kovesi] - Kovesi P. – "MATLAB and Octave Functions for Computer Vision and Image Processing", Online.  
<http://www.csse.uwa.edu.au/~pk/Research/MatlabFns/#match> (09.01.2012)
- [\*\*\* Img, Amaizeing] - \*\*\* -  
[http://www.hs-osnabrueck.de/uploads/media/017\\_Amaizeing.JPG](http://www.hs-osnabrueck.de/uploads/media/017_Amaizeing.JPG) (07.01.2012)
- [\*\*\* Img, BoniRob] - \*\*\* -  
[http://wissen.dradio.de/media/thumbs/9/9eeeb9891e4343efa01641e5de490a4cv1\\_max\\_440x330\\_b3535db83dc50e27c1bb1392364c95a2.jpg](http://wissen.dradio.de/media/thumbs/9/9eeeb9891e4343efa01641e5de490a4cv1_max_440x330_b3535db83dc50e27c1bb1392364c95a2.jpg) (07.01.2012)
- [\*\*\* Img, Eyesonic] - \*\*\* -  
<http://www.eyesonic.nl/Foto%27s/FieldRobotEvent2009/DSC09690%20%28Small%29.JPG> (07.01.2012)
- [\*\*\* Img, Hortibot] - \*\*\* - [http://lh5.ggpht.com/-W4-sHByMG-k/R836H-Eg5cI/AAAAAAAAAMo/huVxJuakJfg/IMG\\_1444.JPG](http://lh5.ggpht.com/-W4-sHByMG-k/R836H-Eg5cI/AAAAAAAAAMo/huVxJuakJfg/IMG_1444.JPG) (07.01.2012)
- [\*\*\* Img, Helios] - \*\*\* -  
[http://www.tu-braunschweig.de/Medien-DB/presse/099\\_08\\_helios.jpg](http://www.tu-braunschweig.de/Medien-DB/presse/099_08_helios.jpg) (07.01.2012)
- [\*\*\* Img, Lukas] - \*\*\* - <http://www.engadget.com/2005/09/17/lukas-the-robot-toils-weeding-in-the-fields/> (07.01.2012)
- [\*\*\* Img, T&H] - \*\*\* -  
<http://www.thtechnology.co.uk/Past%20Projects.html> (07.01.2012)
- [\*\*\* Img, Weedy] - \*\*\* -  
<http://www.unibots.com/Pictures/Weedy.jpg> (07.01.2012)
- [\*\*\* Img, Optimaize Prime] - \*\*\* -  
[http://www.ifm.com/ifmde/web/news\\_091001.htm](http://www.ifm.com/ifmde/web/news_091001.htm) (07.01.2012)
- [\*\*\* Img, Leaf] - \*\*\* -  
<http://en.wikipedia.org/wiki/Leaf> (17.01.2012)
- [\*\*\*Robotica] - \*\*\* - <http://www.vatau.com/cursuri.html> (07.02.2012)
- [MATLAB – Image Processing Toolbox] – „Identifying Round Objects“, Image Processing Toolbox DEMOS



## Anexe

### Anexa A; Programe MATLAB pentru procesarea culorilor din imagine

#### Programul de procesare a culorilor (ordinogramă, secvențe de cod, interfața grafică)

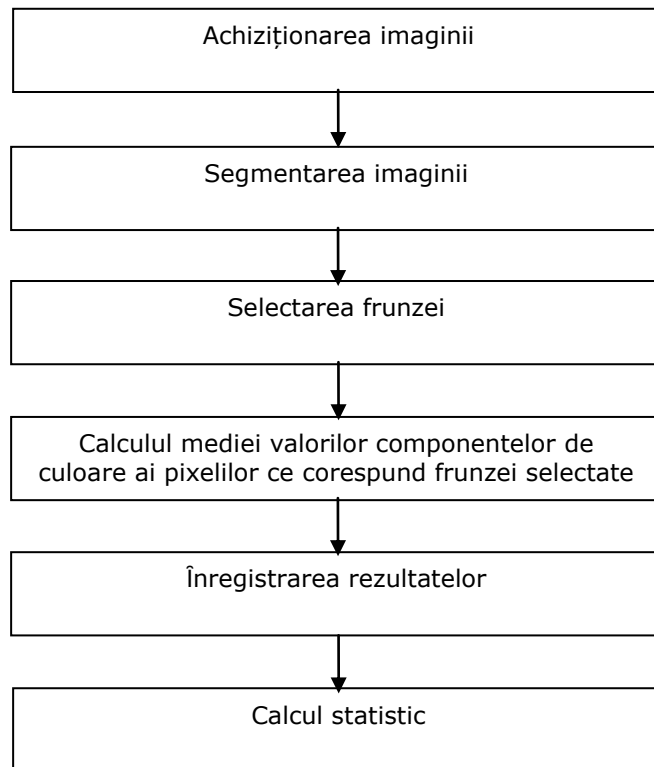


Fig. A. 1. Ordinograma programului de procesare a culorilor

```
% încărcarea unui imaginii ce urmează a fi procesată  
I = uigetfile({'*.jpg;*.tif;*.png;*.gif;*.bmp'});  
J = imread(I);  
% redenumirea imaginii  
imwrite(J, 'imagini\imagine_cameral.jpg');
```

```
image_HSV = rgb2hsv(image_RGB);  
% filtru de culoare verde  
[w,h] = size(image_HSV(:,:,1));
```

## 170 Anexe

```

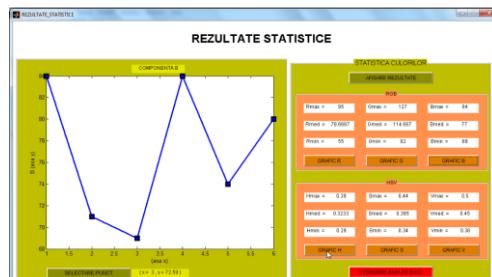
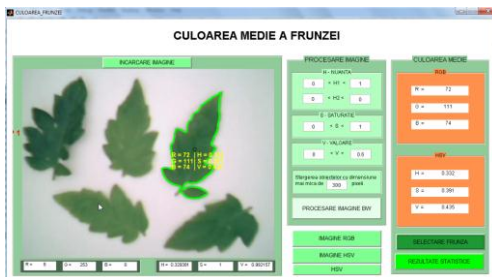
imagine_BW = zeros(w,h);
for i=1:w
    for j=1:h
        if((imagine_HSV(i,j,1) >= H1_min && imagine_HSV(i,j,1) <=
H1_max) || (imagine_HSV(i,j,1) > H2_min && imagine_HSV(i,j,1) <
H2_max)) &&...
            (imagine_HSV(i,j,2) >= S_min && imagine_HSV(i,j,2) <=
S_max) &&...
            (imagine_HSV(i,j,3) >= V_min && imagine_HSV(i,j,3) <=
V_max))
            imagine_BW(i,j)=1;
        else
            imagine_BW(i,j)=0;
        end
    end
end
end
% salvarea imaginii binare filtrate
imwrite(imagine_BW, 'imagini\bw_cameral.jpg');

```

```

imshow(RGB)
% selectarea unei frunze
[x,y] = ginput(1);
BW2 = bwselect(BW1,x,y,4);
obiect = bwlabel(BW2,4);
[randul,coloana] = find(obiect==1);
rc = [randul,coloana];
s=size(rc);
n=s(1);
for i = 1:1:n
    x(i)=randul(i);
    y(i)=coloana(i);
    % extragerea valorilor componentelor de culoare RGB si HSV
    r(i) = RGB(x(i),y(i),1);
    g(i) = RGB(x(i),y(i),2);
    b(i) = RGB(x(i),y(i),3);
    h(i) = HSV(x(i),y(i),1);
    s(i) = HSV(x(i),y(i),2);
    v(i) = HSV(x(i),y(i),3);
end
% calculul mediei pentru fiecare din componentele RGB
R = mean(r);
G = mean(g);
B = mean(b);

```



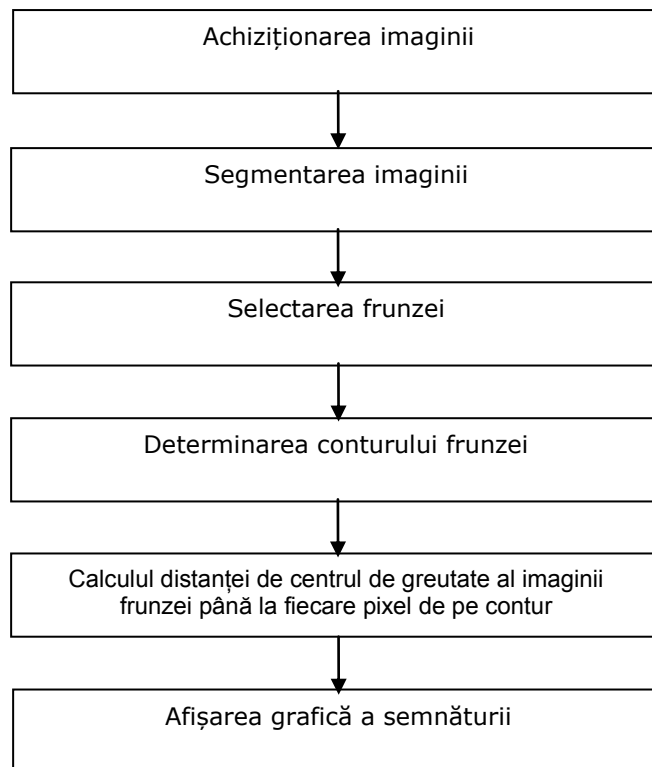
**Anexa B; Programe MATLAB pentru procesarea formelor din imagine****Anexa B1**  
**Programul de calcul a semnăturilor frunzelor**  
**(ordinogramă, secvențe de cod)**

Fig.B. 1. Ordinograma programului de calcul a semnăturilor frunzelor

```
% 'boundary' - returneaza coordonatele pixelilor conturului obiectului  
corespunzator frunzei.  
% 'centroid' - returneaza coordonatele centrului de greutate al  
obiectului corespunzator frunzei.  
% 'distanta' - returneaza valorile distantelor de la centrul de  
greutate al obiectului pana la fiecare pixel al conturului.  
n = length(boundary);  
for i = 1:1:n  
    x_boundary(i) = boundary(i,2);  
    y_boundary(i) = boundary(i,1);  
    x_CG = round(centroid(1,1));  
    y_CG = round(centroid(1,2));
```

```
distanța(i) = sqrt((x_boundary(i) - x_CG)^2 + (y_boundary(i) - y_CG)^2);  
end
```

**Anexa B2**  
**Programul de învățare a formelor**  
**(ordinogramă, secvențe de cod, interfața grafică)**

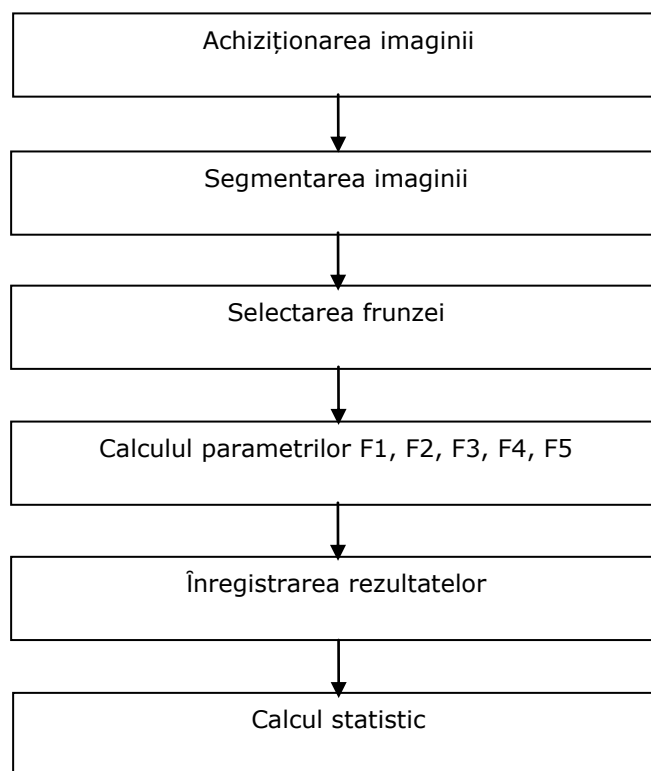
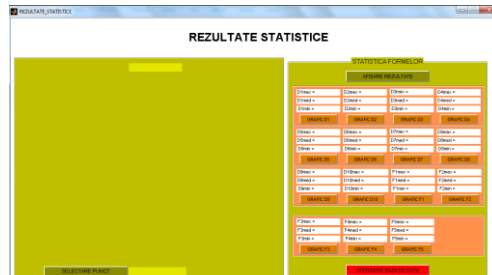
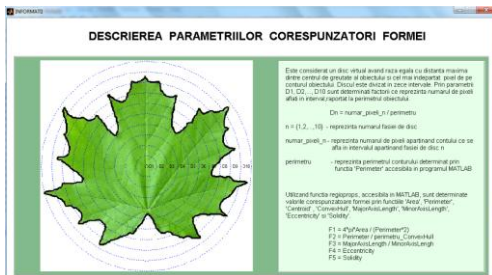
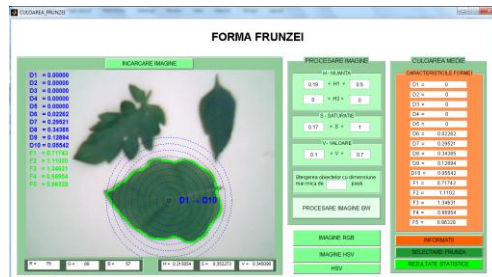
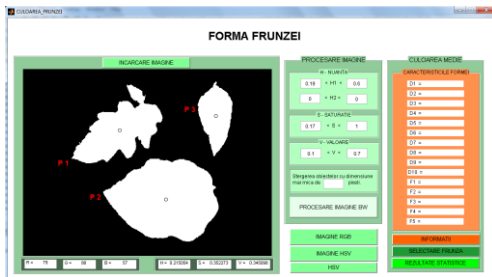
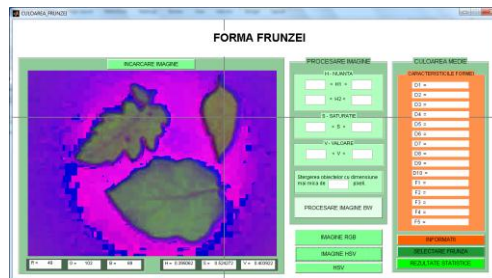
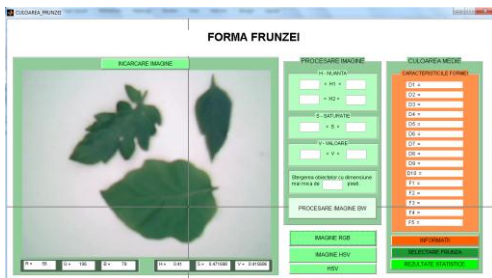
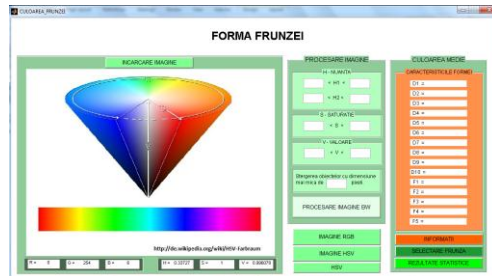
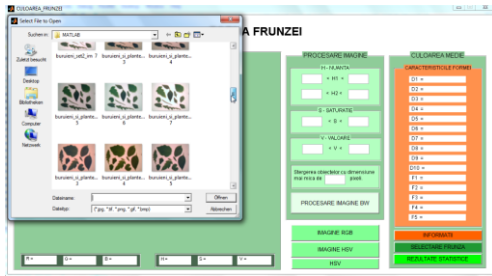


Fig.B. 2. Ordinograma programului de învățare a formelor

```
% determinarea caracteristicilor geometrice  
area = stats.Area;  
perimeter = stats.Perimeter;  
majorAxisLength = stats.MajorAxisLength;  
minorAxisLength = stats.MinorAxisLength;  
eccentricity = stats.Eccentricity;  
convexHull = stats.ConvexHull;  
convexArea = stats.ConvexArea;  
solidity = stats.Solidity;
```

```
% determinarea parametrilor F1, F2, F3, F4, F5
F1 = 4*pi*area/(perimeter)^2;
F3 = majorAxisLength/minorAxisLength;
F4 = eccentricity;
F5 = solidity;
```



**Anexa B3**  
**Programul de recunoaștere a frunzelor**  
**(ordinogramă, secvențe de cod, interfața grafică)**

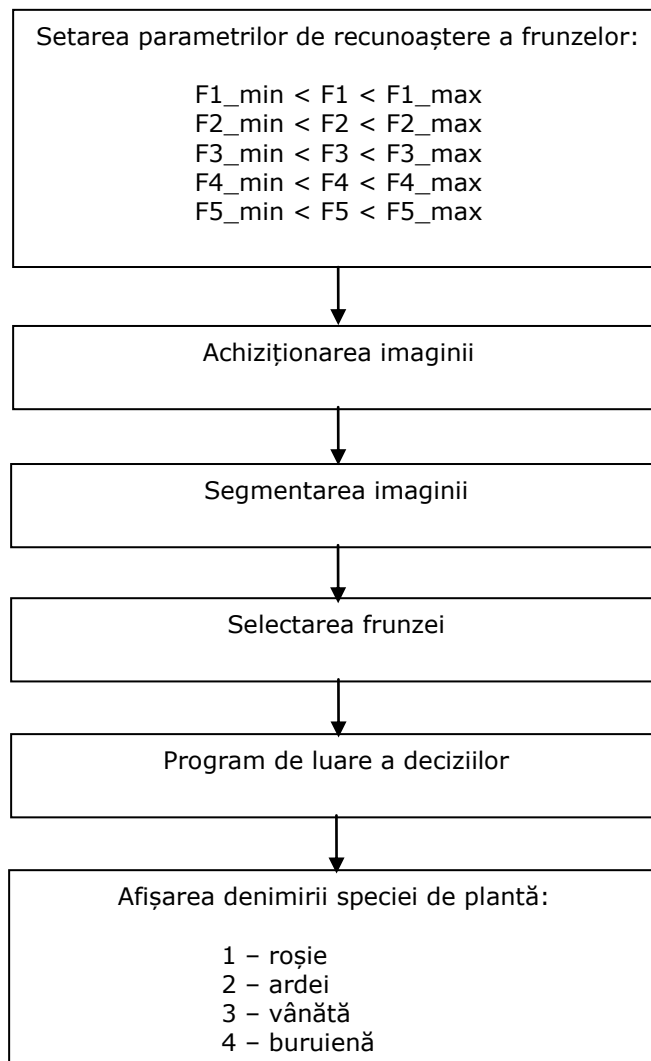


Fig.B. 3. Ordinograma programului de recunoaștere a frunzelor

```

% detectarea frunzei de rosie
    if ((F1 >= (F1_min_rosie-0.0001) && F1 <=
(F1_max_rosie+0.0001))&&...
        (F3 >= (F3_min_rosie-0.0001) && F3 <=
(F3_max_rosie+0.0001))&&...
        (F4 >= (F4_min_rosie-0.0001) && F4 <=
(F4_max_rosie+0.0001))&&...
        (F5 >= (F5_min_rosie-0.0001) && F5 <= (F5_max_rosie+0.0001)))

        hold on
        boundary = B{k};
        plot(boundary(:,2), boundary(:,1), 'y', 'LineWidth', 3)
        centroid = stats(k).Centroid;
        denumire_plante = 'rosie';
        text(centroid(1)-
25,centroid(2)+13,denumire_plante,'Color','y',...
'FontSize',14,'FontWeight','bold');
    end

```

```

% detectarea frunzei de ardei
    if ((F1 >= (F1_min_ardei-0.0001) && F1 <=
(F1_max_ardei+0.0001))&&...
        (F3 >= (F3_min_ardei-0.0001) && F3 <=
(F3_max_ardei+0.0001))&&...
        (F4 >= (F4_min_ardei-0.0001) && F4 <=
(F4_max_ardei+0.0001))&&...
        (F5 >= (F5_min_ardei-0.0001) && F5 <= (F5_max_ardei+0.0001)))
        centroid = stats(k).Centroid;
        hold on
        boundary = B{k};
        plot(boundary(:,2), boundary(:,1), 'y', 'LineWidth', 3)
        denumire_plante = 'ardei';
        text(centroid(1)-
25,centroid(2)+13,denumire_plante,'Color','y',...
'FontSize',14,'FontWeight','bold');
    end

```

```

% detectarea frunzei de vanata
    if ((F1 >= (F1_min_vanata-0.0001) && F1 <=
(F1_max_vanata+0.0001))&&...
        (F3 >= (F3_min_vanata-0.0001) && F3 <=
(F3_max_vanata+0.0001))&&...
        (F4 >= (F4_min_vanata-0.0001) && F4 <=
(F4_max_vanata+0.0001))&&...
        (F5 >= (F5_min_vanata-0.0001) && F5 <= (F5_max_vanata+0.0001)))

        centroid = stats(k).Centroid;

        hold on
        boundary = B{k};
        plot(boundary(:,2), boundary(:,1), 'y', 'LineWidth', 3)
        centroid = stats(k).Centroid;
        denumire_plante = 'vanata';
        text(centroid(1)-
25,centroid(2)+13,denumire_plante,'Color','y',...

```

```
        'FontSize',14,'FontWeight','bold');
    end

% detectarea frunzelor de buruiana
    if ((F1 >= (F1_min_rosie-0.0001) && F1 <=
(F1_max_rosie+0.0001))&&...
        (F3 >= (F3_min_rosie-0.0001) && F3 <=
(F3_max_rosie+0.0001))&&...
        (F4 >= (F4_min_rosie-0.0001) && F4 <=
(F4_max_rosie+0.0001))&&...
        (F5 >= (F5_min_rosie-0.0001) && F5 <= (F5_max_rosie+0.0001)))

        elseif ((F1 >= (F1_min_ardei-0.0001) && F1 <=
(F1_max_ardei+0.0001))&&...
        (F3 >= (F3_min_ardei-0.0001) && F3 <=
(F3_max_ardei+0.0001))&&...
        (F4 >= (F4_min_ardei-0.0001) && F4 <=
(F4_max_ardei+0.0001))&&...
        (F5 >= (F5_min_ardei-0.0001) && F5 <= (F5_max_ardei+0.0001)))

        elseif ((F1 >= (F1_min_vanata-0.0001) && F1 <=
(F1_max_vanata+0.0001))&&...
        (F3 >= (F3_min_vanata-0.0001) && F3 <=
(F3_max_vanata+0.0001))&&...
        (F4 >= (F4_min_vanata-0.0001) && F4 <=
(F4_max_vanata+0.0001))&&...
        (F5 >= (F5_min_vanata-0.0001) && F5 <= (F5_max_vanata+0.0001)))

        else
            hold on
            boundary = B{k};
            plot(boundary(:,2), boundary(:,1), 'r', 'LineWidth', 3)
            centroid = stats(k).Centroid;
            denumire_plante = 'buruiana';
            text(centroid(1)-
25,centroid(2)+13,denumire_plante,'Color','r',...
        'FontSize',10,'FontWeight','bold');
    end
```



STEP1

### STEP1 - SETARI


RECUNOASTERA FRUNZEI DE ROSIE	RECUNOASTERA FRUNZEI DE ARDEI	RECUNOASTERA FRUNZEI DE VANATA
0.3203 < F1 < 0.5201	0.5310 < F1 < 0.6571	0.6905 < F1 < 0.7412
1.1690 < F2 < 1.3519	1.0623 < F2 < 1.0913	1.0728 < F2 < 1.1115
1.8243 < F3 < 2.4142	1.7793 < F3 < 2.0351	1.3201 < F3 < 1.5070
0.8364 < F4 < 0.9102	0.8271 < F4 < 0.8710	0.6528 < F4 < 0.7481
0.8285 < F5 < 0.9043	0.9177 < F5 < 0.9627	0.9442 < F5 < 0.9654

**NEXT**

STEP3

### STEP3 - DETECTAREA SPECIEI DE PLANTA

IMAGINEA



- FRUNZA DE ROSIE**
- FRUNZA DE ARDEI**
- FRUNZA DE VANATA**
- FRUNZA DE BURUJANA**

## Anexa C; Programe MATLAB pentru calibrarea camerelor video

### Anexa C1

#### Programul de determinare a distanței focale (ordinogramă, secvențe de cod, interfață grafică)

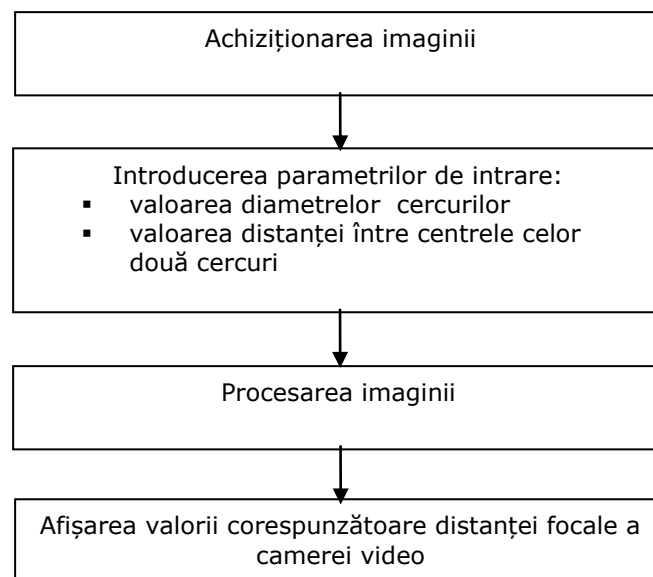


Fig. C. 1. Ordinograma programului de determinare a distanței focale

```

function [] = algoritmi()

d_cerc1_world = importdata('parametrii\d_cerc1_world.txt');
d_cerc2_world = importdata('parametrii\d_cerc2_world.txt');
d_P1P2_world = importdata('parametrii\d_P1P2_world.txt');
d_cerc1_im = importdata('parametrii\d_cerc1_im.txt');
d_cerc2_im = importdata('parametrii\d_cerc2_im.txt');
x_P1_im_cam = importdata('parametrii\x_P1_im_cam.txt');
x_P2_im_cam = importdata('parametrii\x_P2_im_cam.txt');
y_P1_im_cam = importdata('parametrii\y_P1_im_cam.txt');
y_P2_im_cam = importdata('parametrii\y_P2_im_cam.txt');

%% modelul matematic
A = (d_cerc1_world/d_cerc1_im*x_P1_im_cam-
d_cerc2_world/d_cerc2_im*x_P2_im_cam)^2;
B = (d_cerc1_world/d_cerc1_im*y_P1_im_cam-
d_cerc2_world/d_cerc2_im*y_P2_im_cam)^2;
C = d_P1P2_world^2;
N1 = A+B-C;
  
```

```

N2 = -((d_cerc1_world/d_cerc1_im - d_cerc2_world/d_cerc2_im)^2);

%% distanta focala exprimata in pixeli
f = sqrt(abs(N1/N2));

a_str = sprintf('%2.2f',f);
fid = fopen('parametrii\f.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);

```

### Programul interfeței grafice:

```

function varargout = START(varargin)
gui Singleton = 1;
gui_State = struct('gui_Name',      mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @START_OpeningFcn, ...
                  'gui_OutputFcn', @START_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function START_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;
guidata(hObject, handles);

function varargout = START_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;

function pushbutton3_Callback(hObject, eventdata, handles)
imagine_RGB = imread('imagine\imagine.jpg');
imshow(imagine_RGB)
imagine_HSV = rgb2hsv(imagine_RGB);
[y,x] = ginput(1);
x = abs(round(x));
y = abs(round(y));
H = imagine_HSV(x,y,1);
S = imagine_HSV(x,y,2);
V = imagine_HSV(x,y,3);
set(handles.text2, 'String',H)
set(handles.text4, 'String',S)
set(handles.text6, 'String',V)

function pushbutton4_Callback(hObject, eventdata, handles)
imagine_RGB = imread('imagine\imagine.jpg');
imagine_HSV = rgb2hsv(imagine_RGB);
imshow(imagine_HSV)
[y,x] = ginput(1);
x = abs(round(x));
y = abs(round(y));
H = imagine_HSV(x,y,1);
S = imagine_HSV(x,y,2);
V = imagine_HSV(x,y,3);
set(handles.text2, 'String',H)

```

```

set(handles.text4,'String',S)
set(handles.text6,'String',V)

function pushbutton5_Callback(hObject, eventdata, handles)
    imagine_RGB = imread('HSV.jpg');
    imshow(imagine_RGB)
    imagine_HSV = rgb2hsv(imagine_RGB);
    [y,x] = ginput(1);
    x = abs(round(x));
    y = abs(round(y));
    H = imagine_HSV(x,y,1);
    S = imagine_HSV(x,y,2);
    V = imagine_HSV(x,y,3);
    set(handles.text2,'String',H)
    set(handles.text4,'String',S)
    set(handles.text6,'String',V)

function pushbutton2_Callback(hObject, eventdata, handles)
    procesare_imagine()
    algoritmi()
    f = importdata('parametrii\f.txt');
    set(handles.text20,'String',f)
    imshow('imagine\bw_imagine.jpg')
    delete('imagine\*.jpg')

function pushbutton1_Callback(hObject, eventdata, handles)
    bw = imread('imagine\bw_imagine.jpg');
    imshow(bw)
    bw = im2bw(bw);
    [y,x] = ginput(4);
    c = [y(1) y(2) y(3) y(4)];
    r = [x(1) x(2) x(3) x(4)];
    BW1 = roipoly(bw,c,r);
    H = 0;
    bw = roifilt2(H,bw,BW1);
    dimensiune_obiecte_sterse =
    importdata('parametrii\stergere_obiecte_mici_im.txt');
    bw = bwareaopen(bw, dimensiune_obiecte_sterse);
    se = strel('disk',2);
    bw = imclose(bw,se);
    bw = imfill(bw,'holes');
    imshow(bw)
    imwrite(bw,'imagine\bw_imagine.jpg');
    [B,L] = bwboundaries(bw,'noholes');
    hold on
    stats = regionprops(L,'Area','Centroid');
    for k = 1:length(B)
        boundary = B{k};
        P = sprintf('P%2.0f',k);
        centroid = stats(k).Centroid;
        plot(centroid(1),centroid(2),'ko');
        text(boundary(1,2)-
        35, boundary(1,1)+13,P,'Color','r','FontSize',14,'FontWeight','bold');
    end

function edit1_Callback(hObject, eventdata, handles)
    a = str2num(get(hObject,'String'));
    a_str = sprintf('%2.0f',a);
    fid = fopen('parametrii\stergere_obiecte_mici_im.txt','w');
    fprintf(fid, '%s', a_str);
    fclose(fid);
    filtru_culoare();

```

```

imshow('image\bw_image.jpg')

function edit1 CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
set(hObject,'BackgroundColor','white');
end

function edit2 Callback(hObject, eventdata, handles)
a = str2num(get(hObject,'String'));
a_str = sprintf('%2.2f',a);
fid = fopen('parametrii\V_min_im.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
filtru_culoare();
imshow('image\bw_image.jpg')

function edit2_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
set(hObject,'BackgroundColor','white');
end

function edit3 Callback(hObject, eventdata, handles)
a = str2num(get(hObject,'String'));
a_str = sprintf('%2.2f',a);
fid = fopen('parametrii\V_max_im.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
filtru_culoare();
imshow('image\bw_image.jpg')

function edit3_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
set(hObject,'BackgroundColor','white');
end

function edit4_Callback(hObject, eventdata, handles)
a = str2num(get(hObject,'String'));
a_str = sprintf('%2.2f',a);
fid = fopen('parametrii\S_min_im.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
filtru_culoare();
imshow('image\bw_image.jpg')

function edit4_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
set(hObject,'BackgroundColor','white');
end

function edit5 Callback(hObject, eventdata, handles)
a = str2num(get(hObject,'String'));
a_str = sprintf('%2.2f',a);
fid = fopen('parametrii\S_max_im.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
filtru_culoare();
imshow('image\bw_image.jpg')

```

## 182 Anexe

```
function edit5_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit6_Callback(hObject, eventdata, handles)
a = str2num(get(hObject,'String'));
a_str = sprintf('%2.2f',a);
fid = fopen('parametrii\H1_min_im.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
filtru_culoare();
imshow('image\bw_image.jpg')

function edit6_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit7_Callback(hObject, eventdata, handles)
a = str2num(get(hObject,'String'));
a_str = sprintf('%2.2f',a);
fid = fopen('parametrii\H1_max_im.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
filtru_culoare();
imshow('image\bw_image.jpg')

function edit7_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit8_Callback(hObject, eventdata, handles)
a = str2num(get(hObject,'String'));
a_str = sprintf('%2.2f',a);
fid = fopen('parametrii\H2_min_im.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
filtru_culoare();
imshow('image\bw_image.jpg')

function edit8_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit9_Callback(hObject, eventdata, handles)
a = str2num(get(hObject,'String'));
a_str = sprintf('%2.2f',a);
fid = fopen('parametrii\H2_max_im.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
filtru_culoare();
imshow('image\bw_image.jpg')

function edit9_CreateFcn(hObject, eventdata, handles)
```

```

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit10_Callback(hObject, eventdata, handles)
a = str2num(get(hObject,'String'));
a_str = sprintf('%2.2f',a);
fid = fopen('parametrii\d_cerc2_world.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);

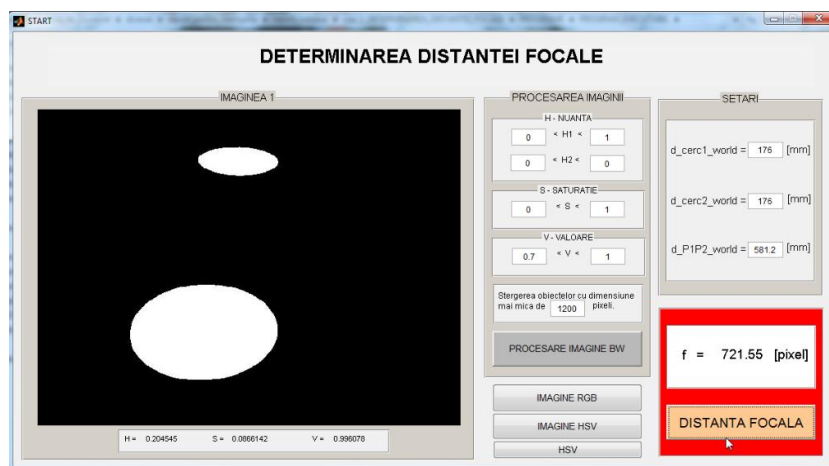
function edit10_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit11_Callback(hObject, eventdata, handles)
a = str2num(get(hObject,'String'));
a_str = sprintf('%2.2f',a);
fid = fopen('parametrii\d_P1P2_world.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);

function edit11_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit12_Callback(hObject, eventdata, handles)
a = str2num(get(hObject,'String'));
a_str = sprintf('%2.2f',a);
fid = fopen('parametrii\d_cerc1_world.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
function edit12_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
end

```



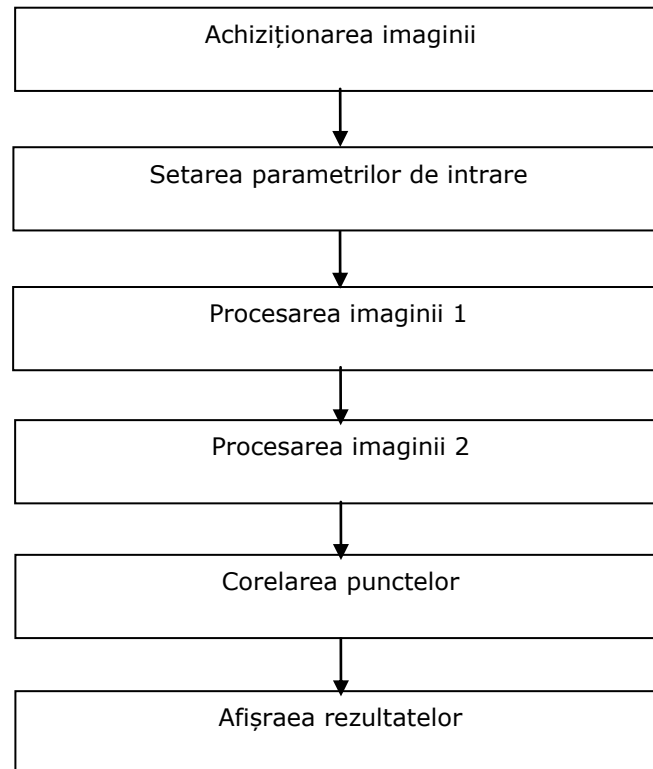
**Anexa C2****Programul de calibrare extrinsecă prin metode analitice  
(ordinogramă, secvențe de cod, interfață grafică)**

Fig. C. 2. Ordinograma programului de calibrare extrinsecă a camerelor video

```
function [] = matrice_transformare_ref1_to_ref2()  
  
%% INPUT CAMERA 1  
f1_x = importdata('parametrii\f1_x.txt');  
f1_y = importdata('parametrii\f1_y.txt');  
x_M_im1 = importdata('parametrii\x_M_im1.txt');  
y_M_im1 = importdata('parametrii\y_M_im1.txt');  
x_M_im1 = abs(round(x_M_im1));  
y_M_im1 = abs(round(y_M_im1));  
x_P1_im1 = importdata('parametrii\x_P1_im1.txt');  
y_P1_im1 = importdata('parametrii\y_P1_im1.txt');  
x_P2_im1 = importdata('parametrii\x_P2_im1.txt');  
y_P2_im1 = importdata('parametrii\y_P2_im1.txt');  
x_P3_im1 = importdata('parametrii\x_P3_im1.txt');  
y_P3_im1 = importdata('parametrii\y_P3_im1.txt');  
x_P4_im1 = importdata('parametrii\x_P4_im1.txt');
```



```

y_P4_im1 = importdata('parametrii\y_P4_im1.txt');
d_F1P1_world = importdata('parametrii\d_F1P1_world.txt');
d_F1P2_world = importdata('parametrii\d_F1P2_world.txt');
d_F1P3_world = importdata('parametrii\d_F1P3_world.txt');
d_F1P4_world = importdata('parametrii\d_F1P4_world.txt');

%% INPUT CAMERA 2
f2_x = importdata('parametrii\f2_x.txt');
f2_y = importdata('parametrii\f2_y.txt');
x_M_im2 = importdata('parametrii\x_M_im2.txt');
y_M_im2 = importdata('parametrii\y_M_im2.txt');
x_M_im2 = abs(round(x_M_im2));
y_M_im2 = abs(round(y_M_im2));

% CORELAREA PUNCTELOR
P1_im2 = importdata('parametrii\P1_im2.txt');
P2_im2 = importdata('parametrii\P2_im2.txt');
P3_im2 = importdata('parametrii\P3_im2.txt');
P4_im2 = importdata('parametrii\P4_im2.txt');
% P1
if P1_im2 ==1
    x_P1_im2 = importdata('parametrii\x_P1_im2.txt');
    y_P1_im2 = importdata('parametrii\y_P1_im2.txt');
    d_F2P1_world = importdata('parametrii\d_F2P1_world.txt');
end
if P1_im2 ==2
    x_P1_im2 = importdata('parametrii\x_P2_im2.txt');
    y_P1_im2 = importdata('parametrii\y_P2_im2.txt');
    d_F2P1_world = importdata('parametrii\d_F2P2_world.txt');
end
if P1_im2 ==3
    x_P1_im2 = importdata('parametrii\x_P3_im2.txt');
    y_P1_im2 = importdata('parametrii\y_P3_im2.txt');
    d_F2P1_world = importdata('parametrii\d_F2P3_world.txt');
end
if P1_im2 ==4
    x_P1_im2 = importdata('parametrii\x_P4_im2.txt');
    y_P1_im2 = importdata('parametrii\y_P4_im2.txt');
    d_F2P1_world = importdata('parametrii\d_F2P4_world.txt');
end
% P2
if P2_im2 ==1
    x_P2_im2 = importdata('parametrii\x_P1_im2.txt');
    y_P2_im2 = importdata('parametrii\y_P1_im2.txt');
    d_F2P2_world = importdata('parametrii\d_F2P1_world.txt');
end
if P2_im2 ==2
    x_P2_im2 = importdata('parametrii\x_P2_im2.txt');
    y_P2_im2 = importdata('parametrii\y_P2_im2.txt');
    d_F2P2_world = importdata('parametrii\d_F2P2_world.txt');
end
if P2_im2 ==3
    x_P2_im2 = importdata('parametrii\x_P3_im2.txt');
    y_P2_im2 = importdata('parametrii\y_P3_im2.txt');
    d_F2P2_world = importdata('parametrii\d_F2P3_world.txt');
end

```

```

end
if P2_im2 ==4
    x_P2_im2 = importdata('parametrii\x_P4_im2.txt');
    y_P2_im2 = importdata('parametrii\y_P4_im2.txt');
    d_F2P2_world = importdata('parametrii\d_F2P4_world.txt');
end
% P3
if P3_im2 ==1
    x_P3_im2 = importdata('parametrii\x_P1_im2.txt');
    y_P3_im2 = importdata('parametrii\y_P1_im2.txt');
    d_F2P3_world = importdata('parametrii\d_F2P1_world.txt');
end
if P3_im2 ==2
    x_P3_im2 = importdata('parametrii\x_P2_im2.txt');
    y_P3_im2 = importdata('parametrii\y_P2_im2.txt');
    d_F2P3_world = importdata('parametrii\d_F2P2_world.txt');
end
if P3_im2 ==3
    x_P3_im2 = importdata('parametrii\x_P3_im2.txt');
    y_P3_im2 = importdata('parametrii\y_P3_im2.txt');
    d_F2P3_world = importdata('parametrii\d_F2P3_world.txt');
end
if P3_im2 ==4
    x_P3_im2 = importdata('parametrii\x_P4_im2.txt');
    y_P3_im2 = importdata('parametrii\y_P4_im2.txt');
    d_F2P3_world = importdata('parametrii\d_F2P4_world.txt');
end
% P4
if P4_im2 ==1
    x_P4_im2 = importdata('parametrii\x_P1_im2.txt');
    y_P4_im2 = importdata('parametrii\y_P1_im2.txt');
    d_F2P4_world = importdata('parametrii\d_F2P1_world.txt');
end
if P4_im2 ==2
    x_P4_im2 = importdata('parametrii\x_P2_im2.txt');
    y_P4_im2 = importdata('parametrii\y_P2_im2.txt');
    d_F2P4_world = importdata('parametrii\d_F2P2_world.txt');
end
if P4_im2 ==3
    x_P4_im2 = importdata('parametrii\x_P3_im2.txt');
    y_P4_im2 = importdata('parametrii\y_P3_im2.txt');
    d_F2P4_world = importdata('parametrii\d_F2P3_world.txt');
end
if P4_im2 ==4
    x_P4_im2 = importdata('parametrii\x_P4_im2.txt');
    y_P4_im2 = importdata('parametrii\y_P4_im2.txt');
    d_F2P4_world = importdata('parametrii\d_F2P4_world.txt');
end

%% ETAPA 1 - CAMERA 1
% Coordonatele proiecțiilor în imaginea 1 ale centrelor cercurilor sunt
raportate la sistemul de referință al imaginii 1.

P1_im1 = [x_P1_im1; y_P1_im1; 0];
P2_im1 = [x_P2_im1; y_P2_im1; 0];

```

```

P3_im1 = [x_P3_im1; y_P3_im1; 0];
P4_im1 = [x_P4_im1; y_P4_im1; 0];

%% ETAPA 1 - CAMERA 2
% Coordonatele proiecțiilor în imaginea 2 ale centrelor cercurilor sunt
raportate la sistemul de referință al imaginii 2.

P1_im2 = [x_P1_im2; y_P1_im2; 0];
P2_im2 = [x_P2_im2; y_P2_im2; 0];
P3_im2 = [x_P3_im2; y_P3_im2; 0];
P4_im2 = [x_P4_im2; y_P4_im2; 0];

%% ETAPA 2 - CAMERA 1
% Calculul distanțelor de la punctul focal F_x respectiv F_y până la
fiecare proiecție a centrelor cercurilor în imaginea 1.
d_F1P1_im1_x = sqrt((x_P1_im1-x_M_im1)^2+((y_P1_im1-
y_M_im1)*f1_x/f1_y)^2+f1_x^2);
d_F1P1_im1_y = sqrt(((x_P1_im1-x_M_im1)*f1_y/f1_x)^2+(y_P1_im1-
y_M_im1)^2+f1_y^2);
d_F1P2_im1_x = sqrt((x_P2_im1-x_M_im1)^2+((y_P2_im1-
y_M_im1)*f1_x/f1_y)^2+f1_x^2);
d_F1P2_im1_y = sqrt(((x_P2_im1-x_M_im1)*f1_y/f1_x)^2+(y_P2_im1-
y_M_im1)^2+f1_y^2);
d_F1P3_im1_x = sqrt((x_P3_im1-x_M_im1)^2+((y_P3_im1-
y_M_im1)*f1_x/f1_y)^2+f1_x^2);
d_F1P3_im1_y = sqrt(((x_P3_im1-x_M_im1)*f1_y/f1_x)^2+(y_P3_im1-
y_M_im1)^2+f1_y^2);
d_F1P4_im1_x = sqrt((x_P4_im1-x_M_im1)^2+((y_P4_im1-
y_M_im1)*f1_x/f1_y)^2+f1_x^2);
d_F1P4_im1_y = sqrt(((x_P4_im1-x_M_im1)*f1_y/f1_x)^2+(y_P4_im1-
y_M_im1)^2+f1_y^2);

%% ETAPA 2 - CAMERA 2
% Calculul distanțelor de la punctul focal F_x respectiv F_y până la
fiecare proiecție a centrelor cercurilor în imaginea 2.
d_F2P1_im2_x = sqrt((x_P1_im2-x_M_im2)^2+((y_P1_im2-
y_M_im2)*f2_x/f2_y)^2+f2_x^2);
d_F2P1_im2_y = sqrt(((x_P1_im2-x_M_im2)*f2_y/f2_x)^2+(y_P1_im2-
y_M_im2)^2+f2_y^2);
d_F2P2_im2_x = sqrt((x_P2_im2-x_M_im2)^2+((y_P2_im2-
y_M_im2)*f2_x/f2_y)^2+f2_x^2);
d_F2P2_im2_y = sqrt(((x_P2_im2-x_M_im2)*f2_y/f2_x)^2+(y_P2_im2-
y_M_im2)^2+f2_y^2);
d_F2P3_im2_x = sqrt((x_P3_im2-x_M_im2)^2+((y_P3_im2-
y_M_im2)*f2_x/f2_y)^2+f2_x^2);
d_F2P3_im2_y = sqrt(((x_P3_im2-x_M_im2)*f2_y/f2_x)^2+(y_P3_im2-
y_M_im2)^2+f2_y^2);
d_F2P4_im2_x = sqrt((x_P4_im2-x_M_im2)^2+((y_P4_im2-
y_M_im2)*f2_x/f2_y)^2+f2_x^2);
d_F2P4_im2_y = sqrt(((x_P4_im2-x_M_im2)*f2_y/f2_x)^2+(y_P4_im2-
y_M_im2)^2+f2_y^2);

%% ETAPA 3 - CAMERA 1

```

```

% Determinarea factorilor de scalare corespunzători fiecărui punct.
Factorii de scalare sunt utilizați la conversia din pixel în mm, iar
valorile acestora sunt distincte pe axele x și y.
s1_cam1_x = d_F1P1_world/d_F1P1_im1_x;
s1_cam1_y = d_F1P1_world/d_F1P1_im1_y;
s2_cam1_x = d_F1P2_world/d_F1P2_im1_x;
s2_cam1_y = d_F1P2_world/d_F1P2_im1_y;
s3_cam1_x = d_F1P3_world/d_F1P3_im1_x;
s3_cam1_y = d_F1P3_world/d_F1P3_im1_y;
s4_cam1_x = d_F1P4_world/d_F1P4_im1_x;
s4_cam1_y = d_F1P4_world/d_F1P4_im1_y;

%% ETAPA 3 - CAMERA 2
% Determinarea factorilor de scalare corespunzători fiecărui punct.
Factorii de scalare sunt utilizați la conversia din pixel în mm, iar
valorile acestora sunt distincte pe axele x și y.
s1_cam2_x = d_F2P1_world/d_F2P1_im2_x;
s1_cam2_y = d_F2P1_world/d_F2P1_im2_y;
s2_cam2_x = d_F2P2_world/d_F2P2_im2_x;
s2_cam2_y = d_F2P2_world/d_F2P2_im2_y;
s3_cam2_x = d_F2P3_world/d_F2P3_im2_x;
s3_cam2_y = d_F2P3_world/d_F2P3_im2_y;
s4_cam2_x = d_F2P4_world/d_F2P4_im2_x;
s4_cam2_y = d_F2P4_world/d_F2P4_im2_y;

%% ETAPA 4 - CAMERA 1
% Determinarea coordonatelor punctelor P1_world, P2_world, P3_world și
P4_world, raportate la sistemul de referință cam1.
x_P1_world_cam1 = s1_cam1_x*(x_P1_im1-x_M_im1);
y_P1_world_cam1 = s1_cam1_y*(y_P1_im1-y_M_im1);
z_P1_world_cam1 = sqrt((d_F1P1_world^2)-(x_P1_world_cam1^2)-
(y_P1_world_cam1^2));
x_P2_world_cam1 = s2_cam1_x*(x_P2_im1-x_M_im1);
y_P2_world_cam1 = s2_cam1_y*(y_P2_im1-y_M_im1);
z_P2_world_cam1 = sqrt((d_F1P2_world^2)-(x_P2_world_cam1^2)-
(y_P2_world_cam1^2));
x_P3_world_cam1 = s3_cam1_x*(x_P3_im1-x_M_im1);
y_P3_world_cam1 = s3_cam1_y*(y_P3_im1-y_M_im1);
z_P3_world_cam1 = sqrt((d_F1P3_world^2)-(x_P3_world_cam1^2)-
(y_P3_world_cam1^2));
x_P4_world_cam1 = s4_cam1_x*(x_P4_im1-x_M_im1);
y_P4_world_cam1 = s4_cam1_y*(y_P4_im1-y_M_im1);
z_P4_world_cam1 = sqrt((d_F1P4_world^2)-(x_P4_world_cam1^2)-
(y_P4_world_cam1^2));

%% ETAPA 4 - CAMERA 2
% Determinarea coordonatelor punctelor P1_world, P2_world, P3_world și
P4_world, raportate la sistemul de referință cam2.
x_P1_world_cam2 = s1_cam2_x*(x_P1_im2-x_M_im2);
y_P1_world_cam2 = s1_cam2_y*(y_P1_im2-y_M_im2);
z_P1_world_cam2 = sqrt((d_F2P1_world^2)-(x_P1_world_cam2^2)-
(y_P1_world_cam2^2));
x_P2_world_cam2 = s2_cam2_x*(x_P2_im2-x_M_im2);
y_P2_world_cam2 = s2_cam2_y*(y_P2_im2-y_M_im2);

```

```

z_P2_world_cam2 = sqrt((d_F2P2_world^2)-(x_P2_world_cam2^2)-
(y_P2_world_cam2^2));
x_P3_world_cam2 = s3_cam2_x*(x_P3_im2-x_M_im2);
y_P3_world_cam2 = s3_cam2_y*(y_P3_im2-y_M_im2);
z_P3_world_cam2 = sqrt((d_F2P3_world^2)-(x_P3_world_cam2^2)-
(y_P3_world_cam2^2));
x_P4_world_cam2 = s4_cam2_x*(x_P4_im2-x_M_im2);
y_P4_world_cam2 = s4_cam2_y*(y_P4_im2-y_M_im2);
z_P4_world_cam2 = sqrt((d_F2P4_world^2)-(x_P4_world_cam2^2)-
(y_P4_world_cam2^2));

%% ETAPA 5 - CAMERA 1
% Vectorul de poziție corespunzător punctelor P1_world, P2_world,
P3_world și P4_world, raportat la sistemul de referință cam1.
P1_world_cam1 = [x_P1_world_cam1; y_P1_world_cam1; z_P1_world_cam1];
P2_world_cam1 = [x_P2_world_cam1; y_P2_world_cam1; z_P2_world_cam1];
P3_world_cam1 = [x_P3_world_cam1; y_P3_world_cam1; z_P3_world_cam1];
P4_world_cam1 = [x_P4_world_cam1; y_P4_world_cam1; z_P4_world_cam1];

%% ETAPA 5 - CAMERA 2
% Vectorul de poziție corespunzător punctelor P1_world, P2_world,
P3_world și P4_world, raportat la sistemul de referință cam2.
P1_world_cam2 = [x_P1_world_cam2; y_P1_world_cam2; z_P1_world_cam2];
P2_world_cam2 = [x_P2_world_cam2; y_P2_world_cam2; z_P2_world_cam2];
P3_world_cam2 = [x_P3_world_cam2; y_P3_world_cam2; z_P3_world_cam2];
P4_world_cam2 = [x_P4_world_cam2; y_P4_world_cam2; z_P4_world_cam2];

%% ETAPA 6 - CAMERA 1
% Determinarea coordonatelor punctelor P1_world, P2_world, P3_world și
P4_world, raportate la sistemul de referință offset_cam1.
P1_world_offset_cam1 = P1_world_cam1 - P4_world_cam1;
x_P1_world_offset_cam1 = P1_world_offset_cam1(1);
y_P1_world_offset_cam1 = P1_world_offset_cam1(2);
z_P1_world_offset_cam1 = P1_world_offset_cam1(3);
P2_world_offset_cam1 = P2_world_cam1 - P4_world_cam1;
x_P2_world_offset_cam1 = P2_world_offset_cam1(1);
y_P2_world_offset_cam1 = P2_world_offset_cam1(2);
z_P2_world_offset_cam1 = P2_world_offset_cam1(3);
P3_world_offset_cam1 = P3_world_cam1 - P4_world_cam1;
x_P3_world_offset_cam1 = P3_world_offset_cam1(1);
y_P3_world_offset_cam1 = P3_world_offset_cam1(2);
z_P3_world_offset_cam1 = P3_world_offset_cam1(3);

%% ETAPA 6 - CAMERA 2
% Determinarea coordonatelor punctelor P1_world, P2_world, P3_world și
P4_world, raportate la sistemul de referință offset_cam2.
P1_world_offset_cam2 = P1_world_cam2 - P4_world_cam2;
x_P1_world_offset_cam2 = P1_world_offset_cam2(1);
y_P1_world_offset_cam2 = P1_world_offset_cam2(2);
z_P1_world_offset_cam2 = P1_world_offset_cam2(3);
P2_world_offset_cam2 = P2_world_cam2 - P4_world_cam2;
x_P2_world_offset_cam2 = P2_world_offset_cam2(1);
y_P2_world_offset_cam2 = P2_world_offset_cam2(2);
z_P2_world_offset_cam2 = P2_world_offset_cam2(3);
P3_world_offset_cam2 = P3_world_cam2 - P4_world_cam2;

```

## 190 Anexe

```
x_P3_world_offset_cam2 = P3_world_offset_cam2(1);
y_P3_world_offset_cam2 = P3_world_offset_cam2(2);
z_P3_world_offset_cam2 = P3_world_offset_cam2(3);

%% ETAPA 7
% Determinarea matricei de rotație utilizate pentru transformarea
valorilor coordonatelor raportate la sistemul de referință offset_cam1,
în valori raportate la sistemul de referință offset_cam2.
A = [x_P1_world_offset_cam1 y_P1_world_offset_cam1
z_P1_world_offset_cam1;...
x_P2_world_offset_cam1 y_P2_world_offset_cam1
z_P2_world_offset_cam1;...
x_P3_world_offset_cam1 y_P3_world_offset_cam1
z_P3_world_offset_cam1];
A1 = [x_P1_world_offset_cam2 y_P1_world_offset_cam1
z_P1_world_offset_cam1;...
x_P2_world_offset_cam2 y_P2_world_offset_cam1
z_P2_world_offset_cam1;...
x_P3_world_offset_cam2 y_P3_world_offset_cam1
z_P3_world_offset_cam1];
A2 = [x_P1_world_offset_cam1 x_P1_world_offset_cam2
z_P1_world_offset_cam1;...
x_P2_world_offset_cam1 x_P2_world_offset_cam2
z_P2_world_offset_cam1;...
x_P3_world_offset_cam1 x_P3_world_offset_cam2
z_P3_world_offset_cam1];
A3 = [x_P1_world_offset_cam1 y_P1_world_offset_cam1
x_P1_world_offset_cam2;...
x_P2_world_offset_cam1 y_P2_world_offset_cam1
x_P2_world_offset_cam2;...
x_P3_world_offset_cam1 y_P3_world_offset_cam1
x_P3_world_offset_cam2];
A4 = [y_P1_world_offset_cam2 y_P1_world_offset_cam1
z_P1_world_offset_cam1;...
y_P2_world_offset_cam2 y_P2_world_offset_cam1
z_P2_world_offset_cam1;...
y_P3_world_offset_cam2 y_P3_world_offset_cam1
z_P3_world_offset_cam1];
A5 = [x_P1_world_offset_cam1 y_P1_world_offset_cam2
z_P1_world_offset_cam1;...
x_P2_world_offset_cam1 y_P2_world_offset_cam2
z_P2_world_offset_cam1;...
x_P3_world_offset_cam1 y_P3_world_offset_cam2
z_P3_world_offset_cam1];
A6 = [x_P1_world_offset_cam1 y_P1_world_offset_cam1
y_P1_world_offset_cam2;...
x_P2_world_offset_cam1 y_P2_world_offset_cam1
y_P2_world_offset_cam2;...
x_P3_world_offset_cam1 y_P3_world_offset_cam1
y_P3_world_offset_cam2];
A7 = [z_P1_world_offset_cam2 y_P1_world_offset_cam1
z_P1_world_offset_cam1;...
z_P2_world_offset_cam2 y_P2_world_offset_cam1
z_P2_world_offset_cam1;...
```

```

    z_P3_world_offset_cam2 y_P3_world_offset_cam1
z_P3_world_offset_cam1];
A8 = [x_P1_world_offset_cam1 z_P1_world_offset_cam2
z_P1_world_offset_cam1;...
x_P2_world_offset_cam1 z_P2_world_offset_cam2
z_P2_world_offset_cam1;...
x_P3_world_offset_cam1 z_P3_world_offset_cam2
z_P3_world_offset_cam1];
A9 = [x_P1_world_offset_cam1 y_P1_world_offset_cam1
z_P1_world_offset_cam2;...
x_P2_world_offset_cam1 y_P2_world_offset_cam1
z_P2_world_offset_cam2;...
x_P3_world_offset_cam1 y_P3_world_offset_cam1
z_P3_world_offset_cam2];

r11 = det(A1)/det(A);
r12 = det(A2)/det(A);
r13 = det(A3)/det(A);
r21 = det(A4)/det(A);
r22 = det(A5)/det(A);
r23 = det(A6)/det(A);
r31 = det(A7)/det(A);
r32 = det(A8)/det(A);
r33 = det(A9)/det(A);

M_Rotatie = [r11 r12 r13; r21 r22 r23; r31 r32 r33];

a_str = sprintf('%2.4f',r11);
fid = fopen('parametrii\r11.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
a_str = sprintf('%2.4f',r12);
fid = fopen('parametrii\r12.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
a_str = sprintf('%2.4f',r13);
fid = fopen('parametrii\r13.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
a_str = sprintf('%2.4f',r21);
fid = fopen('parametrii\r21.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
a_str = sprintf('%2.4f',r22);
fid = fopen('parametrii\r22.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
a_str = sprintf('%2.4f',r23);
fid = fopen('parametrii\r23.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
a_str = sprintf('%2.4f',r31);
fid = fopen('parametrii\r31.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);

```

## 192 Anexe

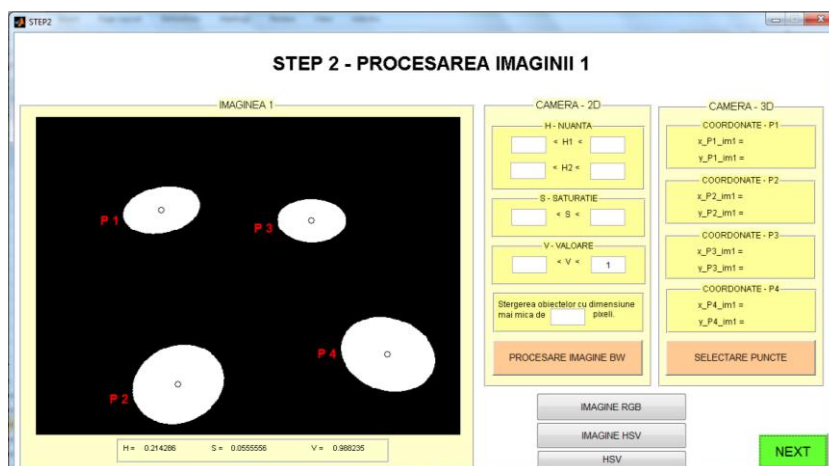
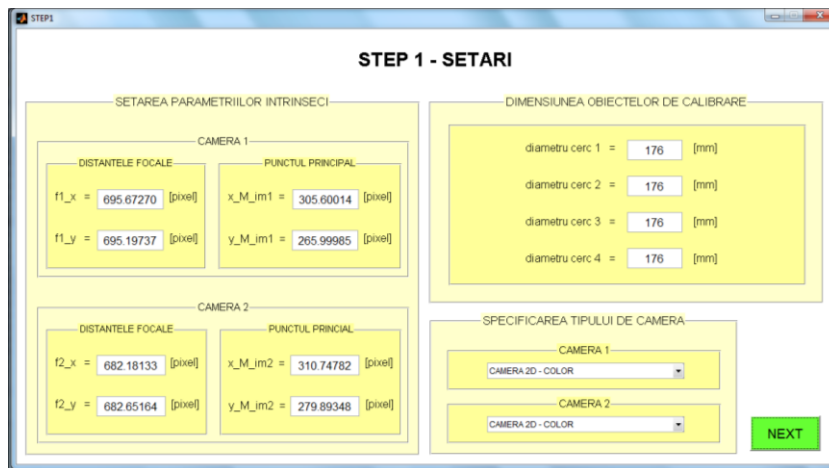
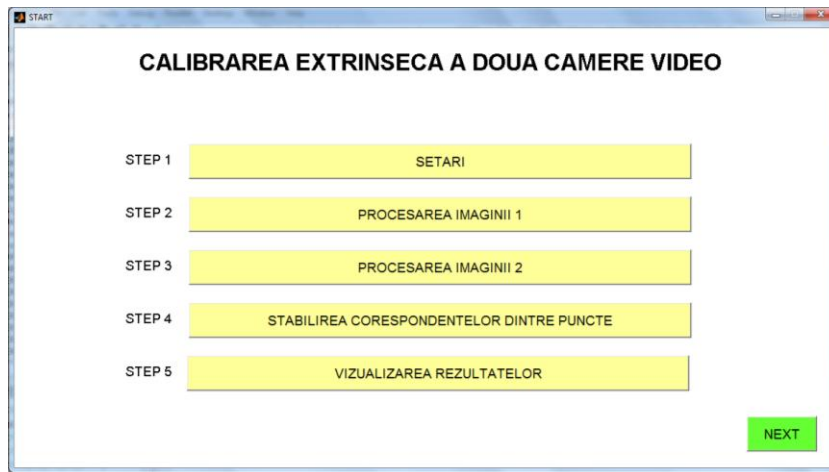
---

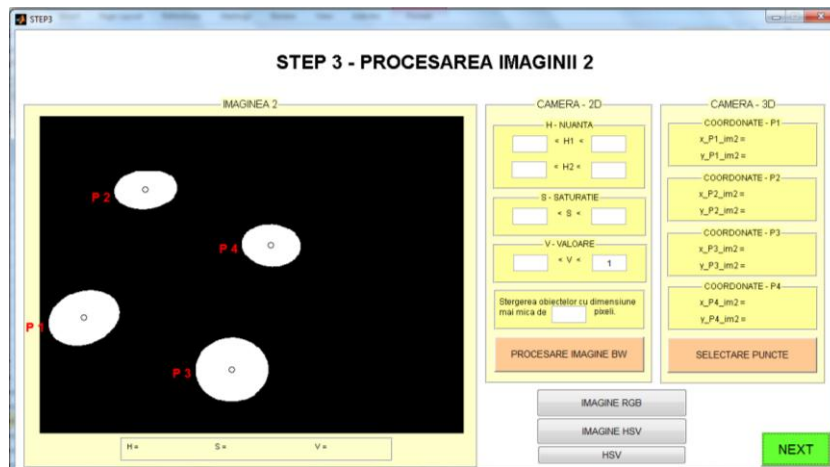
```
a_str = sprintf('%2.4f',r32);
fid = fopen('parametrii\r32.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
a_str = sprintf('%2.4f',r33);
fid = fopen('parametrii\r33.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);

%% ETAPA 8
% Determinarea vectorului de translație utilizat pentru transformarea
valorilor coordonatelor raportate la sistemul de referință cam1, în
valori raportate la sistemul de referință cam2.
M_Translatie = P1_world_cam2-M_Rotatie*P1_world_cam1;
t_x = M_Translatie(1);
t_y = M_Translatie(2);
t_z = M_Translatie(3);

a_str = sprintf('%2.2f',t_x);
fid = fopen('parametrii\t_x.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
a_str = sprintf('%2.2f',t_y);
fid = fopen('parametrii\t_y.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
a_str = sprintf('%2.2f',t_z);
fid = fopen('parametrii\t_z.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
```







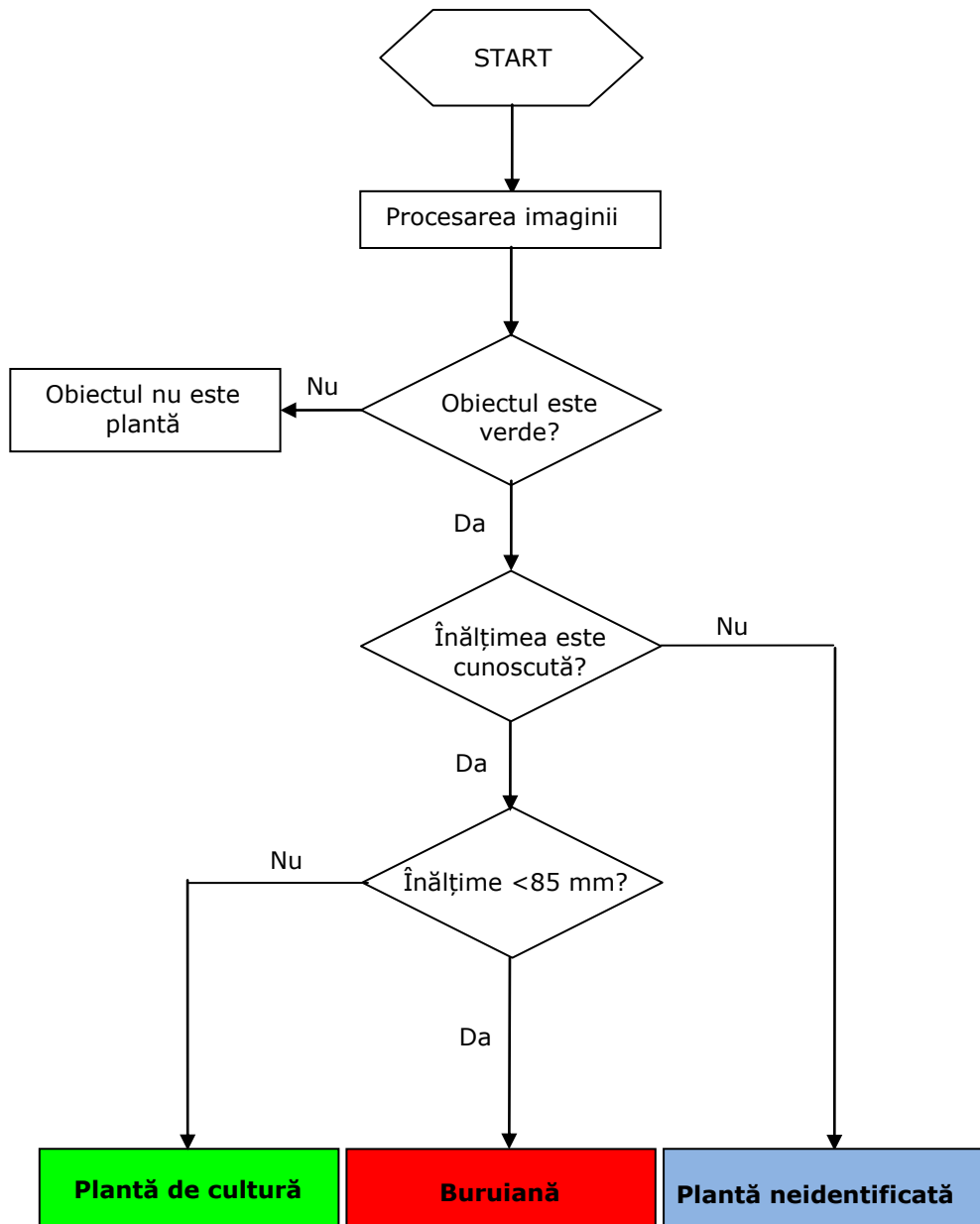
**Anexa D; Programe MATLAB utilizate la diferențierea buruiană/plantă pe baza înălțimilor**

Fig. D. 1. Schema logică a programului pentru deosebire buruiană/plantă

**Secvențe din programul principal:**

```

%% Distribuția valorii orientarilor
s = length(PUNCTE_im1);
for n = 1:1:s
    UNGHI(n) = (atan((PUNCTE_im2(1,n)-
PUNCTE_im1(1,n))/(PUNCTE_im1(2,n)-PUNCTE_im2(2,n)))*180/pi;
    [d_F1P_world(n),
d_F2P_world(n)]=determinarea_distantelor(x_PUNCTE_im1(n),
y_PUNCTE_im1(n), x_PUNCTE_im2(n), y_PUNCTE_im2(n));
end

figure;
x = 1:1:s;
y = UNGHI;
plot(x,y,'m*')
xlabel('puncte')
ylabel('orientare [grade]')
title('Analiza orientării segmentelor')

%% Eliminarea erorilor
MEDIE_UNGHI = mean(UNGHI);
[f1_x, f1_y, x_M_im1, y_M_im1, f2_x, f2_y, x_M_im2, y_M_im2,...
r11, r12, r13, r21, r22, r23, r31, r32, r33, t_x, t_y, t_z, a, b,
c, d] = setari();
distanța_x = abs(t_x);
distanța_y = abs(t_y);
distanța_z = abs(t_z);
distanța_max = max([distanța_x,distanța_y,distanța_z]);
for n = 1:1:s
    if ((UNGHI(n)>MEDIE_UNGHI+25)|| (UNGHI(n)<MEDIE_UNGHI-
25)|| ((abs(d_F1P_world(n)-d_F2P_world(n))>(abs(0.7*distanța_max))))
        PUNCTE_im1(2,n) = 0;
        PUNCTE_im2(2,n) = 0;
        PUNCTE_im1(1,n) = 0;
        PUNCTE_im2(1,n) = 0;
    end
end

x_PUNCTE_im1 = PUNCTE_im1(1,:);
x_PUNCTE_im1_FINAL = x_PUNCTE_im1(find(x_PUNCTE_im1~=0));
y_PUNCTE_im1 = PUNCTE_im1(2,:);
y_PUNCTE_im1_FINAL = y_PUNCTE_im1(find(y_PUNCTE_im1~=0));
x_PUNCTE_im2 = PUNCTE_im2(1,:);
x_PUNCTE_im2_FINAL = x_PUNCTE_im2(find(x_PUNCTE_im2~=0));
y_PUNCTE_im2 = PUNCTE_im2(2,:);
y_PUNCTE_im2_FINAL = y_PUNCTE_im2(find(y_PUNCTE_im2~=0));

%% Afișarea celor două imagini și a punctelor omoloage
figure;
imshow(cam1); hold on;
plot(x_PUNCTE_im1_FINAL,y_PUNCTE_im1_FINAL,'r+');
figure;
imshow(cam2); hold on;
plot(x_PUNCTE_im2_FINAL,y_PUNCTE_im2_FINAL,'g+');

```

```

show(double(im1)+double(im2)), hold on
plot(x_PUNCTE_im1_FINAL,y_PUNCTE_im1_FINAL,'r+');
plot(x_PUNCTE_im2_FINAL,y_PUNCTE_im2_FINAL,'g+');
for n = 1:length(x_PUNCTE_im1_FINAL)
    line([x_PUNCTE_im1_FINAL(n) x_PUNCTE_im2_FINAL(n)],
[y_PUNCTE_im1_FINAL(n) y_PUNCTE_im2_FINAL(n)],'color',[0 0 1])
end

```

```

%% Calculul distanțelor între puncte și planul solului
for i = 1:length(x_PUNCTE_im1_FINAL)
    dist_punct_plan(i)=distanța_punct_plan(x_PUNCTE_im1_FINAL(i),
y_PUNCTE_im1_FINAL(i), x_PUNCTE_im2_FINAL(i), y_PUNCTE_im2_FINAL(i));
end
detectare_plante(x_PUNCTE_im1_FINAL, y_PUNCTE_im1_FINAL,
dist_punct_plan);

```

### Secvențe din programul de detectare a plantelor:

```

[B,L] = bwboundaries(imagine_BW_SELECTARE1,'noholes');
stats = regionprops(L,'Centroid');
for k = 1:length(B)
    boundary = B{k};
    for i = 1:length(c)
        obiect(i)=L(r(i),c(i));
% Detectarea buruienilor
        if (d(i)<prag_inaltime)&&((L(r(i),c(i)))~=0);
            centroid = stats(obiect(i)).Centroid;
            hold on
            plot(centroid(1), centroid(2), 'yo', 'LineWidth', 1)
            boundary = B{obiect(i)};
            plot(boundary(:,2), boundary(:,1), 'r', 'LineWidth', 3)
            denumire_plante = 'buruiena';
            text(centroid(1)-
25,centroid(2)+13,denumire_plante,'Color','y',...
'FontSize',12);
        end
% Detectarea plantelor de cultură
        if (d(i)>prag_inaltime)&&((L(r(i),c(i)))~=0);
            centroid = stats(obiect(i)).Centroid;
            hold on
            plot(centroid(1), centroid(2), 'yo', 'LineWidth', 1)
            boundary = B{obiect(i)};
            plot(boundary(:,2), boundary(:,1), 'g', 'LineWidth', 3)
            denumire_plante = 'planta de cultura';
            text(centroid(1)-
25,centroid(2)+13,denumire_plante,'Color','y',...
'FontSize',12);
        end
    end
end
end

```

## Anexa E; Programe MATLAB pentru corelarea imaginilor

### Programul de corelare a imaginilor (ordinogramă, secvențe de cod, interfața grafică)

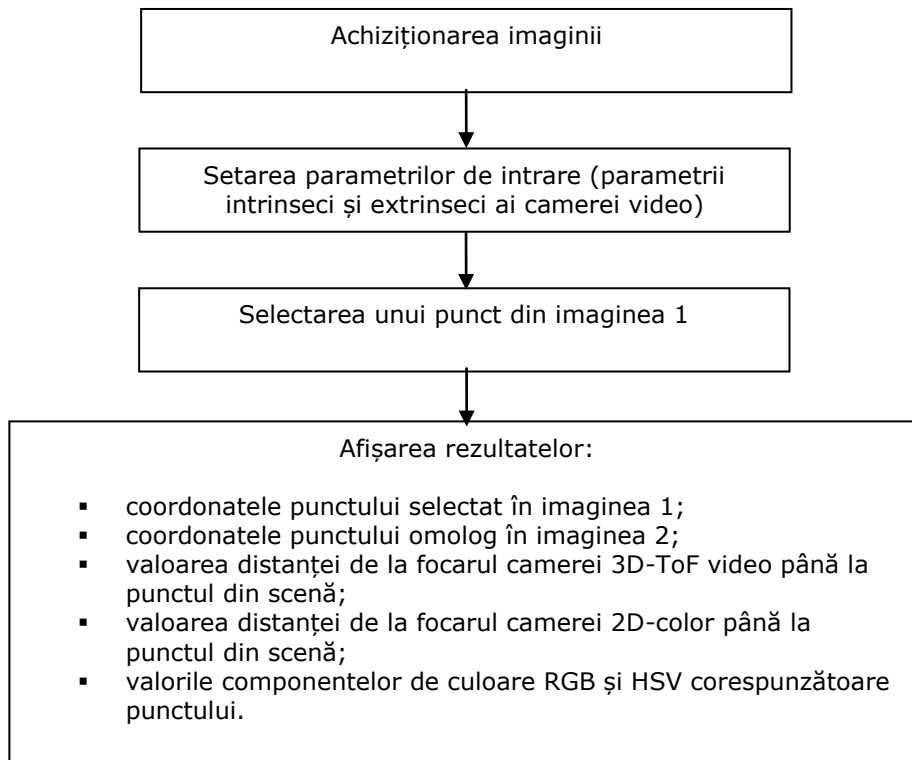


Fig. E. 1. Ordinograma programului de corelare a imaginilor

```

%% ETAPA 1
% Coordonatele proiecțiilor în imaginea 1 ale centrelor cercurilor sunt
raportate la sistemul de referință al imaginii 1.
P_im1 = [x_P_im1; y_P_im1; 0];

%% ETAPA 2
% Calculul distanțelor de la punctul focal F_x respectiv F_y până la
fiecare proiecție a centrelor cercurilor în imaginea 1.
d_F1P_im1_x = sqrt((x_P_im1-x_M_im1)^2+((y_P_im1-
y_M_im1)*f1_x/f1_y)^2+f1_x^2);
d_F1P_im1_y = sqrt(((x_P_im1-x_M_im1)*f1_y/f1_x)^2+(y_P_im1-
y_M_im1)^2+f1_y^2);

%% ETAPA 3
% Determinarea factorilor de scalare corespunzători fiecărui punct.
Factorii de scalare sunt utilizați la conversia din pixel în mm, iar
valorile acestora sunt distincte pe axele x și y.
  
```

```

s_cam1_x = d_F1P_world/d_F1P_im1_x;
s_cam1_y = d_F1P_world/d_F1P_im1_y;

%% ETAPA 4
% Determinarea coordonatelor punctelor P1_world, P2_world, P3_world și
P4_world, raportate la sistemul de referința cam1.
x_P_world_cam1 = s_cam1_x*(x_P_im1-x_M_im1);
y_P_world_cam1 = s_cam1_y*(y_P_im1-y_M_im1);
z_P_world_cam1 = sqrt((d_F1P_world^2)-(x_P_world_cam1^2)-
(y_P_world_cam1^2));
P_world_cam1 = [x_P_world_cam1; y_P_world_cam1; z_P_world_cam1];

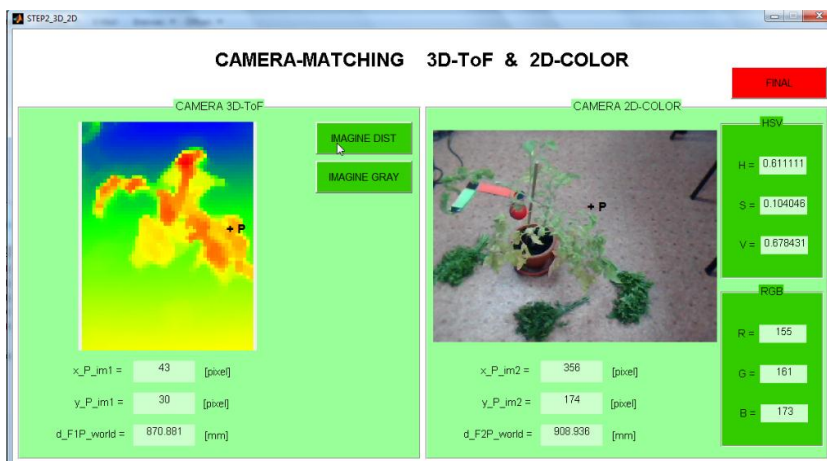
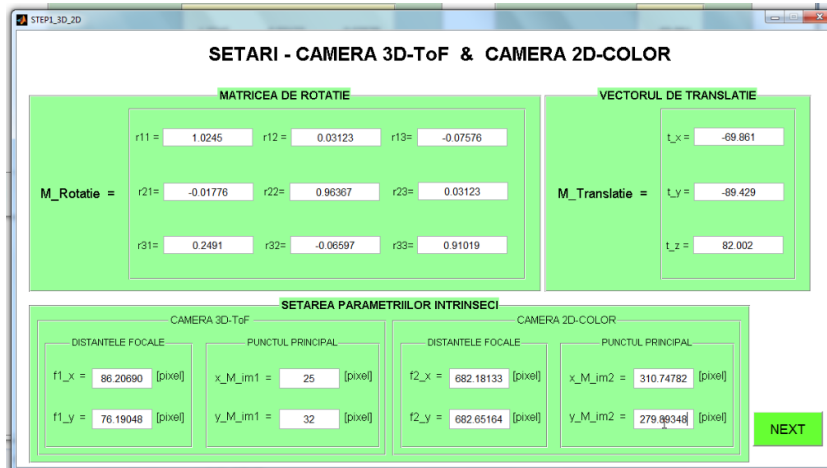
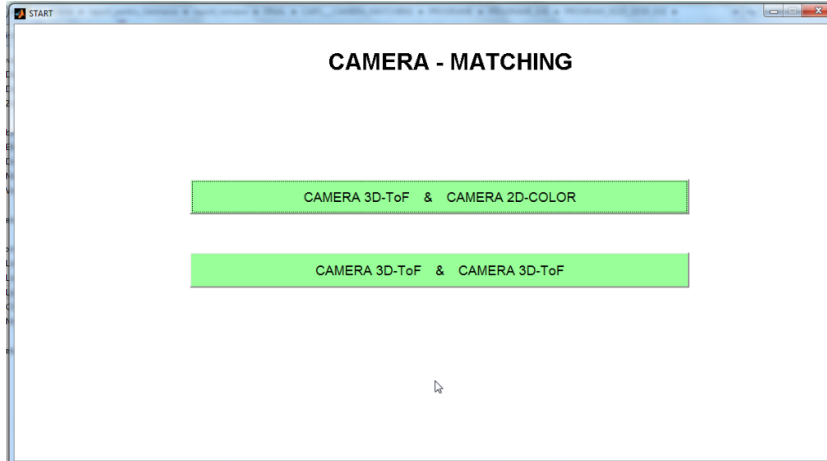
%% ETAPA 5
% Determinarea coordonatelor punctului P_world, raportat la sistemul de
referința cam2. Transformarea din sistemul de referința cam1 în
sistemul de referința cam2 este realizată utilizând matricea de rotație
și vectorul de translație determinate în etapa de calibrare extrinsecă
a camerelor.
M_Rotatie = [r11 r12 r13; r21 r22 r23; r31 r32 r33];
M_Translatie = [t_x; t_y; t_z];
P_world_cam2 = M_Rotatie*P_world_cam1+M_Translatie;
x_P_world_cam2 = P_world_cam2(1);
y_P_world_cam2 = P_world_cam2(2);
z_P_world_cam2 = P_world_cam2(3);

%% ETAPA 6
% Determinarea factorilor de scalare corespunzători punctului P
raportat la camera2. Factorii de scalare sunt utilizați la conversia
din pixel în mm, iar valorile acestora sunt distincte pe axele x și y.
s_cam2_x = z_P_world_cam2/f2_x;
s_cam2_y = z_P_world_cam2/f2_y;

%% ETAPA 7
% Determinarea coordonatelor proiectilor punctului P în imaginea 2,
raportate la sistemul de referință cam2.
x_P_im2_cam2 = x_P_world_cam2/s_cam2_x;
y_P_im2_cam2 = y_P_world_cam2/s_cam2_y;

%% ETAPA 8
% Determinarea coordonatelor proiectilor punctului P în imaginea 2,
raportate la sistemul de referința im2.
x_P_im2 = x_P_im2_cam2+x_M_im2;
y_P_im2 = y_P_im2_cam2+y_M_im2;
d_F2P_world = sqrt(x_P_world_cam2^2 + y_P_world_cam2^2 +
z_P_world_cam2^2);

```





## Anexa F; Programe utilizate la sistemul de navigare al robotului

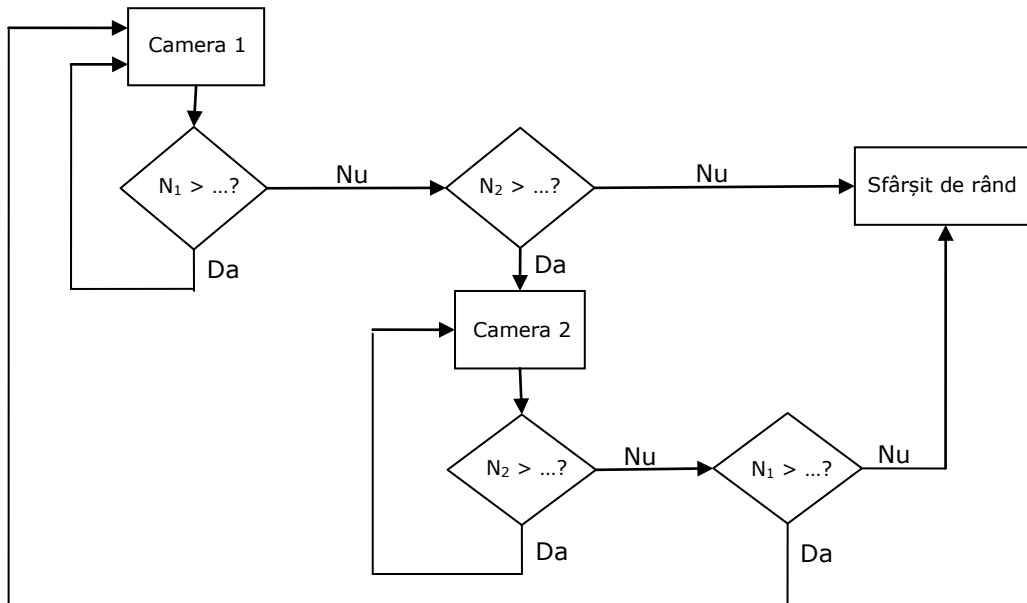


Fig. F. 1. Schema logică a programului de detectare a rândurilor

### Program C++ pentru localizarea rândului de culturi prin metoda regresiei liniare:

```

#include <iostream>
#include <sstream>
#include <string>
#include <fstream>
#include <map>
#include <stdlib.h>
#include <stdio.h>
#include <cmath>
using namespace std;

template <class T>
bool from_string(T& t, const std::string& s, std::ios_base&
(*f)(std::ios_base&))
{
    std::istringstream iss(s);
    return !(iss>>f>>t).fail();
}

void main()
{
    fstream f;
    string str;
  
```

## 202 Anexe

```
float daten[3200];
int i=0;
f.open("E:\\OSNABRUCK\\Field_Robot_Event\\Miha\\soft-
3D\\Licht\\soft.dat");
while (! f.eof() )
{
getline(f, str); // citirea randurilor

from_string<float>(daten[i], str, std::dec);

i++;
}
f.close();

double Hmin = 0.47;
double Hmax = 0.61;
double x;
double y;
int count=0;
int n;
double a1;
double a0;
double m1 = 0;
double m2 = 0;
double m3 = 0;
double m4 = 0;
for (int i = 0; i<3200; i++)
{
    if((daten[i] > Hmin) && (daten[i] < Hmax))
    {
        count++;

        x = i - floor(i/64.0)*64;
        y = floor(i/64.0)+1;

        m1 += x*y;
        m2 += y;
        m3 += x;
        m4 += pow(y,2.0);

        cout<< i << "\t" << count << "\t" << x <<
"\t" << y << "\t" << daten[i] << " \n";
    }
}
n = count;

a1 = (n*m1 - m2*m3)/(n*m4 - pow(m2,2.0));
a0 = (m4*m3 - m2*m1)/(n*m4 - pow(m2,2.0));

cout << "a1 = " << a1 <<endl;
cout << "a0 = " << a0 <<endl;
}
```

## Anexa G; Măsurarea distanțelor utilizând vederea stereo

### Anexa G Programul de măsurarea distanțelor utilizând vederea stereo (ordinogramă, secvențe de cod, interfață grafică)

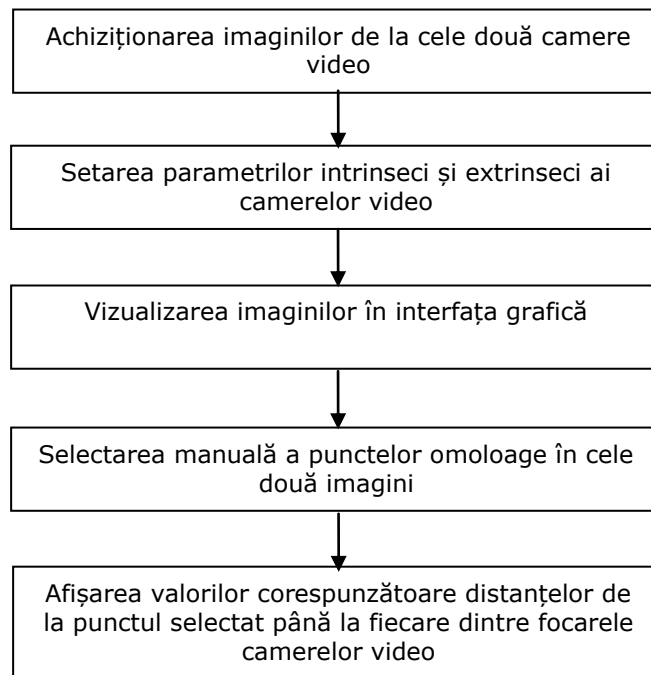


Fig. G. 1. Ordinograma programului de determinare a distanțelor prin vedere stereo

```

function []=determinarea_distantelor()

% Etapa1_camera1
x_P_im1 = importdata('parametrii\x_P_im1.txt');
y_P_im1 = importdata('parametrii\y_P_im1.txt');
z_P_im1 = 0;
% Etapa1_camera2
x_P_im2 = importdata('parametrii\x_P_im2.txt');
y_P_im2 = importdata('parametrii\y_P_im2.txt');
z_P_im2 = 0;
% Etapa2_camera1
I = imread('imagini\image_camera1.jpg');
[y_size_im1, x_size_im1, ii] = size(I);
fl_x = importdata('parametrii\fl_x.txt');
fl_y = importdata('parametrii\fl_y.txt');
x_M_im1 = importdata('parametrii\x_M_im1.txt');
y_M_im1 = importdata('parametrii\y_M_im1.txt');
  
```

## 204 Anexe

```
x_P_im1_cam1 = x_P_im1-x_M_im1;
y_P_im1_cam1 = y_P_im1-y_M_im1;
z_P_im1_cam1 = (f1_x+f1_y)/2;
% Etapa2_camera2
J = imread('imagini\image_camera2.jpg');
[y_size_im2, x_size_im2, ii] = size(J);
f2_x = importdata('parametrii\f2_x.txt');
f2_y = importdata('parametrii\f2_y.txt');
x_M_im2 = importdata('parametrii\x_M_im2.txt');
y_M_im2 = importdata('parametrii\y_M_im2.txt');
x_P_im2_cam2 = x_P_im2-x_M_im2;
y_P_im2_cam2 = y_P_im2-y_M_im2;
z_P_im2_cam2 = (f2_x+f2_y)/2;

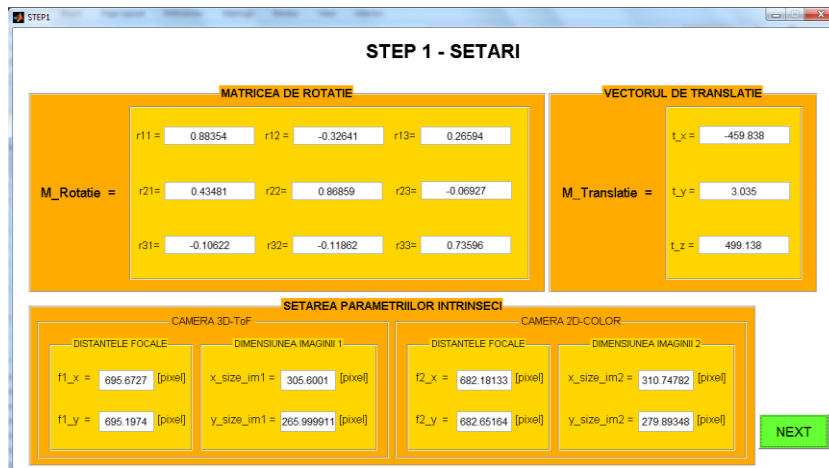
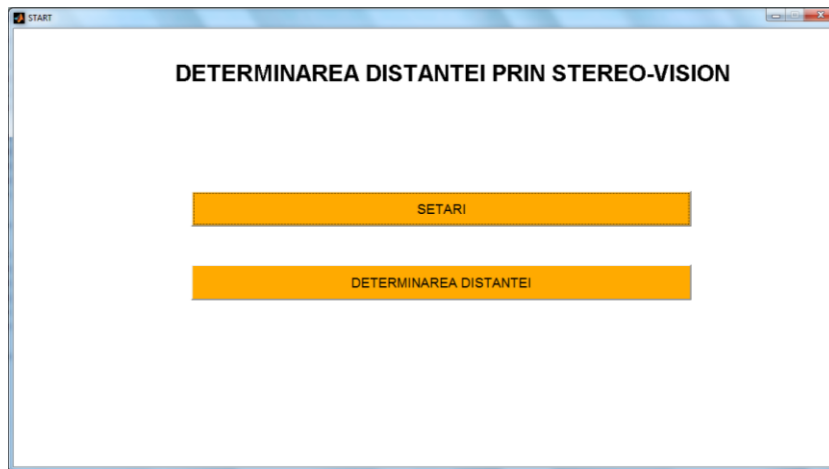
% Etapa3
r11 = importdata('parametrii\r11.txt');
r12 = importdata('parametrii\r12.txt');
r13 = importdata('parametrii\r13.txt');
r21 = importdata('parametrii\r21.txt');
r22 = importdata('parametrii\r22.txt');
r23 = importdata('parametrii\r23.txt');
t_x = importdata('parametrii\t_x.txt');
t_y = importdata('parametrii\t_y.txt');

v1 = r11*x_P_im1_cam1+r12*y_P_im1_cam1+r13*z_P_im1_cam1;
v2 = r21*x_P_im1_cam1+r22*y_P_im1_cam1+r23*z_P_im1_cam1;

A = det([-v1 x_P_im2_cam2; -v2 y_P_im2_cam2]);
A1 = det([t_x x_P_im2_cam2; t_y y_P_im2_cam2]);
A2 = det([-v1 t_x; -v2 t_y]);
s_cam1 = A1/A;
s_cam2 = A2/A;

% Etapa4_camera1
x_P_world_cam1 = s_cam1*x_P_im1_cam1;
y_P_world_cam1 = s_cam1*y_P_im1_cam1;
z_P_world_cam1 = s_cam1*z_P_im1_cam1;
d_F1P_world = sqrt(x_P_world_cam1^2+y_P_world_cam1^2+z_P_world_cam1^2);
a_str = sprintf('%2.3f',d_F1P_world);
fid = fopen('parametrii\d_F1P_world.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);

% Etapa4_camera2
x_P_world_cam2 = s_cam2*x_P_im2_cam2;
y_P_world_cam2 = s_cam2*y_P_im2_cam2;
z_P_world_cam2 = s_cam2*z_P_im2_cam2;
d_F2P_world = sqrt(x_P_world_cam2^2+y_P_world_cam2^2+z_P_world_cam2^2);
a_str = sprintf('%2.3f',d_F2P_world);
fid = fopen('parametrii\d_F2P_world.txt','w');
fprintf(fid, '%s', a_str);
fclose(fid);
```





## Curriculum vitae

Nume:	Tilneac
Prenume:	Mihaela
Data și locul nașterii:	2 octombrie 1982, Timișoara, România
Naționalitate:	Română
Adresă:	Bulevardul Mihai Viteazu, nr.1, Timisoara, cod 300222, România
Telefon:	0040-(0)726471753
E-mail:	<a href="mailto:mihaelatilneac@gmail.com">mihaelatilneac@gmail.com</a> sau <a href="mailto:mihaela.tilneac@mec.upt.ro">mihaela.tilneac@mec.upt.ro</a>

### Studii:

Perioada:	Octombrie 2008 - februarie 2012
Unitatea de învățământ:	Universitatea POLITEHNICA din Timișoara, Facultatea de Mecanică, Departamentul de Mecatronică
Nivel de studii:	Doctorat
Specializarea:	Roboți pentru agricultură
Titlul tezei de doctorat:	Contribuții la utilizarea roboților pentru agricultură în aplicații de combatere a buruienilor

Perioada:	Octombrie 2006- iunie 2008
Unitatea de învățământ:	Universitatea POLITEHNICA din Timișoara, Facultatea de Mecanică, Departamentul de Mecatronică
Nivel de studii:	Master
Specializarea:	Sisteme robotice cu inteligență artificială
Titlul lucrării de disertație:	Aplicații robotizate pentru prelucrarea suprafețelor complexe

Perioada:	Octombrie 2001- iunie 2006
Unitatea de învățământ:	Universitatea POLITEHNICA din Timișoara, Facultatea de Mecanică, Departamentul de Mecatronică
Nivel de studii:	Diplomă
Specializarea:	Roboți industriali (în limba germană)
Titlul lucrării de diplomă:	Sistematizare actualizată a dispozitivelor de prehensiune în vederea proiectării unitare a acestora

### **Experiență profesională:**

Perioada:	Martie 2010- august 2010
Unitatea:	„Hochschule Osnabrück” din Germania
Activitatea:	Stagiu de cercetare
Domeniul:	Roboți pentru agricultură
Perioada:	Decembrie 2006- octombrie 2008
Unitatea:	S.C. STIMEL S.A. , Timișoara
Funcția:	Inginer diplomat
Activitatea:	Proiectare CAD și programare CNC
Perioada:	Iulie 2006- decembrie 2006
Unitatea:	S.C. SoftgateGroup, Timișoara
Funcția:	Inginer diplomat
Activitatea:	Proiectare GIS (Geographic Information System)
Perioada:	Septembrie 2004 - iunie 2006
Unitatea:	Școala cu Clasele I-VIII, Otelec, jud. Timiș
Funcția:	Profesor suplinitor
Disciplina:	Limba germană

### **Aptitudini în domeniul științelor ingineresti:**

Proiectare CAD:	CATIA, Pro/Engineer, SolidWorks, AutoCAD, Cimatron 2D
Limbaje de programare:	Matlab, C++, Visual Basic, HTML, Programare mașini-unelte CNC
Programe de editare:	Microsoft Office, LaTeX
Optoelectronică:	Vedere stereoscopică, procesare digitală a imaginilor
Electronică:	Programare microcontroler BASIC Stamp

### **Limbi străine:**

Germană:	Avansat
Engleză:	Mediu
Franceză:	Mediu
Italiană:	Începător

### **Calificare didactică:**

Absolvirea cursului de pedagogie pentru desfășurarea activității didactice.