

Raportul Stiintific si Tehnic (RST) in extenso

SABIMAS

Etapa III

Realizarea sistemului de prelucrare si analiza a imaginilor radiografice ale osului femural proximal si experimentarea sa in practica medicala

CUPRINS

OBIECTIVE GENERALE.....	3
OBIECTIVELE FAZEI DE EXECUTIE.....	5
REZUMATUL FAZEI.....	6
1. INTRODUCERE	8
1.1. Scopul documentului.....	8
1.2. Obiectivul sistemului de analiza si prelucrare a imaginilor radiografice.....	8
1.3. Publicul tinta.....	9
1.4. Privire de ansamblu asupra documentului	9
2. PRIVIRE DE ANSAMBLU ASUPRA SISTEMULUI	9
2.1. Functionalitatile aplicatiei	9
2.2. Elemente anatomice de baza in artroplastia soldului	10
3. TEHNOLOGII FOLOSITE	12
3.1. Microsoft Visual Studio 2008	12
3.2. C#.....	12
3.3. Biblioteci grafice	12
3.4. MySQL.....	12
3.5. DICOM - OpenDicom	13
4. PROIECTAREA DE ANSAMBLU A APLICATIEI.....	14
4.1. Modulele functionale	14
4.1.1. Modulul pacienti.....	14
4.1.2. Modulul radiografii.....	14
4.1.3. Modulul de manipulare imagini	14
4.1.4. Modulul de masuratori manuale.....	15
4.1.5. Modulul de masuratori automate.....	15
4.2. Diagrama de clase	16
5. PROIECTAREA BAZEI DE DATE	17
5.1. Schema bazei de date.....	17
5.2. Descrierea tabelor bazei de date.....	17
6. PROIECTAREA DE DETALIU	21
6.1. Algoritmi folositi	21
6.1.1. Transformata Hough	25
6.1.2. Transformata Hough pentru linii.....	25
6.1.3. Transformata Hough pentru cercuri.....	26
6.1.4. Adaptarea Transformatei Hough pentru extragerea parametrilor din radiografii	27

6.2.	Descrierea detaliata a claselor	28
6.2.1.	Clasele linie, cerc, masuratori.....	28
6.2.2.	Clasa Form_pacienti	30
6.2.3.	Clasa Dicom	35
6.2.4.	Clasa BitmapPrimitive.....	35
6.2.5.	Clasa Canny.....	37
6.2.6.	Clasa Hough.....	38
6.2.7.	Clasa MasurariAutomate	43
6.2.8.	Clasa Form_radiografii.....	49
7.	INTERFATA UTILIZATOR.....	55
7.1.	Form-ul pacienti	55
7.1.1.	Grid-ul de vizualizare pacienti	56
7.1.2.	Sectiunea de adaugare pacienti/stergere/editare pacienti	57
7.1.3.	Sectiunea de cautare pacienti	60
7.1.4.	Sectiunea Radiografii.....	61
7.2.	Form-ul radiografii.....	63
7.2.1.	Tab-ul de Informatii.....	63
7.2.2.	Tab-ul de Masuratori manuale	64
7.2.3.	Tab-ul masuratori automate	68
7.2.4.	Tab-ul Manipulare fisiere	70
7.2.5.	Zoom.....	71
8.	EXPERIMENTAREA SISTEMULUI IN PRACTICA MEDICALA.....	72
	Studiul descris in acest capitol a fost realizat de partenerii de la SCUB SI UMF.....	72
9.	CONCLUZII SI DIRECTII VIITOARE DE DEZVOLTARE	92
	Bibliografie	94

OBIECTIVE GENERALE

Obiectivul general al proiectului este studierea corelatiei ce trebuie realizata intre forma canalului femural, evaluata prin metode imagistice moderne, si forma protezei. Astfel se va obtine o protezare cu o compatibilitate mecanica cat mai aproape de cea fiziologica, cu o adaptare optima a protezei la osul in care este implantata, permitand recuperarea precoce a pacientului din punct de vedere fizic, psihic si social.

Scopul proiectului este acela de a crea, prin metode informatice avansate, elementele necesare pentru ca protezarea sa aiba caracter personalizat, adaptat fiecarui individ. Din acest punct de vedere, proiectul se adreseaza diversitatii biologice, pe care isi propune sa o conserve, prin fundamentarea stiintifica si dezvoltarea de tehnologii care sa respecte si sa mentina caracterul particular al fiecarui individ.

De asemenea, proiectul isi propune dezvoltarea unor tehnologii care sa creasca eficienta unei metode terapeutice, ducand astfel la eficientizarea serviciilor de sanatate, prin scaderea numarului de reinterventii necesare, dar si la cresterea creativitatii in domeniul sanitar, fiind vorba despre adaptarea protezarii la caracteristicile fiecarui individ.

In acest context, obiectivele specifice ale proiectului se regasesc in:

- implementarea de metode noi de preventie si interventionale la nivel national, arondate la spatiul european de operare, cu aplicatii in domeniul artroplastiei de sold;
- realizarea unui sistem informatic de investigare si prototipare incluzand activitati de analiza, diagnoza si monitorizare computerizata
- dezvoltarea de software specializat pentru prelucrarea / analiza imaginilor radiografice și modelarea tri-dimensională a formei și structurii oaselor pornind de la imagini CT sau MRI
- demonstrarea posibilității de obținere a protezelor personalizate prin sinterizare, folosind modele 3D generate de software-ul dezvoltat in cadrul proiectului.

OBIECTIVELE FAZEI DE EXECUTIE

Etapa a constat in:

- **Realizarea sistemului de prelucrare si analiza a imaginilor radiografice ale osului femural proximal si experimentarea sa in practica medicala**

Activitatile care au stat la baza realizării etapei si de care au raspuns Universitatea POLITEHNICA Bucuresti si Institutul National de Cercetare Dezvoltare pentru Mecatronica si Tehnica Masurarii au fost:

- **Proiectarea sistemului de prelucrare si analiza a imaginilor radiografice ale osului femural proximal (UPB+INCDMTM)**
- **Implementarea si testarea sistemului de prelucrare si analiza a imaginilor radiografice (UPB)**

Activitatile care au stat la baza realizării etapei si de care au raspuns Universitatea de Medicina si Farmacie “CAROL DAVILA” Bucuresti si Spitalul Clinic de Urgenta Floreasca din Bucuresti au fost:

- **Experimentarea sistemului de analiza a imaginilor radiografice in studiul radiografiilor femurale (SCUB+UMF)**

De asemenea, s-a realizat si diseminarea informatiilor : Participare la targuri, expozitii, conferinte si seminarii in domeniu.

REZUMATUL FAZEI

In cadrul etapei s-au realizat:

- Proiectarea de ansamblu a Sistemului de Prelucrare si Analiza a Imaginilor Radiografice
- Proiectarea bazei de date
- Proiectarea de detaliu a sistemului
- Implementarea si testarea Sistemului de Prelucrare si Analiza a Imaginilor Radiografice
- Experimentarea sistemului in studiul radiografiilor femurale si intocmirea Rapoartelor cu concluziile rezultate in urma utilizarii sale
- Diseminarea rezultatelor

In cadrul activitatii de proiectare de ansamblu au fost identificate modulele functionale ale aplicatiei, pornind de la documentul de specificare a cerintelor elaborat in etapa a 2-a. De asemenea, a fost proiectata arhitectura aplicatiei, descrisa in acest document printr-o diagrama de clase.

In etapa de proiectare de detaliu au fost analizati o serie de algoritmi de prelucrare si analiza de imagini, in scopul alegerii celor mai adecvati pentru cerintele aplicatiei. De asemenea, a fost detaliata structura claselor aplicatiei.

In proiectarea bazei de date am avut in vedere:

- Necesitatile de stocare a datelor strict legate de functionalitatile principale ale aplicatiei
- Posibilitatea utilizarii sale pentru gestiunea inter- si intra-institutionala a pacientilor
- Posibilitatea stocarii de informatii medicale pe perioade de timp lungi, pentru urmarirea evolutiei starii de sanatate a pacientilor

Pornind de la acest proiect, se poate crea o retea a tuturor spitalelor locale (de exemplu a tuturor spitalelor din Bucuresti) care sa tina o evidenta completa si unitara a tuturor pacientilor. Aceasta aplicatie ar simplifica foarte mult munca personalului din spitale, datorita existentei unei baze de date globale, cu datele si istoricul medical al fiecarui pacient.

Pastrarea radiografiilor si a masuratorilor lor intr-un format electronic reprezinta un pas inainte pentru sistemul medical din tara noastra. Posibilitatea realizarii de masuratori manuale si automate asupra radiografiilor pacientilor inainte si dupa protezare, cu ajutorul unui software specializat, si salvarea lor intr-o baza de date, deschide noi orizonturi in ce priveste gestiunea pacientilor, urmarirea evolutiei acestora inainte si dupa operatie, realizarea de colaborari intre spitale, etc.

Aplicatia dezvoltata in cadrul acestei etape a fost testata la UPB pe mai multe imagini radiografice in format DICOM si BITMAP.

Analiza imaginilor radiografice in scopul extragerii informatiei de interes este ingreunata de lipsa contrastului dintre tesutul osos si celelalte tesuturi. Acest inconvenient ingreuneaza efectuarea automata a tuturor masuratorilor necesare in artroplastia de sold. Testele pe care le-am efectuat au demonstrat ca in circa 80% dintre cazuri (imagini) masuratorile automate au fost corecte. Din acest motiv, aplicatia permite interventia medicului pentru corectarea parametrilor calculati de aplicatie si efectuarea de masuratori manuale, dar asistate de aplicatie .

Sistemul de Prelucrare si Analiza a Imaginilor Radiografice a fost utilizat in cadrul unui studiu efectuat de partenerii de la SCUB si UMF, pentru a se obtine Rapoartele privind acuratetea rezultatelor produse de sistem.

Studiul s-a efectuat pe un numar de 22 de pacienti (cu varste intre 42- 65 ani) supusi artroplastiei totale necimentate de sold. La acesti pacienti au fost efectuate 20 de artroplastii (4 bilaterale) pentru coxartroza primara (14 pacienti) si coxartroza posttraumatica (2 pacienti).

A fost aplicat acelasi protocol efectuand radiografiile standard antero-posterioare si laterale la 3, 6 si 12 luni postoperator. Studiul radiologic a fost facut prin compararea soldului opus care putea fi normal, cu semne de artroza sau supus unei alte artroplastii.

Diferentele dintre axul protezei si axele anatomice calculate automat de aplicatie si manual de specialistii in ortopedie au fost in ambele cazuri sub 2% in grupul de studiu, demonstrand atat o pozitionare corecta a implantului, cat si o determinare automata corecta a parametrilor.

Directiile de dezvoltare in anul viitor se vor centra pe realizarea de masuratori pe imagini medicale 3D, vizualizarea acestora, precum si extragerea suprafetelor 3D ale oaselor din imagini CT si RMN, in vederea prototiparii protezelor pentru asistarea preoperatorie.

Avand in vedere timpul mare necesar pentru procesarea imaginilor 2D, este de asteptat ca procesarea imaginilor 3D sa dureze si mai mult. Din acest motiv, una dintre directiile de cercetare, in care UPB deja a avansat, este paralelizarea algoritmilor de prelucrare a imaginilor 3D (*Marching Cubes* si *Ray Casting* folosind *CUDA* si *OpenCL*).

Rezultatele obtinute pana la finalul etapei a 3-a au fost publicate in urmatoarele articole:

- International Conference on Signal Processing : Session Image Processing, Catania, Italia, 29-31 Mai 2010, „**Computer Assisted Analysis of Orthopedic Radiographic Images**”, autori: Anca Morar, Florica Moldoveanu, Alin Moldoveanu, Victor Asavei, Alexandru Egner – Indexata ISI.

- International Conference on Telecommunications and Informatics : Session Information Science and Applications, Catania, Italia, 29-31 Mai 2010, „**GPGPU for Cheaper 3D MMO Servers**”, Autori Victor Asavei, Alin Moldoveanu, Florica Moldoveanu, Anca Morar, Alexandru Egner – Indexata ISI.
- Romanian Review Precision Mechanics, Optics and Mechatronics nr. 38/2010 (ISSN 1584-5982, B+ si indexata BDI EBSCO) : “**Joint Motion Area Related To Prosthesis Component Position In Total Hip Arthroplasty**”, autori Adrian Pacioga, Doru Dumitru Palade, Stanca Comşa.
- Buletinul Ştiinţific al UPB, “**Computational Simulation Of Bone - Personalized Hip Prosthesis Assembly**”, autori Adrian Pacioga, Doru Dumitru Palade, Stanca Comşa, lucrare acceptata pentru publicare.

1. INTRODUCERE

1.1. Scopul documentului

Scopul documentului este acela de a descrie proiectarea sistemului de analiza a imaginilor radiografice, de a pune in evidenta detaliile de implementare si tehnicile/algorithmii folositi pentru analiza imaginilor radiografice precum si de a prezenta rezultatele utilizarii sistemului in practica medicala.

1.2. Obiectivul sistemului de analiza si prelucrare a imaginilor radiografice

In prezent, medicii ortopezi din Romania nu dispun de un sistem informatic care sa-i ajute in efectuarea masuratorilor pe radiografiile de sold si femur, in scopul alegerii protezelor, acestea fiind efectuate, de regula, manual, cu rigla. Obiectivul principal al sistemului pe care noi l-am realizat este de a economisi timpul consumat de medici cu masuratorile manuale. Astfel, sistemul calculeaza automat majoritatea parametrilor necesari in alegerea protezelor de sold si da posibilitatea medicului sa interactioneze cu sistemul pentru a masura alti parametri sau a corecta valorile determinate automat.

Un alt obiectiv al sistemului este de a arhiva electronic datele despre pacientii ale caror radiografii au fost analizate. Aceasta va permite medicilor sa urmareasca evolutia in timp a protezarii fiecarui pacient, ajutandu-i in actul medical postoperatoriu. Baza de date care face parte din sistem stocheaza date despre pacienti si radiografiile efectuate inainte si dupa protezare.

1.3. Publicul tinta

Publicul tinta al acestui proiect este reprezentat de specialistii clinici in domeniul ortopediei, in special in Artroplastia de sold. Acestia trebuie sa detina cunostinte minime de utilizare a calculatorului.

1.4. Privire de ansamblu asupra documentului

In capitolul 2 este oferita o privire de ansamblu asupra functionalitatilor sistemului de analiza a radiografiilor si a domeniului sau de aplicabilitate. Capitolul 3 prezinta tehnologiile folosite in implementarea aplicatiei. Capitolul 4 descrie proiectarea de ansamblu a aplicatiei: modulele functionale si diagrama de clase. In capitolul 5 este descrisa baza de date: schema bazei de date si continutul tabelor. Capitolul 6 prezinta detalii de proiectare a aplicatiei: algoritmi selectati pentru a fi implementati in aplicatie, atributele si metodele fiecarei clase. In capitolul 7 se prezinta interfata utilizator, precum si un exemplu de rulare a aplicatiei. In capitolul 8 se prezinta rezultatele utilizarii aplicatiei in practica medicala.

2. PRIVIRE DE ANSAMBLU ASUPRA SISTEMULUI

In acest capitol vom oferi o descriere generala a functionalitatilor aplicatiei de analiza a imaginilor radiografice ortopedice, precum si o prezentare sumara a contextului medical in care aceasta aplicatie va fi folosita.

2.1. Functionalitatile aplicatiei

- Gestiunea pacientilor
 - o Posibilitatea adaugarii de noi pacienti intr-o baza de date, salvand diferite informatii ale acestora
 - o Posibilitatea editarii caracteristicilor pacientilor din baza de date
 - o Posibilitatea stergerii de pacienti din baza de date
- Vizualizarea datelor pacientilor
- Cautarea pacientilor dupa
 - o Nume
 - o Nume doctor
 - o Nume spital
- Gestiunea radiografiilor unui pacient
 - o Posibilitatea de adaugare de noi radiografii (imagini in format DICOM sau BMP)
 - o Posibilitatea stergerii de radiografii din baza de date

- Vizualizarea si analiza imaginilor radiografice
 - o Vizualizarea radiografiilor
 - o Realizarea de masuratori manuale pe imaginile radiografice
 - o Realizarea de masuratori automate pe imaginile radiografice
 - o Salvarea imaginilor radiografice impreuna cu masuratorile manuale si/sau automate realizate pe ele.

2.2. Elemente anatomice de baza in artroplastia soldului

Pentru descrierea modului in care va fi implemetata functionalitatea principala a aplicatiei, aceea de realizare a masuratorilor automate pe imaginile radiografice ortopedice, este nevoie in primul rand de descrierea sumara a elementelor anatomice de baza care se doresc a fi detectate/masurate intr-o radiografie de la nivelul osului, al bazinului, si al soldului.

Articulatia soldului e formata din doua oase: coxalul (care impreuna cu sacrul formeaza bazinul) si femurul (osul coapsei). In figura urmatoare este prezentata o radiografie antero-posterioara de la nivelul soldului, si parametrii urmariti in domeniul Artroplastiei de sold.

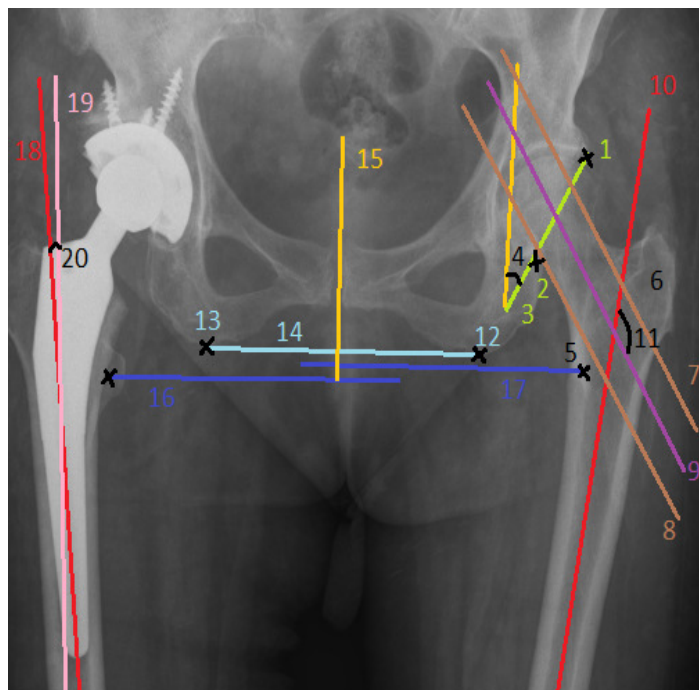


Figura 1. Parametrii important in artroplastie

Parametrii sunt:

- 1) Marginea supero-externa a cotilului
- 2) Marginea infero-interna a cotilului
- 3) Axa capului femural sau axa cotilului
- 4) Unghiul creat de axa cotilului cu axa verticala de referinta sau inclinatia cotilului
- 5) Trohanterul mic
- 6) Trohanterul mare
- 7) Tangenta la corticala superioara a gatului (colului) femural
- 8) Tangenta la corticala inferioara a gatului femural
- 9) Axa colului (gatului) femural (axa cilindrului determinat de tangentele 7 si 8)
- 10) Axa diafizala sau axa corpului femural (axa cilindrului care aproximeaza corpul femural)
- 11) Unghiul cervico-diafizal (unghiul determinat de axa colului si axa diafizala)
- 12) Tuberozitatea ishiadica dreapta (cea mai de jos parte a bazinului)
- 13) Tuberozitatea ishiadica stanga
- 14) Linia bi-tuberozitara (linia determinate de cele doua tuberozitati ishiadice)
- 15) Linia verticala de referinta
- 16) Linia care porneste din centrul trohanterului mic stang, paralela cu linia bi-tuberozitara
- 17) Linia care porneste din centrul trohanterului mic drept, paralela cu linia bi-tuberozitara; distanta dintre liniile 16 si 17 reprezinta distanta verticala intre cele doua oase femurale.

Cel mai important parametru analizat in cadrul masuratorilor automate este unghiul cervico-diafizar. In cazul deformatiilor ce se observa din valoarea acestui parametru (care nu se incadreaza intre 125 si 135 de grade), are loc operatia de artroplastie.

Artroplastia reprezinta o procedura chirurgicala tinand de specialitatea ortopedie si traumatologie prin care fie se inlocuieste o articulatie deteriorata prin artroza sau alt proces patologic cu o proteza, fie se remodeleaza sau realiniaza articulatia printr-o osteotomie sau alta interventie. Inlocuirea cu proteza (numita si endoproteza de sold) presupune inlocuirea atat a acetabulului cat si a capului si colului femural.

Dupa inserarea protezei, trebuie luati in considerare alti parametri:

- 18) Axa diafizala a osului femural
- 19) Axa diafizala a protezei
- 20) Deviatia protezei (unghiul determinat de liniile 18 si 19)

In radiografiile antero-laterale, dupa inserarea protezei, se urmareste determinarea unghiului de ante-versie, care reprezinta unghiul format de axa colului protezei si axa corpului protezei. Acest unghi se determina similar cu unghiul cervico-diafizar, cu singura diferenta ca se alege unghiul ascutit dintre cele doua axe, si nu unghiul obtuz. Daca acest unghi nu are o valoare

aproximativa de 10 grade, proteza se poate luxa. Astfel, se poate observa importanta monitorizarii pacientilor dupa inserarea protezei si a urmaririi evolutiei deviatiei protezei si a unghiului de ante-versie.

Modul in care vor fi determinati parametrii importanti in artroplastie va fi prezentat intr-o sectiune viitoare.

3. TEHNOLOGII FOLOSITE

3.1. Microsoft Visual Studio 2008

Mediul de dezvoltare ales pentru realizarea aplicatiei este IDE-ul de la Microsoft, Visual Studio 2008. Aplicatia este de tip Windows Forms, permitand o interfata prietenoasa cu utilizatorul, prin care se pot realiza toate operatiile de gestiune a pacientilor si de manipulare a imaginilor intr-un mod grafic usor de inteles pentru orice utilizator ce detine cunostinte minime de utilizare a calculatorului.

3.2. C#

Limbajul de programare ales pentru implementarea aplicatiei este C#, deoarece este simplu, modern, orientat pe obiecte.

3.3. Biblioteci grafice

Pentru asistare in procesarea imaginilor s-a ales folosirea A-forge, o biblioteca grafica open source implementata de echipa Google. A-forge permite prelucrarea imaginilor, de exemplu prin aplicarea filtrelor (filtre Gaussiene, filtre Sobel) intr-o maniera rapida.

3.4. MySQL

Pentru crearea si gestiunea bazei de date am ales sistemul de gestiune al bazelor de date MySQL. Acesta ruleaza ca un server ce ofera acces multi-user la un numar de baze de date. Conectarea la baza de date se realizeaza cu un conector numit MySQL ODBC Connector. Astfel, se pot realiza interogari, actualizari si stingeri de elemente din baza de date direct din limbajul C#.

3.5. DICOM - OpenDicom

Pentru gestiunea si interpretarea datelor radiografice intr-o maniera simpla si organizata, este nevoie de un standard. De aceea s-a ales cel mai popular standard pentru imaginile medicale: DICOM (Digital Imaging and Communications in Medicine), care este o specificatie detaliata a codarii si a transferului imaginilor medicale si a informatiilor asociate.

Datele clinice sunt reprezentate intr-o varietate de formate: distantele sunt masurate in milimetri, timpul in secunde, etc. O sectiune a standardului, denumita Data Structure and Their Encoding (Structura datelor si codificarea acestora) defineste 27 de tipuri de date standard, cunoscute ca VR ("value representations" – reprezentari de valori), care includ toate tipurile de date care pot aparea in domeniul medical. Unele dintre cele mai importante date standard sunt: PN (Person Name – Numele persoanei), DT (Data Time – Data si ora), AS (Age String – Varsta)

Extragerea datelor dintr-un fisier DICOM se poate realiza prin folosirea de taguri definite in dictionarul DICOM. Fiecare tag este cautat in fisier, si, in cazul in care este gasit, se interpreteaza.

Unele caracteristici ale fisierelor DICOM, importante in extragerea datelor, sunt:

- Numarul de imagini continute in fisierul DICOM
- Numarul de biti per pixel: 8, 12, 16, 24
- Compresia
- Interpretarea fotometrica: niveluri de gri sau imagini color

Imaginile radiografice (CR – computer radiographic image) stocate in fisierele DICOM sunt insotite de elemente generale de identificare si de cateva informatii specifice.

De exemplu, modulul pacientului contine: numele pacientului, ID-ul acestuia, data nasterii pacientului, etc. Un alt modul, specific pentru radiografii, numit CR Series, contine informatii despre partea corpului examinata in radiografie, pozitia de vizualizare (radiografie antero-laterala, antero-posterioara, etc).

Aplicatia dezvoltata de noi poate interpreta fisiere DICOM (cu extensia .dcm) sau fisiere Bitmap (cu extensia .bmp).

Proiectul openDICOM.NET implementeaza o noua abordare relativa la bibliotecile DICOM. Biblioteca opendicom-sharp, partea principala a proiectului, ofera un API pentru fisierele DICOM in C# pentru Mono si .NET Framework. Intregul continut DICOM poate fi accesat ca o secventa sau un arbore de instante de clase.

4. PROIECTAREA DE ANSAMBLU A APLICATIEI

4.1. Modulele functionale

Modulele principale ale aplicatiei sunt descrise in subsectiunile urmatoare.

4.1.1. Modulul pacienti

Acest modul permite gestiunea pacientilor institutiilor medicale. Acesta este un modul functional care permite:

- Adaugarea de pacienti in baza de date
- Editarea de pacienti
- Stergerea unor pacienti din baza de date
- Cautarea unor pacienti in functie de nume, doctor sau spital

Operatiile se realizeaza prin comunicarea cu baza de date pentru interogari si actiuni de actualizare si stergere.

4.1.2. Modulul radiografii

Acest modul permite gestiunea radiografiilor, prin urmatoarele operatii:

- Asocierea unei radiografii la un anumit pacient
- Inserarea unei noi radiografii pentru un anumit pacient (prin importarea de fisiere DICOM sau BMP)
- Stergerea unei radiografii a unui pacient

Operatiile se realizeaza prin comunicarea cu baza de date pentru interogari si stingeri si prin manipularea de directoare si fisiere. Acest modul este subordonat modulului pacienti, intrucat fiecare radiografie apartine unui pacient; el devine accesibil in interfata utilizator numai dupa ce a fost selectat un pacient pentru procesare.

4.1.3. Modulul de manipulare imagini

Acest modul permite afisarea conturilor in imaginile radiografice, a masuratorilor automate si a celor manuale.

De asemenea, prin intermediul acestui modul se pot salva fisierele reprezentand imaginile radiografice si actiunile executate pe acele imagini, prin alegerea unei latimi si a unei lungimi (in numar de pixeli) pentru fisierul bmp ce urmeaza a fi creat.

Operatiile se realizeaza prin utilizarea modulelor de masuratori automate si manuale, precum si prin manipularea fisierelor.

4.1.4. Modulul de masuratori manuale

Acest modul permite realizarea si gestiunea masuratorilor automate pe imaginea medicala curenta, prin urmatoarele operatii:

- Adaugarea unei linii noi in imagine (salvand pozitiile capetelor si dimensiunea segmentului trasat)
- Adaugarea unui cerc nou in imagine (salvand pozitia centrului si dimensiunea razei)
- Adaugarea unui dreptunghi nou in imagine (salvand pozitiile a doua capete opuse ale dreptunghiului, a lungimii, latimii si diagonalei acestuia)
- Editarea unui element (linie/cerc/dreptunghi) deja creat
- Stergerea unui element (linie/cerc/dreptunghi) deja creat
- Vizualizarea obiectelor (linie/cerc/dreptunghi) ce reprezinta masuratorile manuale

Operatiile se realizeaza prin comunicarea cu baza de date, pentru interogari, actualizari si stingeri si prin operatii pe obiecte din clasa Bitmap prin (crearea unui obiect Bitmap nou sau prin copierea unui alt obiect, schimbarea culorii pixelilor unei imagini reprezentate de un obiect Bitmap, desenarea unei linii, a unui cerc sau a unui dreptunghi intr-o imagine, etc)

4.1.5. Modulul de masuratori automate

Acest modul permite realizarea de masuratori automate si gestionarea parametrilor importanti in artroplastie, descoperiti in radiografia curenta. Daca radiografia nu a fost prelucrata anterior, atunci se va realiza procesul de calculare a parametrilor si salvare a acestora in baza de date. Daca in schimb parametrii au fost determinati anterior, se va realiza procesul de descarcare a informatiilor despre parametrii din baza de date. Modulul permite:

- Vizualizarea parametrilor determinati automat
- Editarea caracteristicilor parametrilor
- Salvarea modificarilor efectuate asupra parametrilor
- Recalcularea anumitor parametri
- Stergerea anumitor parametri

Operatiile se realizeaza prin comunicarea cu baza de date, pentru interogari, actualizari si stingeri si prin operatii pe obiecte din clasa Bitmap (operatiile de la modulul de masuratori automate, precum si operatii mult mai complexe – determinarea conturului unei imagini, extragerea de caracteristici pe imagini etc).

4.2. Diagrama de clase

Figura 14 ilustreaza diagrama clase simplificata a aplicatiei.

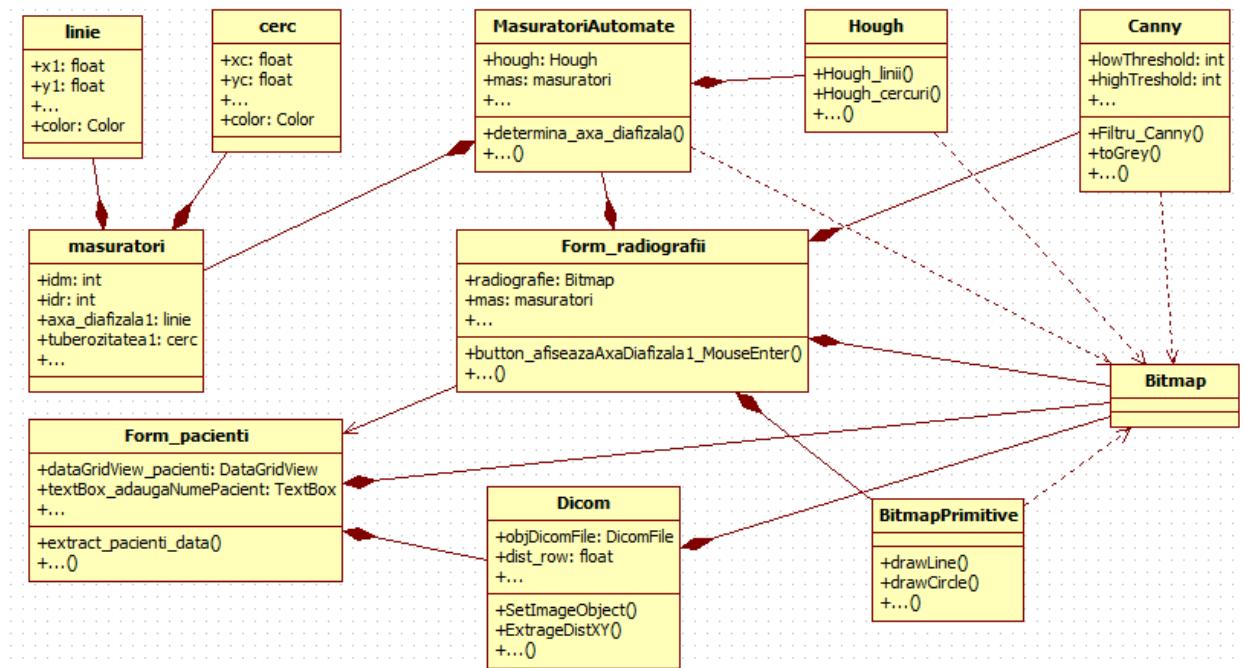


Figura 14. Diagrama de clase simplificata

Clasa *linie* salveaza caracteristicile unui segment de dreapta. Clasa *cerc* pastreaza componentele unui cerc. Clasa *masuratori* este utilizata in pastrarea tuturor parametrilor importanti in artroplastie. Clasa *Form_pacienti* e folosita pentru descrierea interfetei cu utilizatorul in formul principal. Clasa *Dicom* contine metodele pentru manipularea fisierelor dicom. Clasa *BitmapPrimitive* se foloseste pentru operatii pe pixeli in fisierele bitmap (citire valoare intensitate pixel, schimbare valoare intensitate pixel, desenare primitiva (linie,cerc,dreptungi) intr-o culoare data, etc). Clasa *Canny* e utilizata pentru determinarea conturului intr-o imagine prin aplicarea detectorului Canny. Clasa *Hough* contine metodele de aplicare a transformatei Hough pentru determinarea partilor din radiografii care pot fi aproximate prin curbe simple (linii/cercuri). Clasa *MasuratoriAutomate* este utilizata la determinarea tuturor parametrilor importanti in artroplastie, care se pot extrage automat. Clasa *Form_radiografii* este folosita pentru descrierea interfetei cu utilizatorul in formul secundar, dupa ce este selectata o radiografie din formul principal.

5. PROIECTAREA BAZEI DE DATE

5.1. Schema bazei de date

In figura urmatoare sunt ilustrate tabelele bazei de date, impreuna cu relatiile dintre ele.

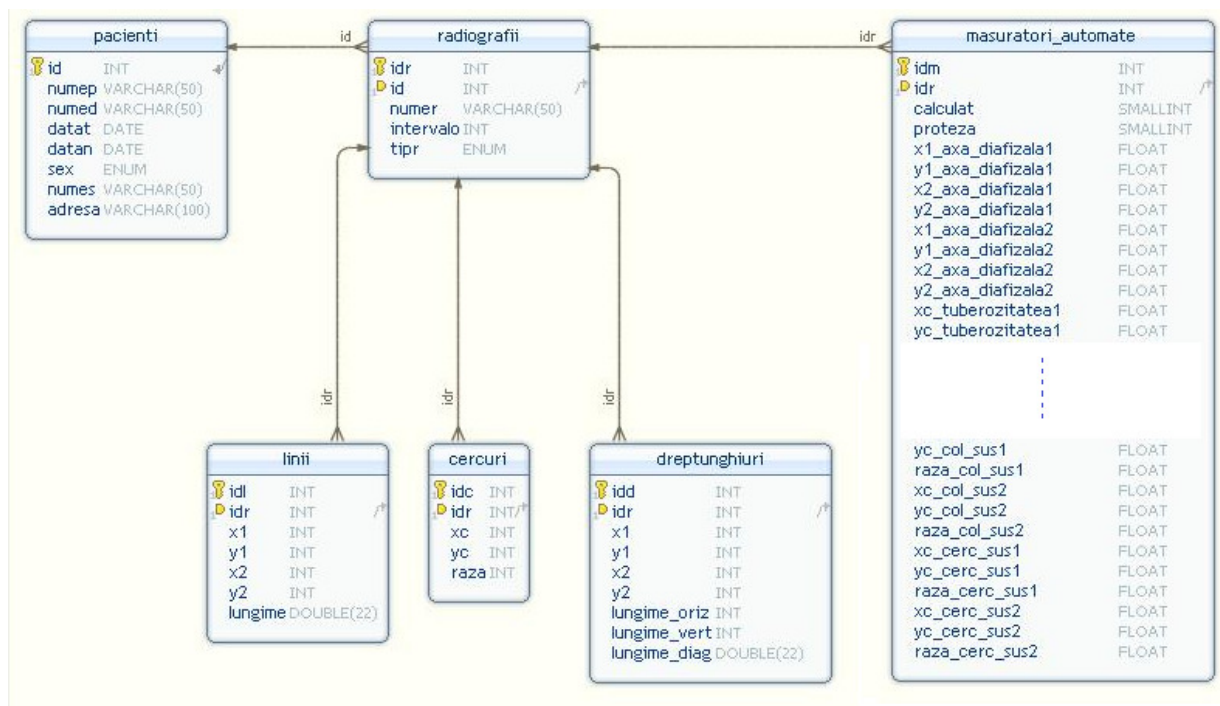


Figura 6. Schema bazei de date

5.2. Descrierea tabelor bazei de date

Tabelul Pacienti

Datele despre pacienti sunt salvate in tabelul pacienti. Informatiile pastrate sunt urmatoarele:

- *id*: numar unic de identificare a pacientilor (cheie primara)
- *numep*: numele pacientului

- *numed*: numele doctorului
- *datat*: data inceperii tratamentului
- *datan*: data nasterii pacientului
- *sex*: sexul pacientului
- *numes*: numele spitalului
- *adresa*: adresa spitalului

Tabelul Radiografii

Informatiile despre radiografii, salvate in tabelul cu acelasi nume, sunt urmatoarele:

- *idr*: numarul unic de identificare al fiecarei radiografii (cheie primara)
- *id*: id-ul pacientului radiografiei respective (cheie externa)
- *numer*: numele radiografiei
- *intervalo*: intervalul (in luni) care a trecut de la operatie
- *tipr*: tipul radiografiei

Tabelul Linii

Modulul de date linii, necesar in modulul de masuratori manuale, se gestioneaza prin tabelul cu acelasi nume. Campurile in tabelul linii sunt urmatoarele:

- *idl*: numarul unic de identificare al liniei (cheie primara)
- *idr*: id-ul radiografiei (cheie externa)
- *x1, y1, x2, y2*: pozitiile capetelor segmentului
- *lungime*: lungimea segmentului (liniei)

Tabelul Cercuri

Similar cu modulul de linii, acest modul este utilizat la masuratorile manuale. Informatiile gestionate prin tabelul cu acelasi nume sunt urmatoarele:

- *idc*: numarul unic de identificare al fiecarui cerc (cheie primara)
- *idr*: id-ul radiografiei (cheie externa)
- *xc, yc*: pozitia centrului cercului
- *raza*: raza cercului

Tabelul Dreptunghiuri

Modulul de date dreptunghiuri, necesar de asemenea la masuratorile manuale, se gestioneaza prin tabelul cu acelasi nume. Campurile in tabelul dreptunghiuri sunt urmatoarele:

- *idd*: identificatorul unic al fiecarui dreptunghi (cheie primara)
- *idr*: id-ul radiografiei (cheie externa)
- *x1, y1, x2, y2*: pozitiile a doua capete opuse ale dreptunghiului
- *lungime_oriz*: dimensiunea unei laturi orizontale a dreptunghiului
- *lungime_vert*: dimensiunea unei laturi verticale a dreptunghiului
- *lungime_diag*: dimensiunea diagonalei dreptunghiului

Tabelul Masuratori automate

Acest modul se foloseste in modulul de masuratori automate, si se gestioneaza prin tabelul cu acelasi nume. Campurile in tabelul *masuratori_automate* sunt urmatoarele:

- *idm*: identificatorul unic al masuratorii automate realizate (cheie primara)
- *idr*: id-ul radiografiei (cheie externa)
- *proteza*: parametru care pastreaza informatii despre protezele din radiografie (este 0 daca radiografia nu contine nici o proteza, 1, daca exista proteza la osul stang, 2, daca exista proteza la osul drept si 3, daca exista proteze la ambele oase)
- *x1_axa_diafizala1, y1_axa_diafizala1, x2_axa_diafizala1, y2_axa_diafizala1*: pozitiile capetelor segmentului ce reprezinta axa diafizala a osului femural stang
- *x1_axa_diafizala2, y1_axa_diafizala2, x2_axa_diafizala2, y2_axa_diafizala2*: pozitiile capetelor segmentului ce reprezinta axa diafizala a osului femural drept
- *xc_tuberozitatea1, yc_tuberozitatea1*: pozitia centrului cercului care reprezinta tuberozitatea ishiadica din partea stanga
- *raza_tuberozitatea1* – raza cercului care reprezinta tuberozitatea ishiadica din partea stanga
- *xc_tuberozitatea2, yc_tuberozitatea2* – pozitia centrului cercului care reprezinta tuberozitatea ishiadica din partea dreapta
- *raza_tuberozitatea2* – raza cercului care reprezinta tuberozitatea ishiadica din partea dreapta
- *xc_capul_femural1, yc_capul_femural1* – pozitia centrului cercului care reprezinta capul femural al osului stang
- *raza_capul_femural1* – raza cercului care reprezinta capul femural al osului stang
- *xc_capul_femural2, yc_capul_femural2* – pozitia centrului cercului care reprezinta capul femural al osului drept
- *raza_capul_femural2* – raza cercului care reprezinta capul femural al osului drept
- *x1_linia_bituberozitara, y1_linia_bituberozitara, x2_linia_bituberozitara, y2_linia_bituberozitara* – pozitiile capetelor segmentului care determina linia bituberozitara

- $xc_trohanterul_mic1, yc_trohanterul_mic1$ – pozitia centrului cercului care reprezinta trohanterul mic al osului stang
- $raza_trohanterul_mic1$ – raza cercului care reprezinta trohanterul mic al osului stang
- $xc_trohanterul_mic2, yc_trohanterul_mic2$ – pozitia centrului cercului care reprezinta trohanterul mic al osului drept
- $raza_trohanterul_mic2$ – raza cercului care reprezinta trohanterul mic al osului drept
- $x1_linia_verticala, y1_linia_verticala, x2_linia_verticala, y2_linia_verticala$ – pozitiile capetelor segmentului care reprezinta linia verticala de referinta
- $diferenta_intre_oase$ – reprezinta distanta pe verticala intre oase
- $x1_axa_protezei1, y1_axa_protezei1, x2_axa_protezei1, y2_axa_protezei1$ – pozitiile capetelor segmentului care determina axa protezei osului stang (daca proteza exista)
- $x1_axa_protezei2, y1_axa_protezei2, x2_axa_protezei2, y2_axa_protezei2$ – pozitiile capetelor segmentului care determina axa protezei osului drept (daca proteza exista)
- $deviatia_protezei1$ – reprezinta deviatia protezei osului stang (daca proteza exista)
- $deviatia_protezei2$ – reprezinta deviatia protezei osului drept (daca proteza exista)
- $x1_axa_capului_femural1, y1_axa_capului_femural1, x2_axa_capului_femural1, y2_axa_capului_femural1$ – pozitiile capetelor segmentului care determina axa capului femural al osului stang
- $x1_axa_capului_femural2, y1_axa_capului_femural2, x2_axa_capului_femural2, y2_axa_capului_femural2$ – pozitiile capetelor segmentului care determina axa capului femural al osului drept
- $unghiul_acetabular1$ – reprezinta unghiul format de axa capului femural al osului stang cu linia verticala de referinta
- $unghiul_acetabular2$ – reprezinta unghiul format de axa capului femural al osului stang cu linia verticala de referinta
- $x1_axa_colului_femural1, y1_axa_colului_femural1, x2_axa_colului_femural1, y2_axa_colului_femural1$ – pozitiile capetelor segmentului care determina axa colului femural al osului stang
- $x1_axa_colului_femural2, y1_axa_colului_femural2, x2_axa_colului_femural2, y2_axa_colului_femural2$ – pozitiile capetelor segmentului care determina axa colului femural al osului drept
- $unghiul_cervico_diafizar1$ – reprezinta unghiul format de axa diafizei cu axa colului osului femural stang
- $unghiul_cervico_diafizar2$ – reprezinta unghiul format de axa diafizei cu axa colului osului femural drept
- $latimea_osului1$ – latimea osului stang la nivelul corpului sau
- $latimea_osului2$ – latimea osului drept la nivelul corpului sau

- xc_col_jos1, yc_col_jos1 – pozitia centrului arcului de cerc care se mapeaza pe conturul de jos al colului femural stang
- $raza_col_jos1$ – raza arcului de cerc care se mapeaza pe conturul de jos al femurului stang
- xc_col_jos2, yc_col_jos2 – pozitia centrului arcului de cerc care se mapeaza pe conturul de jos al colului femural drept
- $raza_col_jos2$ – raza arcului de cerc care se mapeaza pe conturul de jos al femurului drept
- xc_col_sus1, yc_col_sus1 – pozitia centrului arcului de cerc care se mapeaza pe conturul de sus al colului femural stang
- $raza_col_sus1$ – raza arcului de cerc care se mapeaza pe conturul de sus al femurului stang
- xc_col_sus2, yc_col_sus2 – pozitia centrului arcului de cerc care se mapeaza pe conturul de sus al colului femural drept
- $raza_col_sus2$ – raza arcului de cerc care se mapeaza pe conturul de sus al femurului drept
- $xc_cerc_sus1, yc_cerc_sus1$ – pozitia centrului arcului de cerc care se mapeaza pe conturul bazinului de deasupra capului femural al osului stang
- $raza_cerc_sus1$ – raza arcului de cerc care se mapeaza pe conturul bazinului de deasupra capului femural al osului stang
- $xc_cerc_sus2, yc_cerc_sus2$ – pozitia centrului arcului de cerc care se mapeaza pe conturul bazinului de deasupra capului femural al osului drept
- $raza_cerc_sus2$ – raza arcului de cerc care se mapeaza pe conturul bazinului de deasupra capului femural al osului drept

6. PROIECTAREA DE DETALIU

6.1. Algoritmi folositi

In aceasta sectiune vor fi detaliate algoritmi folositi pentru extragerea parametrilor din radiografia, detectorul Canny si detectorul Hough pentru linii si cercuri. Detectorul Canny este utilizat in extragerea conturului oaselor dintr-o radiografie. S-a observat ca o multime de parti ale oaselor din radiografia pot fi comparate cu linii, cercuri sau parti din cercuri. Din acest motiv s-a ales implementarea algoritmului Hough de detectie a liniilor si conicelor.

Detectorul Canny

Operatorul Canny de extragere a frontierelor a fost dezvoltat de John F. Canny în 1986 și folosește un sir de algoritmi pentru a detecta contururile în imagini complexe. Printre elementele pe care Canny și le-a propus în realizarea acestui algoritm sunt:

- scăderea ratei de eroare (detectorul să nu răspundă la puncte “false” și să nu piardă puncte de pe frontieră)
- punctele de pe frontieră să fie bine localizate
- un singur răspuns la un singur punct de pe frontieră.

În vederea împlinirii criteriilor anterioare, detectorul Canny efectuează în primul pas o netezire a imaginii în vederea eliminării zgomotului. În următorul pas este calculat Gradientul imaginii pentru a pune în evidență zonele cu variație bruscă a intensității. În continuare, algoritmul parcurge aceste zone și elimină orice pixel care nu este un maxim local pe direcția gradientului. Matricea gradientului este mai departe redusă prin histerezis.

În următoarele paragrafe vor fi descriși în amănunt pașii algoritmului.

Pasul 1: Se aplică imaginii de intrare un filtru Gaussian (de blur) pentru netezirea acesteia. Cu cât lățimea măștii de convoluție este mai mare, cu atât sensibilitatea la zgomot este mai mică. În același timp, eroarea de localizare a fronturilor crește. Mască de filtrare folosită este următoarea:

$$1/115 * \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}$$

În urma aplicării filtrului, imaginea va avea un aspect mai neted, cu mai puține zgomote. Această metodă este una eficientă în eliminarea pixelilor de front care nu sunt și pixeli de frontieră.

Pasul 2: Se aplică operatorul Sobel imaginii rezultate din pasul 1, obținându-se matricea amplitudinilor gradientului.

Pasul 3: Se calculează direcția gradientului în fiecare punct, obținându-se matricea direcțiilor:

$$\theta(x,y) = \arctg(Dy / Dx)$$

Pasul 4: Se ajusteaza θ la una dintre directiile din spatiul discret al imaginii, si anume cea mai apropiata de valoarea lui θ . In Figura 2 sunt evidentiate cele patru directii posibile in spatiul imaginii, din pixelul p:

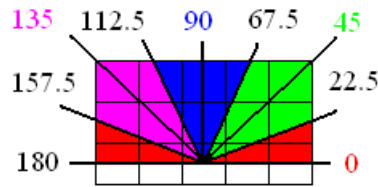


Figura 2. Directia gradientului

Directia asignata gradientului in pixelul p se calculeaza astfel:

Directia gradientului = 0 grade , daca $0 \leq \theta < 22.5$ sau $157.5 \leq \theta \leq 180$

45 grade, daca $22.5 \leq \theta < 67.5$

90 grade, daca $67.5 \leq \theta < 112.5$

135 grade, daca $112.5 \leq \theta < 157.5$

Pasul 5: Pasul 5 poarta denumirea de non-maxima suppression. Detectorul Sobel, ca toti detectorii bazati pe masti de convolutie, produce mai multe puncte de front pentru un acelasi punct de frontiera (mai multe masti pot contine un punct de frontiera). De aceea, matricea amplitudinilor poate contine zone late in jurul frontierei. In acest pas sunt eliminati pixelii care nu au amplitudinea maxima locala (in vecinatatea unui pixel). Sunt declarate puncte de frontiera acele puncte a caror amplitudine este maxima locala pe directia gradientului. Celelalte puncte sunt eliminate (setate la zero).

Pentru fiecare pixel C se iau in considerare pixelii A si B din vecinatatea lui C pe directia gradientului din C.

Daca $D(A) > D(C)$ sau $D(B) > D(C)$, atunci $D(C) = 0$.

Efectul acestui pas este de subtiere a frontierei fara a o intrerupe.

Pasul 6: Ultimul pas poarta denumirea de hysteresis thresholding. In urma aplicarii primilor cinci pasi, imaginea rezultata inca mai are zgomote, care se manifesta ca pixeli de intensitate mica. In acest pas se elimina pixelii falsi. Pentru procedeele de eliminare a zgomotelor se utilizeaza operatia de histerezis. Operatia de histerezis consta in aplicarea a doua praguri, un prag de jos P1

si un prag de sus, P2. Aceste praguri se aplica separat imaginii rezultate in pasul anterior, I5, rezultand 2 imagini binare, T1 si T2. In T1 au valoarea 1 pixelii cu amplitudinea>T1 iar in T2 aceia cu amplitudinea>T2. Imaginea din T2 are intreruperi in frontiera dar contine mai putine puncte false. Punctele din T2 sunt legate in contururi. Urmarirea unei frontiere incepe cu un punct din T2. Se conecteaza puncte din T2 pana cand se ajunge intr-un pixel p care nu mai poate fi conectat cu alt punct din T2 (nu are vecini in T2 pe nici una dintre cele 4 directii). In acest moment, se cauta in T1, in vecinatatea de 8 pixeli a pixelului p, un pixel care poate fi conectat la contur. Se conecteaza la frontiera puncte din T1 pana cand se ajunge la un pixel cu valoare diferita de zero in T2 (sau se ajunge la sfarsit de contur). In acest fel, se completeaza intreruperile de frontiera din T2 cu pixeli din T1. Metoda histerezis rezolva problemele care apar la utilizarea unui singur prag.

In figura 3 se poate observa imaginea initiala reprezentand o radiografie de la nivelul soldului si imaginea finala dupa aplicarea detectorului Canny (se poate observa conturul oaselor).

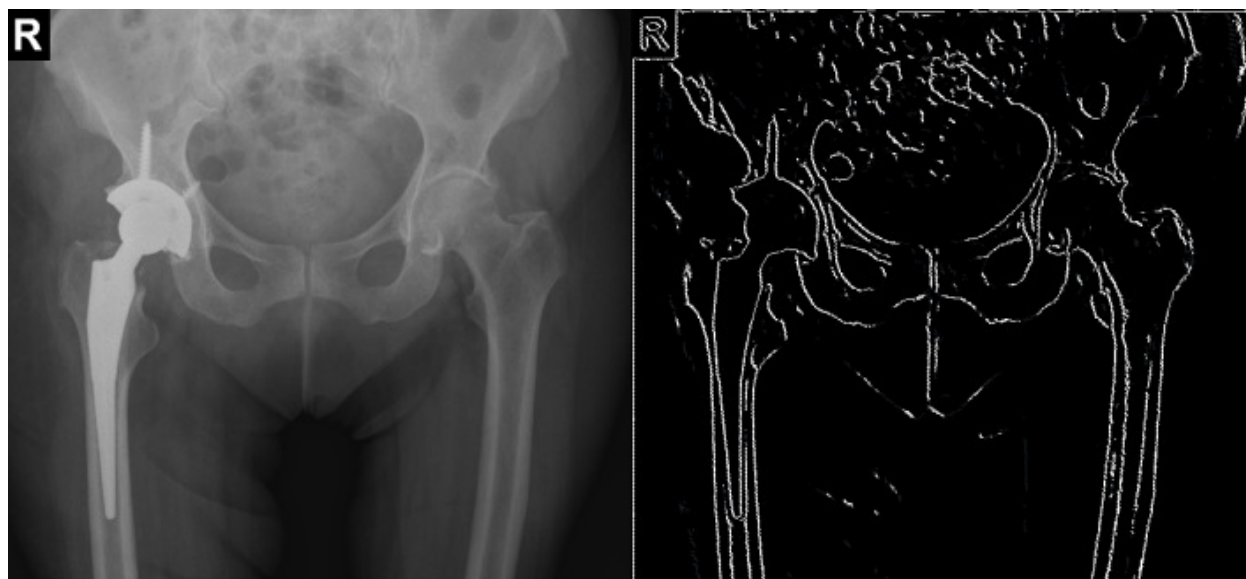


Figura 3. Imaginea radiografica inainte si dupa aplicarea detectorului Canny

Dupa cum se poate observa si in figura anterioara, conturul nu este detectat cu precizie de 100%. Din acest motiv, pentru detectarea parametrilor doriti, se foloseste observatia ca foarte multe parti osoase se pot asemana cu linii sau cercuri. Astfel, se propune algoritmul Hough de extragere a curbilor simple din imagini pentru pozitionarea unor parametrii in radiografie. Implementarile algoritmului vor fi adaptate in functie de parametrii cautati.

6.1.1. Transformata Hough

Transformata Hough este o tehnica de extragere a caracteristicilor folosita in analiza si prelucrarea imaginilor. Scopul acestei tehnici este de a gasi instante imperfecte de obiecte care se incadreaza intr-o anumita clasa de forme printr-o procedura de votare. Aceasta procedura de votare se realizeaza intr-un spatiu parametric, din care obiectele candidat sunt obtinute ca maxime locale intr-un asa-numit spatiu accumulator care este construit explicit de catre algoritmul pentru determinarea transformatei Hough.

Transformata Hough clasica este dedicata identificarii liniilor in imagini, dar mai tarziu transformata Hough s-a extins pentru identificarea de conice, in special cercuri si elipse. Transformata Hough folosita astazi a fost inventata de Richard Duda si Peter Hart in 1972, care au numit-o "Transformata Hough generalizata" dupa Paul Hough.

In analiza automata a imaginilor digitale, problema detectarii formelor simple, cum ar fi liniile drepte, cercurile sau elipsele apare destul de des. In multe cazuri un detector de contur poate fi folosit intr-o faza de pre-procesare pentru a obtine puncte care se afla pe curba dorita in spatiul imaginii. Din cauza imperfectiunilor in imagine sau in detectorul de contur, totusi, pot exista puncte lipsa pe curbele respective, precum si deviatii intre linia/cercul/elipsa ideala si punctele de frontiera obtinute din detectorul de contur. Din aceste motive, este complicata gruparea caracteristicilor de contur extrase in seturi adecvate de linii, cercuri sau elipse. Scopul transformatei Hough este acela de a rezolva aceasta problema prin gruparea punctelor de frontiera printr-o procedura de votare explicita peste un set de obiecte imagine parametrizate.

6.1.2. Transformata Hough pentru linii

Cel mai simplu caz al transformatei Hough il reprezinta transformata liniara pentru detectarea liniilor drepte. In spatiul imagine, liniile drepte pot fi descrise ca $y=mx+b$ si pot fi reprezentate grafic pentru fiecare pereche de puncte imagine (x,y) . In transformata Hough, ideea principala este aceea de a considera caracteristicile liniei nu ca puncte x,y , ci din punctul de vedere al parametrilor sai, in cazul de fata parametrul m si b . Astfel, linia dreapta $y=mx+b$ poate fi reprezentata ca un punct (b,m) in spatiul parametric. Totusi, in cadrul liniilor vertical apar probleme la folosirea parametrilor b si m . Astfel, din motive computationale, este mai adecvata folosirea unui alt set de parametric, numiti r si θ .

Parametrul r reprezinta distanta dintre linie si origine, iar θ este unghiul vectorului de la origine pana la cel mai apropiat punct de pe linie. Folosind aceasta parametrizare, ecuatia liniei poate fi scrisa in felul urmator:

$$y = \left(-\frac{\cos \theta}{\sin \theta} \right) x + \left(\frac{r}{\sin \theta} \right)$$

care poate fi rearanjata astfel: $r = x\cos\theta + y\sin\theta$.

Astfel se poate asocia fiecare linie cu perechea (r, θ) care este unica daca $\theta \in [0, \pi)$ si $r \in \mathbf{R}$ sau daca $\theta \in [0, 2\pi)$ si $r \geq 0$. Planul (r, θ) este adesea numit spatiul Hough pentru setul de linii dreapta in doua dimensiuni. Figura 4 explica modul de determinare a parametrilor r si θ in functie de linie si de origine.

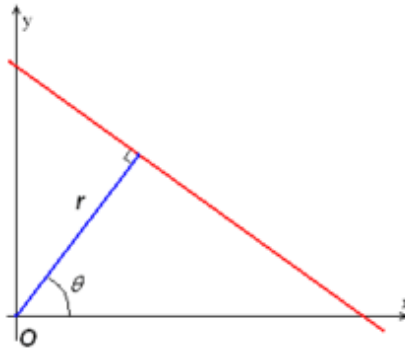


Figura 4. Parametrii r si θ

Transformata Hough foloseste o matrice, numita acumulator, pentru detectarea existentei liniilor $y=mx+b$. Dimensiunea acumulatorului este egala cu numarul de parametric necunoscuti ai problemei transformate Hough. De exemplu, problema Hough liniara are doi parametri necunoscuti: m si b . Cele doua dimensiuni ale acumulatorului corespund valorilor cuantizate pentru m si b . Pentru fiecare pixel si vecinatatea sa, transformata Hough determina daca exista destule dovezi ale existentei unei linii in acel pixel. Daca da, va calcula parametrul acelei linii si va cauta celula acumulatorului pentru r si θ gasiti, incrementand valoarea acelei celule. Gasind celulele cu cele mai mari valori, cele mai probabile linii vor fi extrase. Cea mai simpla modalitate de determinare a celor mai mari valori in acumulator este prin aplicarea de diferite praguri.

6.1.3. Transformata Hough pentru cercuri

Transformata Hough poate fi folosita la determinarea paramerilor unui cerc atunci cand exista un numar de puncte care se afla in zona acelu cerc. Un cerc de raza R cu centrul (a,b) poate fi descris cu urmatoarele ecuatii parametrice:

$$x = a + R \cos(\theta)$$

$$y = b + R \sin(\theta)$$

Cand unghiul θ ia valori intre 0 si 360 de grade, punctele (x,y) urmaresc perimetrul unui cerc. Daca o imagine contine multe puncte, dintre care unele se gasesc in perimetrele unor cercuri, atunci problema este gasirea tripletelor (a,b,R) pentru a descrie fiecare cerc. Una din cele

mai mari probleme este aceea a spatiului 3D, care determina o complexitate computationala foarte mare. Din acest motiv este mai bine daca raza R este cunoscuta, pentru a reduce cautarea intr-un spatiu 2D. In cazul nostru razele cercurilor cautate nu sunt constante, dar se incadreaza in niste limite destul de stricte.

6.1.4. Adaptarea Transformatei Hough pentru extragerea parametrilor din radiografii

In aceasta sectiune vor fi descrise mici modificari pe care noi le-am adus la transformata Hough pentru extragerea unor parametrii importanti in artroplastie.

1. Detectarea axei diafizale

Axa diafizala este axa cilindrului determinat de conturul corpului femural. S-a observat ca liniile conturului sunt drepte, si deci pot fi aproximate cu niste linii drepte. Singura modificare adusa la algoritmul Hough initial pentru detectarea liniilor drepte este limita de cautare in spatiul (r, θ) si in spatiul pixelilor (x, y) . Pentru detectarea conturului corpului femural, se iau in considerare numai pixelii imaginii cu $y < \text{height}/2$ si cu θ in anumite limite incat sa descrie linii aproape verticale.

Dupa detectia liniilor drepte din imagine, se aleg acele linii care reprezinta conturul. Se imparte imaginea in doua (partea stanga si partea dreapta). In fiecare jumatate se aleg liniile extreme (cea mai din stanga si cea mai din dreapta), considerand ca acestea reprezinta liniile conturului. Apoi se determina axa cilindrului determinat de cele doua linii. Aceasta axa este axa diafizala.

2. Detectarea tuberozitatii ishiadice

Algoritmul Hough pentru cercuri este modificat in felul urmatoare:

- Centrele cercurilor vor fi cautate numai in zona inadrata de cele doua corpuri femurale anterior determinate, de jos in sus, algoritmul terminandu-se cand s-a gasit primul cerc. Aceasta metoda duce la rezultate foarte rapide datorita numarului mic de pixeli albi localizati intre corpurile celor doua oase femurale si linia ishiadica, care trebuie inspectati pentru a determina daca formeaza un cerc care corespunde urmatoarelor doua reguli
- Raza cercului nu este constanta, dar este continuta intre $R - \Delta R$ si $R + \Delta R$, unde $\Delta R \ll R$
- Algoritmul ia in considerare numai acei pixeli ai cercului pozitionati in partea de jos a cercului (care determina, impreuna cu centrul cercului, un unghi cu $\sin > 0.5$)

Ceilalti parametrii care aproximeaza arcuri de cerc sunt urmatoarii:

- **Capul femural**

- **Trohanterul mic**
- **Conturul de jos al colului**
- **Conturul de sus al colului**
- **Partea bazinului care se afla deasupra capului femural**

Acesti parametri se determina asemanator cu tuberozitatile ishiadice, cu modificarile de rigoare (in functie de pozitia fiecaruia, de partea de cerc pe care o descrie, etc).

In figura 5 sunt descriși toti parametrii determinati cu transformata Hough modificata:

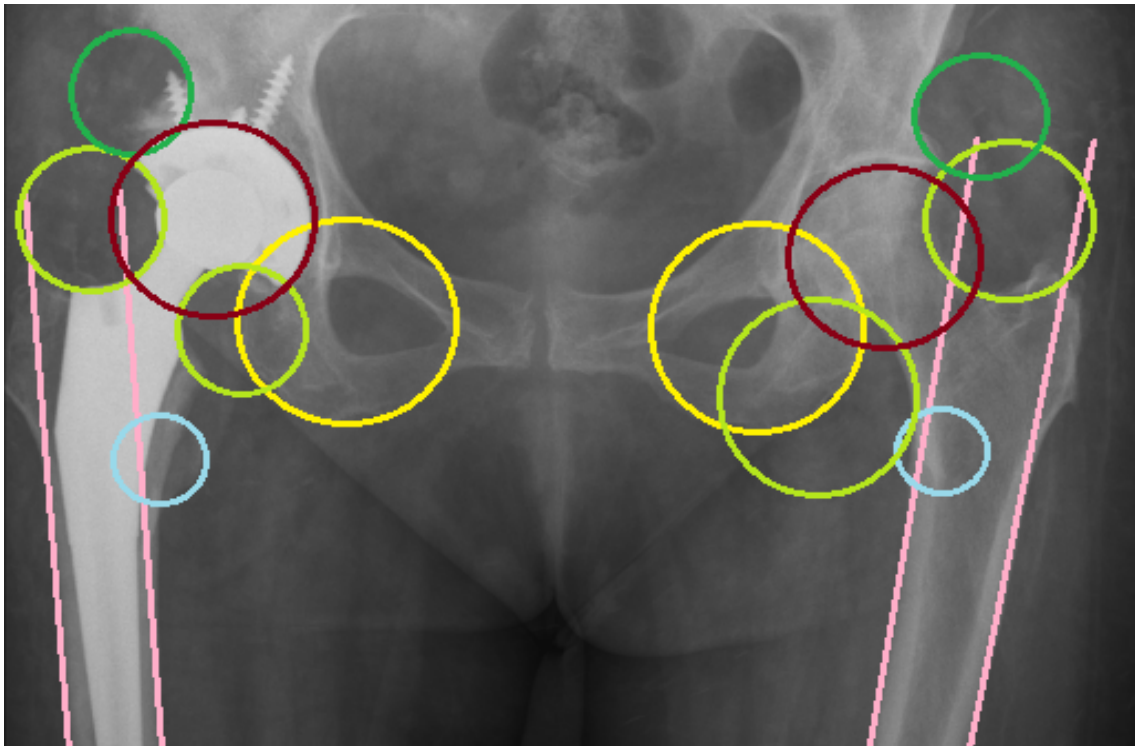


Figura 5. Parti din radiografie care pot fi aproximate prin linii/cercuri

6.2. Descrierea detaliata a claselor

In acest capitol vor fi prezentate clasele folosite, impreuna cu attributele si metodele lor, specificand pentru fiecare clasa modulul in care s-a folosit si tabelele din baza de date la care se relateaza.

6.2.1. Clasele linie, cerc, masuratori

Pentru realizarea masuratorilor automate se folosesc cateva clase care continu numai attribute (tipul struct din C#):

- Clasa **linie**, care salveaza caracteristicile unui segment de dreapta, cu urmatoarele componente:
 - o $x1, y1, x2, y2$ de tip *float* care pastreaza pozitiile capetelor segmentului de dreapta
 - o *afiseaza* de tip *bool* care precizeaza daca segmentul de dreapta este sau nu desenat in imagine
 - o *editeaza* de tip *bool* care precizeaza daca linia este in proces de editare (capetele $(x1,y1)$ si $(x2,y2)$ urmeaza sa sufere modificari)
 - o *modifica_linie1* si *modifica_linie2* de tip *bool* care precizeaza care capat al segmentului $(x1,y1)$ sau $(x2,y2)$ urmeaza a fi modificat
 - o *color* de tip *Color* care pastreaza culoarea in care va fi desenata linia in imagine
- Clasa **cerc**, cu urmatoarele componente:
 - o xc, yc, r de tip *float*, care pastreaza pozitia centrului (xc,yc) si raza r a cercului
 - o *afiseaza* de tip *bool* care precizeaza daca obiectul cerc se va afisa sau nu in imagine
 - o *editeaza* de tip *bool* care precizeaza daca obiectul cerc urmeaza a fi editat
 - o *modifica_cerc_centru*, *modifica_cerc_raza* de tip *bool* care precizeaza ce componenta a cercului urmeaza a fi editata (pozitia centrului (xc, yc) respectiv raza r a acestuia)
 - o *color* de tip *Color* care reprezinta culoarea in care va fi desenat cercul in imagine
- Clasa **masuratori**, utilizata pentru pastrarea tuturor parametrilor importanti in artroplastie, determinati automat prin analiza si procesarea imaginilor radiografice. Aceasta structura contine urmatoarele componente, avand fiecare cate un corespondent in tabelul *masuratori_automate* din baza de date
 - o *idm* de tip *int*, care reprezinta id-ul masuratorii automate
 - o *idr* de tip *int*, care reprezinta id-ul radiografiei
 - o *calculat* de tip *int*, care precizeaza daca au fost realizate masuratorile automate pe radiografia respectiva
 - o *axa_diafizala1* si *axa_diafizala2* de tip *linie*, care salveaza informatiile despre axele diafizale ale celor doua oase femurale
 - o *tuberozitate1* si *tuberozitate2* de tip *cerc*, care salveaza informatiile despre cercurile ale caror arcuri de cerc se mapeaza pe cele doua tuberozitati ishiadice
 - o *cap_femural1* si *cap_femural2* de tip *cerc*, care salveaza informatiile despre cercurile ale carori arcuri de cerc descriu cele doua capete femurale
 - o *trochanter_mic1* si *trochanter_mic2* de tip *cerc*, care salveaza informatiile despre cercurile ale caror arcuri de cerc se mapeaza pe cele doua trohantere ale oaselor femurale

- *linia_bituberozitara* de tip *linie*, care salveaza informatii despre linia bituberozitara
- *linia_verticala* de tip *linie*, care salveaza informatii despre linia verticala de referinta
- *diferenta_oase* de tip *float*, care salveaza distanta pe verticala intre cele doua trohantere mici, numita si distanta pe verticala a oaselor femurale
- *proteza* de tip *int* care precizeaza care dintre cele doua oase femurale contine proteza (0 – nici unul, 1 – osul stang, 2 – osul drept, 3 – ambele oase)
- *axa_proteza1* si *axa_proteza2* de tip *linie*, care salveaza componentele axelor diafizale ale protezelor (in cazul in care acestea exista)
- *col_sus1* si *col_sus2* de tip *cerc*, care salveaza informatiile despre cercurile ale caror arcuri de cerc se mapeaza pe contururile de sus ale gaturilor celor doua oase femurale
- *col_jos1* si *col_jos2* de tip *cerc*, care salveaza informatiile despre cercurile ale caror arcuri de cerc se mapeaza pe contururile de jos ale gaturilor celor doua oase femurale
- *cerc_sus1* si *cerc_sus2* de tip *cerc*, care salveaza informatiile despre cercurile ale caror arcuri de cerc se mapeaza pe contururile bazinului care se afla deasupra capetelor femurale
- *axa_cap_femural1* si *axa_cap_femural2* de tip *linie*, care salveaza segmentele de dreapta care se confunda cu cele doua axe ale capetelor femurale
- *axa_colului1* si *axa_colului2* de tip *linie*, care salveaza informatiile despre segmentele de dreapta care se confunda cu cele doua axe ale gaturilor oaselor femurale
- *unghi_cervico_diafizar1* si *unghi_cervico_diafiza2* de tip *float* care pastreaza informatii despre cele doua unghiuri cervico-diafizare
- *deviatie_proteza1* si *deviatie_proteza2* de tip *float* care salveaza informatiile despre unghiurile de deviatie ale protezelor
- *unghi_acetabular1* si *unghi_acetabular2* de tip *float* care salveaza valorile unghiurilor de inclinatia ale axelor capetelor femurale
- *bone_width1* si *bone_width2* de tip *float*, care salveaza valorile latimilor oaselor femurale la nivelul corpurilor acestora

6.2.2. Clasa Form_pacienti

Aceasta clasa extinde clasa Form din C# si este folosita pentru descrierea interfetei cu utilizatorul in formul principal, acela unde:

- Se vizualizeaza tabelul cu pacienti

- Se realizeaza operatii pe pacienti (adaugare, editare, stergere)
- Se cauta pacienti in baza de date
- Se vizualizeaza tabelul cu radiografii
- Se realizeaza operatii pe radiografiile (adaugare, stergere)
- Se previzualizeaza imaginile radiografice

Clasa `Form_pacienti` permite tratarea evenimentelor de interfatare cu utilizatorul in form-
ul principal al aplicatiei.

Aceasta contine urmatoarele atribute:

- Obiecte vizuale:
 - o `dataGridView_pacienti` si `dataGridView_radiografii` – obiecte de tip `DataGridView` care permit afisarea informatiilor despre pacienti/radiografiile intr-o forma tabelara
 - o obiecte de tip `TextBox`:
 - `textBox_adaugaNumePacient`, `textBox_cautaNumePacient`: pentru afisarea si inserarea
- alte variabile ajutatoare

Aceasta contine urmatoarele metode principale:

- `void extract_pacienti_data()`
 - o extrage toti pacientii din tabelul pacienti din baza de date
 - o salveaza informatiile despre pacienti in `dataGridView_pacienti`, care permite vizualizarea acestora in forma tabelara
- `void extract_pacienti_data_id(List <int> vect_iduri)`
 - o extrage pacientii din tabelul pacienti din baza de date care au id-urile in lista de id-
uri, `vect_iduri`, furnizata ca parametru
 - o aceasta metoda este folosita in modulul de cautare al unor pacienti dupa mai
multe criterii (cautare dupa nume si/sau doctor si/sau nume spital)
- `void modifica_delete_pacienti_tab()`
 - o se apeleaza atunci cand se selecteaza un pacient din gridul de vizualizare (din
obiectul `dataGridView_pacienti`) si apoi se selecteaza tab-ul de stergere pacienti
 - o metoda permite modificarea elementelor din tab-ul de stergere pacienti
 - salveaza in componenta `Text` a obiectului grafic `textBox_stergeID` id-ul
pacientului selectat (pentru stergere)
 - salveaza in componenta `Text` a obiectului `textBox_stergeNumePacient`
numele pacientului selectat (pentru stergere)

- **void** modifica_edit_pacienti_tab()
 - se apeleaza atunci cand se selecteaza un pacient din grid-ul de vizualizare pacienti si cand este selectat tab-ul de editare pacienti
 - similar ca la metoda *modifica_delete_pacienti_tab()*, salveaza in componentele *Text* ale obiectelor grafice de tip *TextBox* urmatoarele informatii despre pacient: numele pacientului, numele doctorului, numele spitalului, adresa spitalului
 - metoda modifica si alte obiecte grafice din tab-ul de editare:
 - componenta *SelectedIndex* a obiectului *comboBox_editeazaSex* (aceasta are valoarea 0 daca sexul este feminin si valoarea 1 in caz contrar)
 - componentele *SelectionEnd* si *SelectionStart* ale obiectelor *monthCalendar_editeazaDataTratament* si *monthCalendar_editeazaDataNasterii* pentru a putea selecta data nasterii pacientului si data inceperii tratamentului
- **void** button_adaugaPacient_Click(**object** sender, **EventArgs** e)
 - aceasta metoda reprezinta se foloseste la tratarea unui eveniment declansat de utilizator (la apasarea butonului “*Adauga*” din tab-ul de adaugare pacienti)
 - se incearca salvarea informatiilor completate in *TextBox*-urile pentru nume pacient, nume doctor, nume spital, adresa spital, in *ComboBox*-ul pentru sexul pacientului si in *MonthCalendar*-urile pentru data nasterii si data inceperii tratamentului, printr-o actiune de adaugare de componente noi la tabelul pacienti din baza de date
- **void** dataGridView_pacienti_CellClick(**object** sender, **DataGridViewCellEventArgs** e)
 - aceasta metoda se apeleaza la aparitia evenimentului de selectare a unei celule din grid-ul de vizualizare al pacientilor (eveniment declansat de utilizator)
 - la selectarea oricarei celule reprezentand o caracteristica a unui pacient, se considera ca pacientul respectiv este selectat (pentru prelucrari ulterioare)
- **void** button_cauta_Click(**object** sender, **EventArgs** e)
 - aceasta metoda se utilizeaza la tratarea evenimentului de apasare a butonului “*Cauta*”
 - In functie de informatiile din componentele *Text* ale obiectelor *textBox_cautaNumeSpital*, *textBox_cautaNumeDoctor* si *textBox_cautaNumePacient*, se realizeaza operatia de interogare a tabelului pacienti din baza de date pentru a intoarce id-urilor pacientilor care corespund criteriilor de cautare (in vederea afisarii lor ulterioare in gridul de vizualizare pacienti)
- **void** button_afiseaza_toti_pacientii_Click(**object** sender, **EventArgs** e)

- aceasta metoda trateaza evenimentul de apasare al butonului “*Afiseaza toti pacientii*”
- se reapeleaza metoda de afisare a tuturor pacientilor din baza de date in grid-ul de vizualizare pacienti
- **void** `button_stergePacient_Click(object sender, EventArgs e)`
 - se apeleaza la aparitia evenimentului de apasare a butonului “*Sterge*” din tab-ul de stergere pacienti (eveniment declansat de utilizator)
 - se realizeaza operatia de stergere a pacientului cu id-ul obtinut din componenta *Text* a obiectului `textBox_stergeID` din tabelul pacienti din baza de date
- **void** `button_editeazaPacienti_Click(object sender, EventArgs e)`
 - se apeleaza la aparitia evenimentului de apasare a butonului “*Editeaza*” din tab-ul de editare pacienti (eveniment declansat de utilizator)
 - se realizeaza operatia de actualizare in tabelul pacienti din baza de date a informatiilor despre pacienti, obtinute din obiectele *TextBox* (pentru numele pacientului, numele doctorului, numele spitalului, adresa spitalului), *ComboBox* (pentru sexul pacientului) si *MonthCalendar* (pentru data nasterii si data inceperii tratamentului)
- **void** `extract_radiografii_data(int id)`
 - extrage radiografiile din baza de date pentru un pacient anume
 - metoda primeste ca parametru id-ul pacientului si realizeaza o actiune de interogare in baza de date (pe tabelul pacienti) pentru a returna toate radiografiile acelui pacient. Datele radiografice se pastreaza in `dataGridView_radiografii` (care permite afisarea informatiilor despre radiografii intr-o forma tabelara)
- **void** `sterge_grid_radiografii()`
 - sterge toate elementele (toate radiografiile) din obiectul `dataGridView_radiografii` (elibereaza tabelul de vizualizare a radiografiilor)
- **void** `afisare_radiografie(int tip, int idr, int id)`
 - permite previzualizarea imaginii radiografice selectate din tabelul de vizualizare a radiografiilor. Acesta salveaza imaginea in `pictureBox_previzualizare`.
 - Metoda primeste ca parametri
 - *tip* – tipul fisierului care reprezinta imaginea radiografica (bmp sau dcm)
 - *idr* – id-ul radiografiei
 - *id* – id-ul pacientului
 - Acesti parametri sunt folositi pentru a salva fiecare imagine importata intr-un fisier din director al aplicatiei, folosind urmatoarea regula de salvare a fisierelor: Pentru fiecare pacient se creeaza un director cu numele pacientid (de exemplu pacient6 pentru pacientul cu id-ul 6). Fiecare radiografie este salvata intr-un fisier,

- in directorul pacientului respectiv, cu numele radidr.tip (de exemplu rad8.bmp pentru radiografia cu idr 8 care este salvata in formatul bmp).
- Datorita acestui sistem de salvare a fisierelor importate, este nevoie de cei trei parametri de asemenea pentru afisarea imaginii care reprezinta radiografia curenta.
 - **void** modifica_sterge_raadiografie()
 - se apeleaza atunci cand se doreste stergerea imaginii pentru previzualizarea radiografiei
 - aceasta metoda seteaza pe null componenta *Image* din *pictureBox_previzualizare*
 - **void** modifica_afisare_radiografie()
 - se apeleaza atunci cand a fost selectata o noua radiografie din grid-ul de vizualizare al radiografiilor
 - aceasta apeleaza metoda *afisare_radiografie* pentru modificarea componentei *Image* din *pictureBox_previzualizare*
 - **void** pictureBox_previzualizare_DoubleClick(**object** sender, **EventArgs** e)
 - aceasta metoda se foloseste pentru tratarea evenimentului de *DoubleClick* pe *pictureBox_previzualizare* (eveniment declansat de utilizator).
 - La declansarea evenimentului se va deschide un nou *Form*, numit *Form_radiografii*, pentru manipularea radiografiei curente.
 - **void** dataGridView_radiografii_CellClick(**object** sender, **DataGridViewCellEventArgs** e)
 - aceasta metoda se apeleaza la aparitia evenimentului de selectare a unei celule din grid-ul de vizualizare al radiografiilor (eveniment declansat de utilizator)
 - la selectarea oricarei celule reprezentand o caracteristica a unei radiografii, se considera ca radiografia respectiva este selectata (pentru prelucrari ulterioare)
 - **void** button_importaDCM_Click(**object** sender, **EventArgs** e)
 - metoda se utilizeaza la tratarea evenimentului de apasare a butonului “*Importa DCM*” (declansat de utilizator).
 - La apelarea acestei metode se va deschide un dialog vizual (*openFileDialog*) care va permite selectarea de catre utilizator a fisierului care se doreste a fi importat (verifica daca fisierul selectat este in formatul dicom si da eroare in caz contrar)
 - **void** button_importaBMP_Click(**object** sender, **EventArgs** e)
 - metoda se utilizeaza la tratarea evenimentului de apasare a butonului “*Importa BMP*” (declansat de utilizator).
 - La apelarea acestei metode se va deschide un dialog vizual (*openFileDialog*) care va permite selectarea de catre utilizator a fisierului care se doreste a fi importat (verifica daca fisierul selectat este in format bitmap si da eroare in caz contrar)
 - **void** button_adaugaRadiografie_Click(**object** sender, **EventArgs** e)

- metoda se foloseste pentru tratarea evenimentului de apasare a butonului *Adauga* din tab-ul de adaugare radiografii
- se realizeaza operatia de inserare in tabelul radiografii a fisierului (dicom sau bitmap) anterior importat
- `void button_stergeRadiografie_Click(object sender, EventArgs e)`
 - metoda se foloseste pentru tratarea evenimentului de apasare a butonului *Sterge* din tab-ul de stergere radiografii
 - se realizeaza operatia de stergere din baza de date a radiografiei selectate in grid-ul de vizualizare a radiografiilor

6.2.3. Clasa Dicom

Clasa Dicom este folosita pentru manipularea fisierelor DICOM. Principalele attribute ale clasei sunt urmatoarele:

- *objDicomFile* de tipul *DicomFile* (pentru incarcarea fisierului dicom intr-o structura parsabila)
- *dist_row* de tipul *float* (distanța între 2 pixeli consecutivi între randuri în mm)
- *dist_col* de tipul *float* (distanța între 2 pixeli consecutivi între coloane în mm)
- *picture* de tipul *Bitmap* (imaginea din fisierul dicom)

Metodele principale ale clasei Dicom sunt:

- `Bitmap SetImageObject(string strFileName)`
 - Deschide fisierul dicom in functie de parametrul *strFileName* si intoarce un *Bitmap* care reprezinta imaginea radiografica
- `void ExtrageDistXY()`
 - extrage informatiile despre distanta in mm pe verticala si orizontala între doi pixeli
- `Taguri[] FindImpTags()`
 - Cauta in fisierul Dicom si intoarce cele mai importante taguri
- `Module[] ListaCompletaModuleTaguri()`
 - Intoarce lista de taguri si valori corespunzatoare

Clasa *Dicom* foloseste doua clase ajutatoare, numite *Taguri* si *Module*, pentru a putea manipula tagurile si valorile acestora in fisierele DICOM.

6.2.4. Clasa BitmapPrimitive

Aceasta clasa se foloseste pentru operatii pe pixeli in fisierele bitmap (citire valoare intensitate pixel, schimbare valoare intensitate pixel, desenare primitiva (linie,cerc,dreptungi) intr-o culoare data, etc)

Principalele metode ale clasei sunt urmatoarele:

- **int** get_pixelValue_prim(BitmapData ImgData, **int** x, **int** y)
 - o aceasta metoda intoarce valoarea intensitatii pixelului (x,y) din obiectul de tip *BitmapData ImgData*
- **int** get_pixelValue(Bitmap poza, **int** x, **int** y)
 - o aceasta metoda intoarce valoarea intensitatii pixelului (x,y) din imaginea reprezentata de obiectul poza.
- **void** drawLine_prim(BitmapData ImgData, **int** x1, **int** y1, **int** x2, **int** y2, **Color** cul, **int** width)
 - o aceasta metoda deseneaza un segment de dreapta cu capetele (x1,y1) – (x2,y2) in culoarea *cul*, de latimea *width*, in obiectul *ImgData*
- **void** drawLine(Bitmap poza,int x1,int y1,int x2,int y2,Color cul,int width)
 - o metoda *drawLine* deseneaza un segment de dreapta cu capetele (x1,y1) – (x2,y2) in culoarea *cul*, de latimea *width*, in imaginea reprezentata de obiectul *Bitmap poza*
 - o aceasta metoda apeleaza *drawLine_prim* (dupa ce obtine obiectul de tip *BitmapData* din obiectul poza de tip *Bitmap*)
- **void** drawLine(Bitmap poza, **linie** l, **Color** cul, **int** width)
 - o similara cu metoda anterioara, dar in loc de parametrii *x1*, *y1*, *x2*, *y2*, ofera parametrul *linie l* (care are componentele *l.x1*, *l.y1*, *l.x2*, *l.y2*)
 - o apeleaza metoda anterioara
- **void** drawCircle_prim(BitmapData ImgData, **int** xc, **int** yc, **int** raza, **Color** cul, **int** N, **int** width)
 - o deseneaza un cerc cu centrul in punctul (xc,yc), de raza *raza*, in culoarea *cul*, cu *N* segmente de dreapta cu latimea *width* care aproximeaza cercul, in obiectul *ImgData*
- **void** drawCircle(Bitmap poza, **int** xc, **int** yc, **int** raza,Color cul, **int** N,int width)
 - o deseneaza un cerc cu centrul in punctul (xc,yc) de raza *raza*, in culoarea *cul*, cu *N* segmente de dreapta cu latimea *width* care aproximeaza cercul, in imaginea reprezentata de obiectul *Bitmap poza*
 - o apeleaza metoda *drawCircle_prim*
- **void** drawCircle(Bitmap poza, **cerc** c, **Color** cul, **int** N, **int** width)
 - o similara cu metoda anterioara, dar in loc de parametrii *xc*, *yc*, *raza*, ofera parametrul *cerc c* (care are componentele *c.xc*, *c.yc*, *c.r*)
 - o apeleaza metoda anterioara
- **void** drawRectangle(Bitmap poza, **int** x1, **int** y1, **int** x2, **int** y2, **Color** cul, **int** width)
 - o deseneaza un dreptunghi cu capetele diagonal opuse in (x1,y1), (x2,y2), in culoarea *cul*, cu latimea liniei *width*

- apeleaza metoda *drawLine* de 4 ori (pentru fiecare latura a dreptunghiului)
- **void** drawFilledRectangle(**Bitmap** poza, **int** x1, **int** y1, **int** x2, **int** y2, **Color** cul, **int** width)
 - deseneaza un dreptunghi plin cu capetele diagonal opuse in $(x1,y1)$, $(x2,y2)$, in culoarea cul, cu latimea liniei *width*
 - apeleaza metoda *drawLine* de $x2-x1$ ori
- **void** selectLine(**Bitmap** poza, **int** x1, **int** y1, **int** x2, **int** y2, **Color** cul, **int** width)
 - deseneaza in capetele segmentului de dreapta, determinat de $(x1,y1)$ - $(x2,y2)$, doua dreptunghiuri umplute in culoarea *cul*, scotand in evidenta acea linie
- **void** selectLine(**Bitmap** poza, **linie** l, **Color** cul, **int** width)
 - similara cu metoda anterioara, dar in loc de parametrii *x1*, *y1*, *x2*, *y2*, da ca parametru obiectul *l* de tip *linie* (care are componentele *l.x1*, *l.y1*, *l.x2*, *l.y2*)
 - apeleaza metoda anterioara
- **void** selectCircle(**Bitmap** poza, **int** xc, **int** yc, **int** raza, **Color** cul, **int** width)
 - aceasta metoda deseneaza in centrul cercului si in punctele cardinale ale cercului (sus, jos, stanga, dreapta), cate un dreptunghi umplut in culoarea *cul*, scotand in evidenta acel cerc
- **void** selectCircle(**Bitmap** poza, **cerc** c, **Color** cul, **int** width)
 - similara cu metoda anterioara, dar in loc de parametrii *xc,yc*, *raza*, ofera parametrul *c* de tip *cerc* (care are componentele *c.xc*, *c.yc*, *c.r*)
 - apeleaza metoda anterioara
- **public void** selectRectangle(**Bitmap** poza, **int** x1, **int** y1, **int** x2, **int** y2, **Color** cul, **int** width)
 - deseneaza in doua capete diagonal opuse cate un dreptunghi umplut in culoarea *cul*, scotand in evidenta dreptunghiul respectiv.

6.2.5. Clasa Canny

Aceasta clasa se foloseste pentru determinarea conturului intr-o imagine prin aplicarea detectorului Canny.

Cele mai importante atribute ale clasei sunt urmatoarele:

- *lowThreshold* (*int*) – pragul cel mic pentru aplicarea pasului de histerezis
- *highThreshold* (*int*) – pragul cel mare pentru aplicarea pasului de histerezis
- *xKernel*, *yKernel* – matrice de 3x3 int – kerneluri Sobel

Metodele principale ale clasei Canny sunt:

- **void** Filtru_Canny(**Bitmap** srcImg, **ref Bitmap** dstImg, **double** zoom)

- aceasta metoda realizeaza aplicarea filtrului Canny pornind de la imaginea *srcImg* si salvand conturul in imaginea *dstImg*
- `public Bitmap toGrey(Bitmap sourceImage, double gr, double gg, double gb)`
 - metoda *toGrey* porneste de la imaginea *sourceImage* si intoarce o imagine in niveluri de gri (in functie de coeficientii *gr*, *gg*, *gb* pentru rosu, verde si albastru)
- `Bitmap Blur(Bitmap sourceBitmapImage)`
 - Metoda porneste de la imaginea *sourceBitmapImage* si aplica un filtru de Blur (un filtru Gaussian) , intorcand o noua imagine, blurata
- `Bitmap Filtru(Bitmap sourceBitmapImage)`
 - Metoda aceasta este cea care realizeaza toti pasii detectorului Canny, fiind apelata de catre *Filtru_Canny*
 - Pasii pe care metoda ii urmareste sunt:
 - Pasul 1 – aplica un filtru de blur asupra imaginii
 - Pasul 2 – aplica un filtru Sobel
 - Pasul 3 – calculeaza directia gradientului
 - Pasul 4 – ajusteaza directia gradientului
 - Pasul 5 – non maxima suppression – inlatura din pixelii de front acei pixeli care nu reprezinta un maxim local
 - Pasul 6 – histerezis – aplica doua praguri pe imaginea rezultata din pasii anteriori. Toti pixelii cu valoarea intensitatii mai mica decat primul prag devin negrii. Toti pixelii cu valoarea intensitatii mai mare decat al doilea prag devini albi (pixeli de contur sau pixeli de frontiera). Pixelii care se afla intre cele doua praguri devin pixeli de contur daca se afla pe drumul dintre doi pixeli albi.

6.2.6. Clasa Hough

Clasa Hough contine metodele de aplicare a transformatei Hough pentru determinarea partilor din radiografii care pot fi approximate prin curbe simple (linii/cercuri).

Metodele principale ale clasei sunt urmatoarele:

- `public void Hough_linii(ref Bitmap src, int prag_intensitate, int prag_acum, limite_cautare lc, ref List<pereche_linie> lista_linii)`
 - aceasta metoda realizeaza implementarea transformatei Hough pentru linii, pe imaginea *src* (binarizata, dupa aplicarea algoritmului Canny)
 - *prag_intensitate* reprezinta pragul de la care se iau in considerare pixelii (*x,y*) pentru a fi verificati daca fac parte sau nu dintr-o linie

- *prag_acum* reprezinta pragul de la care o valoare a unui prag determina existenta unei linii
- *lista_linii* reprezinta lista de perechi (*r,teta*) determinate dupa aplicarea algoritmului Hough pentru linii
- `public void Hough_axa_diafizala(ref Bitmap src, int prag_intensitate, int prag_acum, limite_cautare lc, ref List<perche_linie> lista_linii)`
 - aceasta metoda salveaza in *lista_linii* acele perechi candidat ce determina liniile conturului corpului femural.
 - Parametrul *lc* specifica limitele de cautare (se cauta numai pixelii (*x,y*) din jumatatea de jos a imaginii si se iau in considerare numai acele unghiuri *teta* care determina linii aproximativ verticale.
 - se apeleaza metoda *Hough_linii* cu limitele de cautare specificate
 - la executia metodei *Hough_axa_diafizala*, lista de linii candidat este urmatoarea:

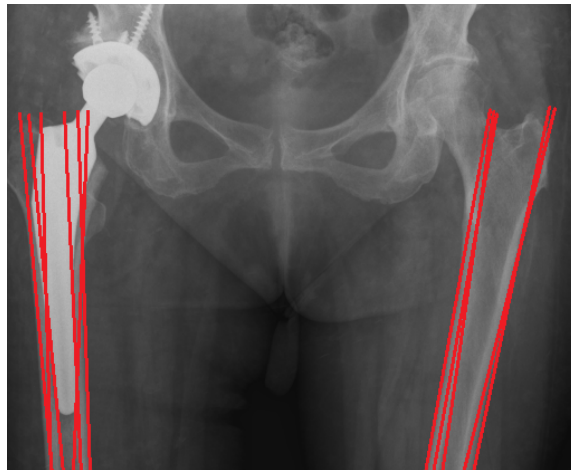


Figura 7. Liniile candidat pentru contururile corpurilor oaselor femurale

- `public void Hough_axa_protezei(ref Bitmap src, int prag_intensitate, int prag_acum, linie_axa_diafizala, float bone_width, ref List<perche_linie> lista_linii)`
 - aceasta metoda salveaza in *lista_linii* acele perechi candidat ce determina liniile conturului protezei.
 - Parametrul *axa_diafizala* reprezinta axa diafizala a femurului pentru care se cauta axa protezei (daca aceasta exista)
 - Parametrul *bone_width* specifica latimea corpului femural
 - In functie de *axa_diafizala* si *bone_width* se determina limitele de cautare a liniilor conturului protezei (se cauta doar in spatiul imagine determinat de conturul femurului)
 - Se apeleaza metoda *Hough_linii* cu limitele de cautare specificate

- `public void Hough_cercuri(ref Bitmap src, int prag_intensitate, int prag_acum, limite_cautare lc, ref List <cerc> lista_cercuri)`
 - o aceasta metoda salveaza in *lista_cercuri* toate cercurile care corespund limitelor de cautare *lc* (*lc* limiteaza atat pozitia pixelilor (*x,y*), cat si raza cercurilor cautate)
- `public void Hough_tuberozitati(ref Bitmap src, int prag_intensitate, int prag_acum, limite_cautare lc, ref cerc pc)`
 - o metoda *Hough_tuberozitati* salveaza in parametrul *pc* primul cerc (pozitia *xc,yc* a centrului si raza cercului), cautand de jos in sus , cu limitele de cautare *lc* determinate de corpurile femurale anterior calculate, care are un numar >*prag_acum* de pixeli de front(cu valoarea >*prag_intensitate*) cu proprietatea ca *y_pixel > yc* (se ia in considerare numai partea de jos a cercului)
 - o In figura urmatoare este un exemplu de cercuri intoarse la apelarea metodei *Hough_tuberozitati* (apelata de doua ori, pentru aflarea ambelor tuberozitati). Arcul de cerc rosu contine acei pixeli care sunt luati in considerare la determinarea unei tuberozitati

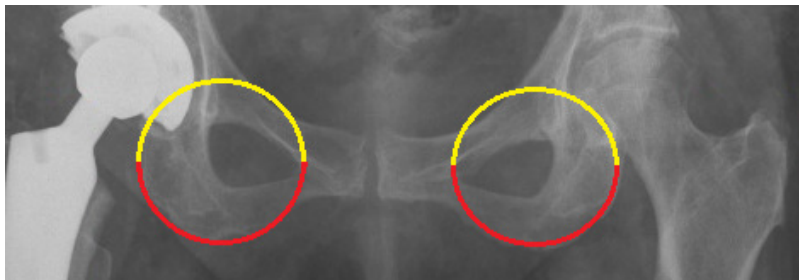


Figura 8. Cercurile reprezentand tuberozitatile ishiadice

- `public void Hough_cap_femural1(ref Bitmap src, int prag_intensitate, int prag_acum, limite_cautare lc, ref cerc c)`
 - o aceasta metoda salveaza in parametrul cerc primul cerc care corespunde capului femural
 - o se cauta de jos in sus, dinspre exterior spre interior (de la stanga spre dreapta), in limitele de cautare trasate prin parametrul *lc*, si se iau in considerare numai acei pixeli de front care se afle in partea de nord-est a cercului
- `public void Hough_cap_femural2(ref Bitmap src, int prag_intensitate, int prag_acum, limite_cautare lc, ref cerc c)`
 - o metoda similara cu *Hough_cap_femural1* – singura diferenta este directia de cautare a cercului (de la dreapta spre stanga) si pozitia pixelilor unui cerc candidat care sunt luati in considerare (cei din partea de nord-vest)

- in figura de mai jos sunt descrise cercurile salvate in parametrul c (cate un cerc pentru fiecare metoda $Hough_cap_femural$). Arcurile de cerc rosii contin pixelii care au fost luati in considerare in cautarea capetelor femurale.



Figura 9. Cercurile reprezentand capetele femurale

- `public void Hough_trochanter_mic1(ref Bitmap src, int prag_intensitate, int prag_acum, limite_cautare lc, linie l, float bone_width, ref cerc c)`
 - salveaza in parametrul c primul cerc care corespunde trohanterului mic stang
 - similar cu metodele anterioare, cu urmatoarele diferente:
 - se cauta de jos in sus, de la dreapta spre stanga (dinspre centrul imaginii inspre margini)
 - se iau in considerare numai pixelii din dreapta centrului cercului (se cauta un arc de cerc care are $x_pixel > x_centru$)
- `public void Hough_trochanter_mic2(ref Bitmap src, int prag_intensitate, int prag_acum, limite_cautare lc, linie l, float bone_width, ref cerc c)`
 - metoda aduce urmatoarele modificari la $Hough_trochanter_mic1$:
 - se cauta de jos in sus, de la stanga spre dreapta
 - se iau in considerare numai pixelii din stanga centrului cercului (se cauta un arc de cerc care are $x_pixel < x_centru$)
 - in figura de mai jos sunt descrise cercurile salvate in parametrul c (cate un cerc pentru fiecare metoda $Hough_trochanter_mic$). Arcurile de cerc rosii contin pixelii care au fost luati in considerare in cautarea trohanterelor mici.



Figura 10. Cercurile reprezentand trohanterele mici

- `public void Hough_col_sus1(ref Bitmap src, int prag_intensitate, int prag_acum, limite_cautare lc, ref cerc c)`
 - o salveaza in parametrul *c* primul cerc al carui arc de cerc din partea de sud-est reprezinta conturul din partea de sus a colului femural al osului stang
- `public void Hough_col_sus2(ref Bitmap src, int prag_intensitate, int prag_acum, limite_cautare lc, ref cerc c)`
 - o salveaza in parametrul *c* primul cerc al carui arc de cerc din partea de sud-vest corespunde conturului din partea de sus a colului femural al osului drept
 - o in figura de mai jos sunt descrise cercurile salvate in parametrii *c* (cate un cerc pentru fiecare metoda *Hough_col_sus*). Arcurile de cerc rosii contin pixelii care au fost luati in considerare in cautarea conturilor din partea de sus a gaturilor oaselor femurale.



Figura 11. Cercurile reprezentand contururile de sus ale gaturilor oaselor femurale

- `public void Hough_col_jos1(ref Bitmap src, int prag_intensitate, int prag_acum, limite_cautare lc, ref cerc c)`
 - o salveaza in parametrul *c* primul cerc al carui arc de cerc din partea de nord-vest corespunde conturului din partea de jos a colului femural al osului stang
- `public void Hough_col_jos2(ref Bitmap src, int prag_intensitate, int prag_acum, limite_cautare lc, ref cerc c)`
 - o salveaza in parametrul *c* primul cerc gasit al carui arc de cerc din partea de nord-est corespunde conturului din partea de jos a colului femural al osului drept
 - o in figura de mai jos sunt descrise cercurile salvate in parametrii *c* (cate un cerc pentru fiecare metoda *Hough_col_jos*). Arcurile de cerc rosii contin pixelii care au fost luati in considerare in cautarea conturilor din partea de jos a gaturilor oaselor femurale.

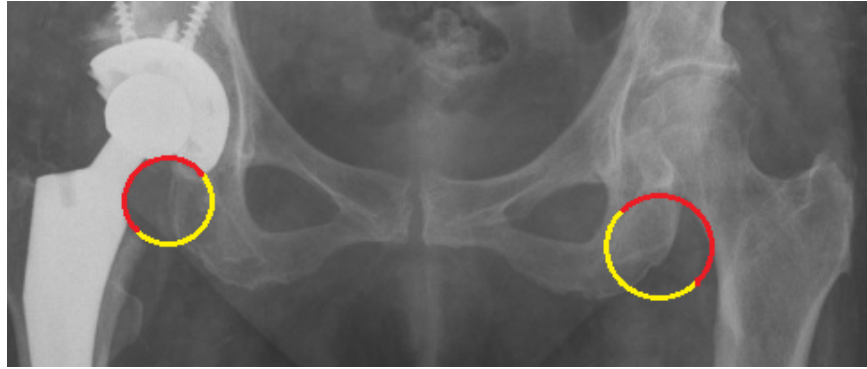


Figura 12. Cercurile reprezentand contururile de jos ale gaturilor oaselor femurale

- `public void Hough_cerc_sus1(ref Bitmap src, int prag_intensitate, int prag_acum, limite_cautare lc, ref cerc c)`
 - o salveaza in parametrul *c* primul cerc gasit al carui arc de cerc din partea de est corespunde conturului bazinului de deasupra capului femural al osului stang
- `public void Hough_cerc_sus2(ref Bitmap src, int prag_intensitate, int prag_acum, limite_cautare lc, ref cerc c)`
 - o salveaza in parametrul *c* primul cerc gasit al carui arc de cerc din partea de est corespunde conturului bazinului de deasupra capului femural al osului drept
 - o in figura de mai jos sunt descrise cercurile salvate in parametrii *c* (cate un cerc pentru fiecare metoda *Hough_cerc_sus*). Arcurile de cerc rosii contin pixelii care au fost luati in considerare in cautarea contururilor bazinului de deasupra capetelor femurale.



Figura 13. Cercurile reprezentand contururile bazinului de deasupra capetelor femurale

6.2.7. Clasa MasurariAutomate

Clasa MasurariAutomate este utilizata la determinarea tuturor parametrilor importanti in artroplastie, care se pot extrage automat.

Cele mai importante atribute ale clasei sunt urmatoarele:

- Obiectul *hough* de tip *Hough* (pentru transformatele Hough in vederea determinarii liniilor si cercurilor din imaginea radiografica)
- Obiectul *mas* de tip *masuratori* (pentru salvarea masurarilor automate realizate)

Principalele metode ale clasei sunt:

- **void** determina_axa_diafizala1(**ref** *Bitmap* src, **ref** *linie* axa1, **ref** *float* bone_width)
 - o aceasta metoda determina axa_diafizala si latimea osului stang pe imaginea radiografica salvata in obiectul *src* de tip *Bitmap*
 - o apeleaza metoda *Hough_axa_diafizala*, determinand lista de linii candidat pentru conturul corpului osului femural stang
 - o retine liniile extreme (cea mai din stanga si cea mai din dreapta) ca fiind liniile reprezentand conturul osului femural stang
 - o determina axa cilindrului determinat de cele doua linii anterior extrase. Aceasta axa se salveaza in parametrul *axa1*
 - o latimea cilindrului determinat de cele doua linii extreme se salveaza in parametrul *bone_width*
- **void** determina_axa_diafizala2(**ref** *Bitmap* src, **ref** *linie* axa2, **ref** *float* bone_width)
 - o metoda similara celei anterioare, singura diferenta fiind ca operatiile se realizeaza pe jumatatea din partea dreapta a imaginii
- **void** determina_axa_proteza1(**ref** *Bitmap* src, *linie* axa_diafizala, *float* bone_width, **ref** *linie* axa_protezei)
 - o salveaza axa protezei in parametrul *axa_protezei*, pornind de la informatiile despre axa diafizala (parametrul *axa_diafizala*) si latimea (*bone_width*) osului stang, din imaginea radiografica salvata in obiectul *src* de tip *Bitmap*
- **void** determina_axa_proteza2(**ref** *Bitmap* src, *linie* axa_diafizala, *float* bone_width, **ref** *linie* axa_protezei)
 - o similara cu metoda anterioara, singura diferenta fiind faptul ca extrage axa protezei osului drept
- **void** determina_tuberozitate(**ref** *Bitmap* src, *limite_cautare* lc, **ref** *cerc* c)
 - o apeleaza metoda *Hough_tuberozitati* pe imaginea *src*, cu limitele de cautare din *lc*, salvand cercul determinat in parametrul *c*
- **void** determina_cap_femural1(**ref** *Bitmap* src, *limite_cautare* lc, **ref** *cerc* c)
- **void** determina_cap_femural2(**ref** *Bitmap* src, *limite_cautare* lc, **ref** *cerc* c)
- **void** determina_trochanter_mic1(**ref** *Bitmap* src, *limite_cautare* lc, *linie* l, *float* bone_width, **ref** *cerc* c)
- **void** determina_trochanter_mic2(**ref** *Bitmap* src, *limite_cautare* lc, *linie* l, *float* bone_width, **ref** *cerc* c)

- void determina_col_sus1(ref Bitmap src, limite_cautare lc, ref cerc c)
- void determina_col_sus2(ref Bitmap src, limite_cautare lc, ref cerc c)
- void determina_col_jos1(ref Bitmap src, limite_cautare lc, ref cerc c)
- void determina_col_jos2(ref Bitmap src, limite_cautare lc, ref cerc c)
- void determina_cerc_sus1(ref Bitmap src, limite_cautare lc, ref cerc c)
- void determina_cerc_sus2(ref Bitmap src, limite_cautare lc, ref cerc c)
 - o toate aceste metode apeleaza functiile corespondente din clasa Hough pe imaginea *src* cu limitele de cautare din *lc*, salvand cercurile determinate in parametrul *c*
- void intersectie_linie_cerc(float x_linie, float y_linie, float panta, cerc c, ref float x1, ref float y1, ref float x2, ref float y2)
 - o determina punctele de intersectie (salvate in $(x1,y1)$ si $(x2,y2)$) ale unei linii (determinata de un punct (x_linie,y_linie) si *panta* acestuia) cu un cerc *c*
- void intersectie_cercuri(Bitmap src, cerc c1, cerc c2, ref float x1, ref float y1, ref float x2, ref float y2)
 - o determina punctele de intersectie (salvate in $(x1,y1)$ si $(x2,y2)$) ale doua cercuri *c1* si *c2*
- void initializeaza_masuratori(ref masuratori mas)
 - o metoda aceasta seteaza parametrii tuturor componentelor unui obiect *mas* de tip *masuratori*
- void calculeaza_axa_diafizala1(ref Bitmap src,ref linie axa_diafizala1, ref float bone_width)
 - o initializeaza componentele axei diafizale si latimea si apeleaza metoda *determina_axa_diafizala1*
- public void calculeaza_axa_diafizala2(ref Bitmap src, ref linie axa_diafizala2, ref float bone_width)
 - o similara cu metoda anterioara, apeland insa metoda *determina_axa_diafizala2*
- public void calculeaza_tuberozitate1(ref Bitmap src, ref cerc tuberozitate1, linie axa_diafizala1, float bone_width1, float bone_width2)
 - o aceasta metoda apeleaza metoda *determina_tuberozitati*, dar numai daca *axa_diafizala1* exista (nu are $x1,x2,y1,y2$ toate egale cu 0). In concluzie, tuberozitatile se determina numai dupa ce au fost extrase anterior axele diafizale
 - o in functie de *axa_diafizala1* si de *bone_width1*, seteaza limitele de cautare pentru metoda *determina_tuberozitati*
- void calculeaza_tuberozitate2(ref Bitmap src, ref cerc tuberozitate2, linie axa_diafizala2, float bone_width1, float bone_width2)
 - o similara cu metoda anterioara, dar determina tuberozitatea din partea dreapta

- **void** calculeaza_cap_femural1(**ref Bitmap** src, **ref cerc** cap_femural1, **linie** axa_diafizala1, **cerc** tuberozitate1)
 - o aceasta metoda apeleaza metoda *determina_cap_femural1*, dar numai daca *axa_diafizala1* si *tuberozitate1* exista (nu au componentele egale cu 0). In concluzie, capetele femurale se determina numai dupa ce (si daca) au fost determinate axele diafizale si tuberozitatile
 - o In functie de *axa_diafizala1* si *tuberozitate1* se seteaza limitele de cautare pentru metoda *determina_cap_femural1*
- **void** calculeaza_cap_femural2(**ref Bitmap** src, **ref cerc** cap_femural2, **linie** axa_diafizala2, **cerc** tuberozitate2)
 - o similara cu metoda anterioara
- **void** calculeaza_trohanter_mic1(**ref Bitmap** src, **ref cerc** trohanter_mic1, **linie** axa_diafizala1, **cerc** cap_femural1, **cerc** tuberozitate1, **float** bone_width)
 - o aceasta metoda apeleaza functia *determina_trohanter_mic1*, dar numai daca *axa_diafizala1*, *cap_femural1*, *tuberozitate1*, *bone_width* exista (nu au componentele egale cu 0). In concluzie, trohanterele mici se extrag numai dupa ce (si daca) au fost determinate anterior axele diafizale, tuberozitatile ishiadice si capetele femurale
 - o In functie de acesti parametri, se seteaza limitele de cautare pentru metoda *determina_trohanter_mic1*, si se salveaza in parametrul *trohanter_mic1* cercul determinat
- **void** calculeaza_trohanter_mic2(**ref Bitmap** src, **ref cerc** trohanter_mic2, **linie** axa_diafizala2, **cerc** cap_femural2, **cerc** tuberozitate2, **float** bone_width2)
 - o similara cu metoda anterioara
- **void** calculeaza_linia_bituberozitara(**ref Bitmap** src, **ref linie** linia_bituberozitara, **cerc** tuberozitate1, **cerc** tuberozitate2)
 - o aceasta metoda salveaza in parametrul *linia_bituberozitara* segmentul de dreapta care uneste extremitatile de jos ale cercurilor salvate in parametrii *tuberozitate1* si *tuberozitate2* (daca acestia au componentele diferite de 0). Deci, linia bituberozitara se calculeaza numai daca anterior au fost extrase in imaginea radiografica tuberozitatile ishiadice
- **void** calculeaza_linia_verticala(**ref Bitmap** src, **ref linie** linia_verticala, **linie** linia_bituberozitara)
 - o salveaza in parametrul *linia_verticala* mediatoarea segmentului reprezentat de parametrul *linia_bituberozitara*. Deci, linia verticala va fi extrasa numai daca anterior a fost determinata linia bituberozitara

- **void** calculeaza_diferenta_oase(**ref** **Bitmap** src, **ref** **float** diferenta_oase, **linie** linia_verticala, **linie** linia_bituberozitara, **cerc** trochanter_mic1, **cerc** trochanter_mic2)
 - o aceasta metoda salveaza in parametrul *diferenta_oase*, distanta pe verticala a trohanterelor mici. Pentru aceasta este necesara extragerea anterioara a liniei bituberozitare, a liniei verticale si a celor doua trohantere mici
- **void** calculeaza_proteze(**Bitmap** notcanny, **ref** **int** proteza, **linie** axa_diafizala1, **linie** axa_diafizala2)
 - o salveaza in parametrul *proteza* informatia despre protezele din imaginea radiografica (cate proteze sunt, si in dreptul carui femur). Pentru aceasta, este nevoie de extragerea anterioara a celor doua axe diafizale
- **void** calculeaza_proteza1(**ref** **Bitmap** src, **ref** **linie** axa_proteza1, **linie** axa_diafizala1, **int** proteza, **float** bone_width1)
 - o aceasta metoda salveaza in *axa_proteza1* informatiile despre axa protezei osului stang prin apelarea metodei *determina_axa_proteza1*, dar numai daca axa diafizala a fost anterior determinata si daca parametrul *proteza* indica faptul ca osul femural stang contine proteza
- **void** calculeaza_proteza2(**ref** **Bitmap** src, **ref** **linie** axa_proteza2, **linie** axa_diafizala2, **int** proteza, **float** bone_width2)
 - o similara cu metoda anterioara
- **void** calculeaza_col_sus1(**ref** **Bitmap** src, **ref** **cerc** col_sus1, **cerc** cap_femural1)
 - o salveaza in parametrul *col_sus1* informatiile despre conturul de sus al colului osului stang, in functie de capul femural stang
 - o apeleaza metoda *determina_col_sus1*, cu limitele de cautare stabilite in functie de parametrul *cap_femural1*
- **void** calculeaza_col_sus2(**ref** **Bitmap** src, **ref** **cerc** col_sus2, **cerc** cap_femural2)
 - o similara cu metoda anterioara
- **void** calculeaza_col_jos1(**ref** **Bitmap** src, **ref** **cerc** col_jos1, **cerc** cap_femural1)
 - o determina cercul (salvat in parametrul *col_jos1*) care se mapeaza pe conturul de jos al colului femural stang in functie de parametrul *cap_femural1*
- **void** calculeaza_col_jos2(**ref** **Bitmap** src, **ref** **cerc** col_jos2, **cerc** cap_femural2)
 - o similara cu metoda anterioara
- **void** calculeaza_axa_colului1(**ref** **Bitmap** src, **ref** **linie** axa_colului1, **cerc** col_sus1, **cerc** col_jos1)
 - o metoda salveaza in *axa_colului1* informatiile despre axa colului femural stang, dar numai daca au fost anterior determinate *col_sus1* si *col_jos1*
 - o aceasta axa reprezinta mediatoarea celui mai scurt segment care uneste cercurile *col_sus1* si *col_jos1*

- **void** calculeaza_axa_colului2(**ref** Bitmap src, **ref** linie axa_colului2, **cerc** col_sus2, **cerc** col_jos2)
 - o similara cu metoda *calculeaza_axa_colului1*
- **void** calculeaza_unghiul_cervico_diafizar1(**ref** float unghi_cervico_diafizar1, **linie** axa_diafizala1, **linie** axa_colului1)
 - o aceasta metoda salveaza unghiul cervico-diafizar in parametrul *unghi_cervico_diafizar1* in functie de *axa_diafizala1* si *axa_colului1* (se poate determina numai daca axa diafizala si axa colului au fost anterior extrase)
- **void** calculeaza_unghiul_cervico_diafizar2(**ref** float unghi_cervico_diafizar2, **linie** axa_diafizala2, **linie** axa_colului2)
 - o similara cu metoda anterioara
- **void** calculeaza_cerc_sus1(**ref** Bitmap src, **ref** cerc cerc_sus1, **cerc** col_sus1)
 - o aceasta metoda salveaza in parametrul *cerc_sus1* informatiile despre cercul al carui arc de cerc se mapeaza pe conturul bazinului de deasupra capului femural
 - o *cerc_sus1* se determina numai daca anterior a fost extras *col_sus1*
- **void** calculeaza_cerc_sus2(**ref** Bitmap src, **ref** cerc cerc_sus2, **cerc** col_sus2)
 - o similara cu metoda anterioara
- **void** calculeaza_axa_capului_femural1(**ref** Bitmap src, **ref** linie axa_cap_femural1, **cerc** cerc_sus1, **cerc** col_sus1, **cerc** cap_femural1, **cerc** col_jos1, **cerc** tuberozitate1)
 - o aceasta metoda salveaza in parametrul *axa_cap_femural1* informatiile despre axa capului femural stang, in functie de parametrii anterior determinati reprezentati de *cerc_sus1*, *col_sus1*, *cap_femural1*, *col_jos1*, *tuberozitate1*
 - o primul capat al axei capului femural (*axa_cap_femural1.x1*, *axa_cap_femural1.y1*) se determina ca media aritmetica a pozitiilor a trei puncte:
 - punctul de intersectie din dreapta al cercurilor *cerc_sus1* si *col_sus1*
 - punctul de intersectie de jos al cercurilor *cerc_sus1* si *cap_femural1*
 - punctul de intersectie de sus al cercurilor *cap_femural1* si *col_sus1*
 - o al doilea capat al axei capului femural (*axa_cap_femural1.x2*, *axa_cap_femural1.y2*) se determina ca media aritmetica a pozitiilor a trei puncte:
 - punctul de intersectie din dreapta al lui *cap_femural1* cu *col_jos1*
 - punctul de intersectie de sus dintre *tuberozitate_1* si *col_jos1*
 - punctul de intersectie de jos dintre *cap_femural1* si *tuberozitate1*
- **void** calculeaza_axa_capului_femural2(**ref** Bitmap src, **ref** linie axa_cap_femural2, **cerc** cerc_sus2, **cerc** col_sus2, **cerc** cap_femural2, **cerc** col_jos2, **cerc** tuberozitate2)
 - o similara cu metoda anterioara
- **void** calculeaza_deviatia_protezei1(**ref** float deviatie_proteza1, **int** proteza, **linie** axa_diafizala1, **linie** axa_proteza1)

- aceasta metoda calculeaza in parametrul *deviatie_proteza1* unghiul format de axa diafizala (parametrul *axa_diafizala1*) cu axa protezei (parametrul *axa_proteza1*) a oslui stang
- **void** calculeaza_deviatia_protezei2(**ref float** deviatie_proteza2, **int** proteza, **linie** axa_diafizala2, **linie** axa_proteza2)
 - similara cu metoda anterioara
- **void** calculeaza_unghi_acetabular1(**ref float** unghi_acetabular1, **linie** axa_cap_femural1, **linie** linia_verticala)
 - aceasta metoda calculeaza in parametrul *unghi_acetabular1* unghiul format de axa capului femural stang (din parametrul *axa_cap_femural1*) cu linia verticala de referinta (din parametrul *linia_verticala*)
- **void** calculeaza_unghi_acetabular2(**ref float** unghi_acetabular2, **linie** axa_cap_femural2, **linie** linia_verticala)
 - similara metodei anterioare
- **void** masuratori(**Bitmap** notcanny, **ref Bitmap** src, **ref masuratori** mas)
 - salveaza in parametrul *mas* toate masuratorile automate (realizeaza initializarea masuratorilor si calcularea lor prin apelarea tuturor metodelor anterioare care au prefixul calculeaza)

6.2.8. Clasa Form_radiografii

Aceasta extinde clasa Form din C# si este folosita pentru descrierea interfetei cu utilizatorul in formul secundar, dupa ce este selectata o radiografie din formul principal. Acest form permite urmatoarele functionalitati:

- Afisarea informatiilor despre radiografia curenta
- Realizarea de masuratori manuale
- Realizarea de masuratori automate
- Manipularea radiografiei (afisarea conturului obtinut cu detectorul Canny, afisarea masuratorilor manuale efectuate, afisarea masuratorilor automate, salvarea imaginii rezultate intr-un fisier bmp)

Principalele atribute ale clasei sunt:

- *radiografie* de tip *Bitmap* care pastreaza radiografia la marimea ei initiala
- *radiografie_ajustata* de tip *Bitmap* care pastreaza radiografia la o marime ajustata (in functie de factorul de zoom, pastrand proportiile intre lungime si latime)
- *cannyImg_final* de tip *Bitmap* care pastreaza radiografia continand conturul imaginii radiografice, la o marime ajustata (in functie de factorul de zoom)
- alte obiecte de tip *Bitmap* ajutatoare pentru manipularea imaginilor

- *mas* de tip *masuratori* care pastreaza masuratorile din imaginea ajustata (in functie de factorul de zoom)
- *mas_img_init* de tip *masuratori* care pastreaza masuratorile din imaginea initiala (in functie de factorul de zoom)
- *zoomFactor* de tip *float*, factorul cu care se scaleaza imaginea, dupa ce a fost scalata ca sa intre 100% in panoul de vizualizare al radiografiei
- alte variabile ajutatoare

Metodele principale ale clasei sunt prezentate in paginile de mai jos.

Metodele care trateaza evenimentele declansate de utilizator sunt urmatoarele:

- *MouseEnter* pe un buton din tab-ul de masuratori automate: apar *tool-tips* cu un text explicativ despre functionalitatea fiecarui buton (afisare/ editare/ salvare/ recalculare/ stergere ale fiecarui parametru determinat in masuratorile automate – axe diafizale, tuberozitati, linia bituberozitara, capete femurale, trohantere mici, linia verticala, distanta pe verticala intre oase, axele protezelor, deviatiile protezelor, axele capetelor femurale, unghiurile acetabulare, axele colurilor femurale, unghiurile cervico-diafizare):
 - o `void button_afiseazaAxaDiafizala1_MouseEnter(object sender, EventArgs e)`
 - o ...
 - o `void button_stergeUnghiulCervicoDiafizar2_MouseEnter(object sender, EventArgs e)`
- *MouseEnter* pe un buton din tab-ul de masuratori manuale: apar *tool-tips* cu un text explicativ despre functionalitatea fiecarui buton (adaugare obiect nou/ editare/ stergere pentru obiectele din masuratori manuale – linie, cerc, dreptunghi)
- *Click* pe un buton de Afisare din tab-ul de masuratori automate:
 - o Modifica parametrul afiseaza pentru componenta din masuratori care corespunde butonului respectiv.
- *Click* pe un buton de Editare din tab-ul de masuratori automate:
 - o Seteaza pe *true* componenta *Enabled* din obiectele de tip *TextBox* ale parametrului din masuratori care corespunde butonului respectiv (ca sa se poata schimba valorile componentelor parametrului respectiv prin modificarea *textbox-urilor*)
 - o Selecteaza parametrul respectiv pentru a-l scoate in evidenta pe ecran (daca e linie ii scoate in evidenta capetele, daca e cerc ii scoate in evidenta centrul si punctele cardinale) – cu metodele *selectLine* si *selectCircle* din clasa *BitmapPrimitive*
- *Click* pe un buton de Salvare din tab-ul de masuratori automate

- Setează pe false component *Enabled* din obiectele de tip *TextBox* ale parametrului din măsuratori care corespunde butonului respectiv (nu se mai pot edita *textbox*-urile)
- Actualizează în baza de date, în tabelul *masuratori_automate*, valorile componentelor parametrului corespunzător butonului, cu valorile din *textbox*-uri
- *Click* pe un buton de Recalculare din tab-ul de măsuratori automate
 - Apelează din nou metodele din clasa *MasuratoriAutomate* care au prefixul *calculeaza* și corespund parametrului din măsuratori automate al cărui buton a fost apăsător
 - Actualizează în baza de date, în tabelul *masuratori_automate*, valorile componentelor parametrului corespunzător butonului
- *Click* pe un buton de Stergere din tab-ul de măsuratori automate
 - Modifică textul din fiecare obiect de tip *TextBox* aferent parametrului corespunzător butonului de Stergere apăsător (textul devine 0)
 - Actualizează în baza de date, în tabelul *masuratori_automate*, valorile componentelor parametrului corespunzător butonului (le pune pe 0)
 - Apelează din nou metoda de afișare a măsurătorilor automate. În felul acesta, parametrul corespunzător nu va mai fi afișat
- *Click* pe un buton de obiect nou din tab-ul de măsuratori manuale
 - Setează pe *true* un parametru *bool* corespunzător tipului de obiect nou creat (linie, cerc sau dreptunghi) și pe false parametrii corespunzător celorlalte tipuri (de exemplu la crearea unei linii noi, *linie_noua=true*, *cerc_nou=false* și *dreptunghi_nou=false*)
- *Click* pe un buton de stergere obiect din tab-ul de măsuratori manuale
 - Se șterge din tabelul linii, cercuri sau dreptunghiuri (în funcție de tipul obiectului ales pentru ștergere din grid-ul de vizualizare al obiectelor desenate manual) acel obiect selectat.
- *Click* pe un buton de editare obiect din tab-ul de măsuratori manuale
 - Se setează pe *true* un parametru *bool* corespunzător tipului de obiect editat (de exemplu, la editarea unei linii, *linie_edit=true*, *cerc_edit=false* și *dreptunghi_edit=false*)
- `void` *tabPage_masuratoriManuale_Enter*(`object` sender, `EventArgs` e)
 - se apelează când se intră în tab-ul de măsuratori manuale
 - setează anumite atribute, pentru simplificarea metodelor specifice tabului de măsuratori manuale
- `void` *tabPage_masuratoriAutomate_Enter*(`object` sender, `EventArgs` e)
 - se apelează când se intră în tab-ul de măsuratori automate

- `void tabPage_manipulareFisiere_Enter(object sender, EventArgs e)`
 - o se apeleaza cand se intra in tab-ul de manipulare fisiere
- `void tabPage_informatii_Enter(object sender, EventArgs e)`
 - o se apeleaza cand se intra in tab-ul de informatii pacient
- `void trackBar_zoomControl_Scroll(object sender, EventArgs e)`
 - o determina factorul de zoom al imaginii
- `void dataGridView_linii_CellContentClick(object sender, DataGridViewCellEventArgs e)/
dataGridView_cercuri_CellContentClick()/
dataGridView_dreptunghiuri_CellContentClick()`
 - o se apeleaza atunci cand utilizatorul selecteaza o celula din grid-ul de vizualizare al liniilor/ cercurilor/ dreptunghiurilor
 - o se selecteaza obiectul de pe linia celulei selectate, pentru actiuni ulterioare
 - o se utilizeaza in tab-ul de masuratori manuale
- `void pictureBox_rad_MouseDown(object sender, System.Windows.Forms.MouseEventArgs e)`
 - o Aceasta metoda se apeleaza atunci cand utilizatorul incepe *MouseDown* in panoul de afisare a imaginii radiografice
 - o In functie de tab-ul activ (cel de masuratori automate sau cel de masuratori manuale), in functie de obiectul selectat (linie/cerc/dreptunghi curent din masuratori manuale sau axa diafizala/tuberozitatea/.../axa capului femural selectata din masuratori automate) si in functie de operatia selectata (de desenare obiect nou sau de editare obiect selectat), se incepe operatia de desenare/editare obiect
- `void pictureBox_rad_MouseMove(object sender, System.Windows.Forms.MouseEventArgs e)`
 - o aceasta metoda se apeleaza atunci cand utilizatorul realizeaza *MouseMove* in panoul de afisare a imaginii radiografice
 - o In functie de tab-ul activ (cel de masuratori automate sau cel de masuratori manuale), in functie de obiectul selectat (linie/cerc/dreptunghi curent din masuratori manuale sau axa diafizala/tuberozitatea/.../axa capului femural selectata din masuratori automate) si in functie de operatia selectata (de desenare obiect nou sau de editare obiect selectat), se realizeaza operatia de desenare/editare obiect
- `void pictureBox_rad_MouseUp(object sender, System.Windows.Forms.MouseEventArgs e)`
 - o aceasta metoda se apeleaza atunci cand utilizatorul realizeaza *MouseUp* in panoul de afisare a imaginii radiografice

- In functie de tab-ul activ (cel de masuratori automate sau cel de masuratori manuale), in functie de obiectul selectat (linie/cerc/dreptunghi curent din masuratori manuale sau axa diafizala/tuberozitatea/.../axa capului femural selectata din masuratori automate) si in functie de operatia selectata (de desenare obiect nou sau de editare obiect selectat), se incheie operatia de desenare/editare obiect
- **void** button_afiseazaContur_Click(**object** sender, **EventArgs** e)
 - aceasta metoda se apeleaza in tab-ul de manipulare radiografii, atunci cand utilizatorul apasa butonul de afisare a conturului imaginii
 - metoda interschimba imaginile radiografie initiala fata de conturul radiografiei initiale
- **void** button_afiseazaMasAutomate_Click(**object** sender, **EventArgs** e)
 - aceasta metoda se apeleaza in tab-ul de manipulare fisiere/radiografii, atunci cand utilizatorul apasa butonul de afisare a masuratorilor automate
 - afiseaza/ascunde masuratorile automate
- **void** button_afiseazaMasManuale_Click(**object** sender, **EventArgs** e)
 - aceasta metoda se apeleaza in tab-ul de manipulare fisiere/radiografii, atunci cand utilizatorul apasa butonul de afisare a masuratorilor manuale
 - afiseaza/ascunde masuratorile manuale
- **void** button_salveaza_Click(**object** sender, **EventArgs** e)
 - salveaza imaginea intr-un fisier bmp selectat de utilizator printr-un obiect de tip obiect de tip saveFileDialog, in dimensiunea aleasa de utilizator (prin textbox-urile textBox_latime si textBox_lungime), cu posibilitatea pastrarii proportiilor dimensiunii initiale

Celelalte metode importante ale clasei sunt:

- **Form_radiografie(Bitmap** poza,**int** idp,**int** idr)
 - Constructorul clasei
 - *poza* de tip *Bitmap* reprezinta obiectul in care sunt salvati pixelii imaginii radiografice
 - *idp* de tip *int* reprezinta id-ul pacientului al carei radiografii este analizata
 - *idr* este id-ul radiografiei analizate
- **afla_line_width(Bitmap** img, **ref int** line_width)
 - in functie de dimensiunile imaginii img se determina latimea liniei cu care ar trebui desenate primitivele (cerc, linie, dreptunghi)
- **Bitmap** ResizeBitmap(**Bitmap** b, **int** nWidth, **int** nHeight)
 - Redimensioneaza imaginea *b* in functie de latime (*nWidth*) si inaltime (*nHeight*)

- **void** `adjustSizeToScreen(Bitmap radiografie, ref Bitmap masManualeImg, float zoom, float zoomFactor)`
 - redimensioneaza imaginea *masManualeImg* in functie de *radiografie*, *zoom* si *zoomFactor*
 - *radiografie* reprezinta radiografia la dimensiunile initiale
 - *zoom* reprezinta factorul de scalare care redimensioneaza imaginea initiala pentru a incapa 100% in panoul de vizualizare al radiografiilor
 - *zoomFactor* este dat de la utilizator (factorul de scalare al imaginii) salvat in componenta *Value* a unui obiect de tip *TrackBar*
- **void** `afla_info_pacient()`
 - salveaza in niste variabile globale informatiile despre pacient, in functie de id-ul pacient (salvat in atributul *idp*)
 - realizeaza o operatie de interogare a tabelului pacienti din baza de date
- **void** `afiseaza_informatii()`
 - afiseaza in obiecte de tip *TextBox* informatiile despre pacient obtinute in metoda anterioara
 - folosita in tab-ul de informatii pacient
- **void** `salvez_linie_noua()`
 - salveaza in tabelul linii din baza de date o noua linie
 - folosita in tab-ul de masuratori manuale
- **void** `salvez_cerc_nou()`
 - salveaza in tabelul cercuri din baza de date un cerc nou
 - folosita in tab-ul de masuratori manuale
- **void** `salvez_dreptunghi_nou()`
 - salveaza in tabelul dreptunghiuri din baza de date un dreptunghi nou
 - folosita in tab-ul de masuratori manuale
- **void** `update_linie()/update_cerc()/update_dreptunghi()`
 - realizeaza operatia de actualizare a unei inregistrari din tabelul de linii/ cercuri/ dreptunghiuri
- **void** `extrage_linii()/extrage_cercuri()/extrage_dreptunghiuri`
 - extrage din baza de date toate liniile/ cercurile/ dreptunghiurile din tabelul linii/ cercuri/ dreptunghiuri si afiseaza informatiile despre ele in gridul de vizualizare al liniilor/ cercurilor/ dreptunghiurilor
- **void** `afiseaza liniile_radiografie (ref Bitmap img)/ afiseaza_cercurile_radiografie (ref Bitmap img)/ afiseaza_dreptunghiurile_radiografie (ref Bitmap img)`

- deseneaza in imaginea *img* toate liniile/ cercurile/ dreptunghiurile din grid-ul de vizualizare al liniilor/ cercurilor/ dreptunghiurilor (tinand cont de factorul de zoom)
- **void** completeaza_textbox_masuratori_automate()
 - completeaza in *textbox*-urile aferente parametrilor masuratorilor automate, componentele parametrilor corespunzatori
- **void** afiseaza_masuratori_automate(ref **Bitmap** *img*)
 - deseneaza in imaginea *img* toti parametrii masuratorilor automate, in functie de componenta afiseaza de tip *bool* si in culoarea *color* a acestor parametri (care sunt de tip *linie* sau *cerc*)
- **bool** extrage_masuratori_automate_baza_date()
 - salveaza in atributul *mas_img_init* de tip *MasuratoriAutomate* informatiile despre parametrii determinati in masuratorile automate
 - realizeaza o operatie de interogare a bazei de date, pe tabelul *masuratori_automate*
- **void** salvez_masuratori_automate_baza_date()
 - salveaza in baza de date, in tabelul *masuratori_automate*, componentele parametrilor importanti in artroplastie determinate automat (pastrate in atributul *mas_img_init*)
- **void** updatez_masuratori_automate_baza_date()
 - realizeaza o operatie de actualizare a tabelului *masuratori_automate* din baza de date, cu parametrii modificati din obiectul *mas_img_init*

7. INTERFATA UTILIZATOR

In aceasta sectiune va fi prezentata interfata utilizator si modul de functionare a aplicatiei.

7.1. Form-ul pacienti

La rularea aplicatiei apare form-ul principal, numit si Form pacienti.

Acesta este impartit in patru sectiuni:

- In partea din stanga sus se afla gridul de vizualizare al pacientilor. Initial nu este selectat nici un pacient
- In partea din stanga sus se afla sectiunea de adaugare/stergere/editare pacienti, cu cate un tab corespunzator fiecărei operatii
- In partea din dreapta sus se afla sectiunea de cautare pacienti

- In partea din dreapta jos se afla gridul de vizualizare pacienti si sectiunea de stergere/adaugare radiografii. Aceasta sectiune devine activa numai daca este selectat un pacient in gridul de vizualizare pacienti/

In figura 15 este prezentat formularul la deschiderea aplicatiei

The screenshot shows the 'Radiografii' application window. It features a table of patients, search fields for patient details, and forms for adding or editing patient data and radiographs.

Id	Nume Pacient	Nume Doctor	Data tratament	Data nasterii	Sex	Nume Spital
1	Morar Anca	Popescu Dana	2010-07-20	1996-07-18	m	Spital
2	Morar Beni	Popescu Dana	2010-07-13	1983-07-06	m	Spital
3	Manea Maria	Oprescu Daniela	2010-07-12	1985-07-24	f	Spital
4	Manea Mihai	Oprescu Daniela	2010-07-21	1961-07-25	m	Spital
5	Morar Gavrilă	Maris Daniela	2010-07-13	1954-07-20	m	Spital
6	Morar Rodica	Maris Daniela	2010-07-22	2010-07-07	f	Spital
7	Matei Rares	Maris Daniela	2010-07-14	1995-07-20	m	Spital
8	Matei Tudor	Maris Daniela	2009-07-14	2000-07-27	m	Spital
9	Berca Radu	Matei Dan	2010-02-16	1973-07-19	m	Spital
10	Berca Emanuela	Matei Dan	2010-07-15	1968-07-25	f	Spital
11	Berca Teodora	Matei Gabriela	2010-04-12	2002-07-24	f	Spital
12	Berca Alexandra	Matei Gabriela	2010-07-07	2005-07-26	f	Spital

The interface also includes a search section with fields for 'Nume Pacient', 'Nume Doctor', and 'Nume Spital', and buttons for 'Cauta' and 'Afiseaza toti pacientii'. Below the table are sections for 'Adauga/Sterge/Editeaza Pacienti' and 'Adauga/Sterge Radiografii', each with various input fields and buttons like 'Adauga', 'Importa DCM', and 'Importa BMP'.

Figura 15. Form-ul pacienti

7.1.1. Grid-ul de vizualizare pacienti

Acesta contine informatiile despre toti pacientii din baza de date in forma tabelara. Grid-ul de vizualizare se modifica la realizarea urmatoarelor operatii:

- Selectare pacient (se selecteaza intreaga linie in grid-ul de vizualizare, la selectia oricarei celule corespunzatoare unui pacient)
- Adaugare pacient nou (se adauga o noua linie in grid-ul de vizualizare, la sfarsit, cu informatiile noului pacient)

- Editare pacient (se actualizeaza informatiile pacientului selectat pentru editare)
- Stergere pacient (dispare linia din grid-ul de vizualizare, corespunzatoare pacientului sters)
- Cautare pacient (apar numai pacientii care se incadreaza in criteriile de cautare)

7.1.2. Sectiunea de adaugare/pacienti/stergere/editare pacienti

7.1.2.1. Adaugare pacienti

La lansarea aplicatiei in aceasta sectiune este vizibil tab-ul de adaugare pacienti.

Aici se pot introduce informatiile despre pacientul care se doreste a fi inserat in baza de date:

The screenshot shows a software interface for patient management. The title bar reads "Adauga/Sterge/Editeaza Pacienti". On the left, a vertical sidebar contains three buttons: "Adauga", "Sterge", and "Editeaza". The main content area is a form with the following fields and controls:

- Nume Pacient:** Text input field containing "Popescu".
- Nume Doctor:** Text input field containing "Ion".
- Data inceperii tratamentului:** A calendar for November 2010. The date "8" is selected and highlighted with a blue box. Below the calendar, it says "Today: 11/8/2010".
- Data nasterii:** A calendar for December 1956. The date "20" is selected and highlighted with a blue box. Below the calendar, it says "Today: 11/8/2010".
- Sex:** A dropdown menu currently showing "Masculin".
- Nume Spital:** Text input field containing "Spitalul Fundeni".
- Adresa Spital:** Text input field containing "Fundeni".
- Adauga:** A large, grey button at the bottom center of the form.

Figura 16. Introducerea unui nou pacient in baza de date

La apasarea butonului "Adauga", apare pe ecran urmatorul *MessageBox*:

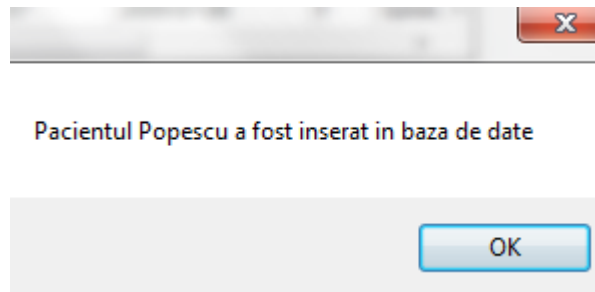


Figura 17. Mesaj de confirmare a adaugarii unui nou pacient

Dupa apasarea butonului “*Adauga*”, pacientul va putea fi vizualizat in grid-ul de vizualizare pacienti.

7.1.2.2. Editare pacienti

Atunci cand un pacient din grid-ul de vizualizare pacienti este selectat, si se intra in tab-ul de Editare pacienti, in textbox-uri sunt salvate informatiile despre pacientul respectiv.

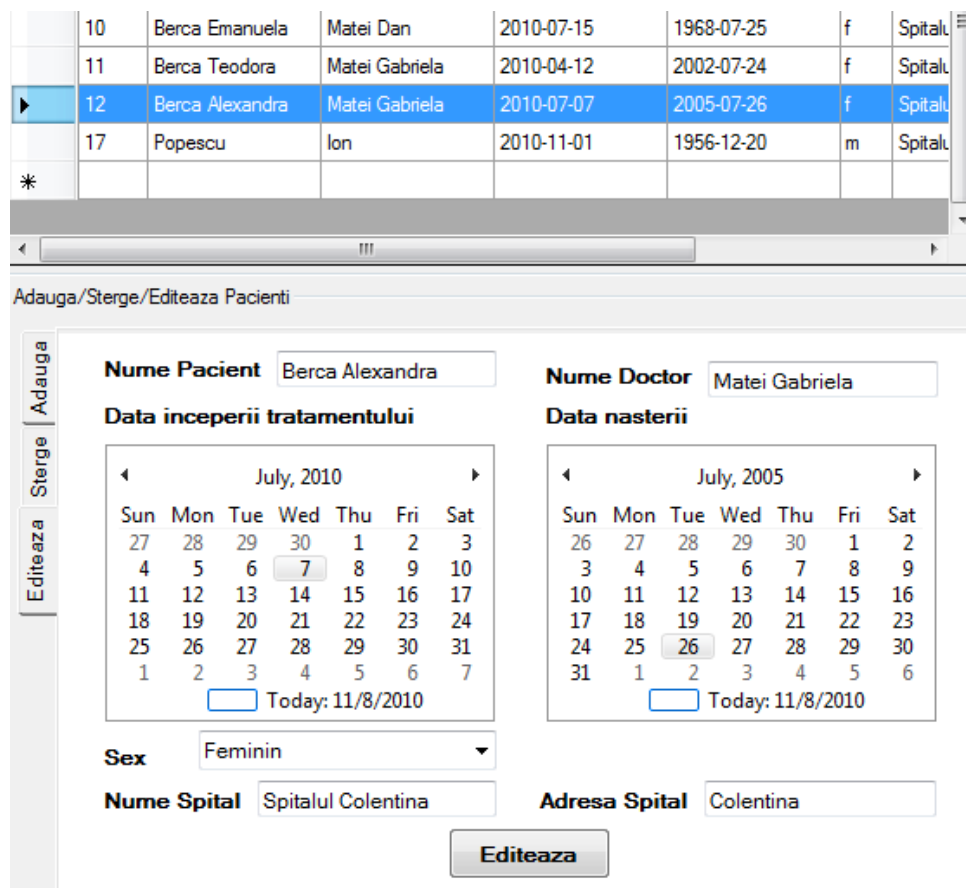


Figura 18. Editarea pacientului selectat din grid-ul de vizualizare pacienti

Din acest moment se poate modifica orice informatie din tab-ul de Editare (nume pacient, nume doctor, etc). Dupa ce au fost modificate *textbox*-urile, *monthcalendar*-urile si/sau *combobox*-urile, la apasarea butonului “*Editeaza*”, apare un *MessageBox* asemanator cu cel din Figura 17 cu mesajul “Datele pacientului Berca Alexandra au fost reactualizate”, iar in grid-ul de vizualizare se pot observa noile informatii

7.1.2.3. Stergere pacienti

La selectarea unui pacient din grid-ul de vizualizare si la intrarea in tab-ul de stergere pacienti, form-ul va arata in felul urmatoar:

The screenshot shows a web application interface. At the top, there is a grid of patients with the following data:

ID	Nume Pacient	Nume Doctor	Data Inregistrari	Data Nasterii	Sex	Locul de Muncă
11	Berca Teodora	Matei Gabriela	2010-04-12	2002-07-24	f	Spitalu
12	Berca Alexia	Matei Gabriela	2010-07-07	2005-07-26	f	Spitalu
17	Popescu	Ion	2010-11-01	1956-12-20	m	Spitalu

Below the grid, there is a form titled "Adauga/Sterge/Editeaza Pacienti". The form has three tabs: "Adauga", "Sterge", and "Editeaza". The "Sterge" tab is selected. The form contains the following fields and buttons:

- ID Pacient**:
- Nume Pacient**:
- Sterge**:

Figura 19. Stergerea unui pacient

Acelasi rezultat in tab-ul de stergere pacienti se poate obtine si fara selectarea unui pacient din grid-ul de vizualizare, prin inserarea id-ului pacient in *textbox*-ul corespunzator (urmand ca *textbox*-ul corespunzator numelui pacientului sa se actualizeze in functie de *id*, neputand fi modificat de catre utilizator).

La apasarea butonului “*Sterge*”, pe ecran va aparea urmatorul *MessageBox*:

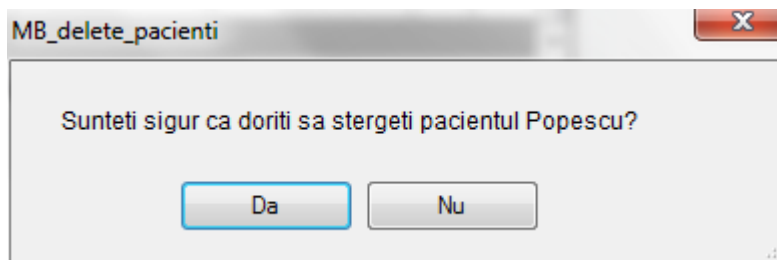


Figura 20. Mesaj de interogare a utilizatorului cu privire la stergerea unui pacient

Daca este apasat butonul “Da”, atunci pacientul este sters din baza de date, apare un mesaj de confirmare a stergerii, asemanator cu cel din figura 17, iar linia corespunzatoare pacientului nu mai este vizibila in grid-ul de vizualizare pacienti

7.1.3. Sectiunea de cautare pacienti

In aceasta sectiune este posibila cautarea pacientilor dupa urmatoarele criterii:

- Nume pacient
- Nume doctor
- Nume spital

La completarea textboxurilor cu aceste informatii (pacient si/sau doctor si/sau spital), in grid-ul de vizualizare pacienti, dupa apasarea butonului “Cauta”, vor aparea numai acei pacienti care corespund criteriilor de cautare. In figura 21 se poate observa un exemplu de utilizare a acestei sectiuni.

	Id	Nume Pacient	Nume Doctor	Data tratament	Data nasterii	Sex	Nume Sp
▶	1	Morar Anca	Popescu Dana	2010-07-20	1996-07-18	m	Spitalul de
	2	Morar Beni	Popescu Dana	2010-07-13	1983-07-06	m	Spitalul de
	5	Morar Gavriela	Maris Daniela	2010-07-13	1954-07-20	m	Spitalul G
	6	Morar Rodica	Maris Daniela	2010-07-22	2010-07-07	f	Spitalul G
*							

Figura 21. Cautare pacienti

La apasarea butonului “Afiseaza toti pacientii” vor aparea in grid-ul de vizualizare din nou toti pacientii din baza de date.

7.1.4. Sectiunea Radiografii

La selectarea unui pacient din grid-ul de vizualizare pacienti, sectiunea radiografii devine activa, continand radiografiile pentru pacientul respectiv.

Dupa cum se poate observa in Figura 22, in grid-ul de vizualizare radiografii se pot vedea toate radiografiile pacientului selectat, impreuna cu informatiile asociate.

The screenshot displays a web-based interface for patient management. At the top left, a table lists patients with columns for ID, name, last name, birth date, admission date, gender, and hospital. Row 7 is selected. To the right, a search form for the hospital name is visible, with a 'Cauta' button and an 'Afiseaza toti pacientii' button. Below the patient table, there are two main panels. The left panel, titled 'Adauga/Sterge/Editeaza Pacienti', has a sidebar with 'Adauga', 'Sterge', and 'Editeaza' options. The 'Sterge' option is active, showing fields for 'ID Pacient' (value: 7) and 'Nume Pacient' (value: Matei Rares), with a 'Sterge' button below. The right panel, titled 'Adauga/Sterge Radiografii', has a sidebar with 'Adauga' and 'Sterge' options. The 'Adauga' option is active, showing a 'Radiografii' table with columns for 'Id rad', 'Id pac', 'Nume rad', 'Interv op (luni)', and 'Tip rad'. The table contains four rows of radiograph data for patient ID 7. Below the table are buttons for 'Importa DCM' and 'Importa BMP', an 'Interval de la operatie' field with a unit selector set to 'luni', and an 'Adauga' button.

	6	7	8	9	10	11	12
	Morar Rodica	Matei Rares	Matei Tudor	Berca Radu	Berca Emanuela	Berca Teodora	Berca Alexia
	Maris Daniela	Maris Daniela	Maris Daniela	Matei Dan	Matei Dan	Matei Gabriela	Matei Gabriela
	2010-07-22	2010-07-14	2009-07-14	2010-02-16	2010-07-15	2010-04-12	2010-07-07
	2010-07-07	1995-07-20	2000-07-27	1973-07-19	1968-07-25	2002-07-24	2005-07-26
	f	m	m	m	f	f	f
	Spital	Spital	Spital	Spital	Spital	Spital	Spital

	Id rad	Id pac	Nume rad	Interv op (luni)	Tip rad
	7	7	20090520746501.dcm	0	dcr
	10	7	20090520748067.dcm	3	dcr
	11	7	200905181145371.dcm	2	dcr
	12	7	200905181147097.dcm	2	dcr

Figura 22. Activarea sectiunii radiografii

La selectarea unei radiografii, in panoul din dreapta jos, apare o imagine pentru previzualizarea imaginii radiografice selectate. La *dublu click* pe aceasta imagine, se deschide form-ul de radiografii.

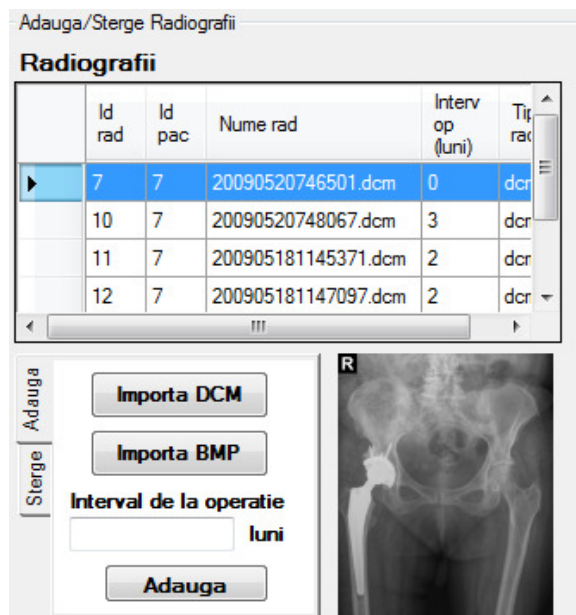


Figura 23. Selectarea unei radiografii

7.1.4.1. Importarea unei noi radiografii

Initial, in sectiunea radiografii, este activ tab-ul de adaugare radiografii. La apasarea butonului “Importa DCM”/”Importa BMP”, apare un *openDialogFile* pentru selectarea fisierului de importat, ca in figura 24

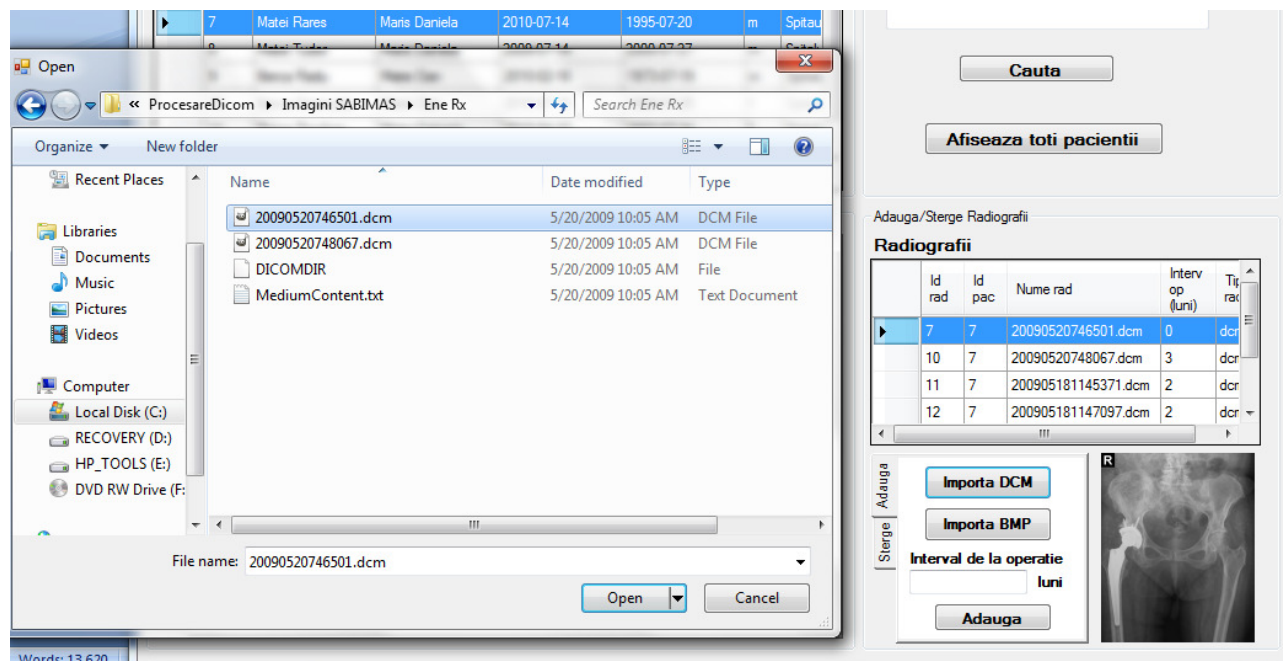


Figura 24. Importarea unui fisier care contine o radiografie

La apasarea butonului “*Adauga*”, radiografia este inserata in baza de date , apare un mesaj de confirmare asemanator cu cel din figura 17, iar radiografia poate fi observata in grid-ul de vizualizare radiografii.

7.1.4.2. Stergerea unei radiografii

La selectarea unei radiografii din grid-ul de vizualizare radiografii, in tab-ul de stergere, la apasarea butonului “*Sterge*”, pe ecran va apare mesajul de interogare a utilizatorului cu privire la stergerea radiografiei. Daca se apasa butonul “*Da*”, atunci apare un mesaj de confirmare asemanator celui din figura 17, iar radiografia este stearsa din baza de date si nu mai poate fi vizualizata in grid-ul de vizualizare pacienti

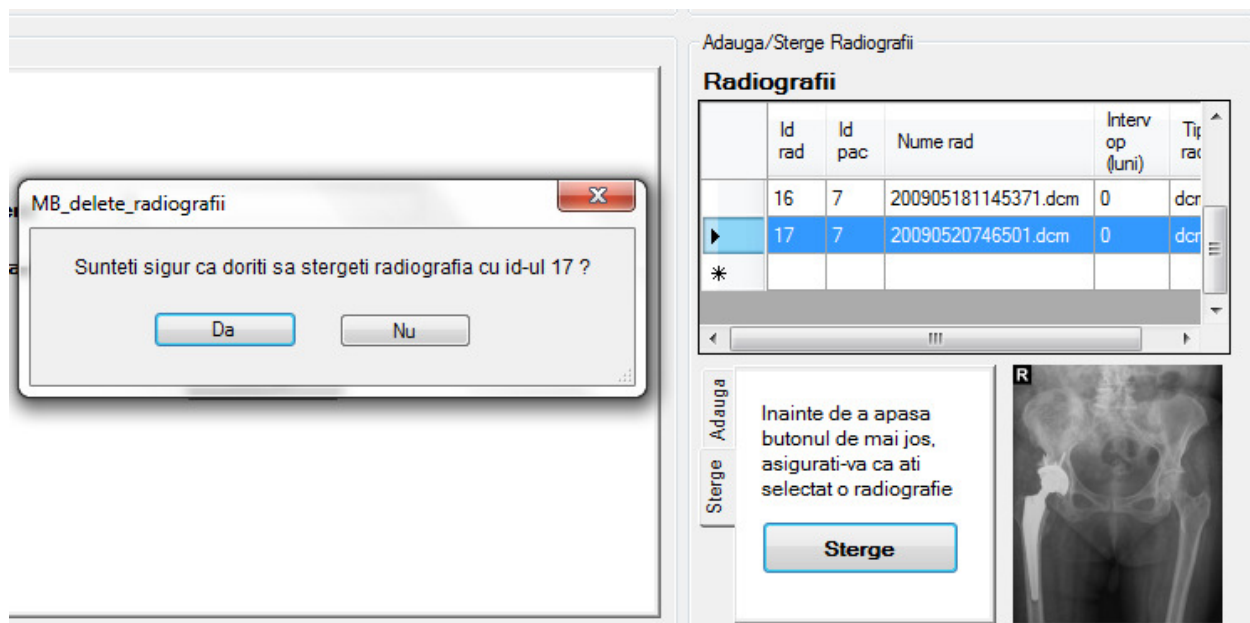


Figura 25. Stergerea unei radiografii

7.2. Form-ul radiografii

La selectarea unei radiografii din grid-ul de vizualizare radiografii si la *dublu click* pe imaginea de previzualizare, se deschide un nou form, numit Form-ul de radiografii.

7.2.1. Tab-ul de Informatii

Initial este activ tab-ul de Informatii, unde in partea stanga sunt afisate informatii despre pacient (extrase din baza de date), iar in partea dreapta este afisata imaginia radiografica initiala.

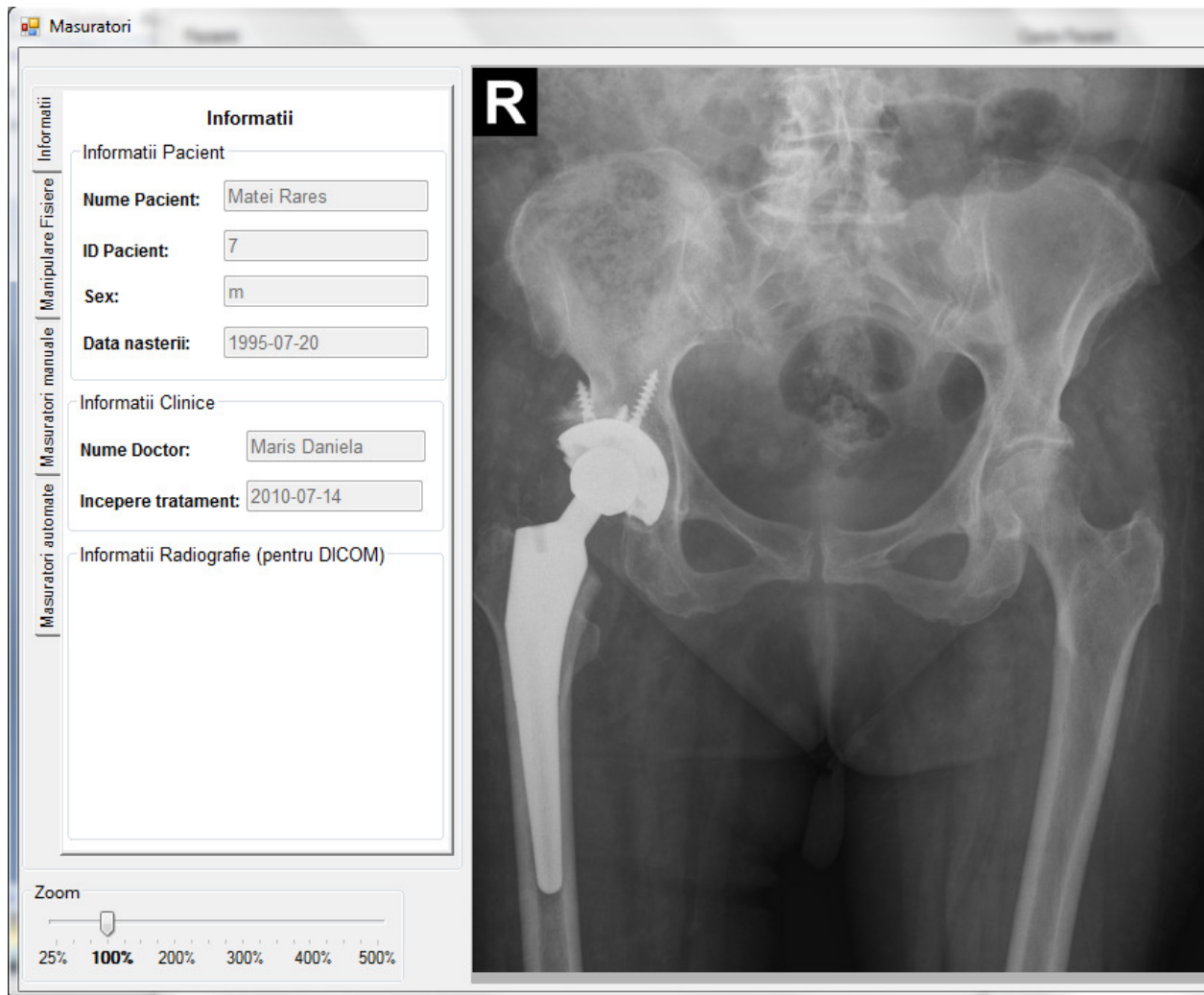


Figura 26. Tab-ul informatii

7.2.2. Tab-ul de Masuratori manuale

La intrarea in tab-ul de masuratori manuale, se afiseaza in imagine toate obiectele din baza de date, din tabelele linii, cercuri si dreptunghiuri.

Exista trei grid-uri de vizualizare, cate unul pentru fiecare tip de obiect.

Fiecare grid de vizualizare are asociate trei butoane, unul pentru adaugare de obiect nou (care are asociata o imagine “plus”), unul pentru editarea obiectului selectat (care are asociata o imagine cu un creion) si unul pentru stergerea obiectului selectat (care are asociata o imagine cu un “x”).

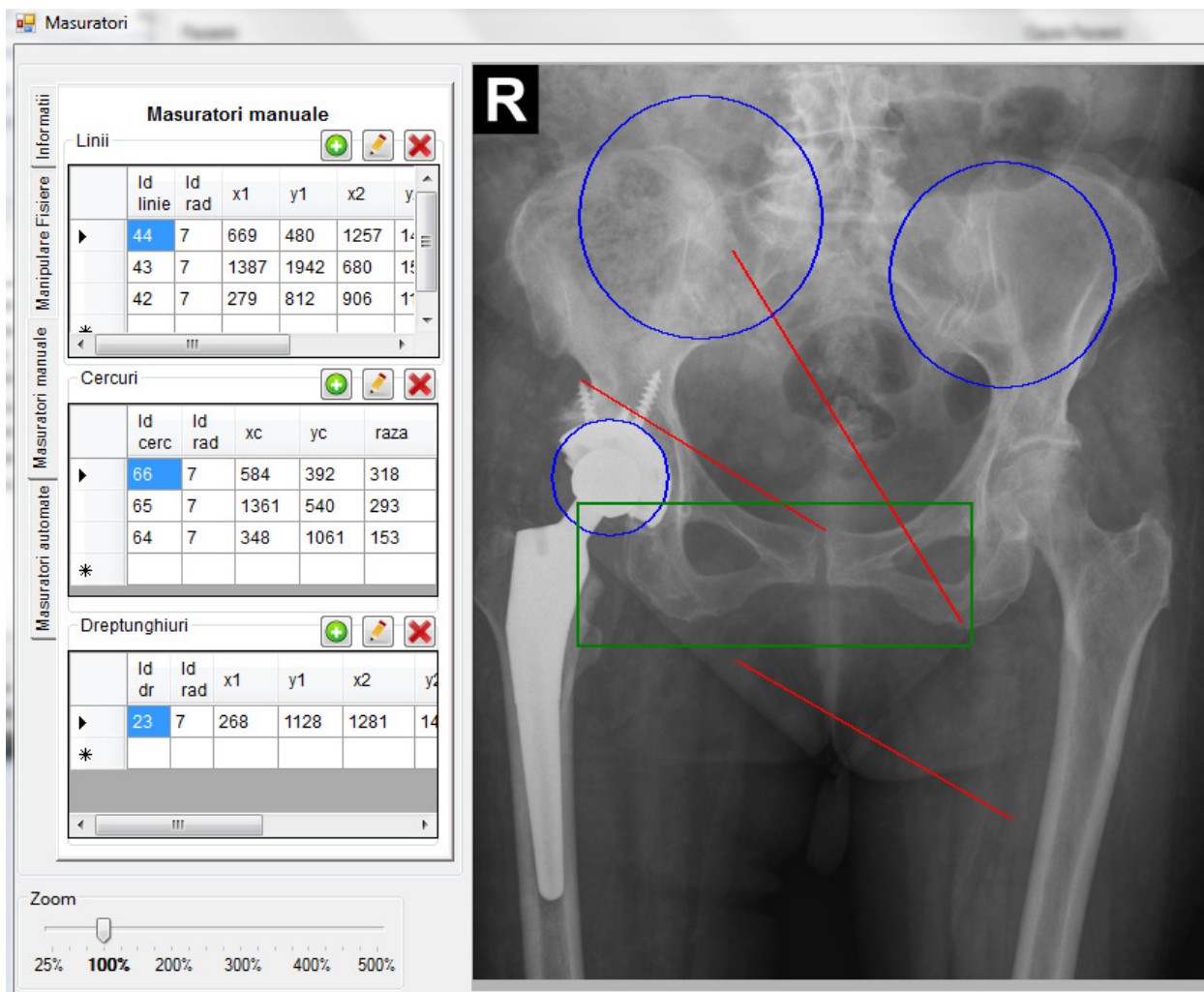


Figura 27. Tab-ul de Masuratori Manuale

7.2.2.1. Adaugarea unui obiect nou

Presupunem ca dorim adaugarea unei linii noi in imagine, de exemplu care sa aiba capetele in doua colturi ale imaginii. Pentru aceasta se apasa butonul de Adaugare linie noua, corespunzator grid-ului de vizualizare linii. Dupa ce a fost apasat butonul, cu *drag* de mouse se poate trasa noua linie. Figura 28 prezinta rezultatul acestei operatii:

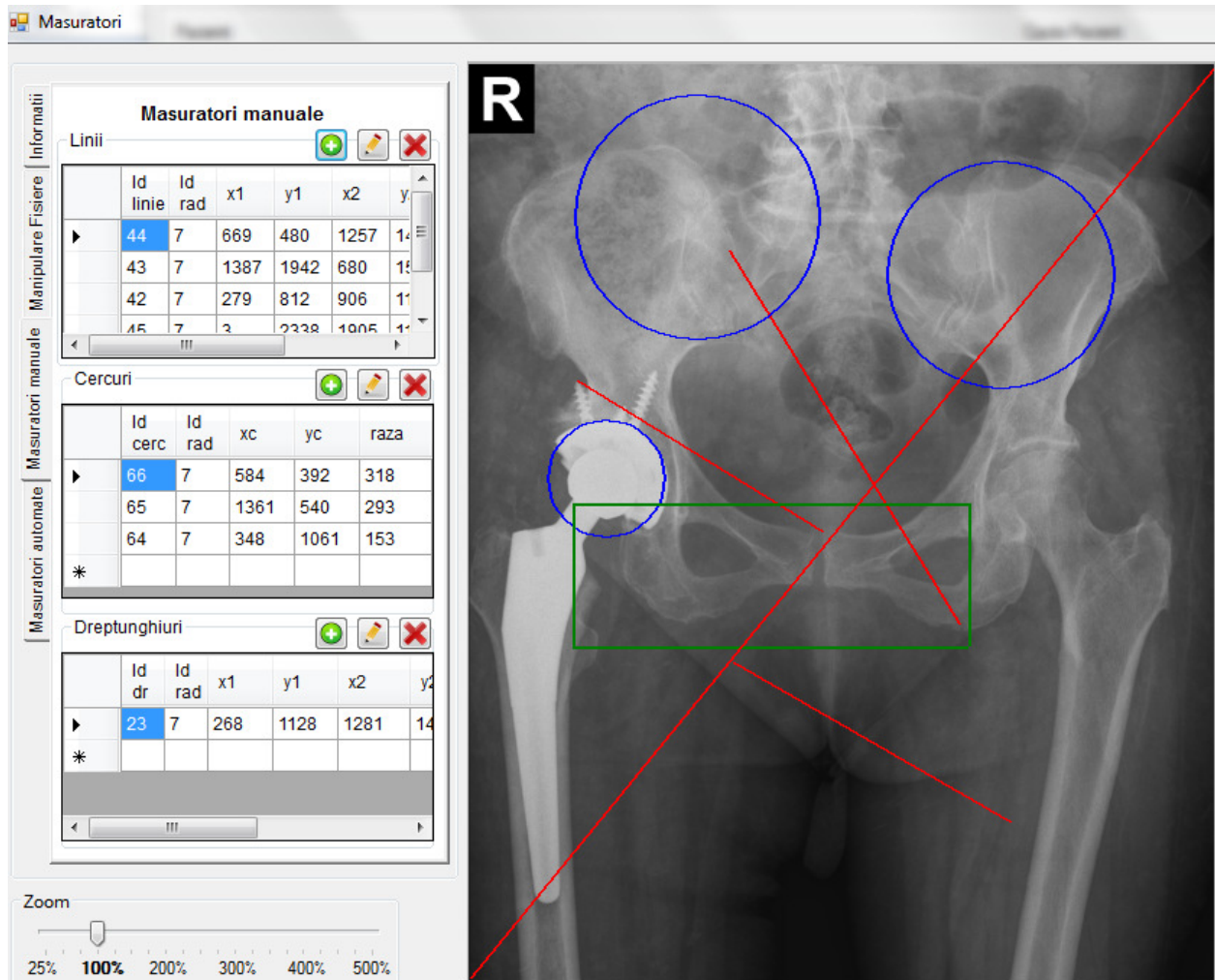


Figura 28. Adaugarea unei linii noi

7.2.2.2. Editarea unui obiect selectat

Pentru a edita un obiect, acesta trebuie anterior selectat. Presupunem ca dorim sa selectam cercul din stanga sus pentru a-l micsora raza.

La selectare din grid-ul de vizualizare cercuri, acesta va fi scos in evidenta prin desenarea de dreptunghiuri umplute in dreptul centrului si al celor puncte cardinal, ca in Figura 29:

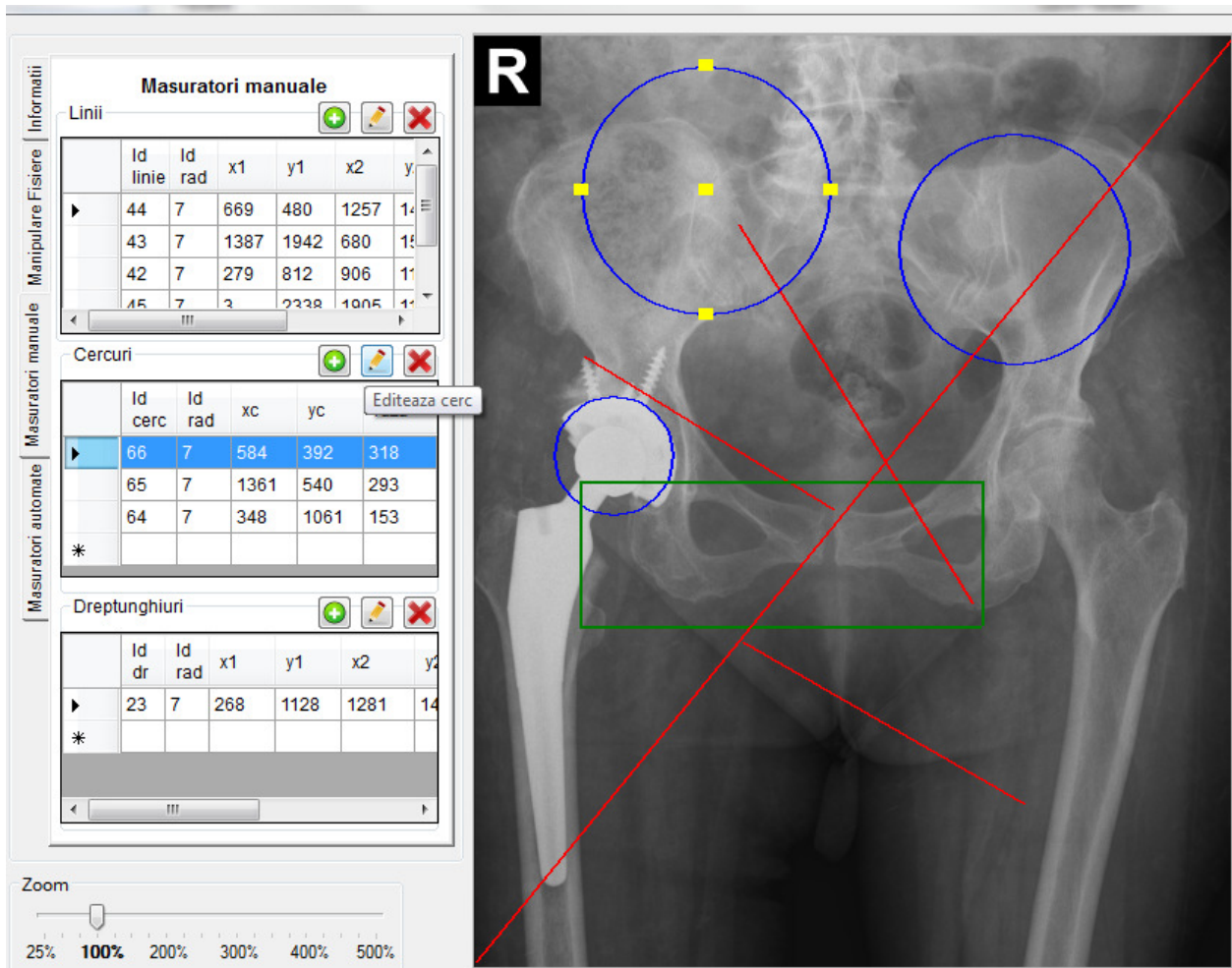


Figura 29. Selectarea unui cerc in vederea editarii

La apasarea butonului de Editare si prin *mouse dragged*, se poate modifica raza cercului, ca in figura 30. Modificarile se regasesc si in grid-ul de vizualizare cercuri, si in baza de date.

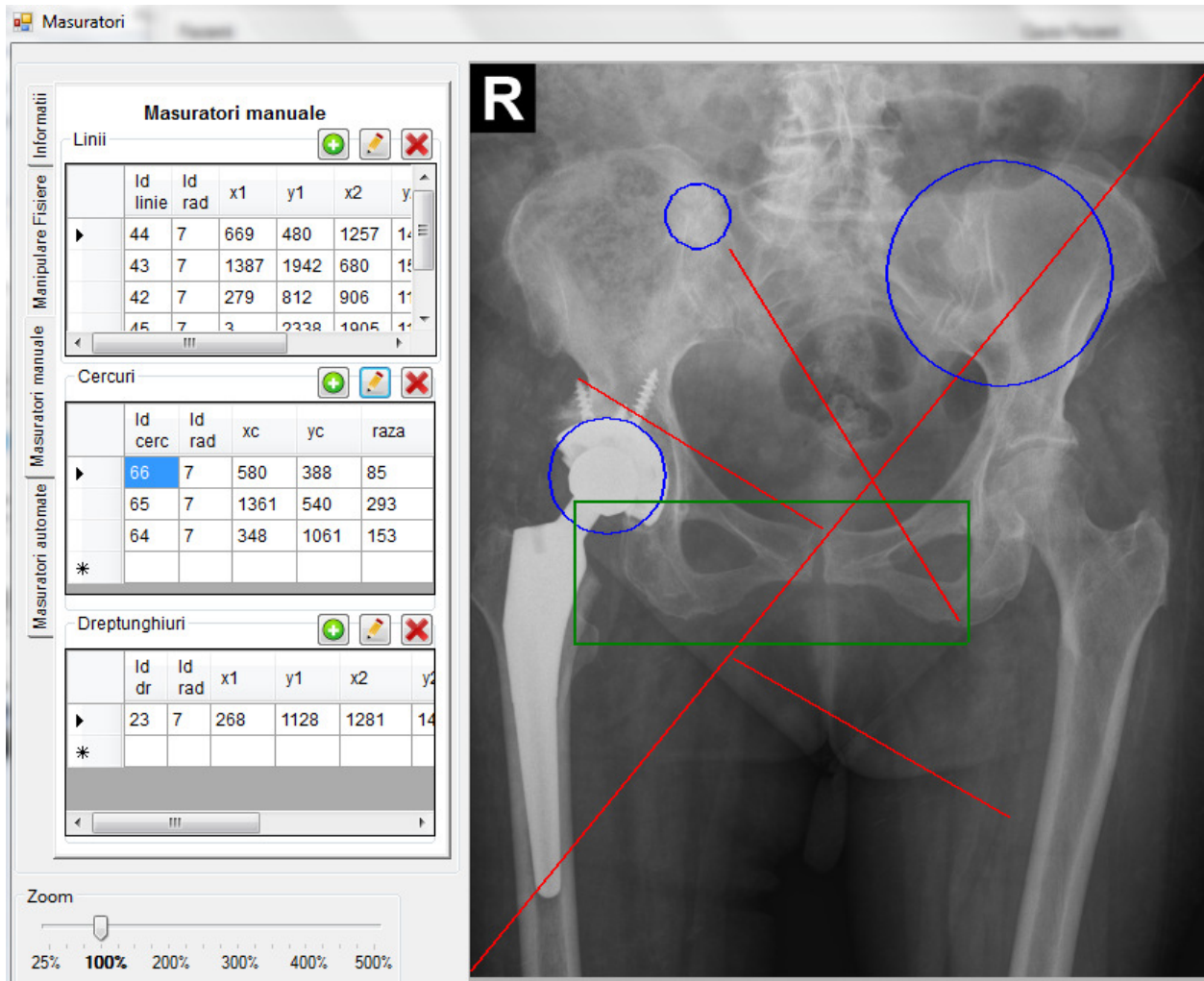


Figura 30. Editarea unui cerc

7.2.2.3. Stergerea unui obiect

Pentru stergerea unui obiect, este nevoie ca acesta sa fie anterior selectat din grid-ul de vizualizare obiecte. La apasarea butonului de Stergere, acest obiect va fi sters din imagine, din baza de date si din grid-ul de vizualizare obiecte

7.2.3. Tab-ul masuratori automate

La selectarea tab-ului de masuratori automate, va aparea o fereastră impartita in doua, ca in Figura 31:

- Componentele in text ale parametrilor masuratorilor automate

- Obiectele ce reprezinta parametrii masuratorilor automate, extrase din baza de date din tabelul masuratori_automate, sau calculate pentru prima oara, daca nu au mai fost procesate, desenate in figura din partea dreapta

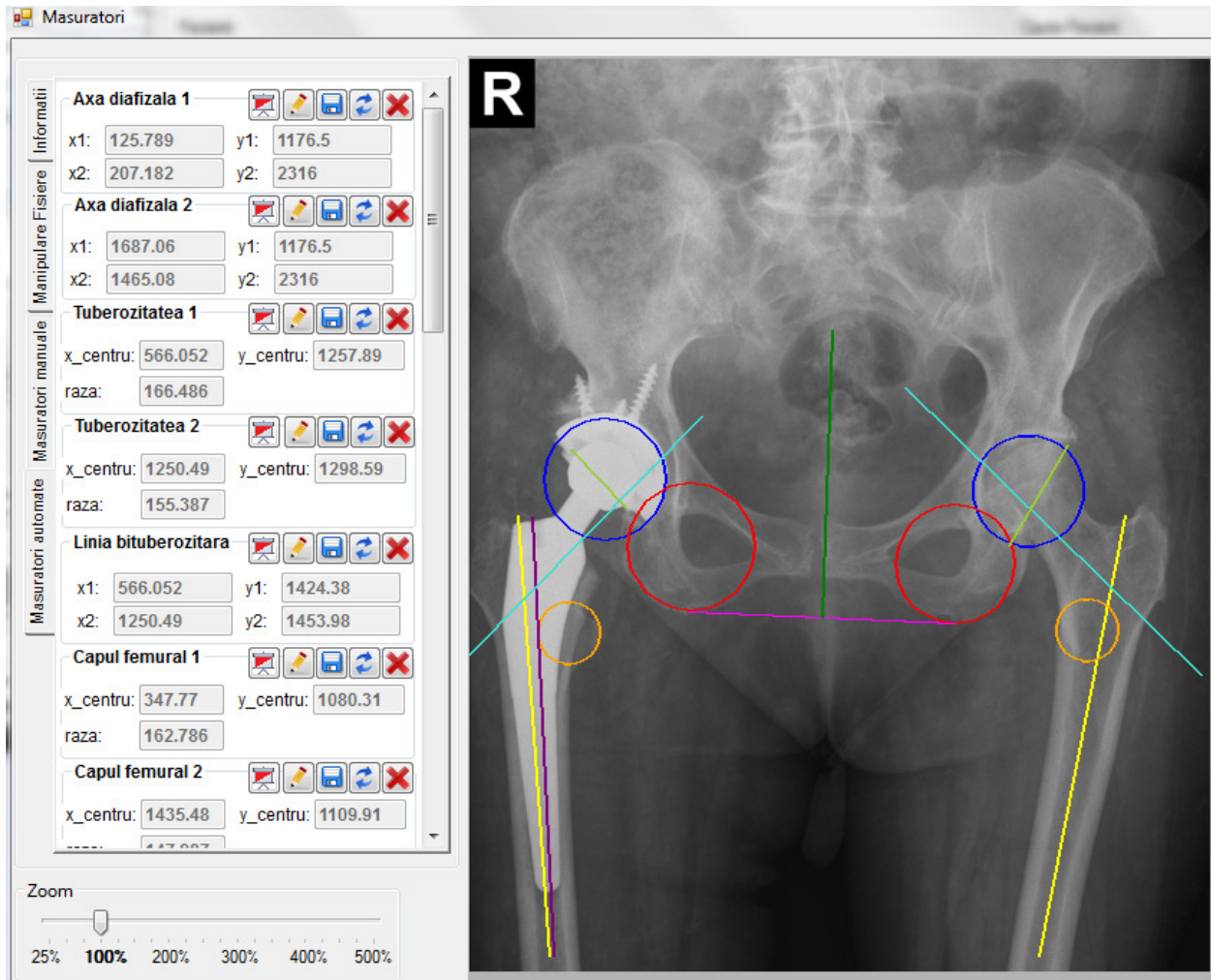


Figura 31. Masuratorile automate

Dupa cum se poate observa si in figura, fiecare parametru are asociate 5 butoane:

- “Afiseaza/ascunde” parametrul selectat – afiseaza sau nu primitive grafica (linie, cerc) care corespunde parametrului respectiv
- “Editeaza” parametrul selectat – exista doua posibilitati de editare:
 - o Modificarea textbox-urilor din partea stanga
 - o Editarea in modul vizual, ca in tab-ul de masuratori automate
- “Salveaza” caracteristicile parametrului curent – salveaza modificarile facute la apasarea butonului de Editare

- *“Recalculeaza”* parametrul curent – recalculeaza parametrul curent, in functie de parametrii anterior determinati (sau editati) de care depinde
- *“Sterge”* parametrul curent – sterge din baza de date si de pe ecran parametrul curent

7.2.4. Tab-ul Manipulare fisiere

La activarea tab-ului de manipulare fisiere, va apare urmatoarea fereastra:

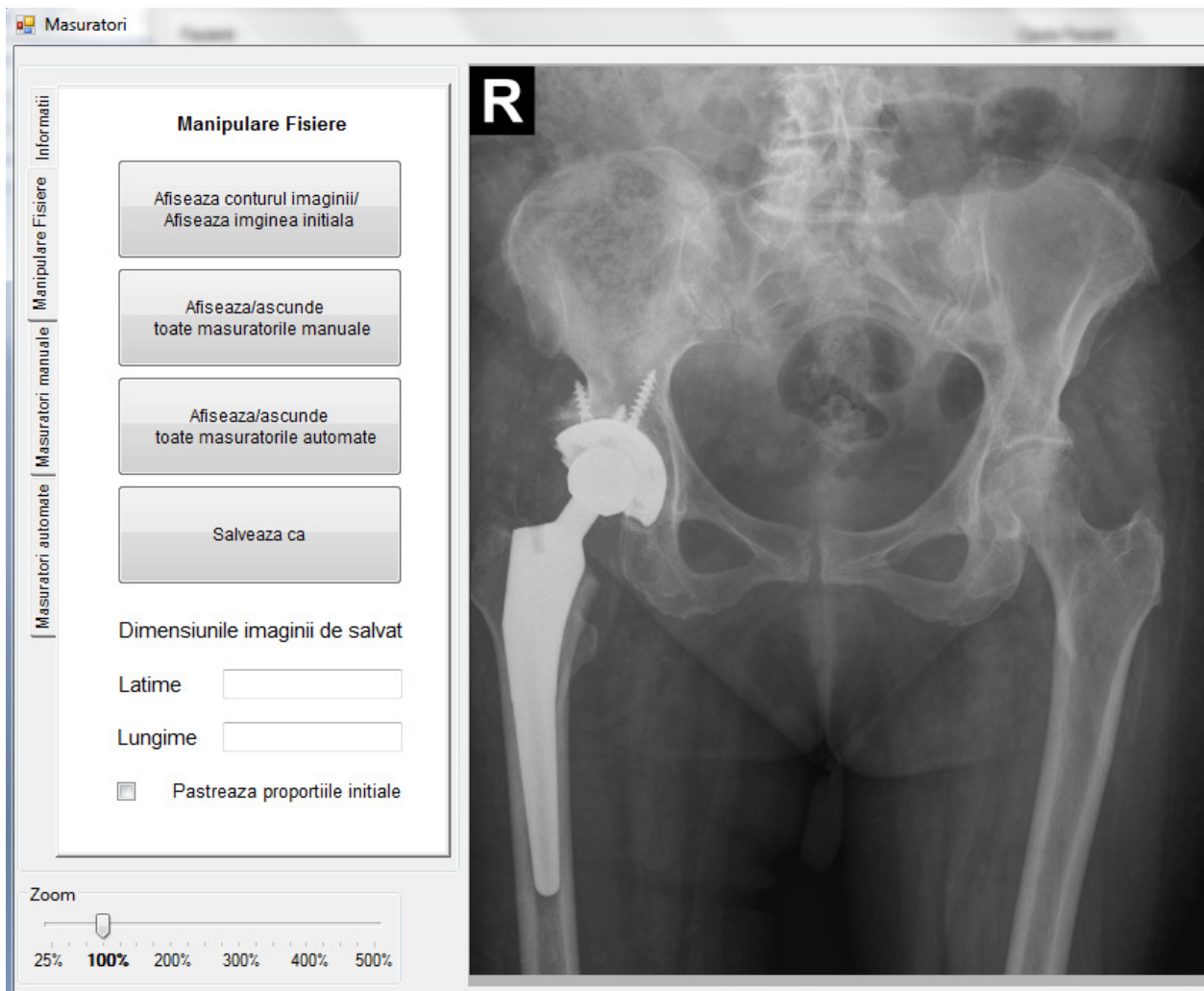


Figura 32. Manipularea fisierelor

Aici exista urmatoarele posibilitati, in functie de butoanele din tab:

- *“Afiseaza conturul imaginii/Afiseaza imaginea initiala”*
- *“Afiseaza/ascunde masuratorile manuale”*
- *“Afiseaza/ascunde masuratorile automate”*

Figura 33 prezinta conturul imaginii si masuratorile manuale:

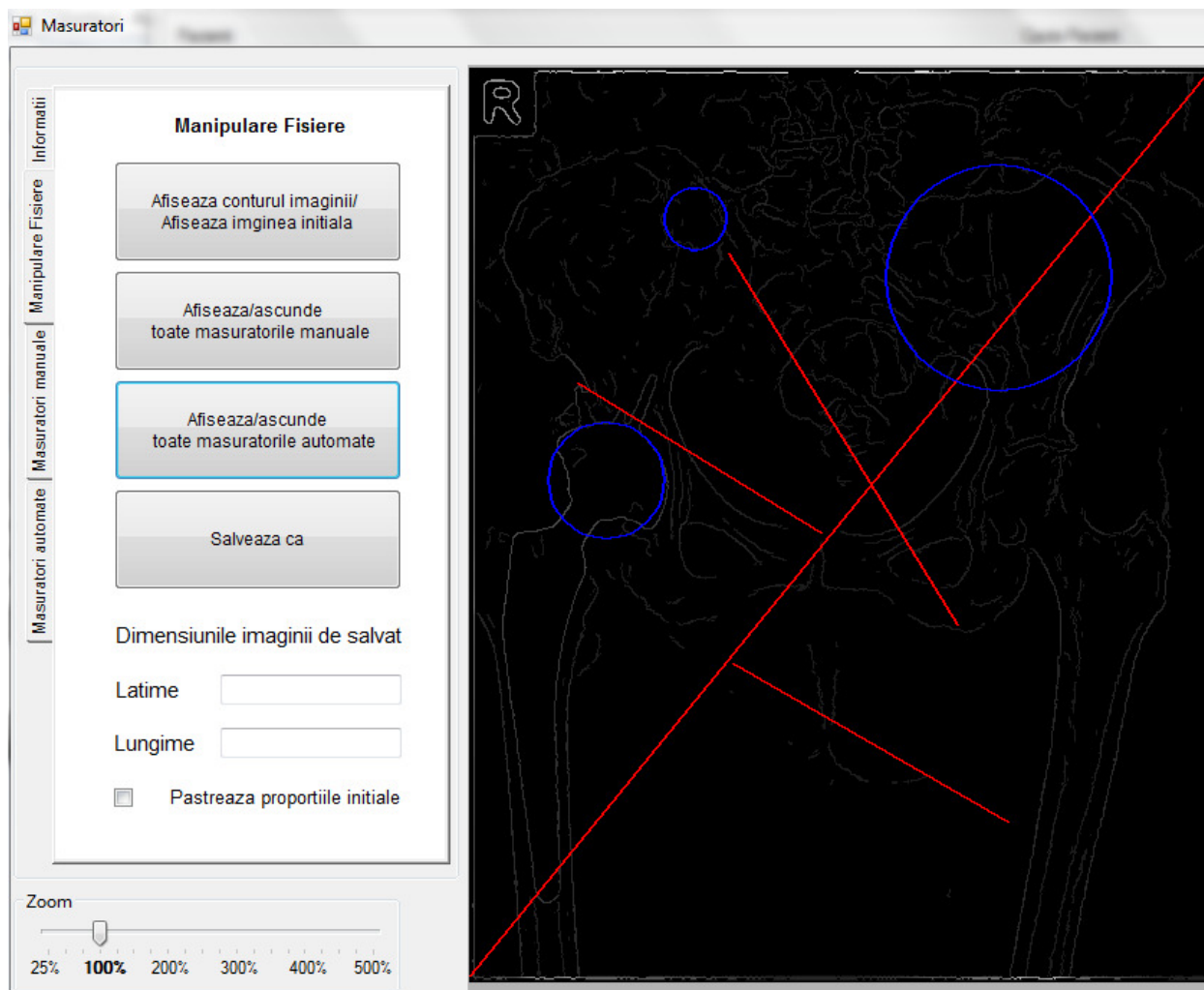


Figura 33. Afisarea conturului imaginii si a masuratorilor manuale

Butonul “*Salveaza ca*” salveaza intr-un fisier ales prin intermediul unui *saveDialogFile*, imaginea din partea dreapta, in functie de latimea si lungimea scrise in cele doua *textbox*-uri din partea din stanga jos (cu posibilitatea pastrarii proportiilor).

7.2.5. Zoom

Aplicatia ofera posibilitatea de scalare a imaginii, de la 25% pana la 500%, prin selectarea de catre utilizator a factorului de zoom (partea stanga-jos), in oricare din taburile Manipulare fisiere, Masuratori manuale sau Masuratori automate.

In figura 34 este afisata radiografia scalata la 275%.

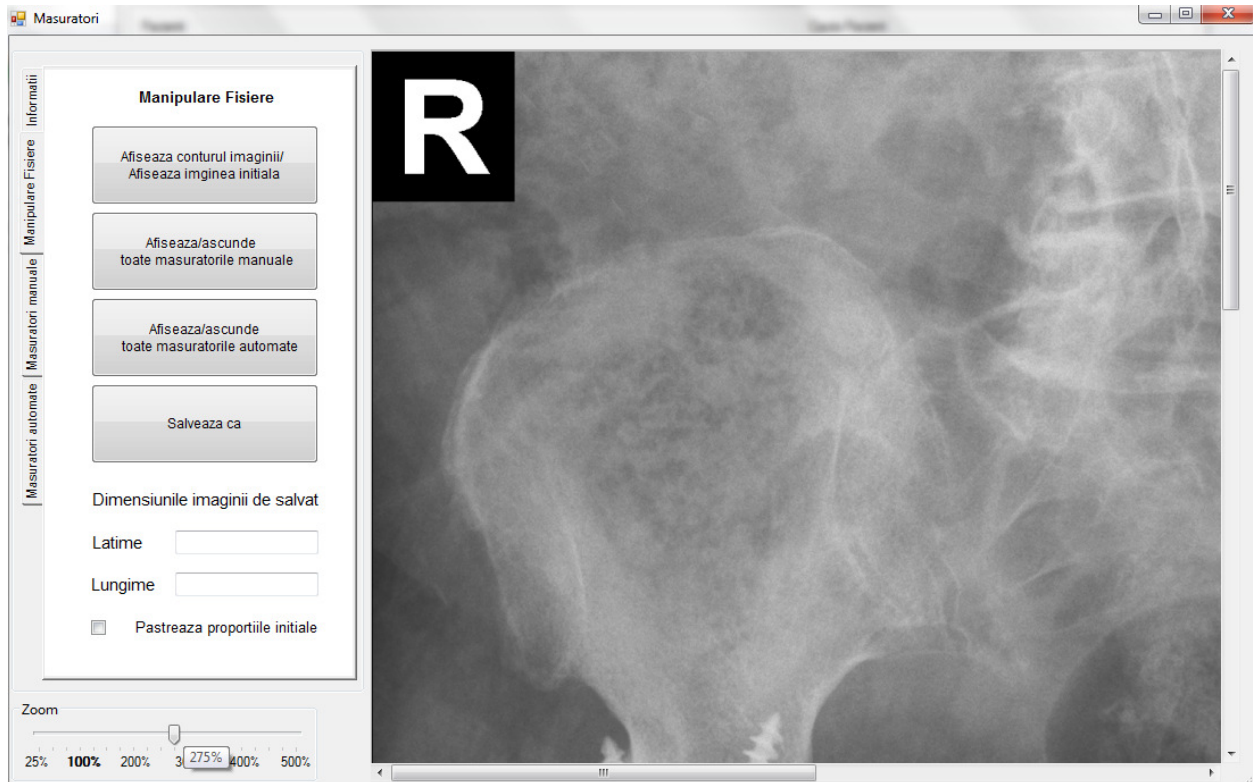


Figura 34. Scalarea imaginii la 275%

8. EXPERIMENTAREA SISTEMULUI IN PRACTICA MEDICALA

Studiul descris in acest capitol a fost realizat de partenerii de la SCUB SI UMF.

Numarul artroplastiiilor totale de sold este in continua crestere de la an la an cu precadere in randul persoanelor tinere, active.

In ideea de a creste durata de viata a unei proteze totale necimentate de sold, trebuie sa obtinem o cat mai buna adaptare a implantului la os.

In acest scop, ne propunem cu ajutorul programului creat de UPB, sa utilizam modele bi si tridimensionale ale articulatiei soldului generate pe baza radiografiilor standard si a tomografiilor computerizate.

Rezultatul ar consta in producerea unui implant care va avea o mobilitate minima si un transfer optim al sarcinilor de incarcare la nivelul osului.

Aplicatia experimentata cuprinde interfata cu utilizatorul-introducerea datelor legate de pacient, analiza sistemului osos si articular al soldului si efectuarea de masuratori pre si postoperatorii pe baza radiografiilor obtinute.

Imediat dupa instalarea programului si deschiderea acestuia apare o interfata ce ii permite utilizatorului sa introduca date despre pacient. Interfata este usor de folosit fiind accesibila oricarei persoane, nu necesita o pregatire speciala si nici timp prea mult.

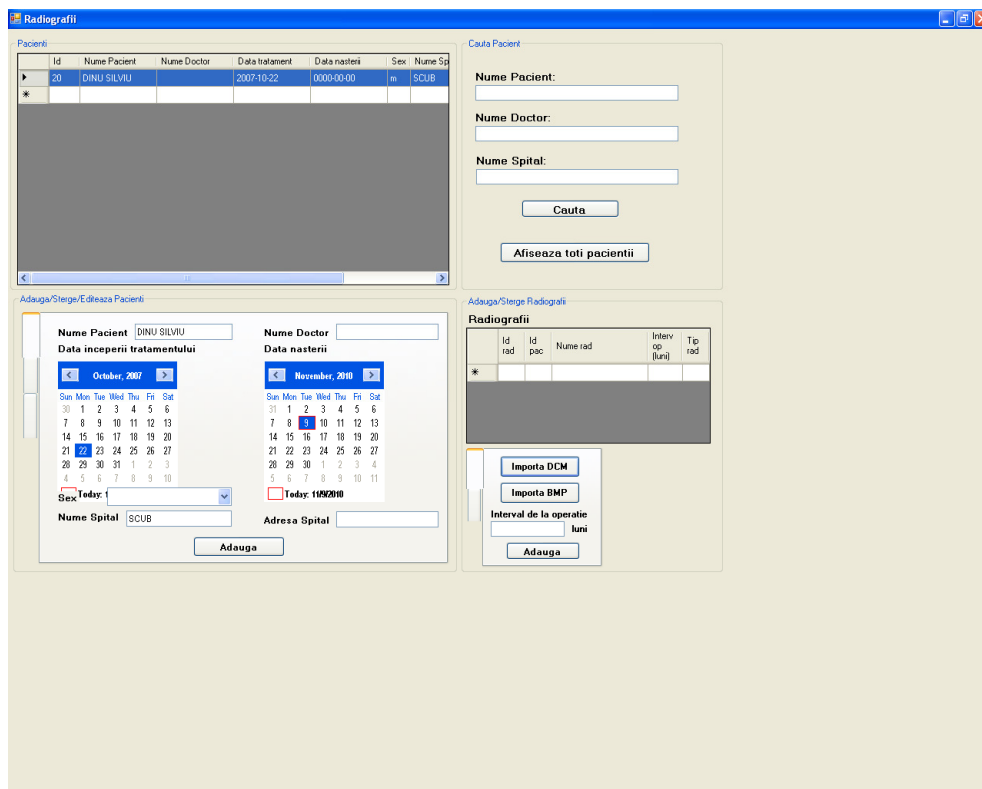


Figura 35. Interfata de introducere date pacienti

Datele despre pacient, situate in partea stanga a ecranului, pot fi introduce pentru prima data sau pot fi modificate ca urmare a unor schimbari ce s-au petrecut de la ultima introducere a datelor cu ajutorul butoanelor pentru editare, stergere, adaugare.

Date generale despre pacient contin: nume, prenume, data nasterii, sex. Poate ar fi de adaugat CNP-ul pacientului. Tot in tabelul din partea stanga se introduc date care se refera la numele medicului si spitalului, precum si la data curenta si data inceperii tratamentului.

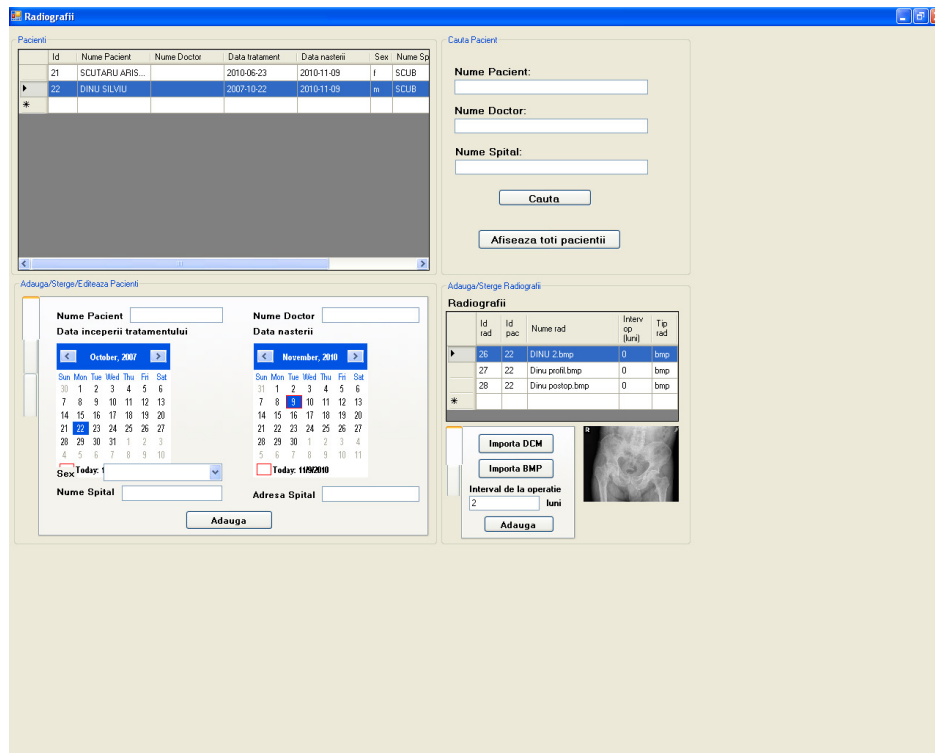


Figura 36. Introducerea datelor despre medic si spital

In partea din dreapta a ecranului, dupa cum se observa, am introdus date de imagistica. Astfel, pentru fiecare pacient in parte am introdus radiografiile pre si postoperatorii, in incidentele antero-posterioara si laterala. Programul ne permite importul radiografiilor atat sub forma fisierelor format dicom cat si bitmap. Se poate preciza si intervalul scurs de la data operatiei.

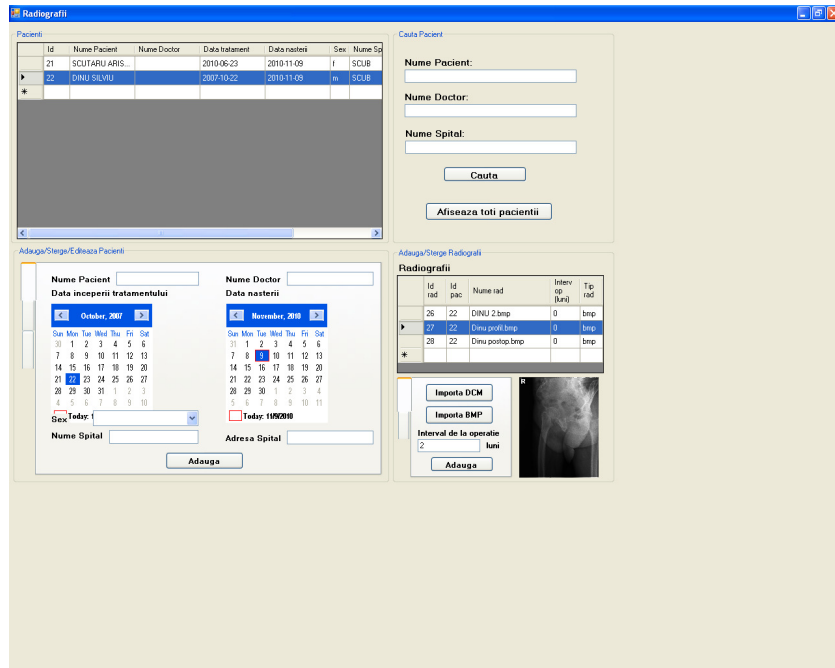


Figura 36. Importarea fisierelor tip bmp; Rx pre si postoperator

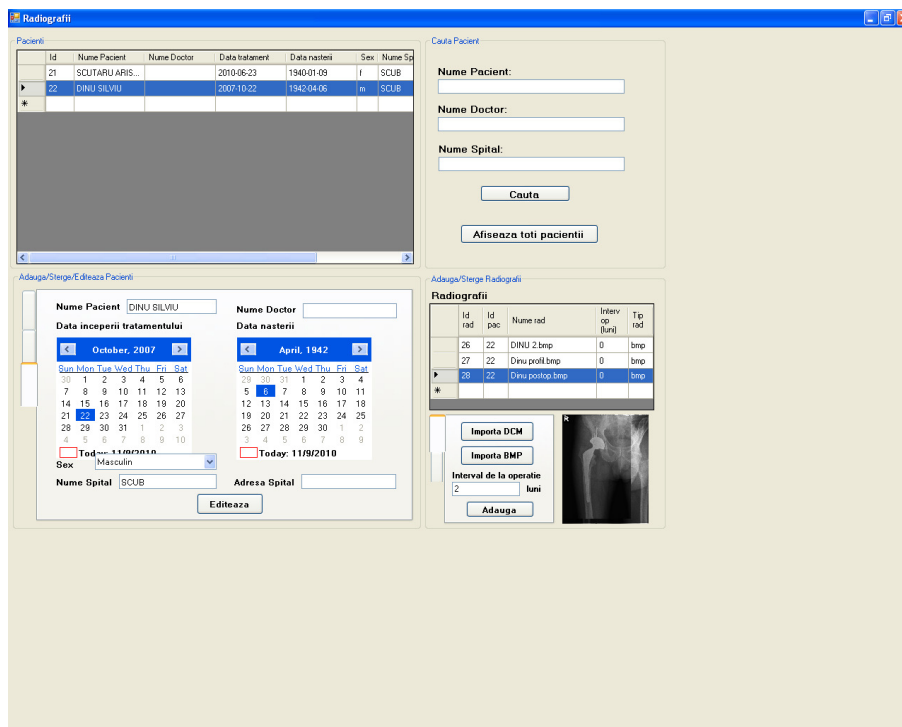


Figura 37. Importarea fisierelor tip DICOM

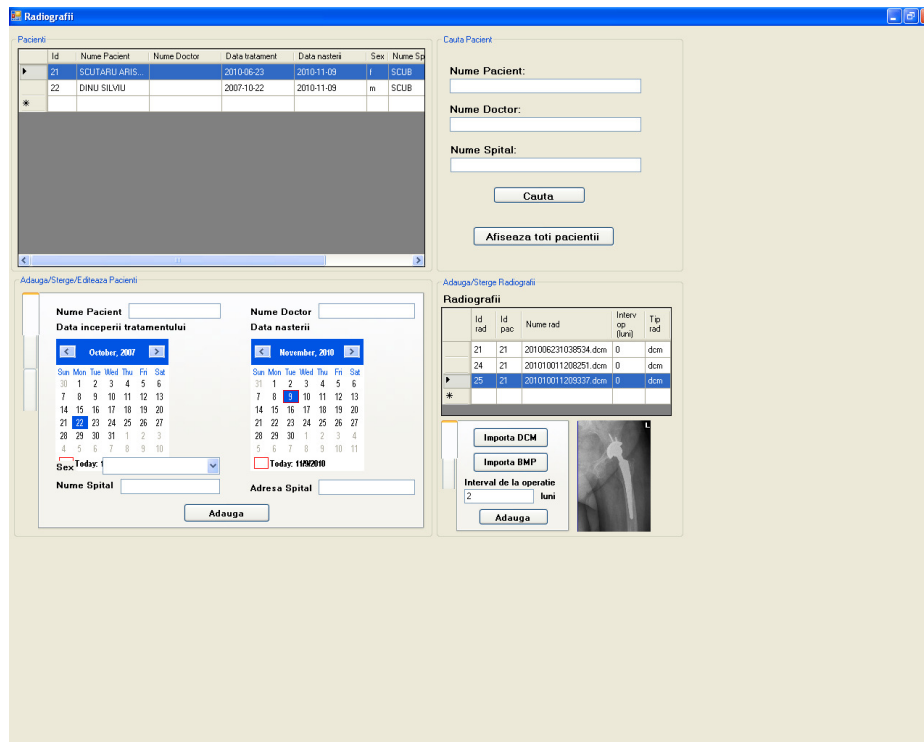


Figura 38.

In continuare, programul permite deschiderea unei noi ferestre dedicate masuratorilor radiologice, nu inainte de a avea acces la cateva informatii clinice, despre radiografii (in cazul fisierelor dicom), precum si la cateva informatii despre pacientul caruia apartine radiografia pre sau postoperatorie.

Cu ajutorul unor butoane de comanda putem vizualiza numai conturul imaginii, sau putem accesa masuratorile manuale sau automate.

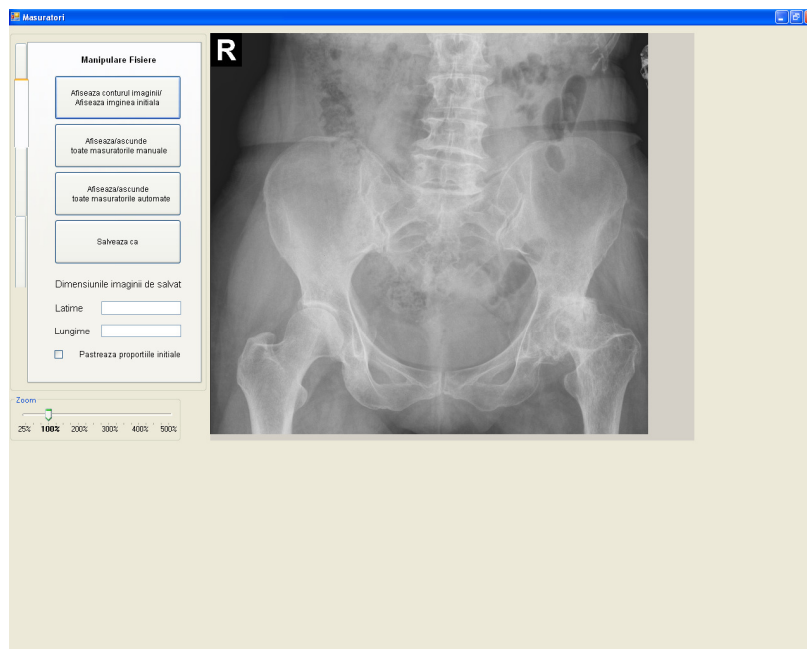
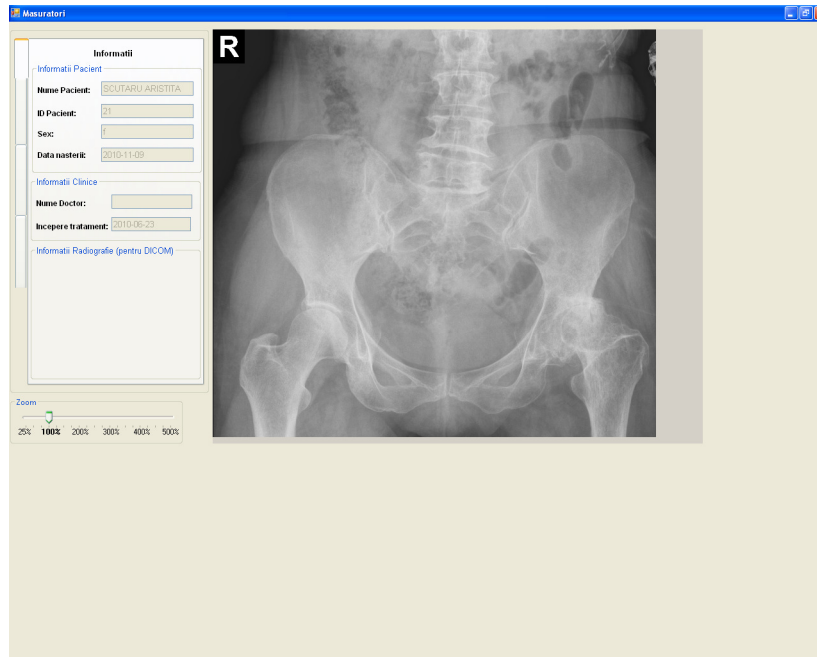


Figura 39. Vizualizarea conturilor din imagine si a masuratorilor

In cadrul masuratorilor manuale, putem folosi o serie de elemente geometrice care vor defini repere osoase, unghiuri, arii utile in stabilirea dimensiunilor cat mai corecte ale impantului.

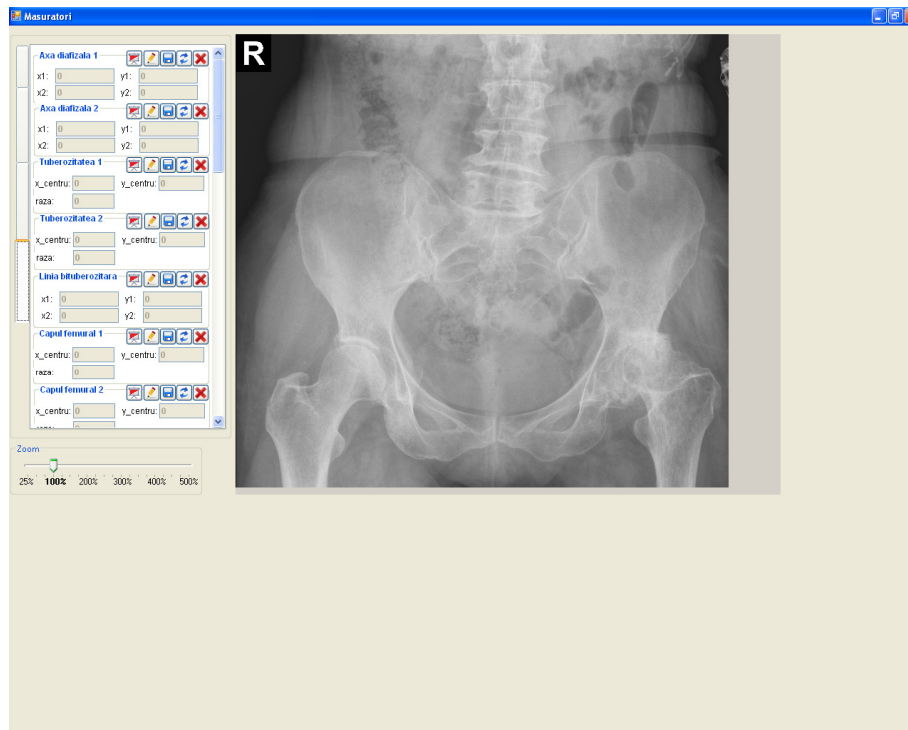
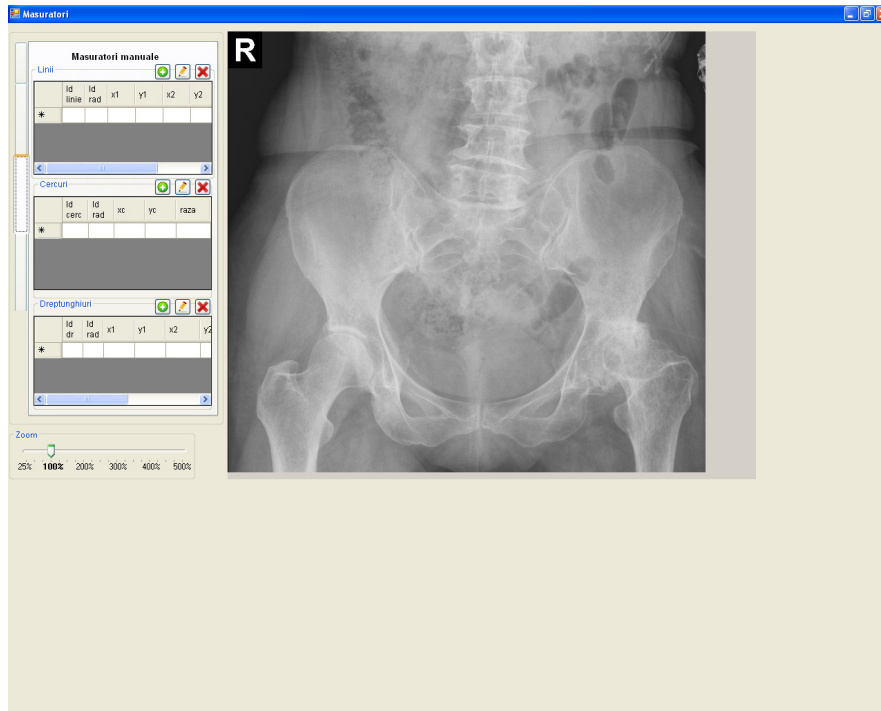


Figura 40. Efectuarea masuratorilor manuale

Tabul “Masuratori automate” calculeaza urmatoorii parametrii: axa diafizara, linia bituberozitară, diametrul capului femural, axa protezei, deviatia protezei, axul capului femoral, unghiul acetabular, axa colului femural si unghiul cervico-diafizar.

Aceste masuratori se efectueaza atat pe radiografia pre, cat si postoperatorie.

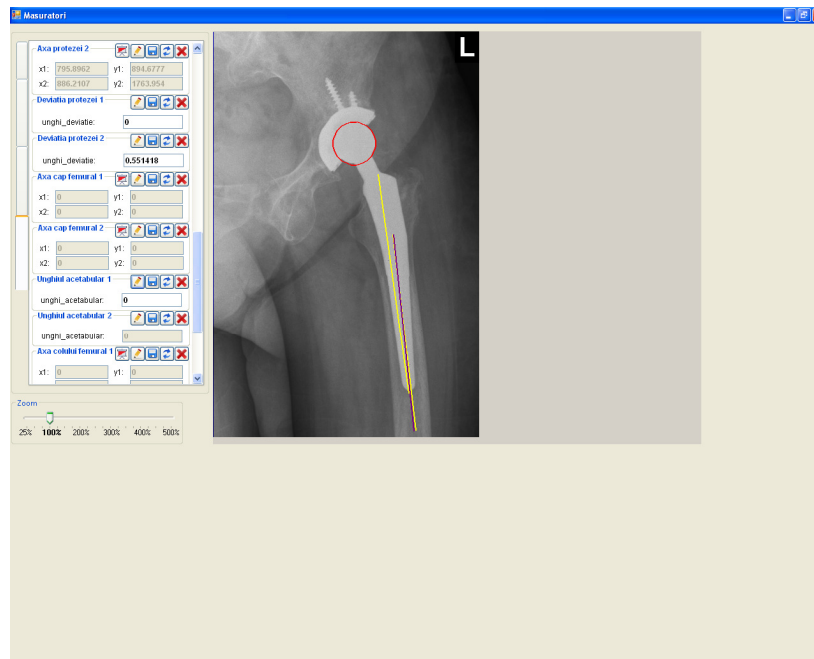


Figura 41. Masuratori pre si post operatorii

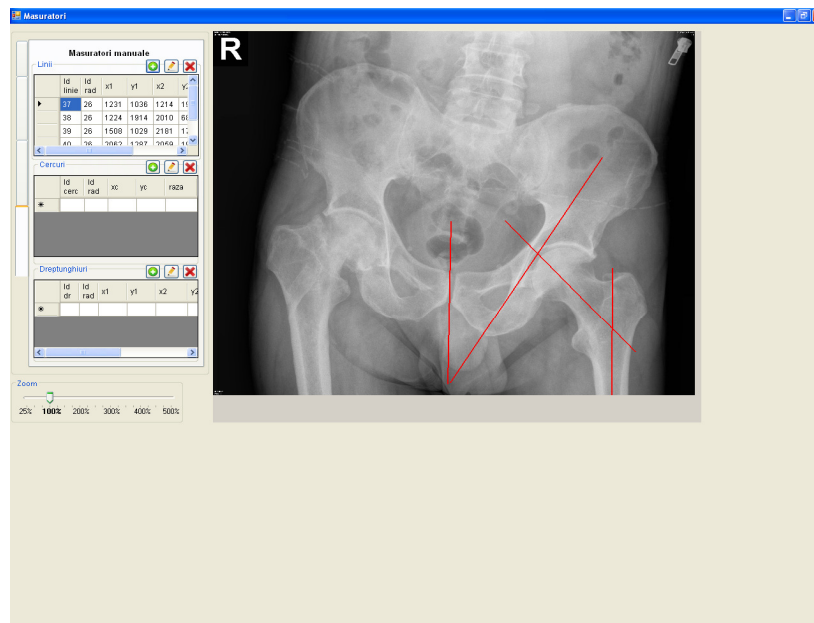


Figura 42. Masuratori manuale la nivelul soldului sanatos

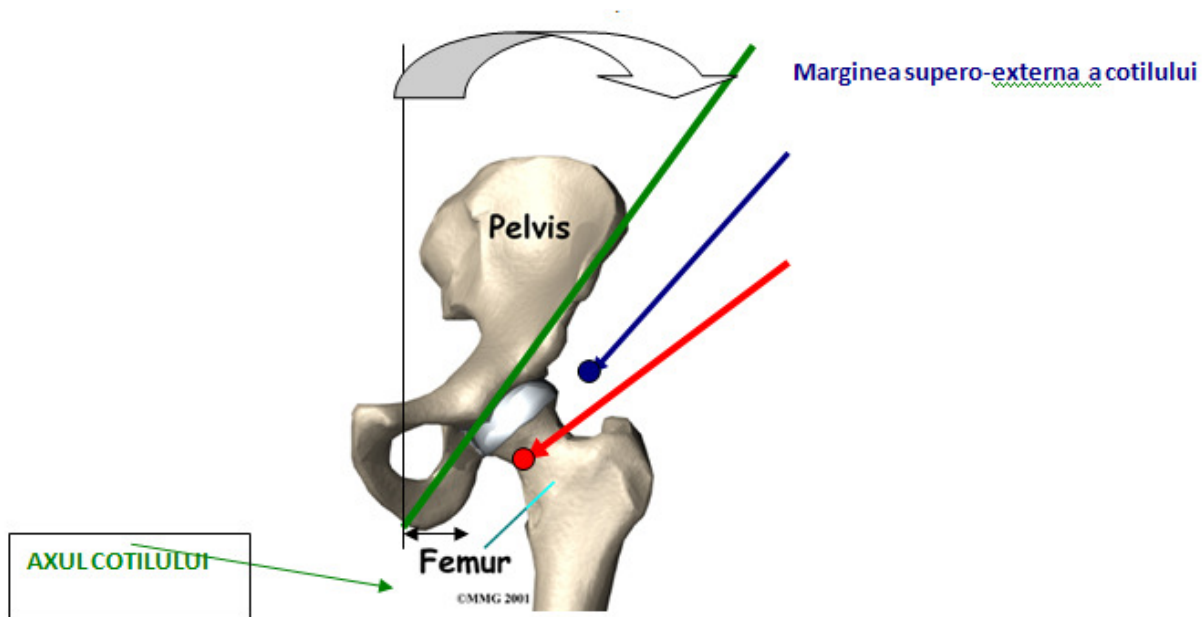


Figura 43. Reperete osoase si liniile folosite la calcularea parametrilor doriti.

Exista , pentru evaluarea echilibrului bazinului si a egalitatii (sau inegalitatii) de lungime a membrului inferioare, o serie de linii care trebuie, daca bazinul este normal, sa fie paralele intre ele, si perpendiculare pe verticala:

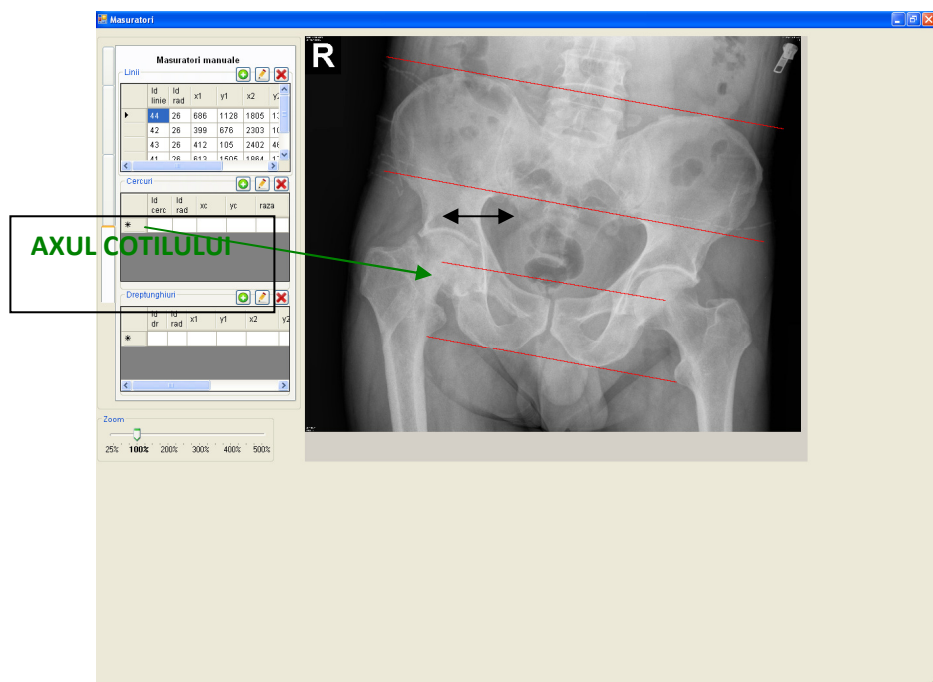


Figura 44.

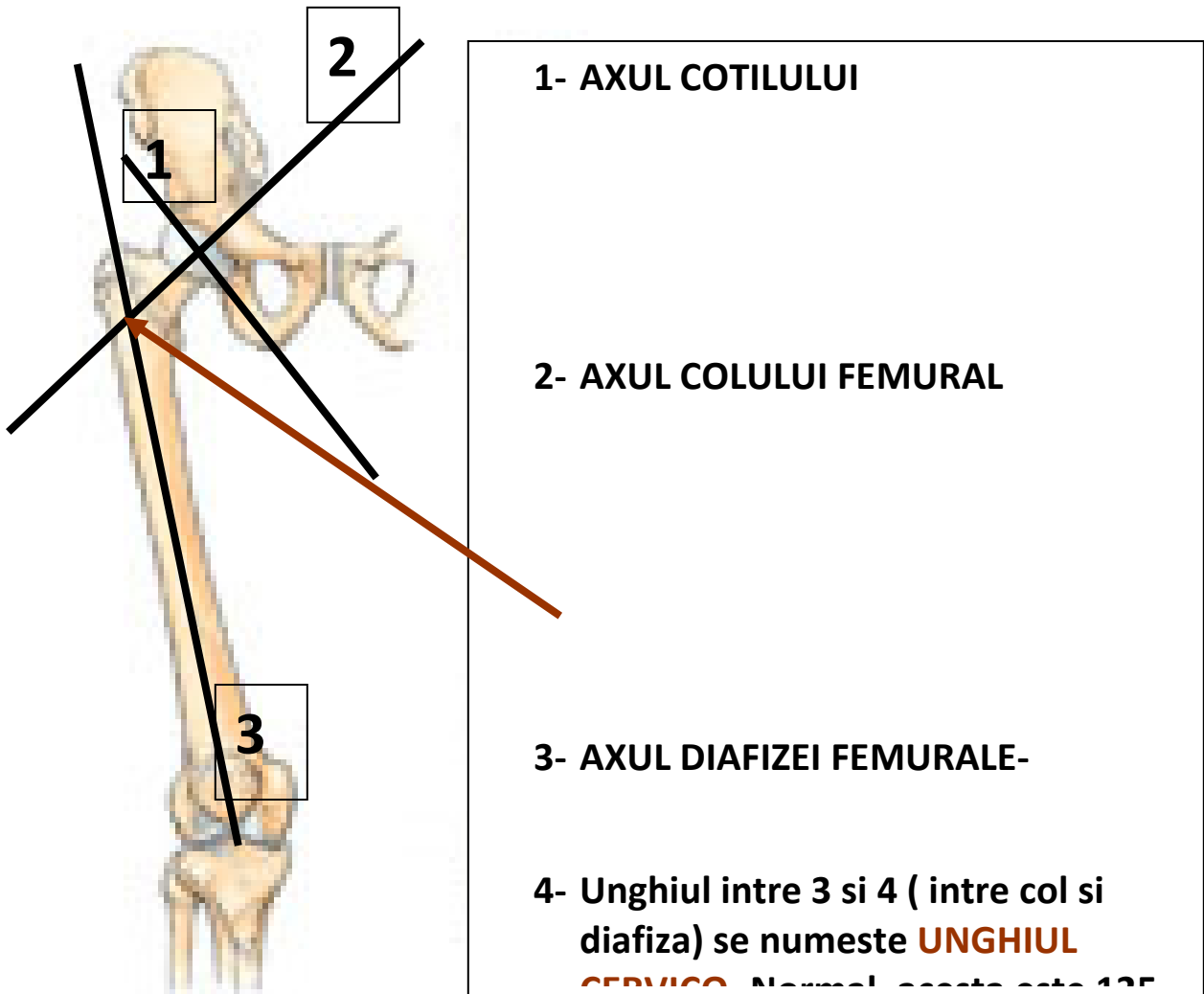


Figura 45.

4- LINIA BI-TUBEROZITARA- care trece tangent la tuberozitatile ischiadice- sunt punctele cele mai joase ale oaselor bazinului. In mod normal, LINIA BI-TUBEROZITARA TRECE PRIN TROHANTERELE MICI (de ambele parti) !!!, aceasta insemnand ca membrele inferioare sunt egale in lungime. Exista cazuri cand , conformational, aceasta linie nu trece prin trohantere, ci mai sus sau maijos, dar trece LA ACELASI NIVEL prin ambele femure, altfel, picioarele sunt inegale. Atunci avem doua linii—una care trece prin micile trohantere si una care trece tangent la tuberozitati. Acestea, in mod normal, SUNT PARALELE.

5- LINIA BI-SPINOASA- care trece prin spinele iliace antero-superioare- sunt punctele palpabile anterior de la nivelul oaselor bazinului

6- LINIA BI-CRESTA- care trece tangent la crestele iliace, punctele palpabile lateral ale oaselor bazinului

7- Linia care uneste centrele capetelor femurale.

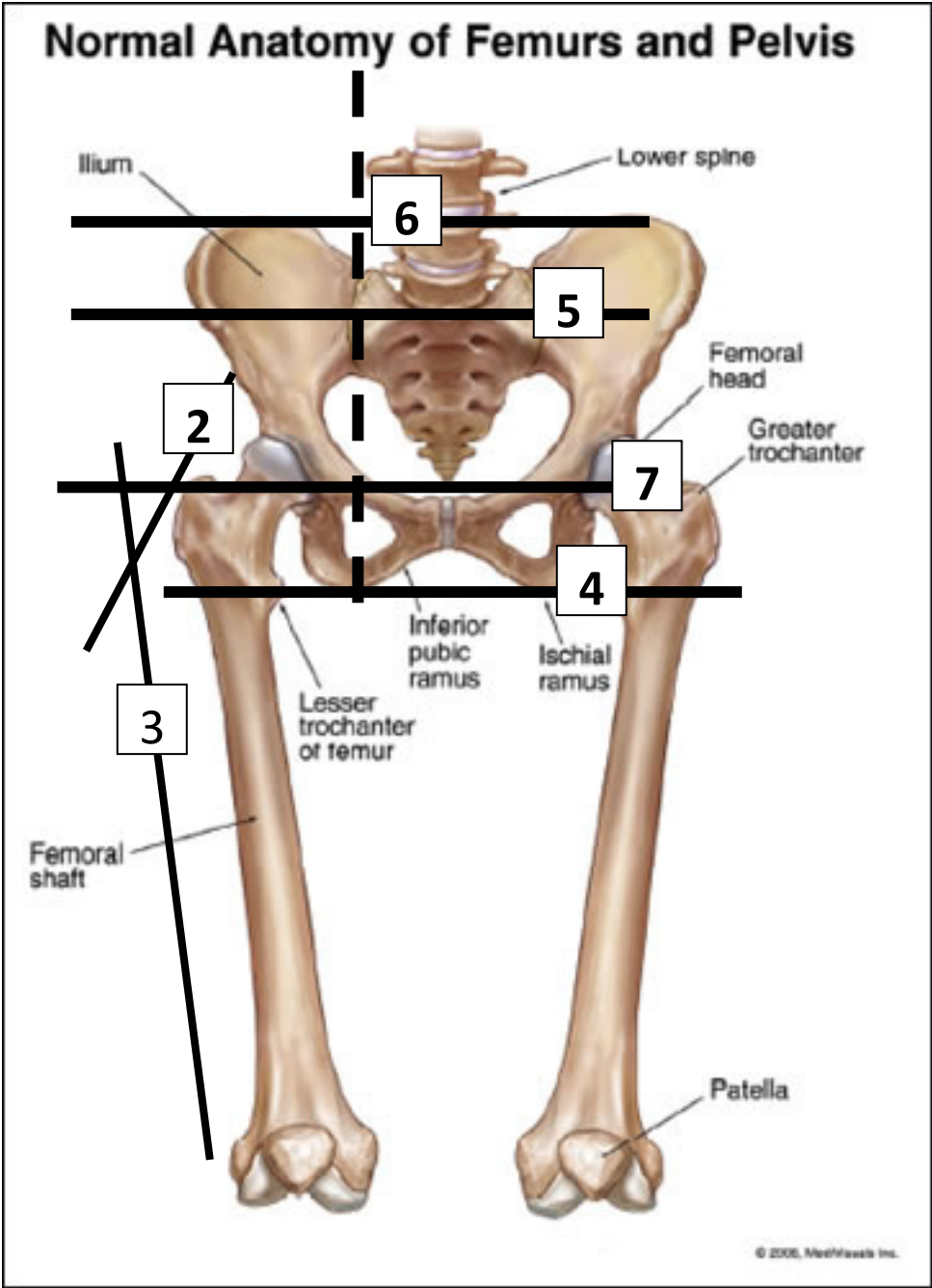


Figura 46.

Studiul a luat in calcul un numar de 22 de pacienti (cu varste intre 42- 65 ani) supusi artroplastiei totale necimentate de sold.

La acesti pacienti au fost efectuate 20 de artroplastii (4 bilaterale) pentru coxartroza primara (14 pacienti) si coxartroza posttraumatica (2 pacienti).

Am aplicat acelasi protocol efectuand radiografiile standard antero-posterioare si laterale la 3, 6 si 12 luni postoperator.

Studiul radiologic incepe prin compararea soldului opus care putea fi normal, cu semne de artroza sau supus unei alte artroplastii.

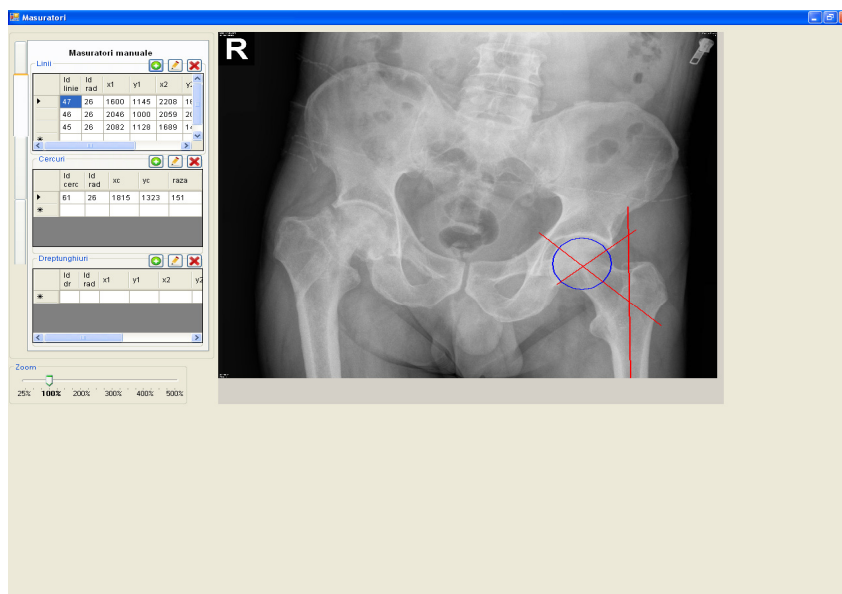


Figura 47. Compararea cu soldul opus, normal

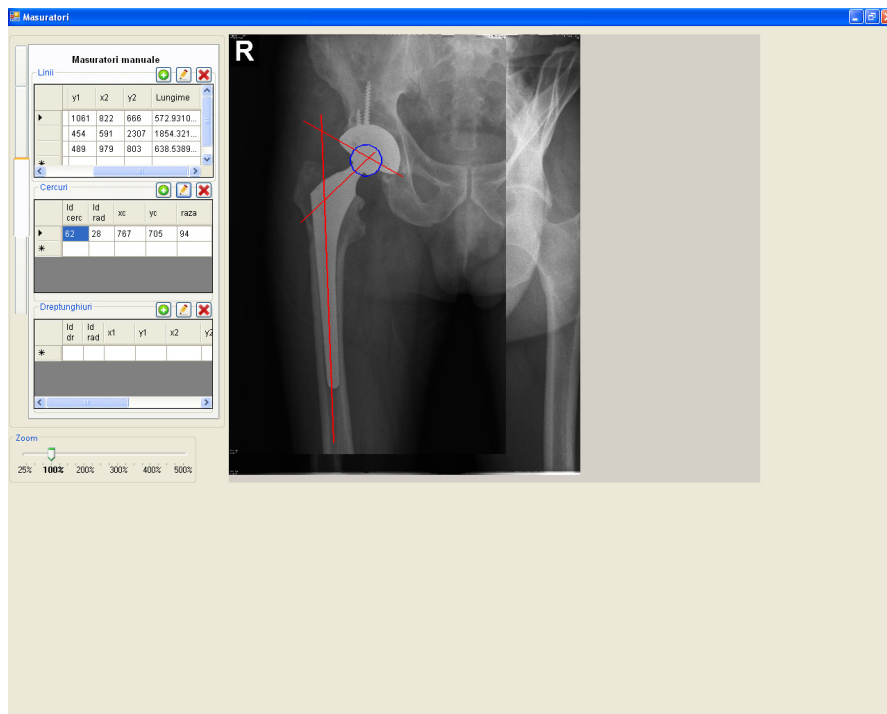
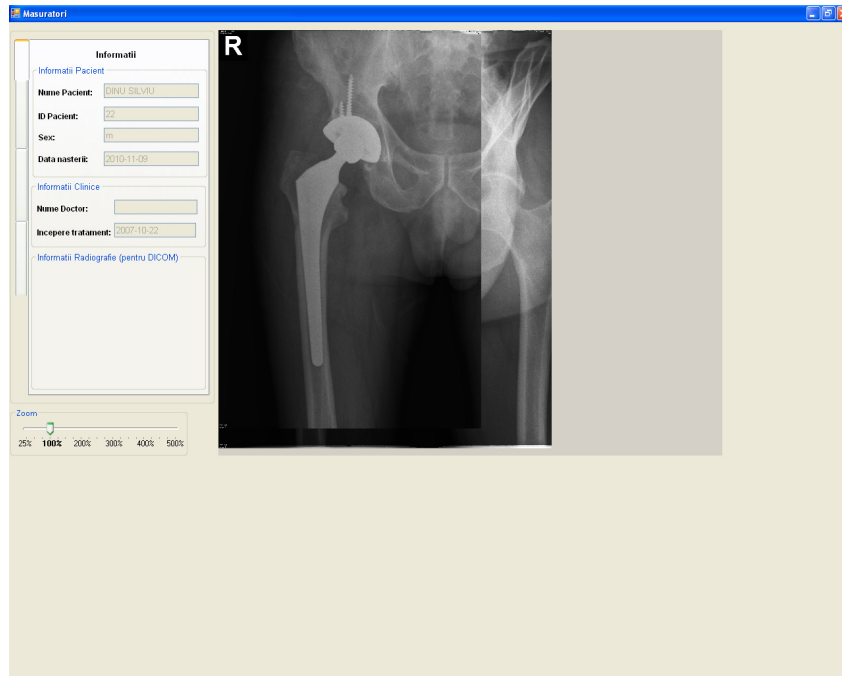


Figura 48. Analiza pozitiei protezei fata de anatomia locala

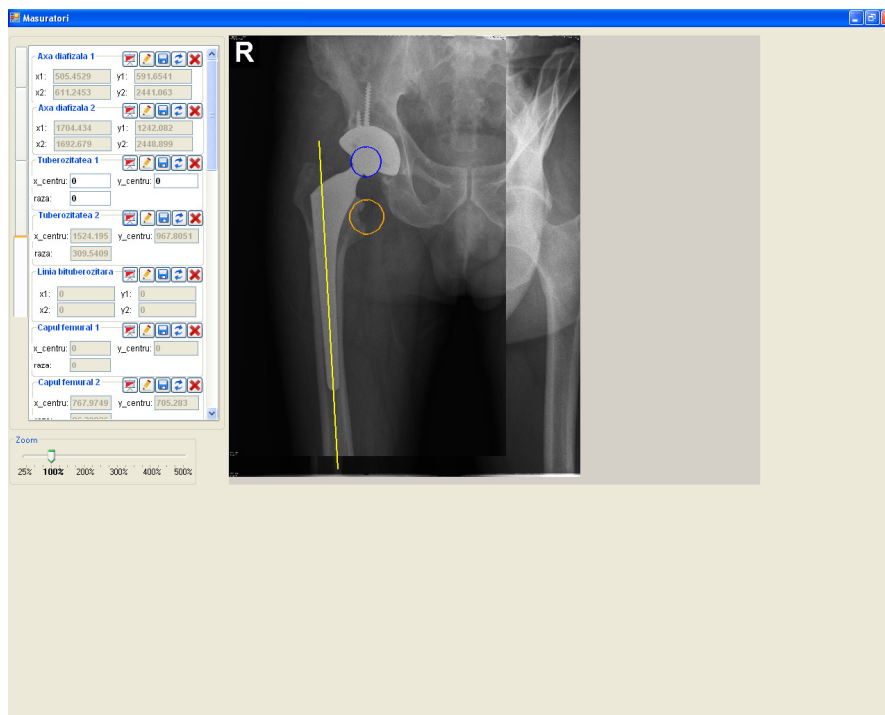


Figura 49. Analiza pozitiei protezei fata de anatomia locala

Radiografiile au fost apoi analizate referitor la:

- pozitia protezei fata de anatomia locala (axul acetabulului, unghiul dintre stem si axul femural)
- calitatea osului, folosind parametrii de imagine in conditii de expunere standard si densitometria osoasa.

Comparativ cu soldul contralateral, am evaluat pe radiografiile antero-posterioare axul acetabular, unghiul dintre axul protezei si axul diafizei precum si pozitia micului trohanter, iar pe radiografiile de profil- unghiul dintr axul protezei si axul diafizei femurale, precum si gradul de anteversie al protezei.

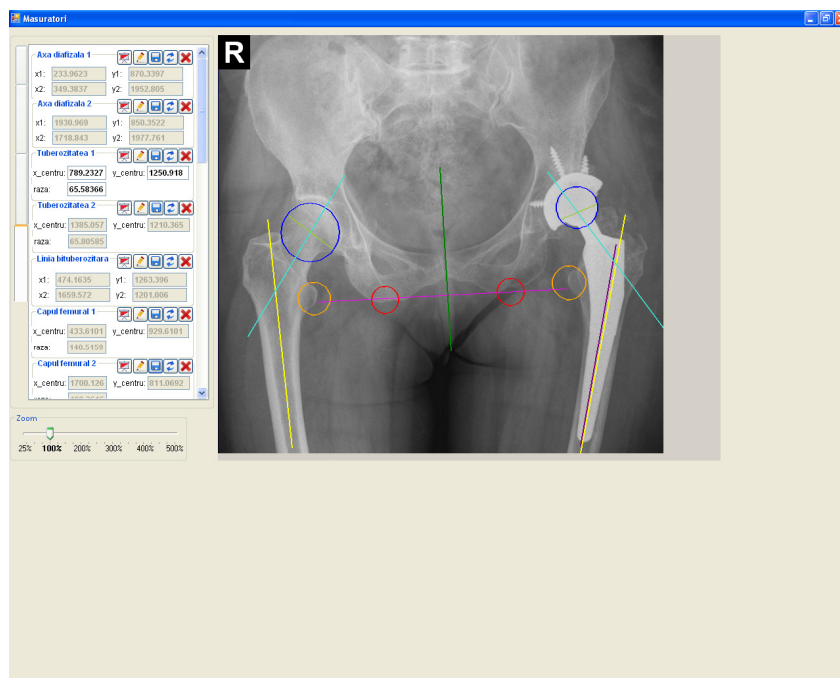
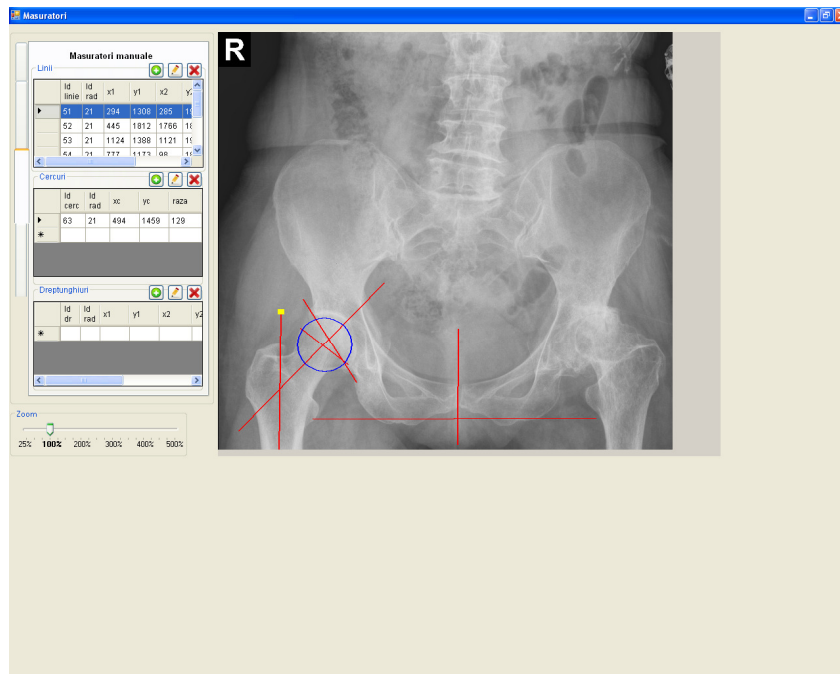


Figura 50. Masuratori efectuate pe radiografii cu ajutorul aplicatiei de analiza

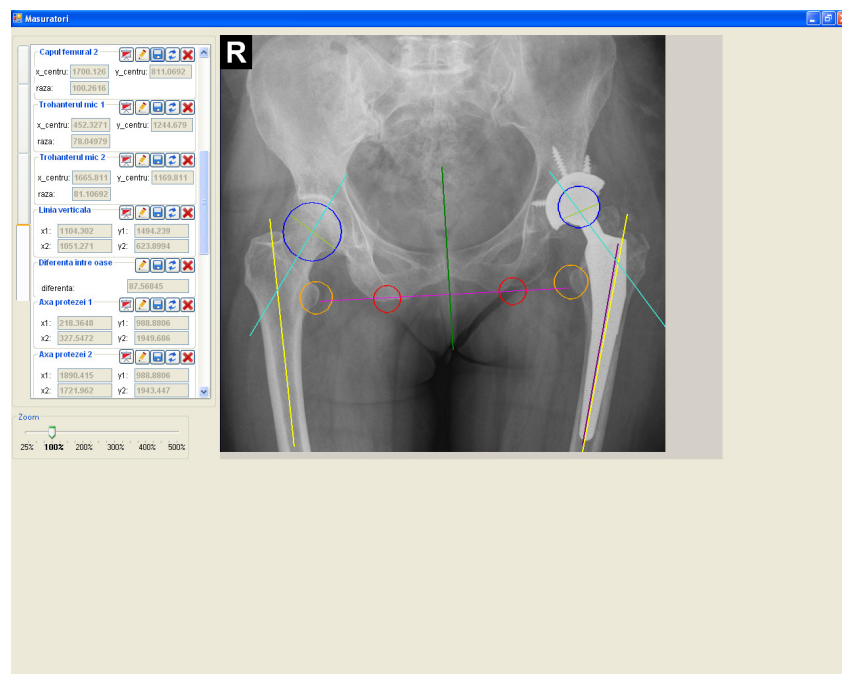
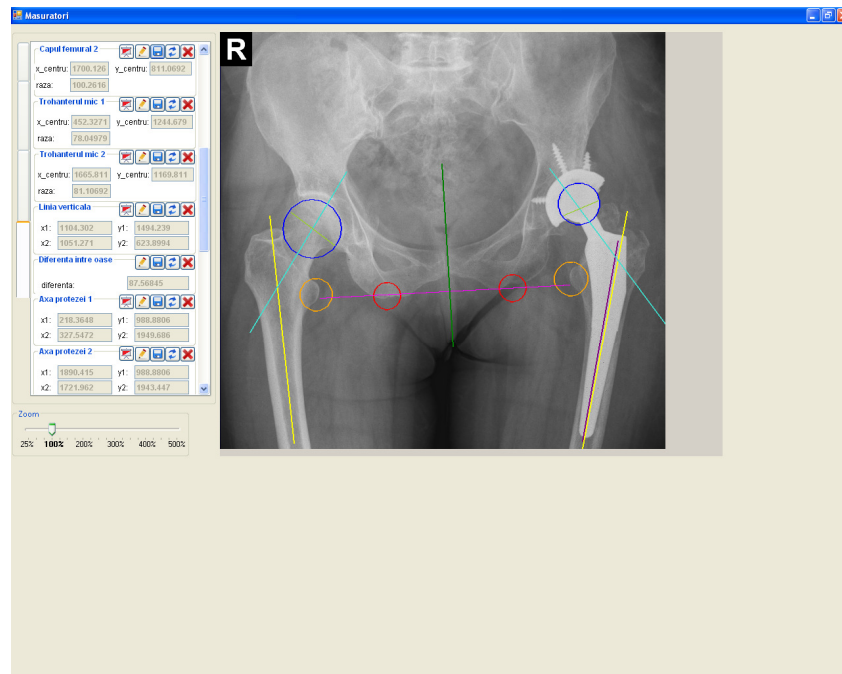


Figura 51. Masuratori efectuate pe radiografii cu ajutorul aplicatiei de analiza

Diferentele dintre axul protezei si axele anatomice au fost sub 2% in grupul de studiu, demonstrand o pozitionare corecta a implantului.

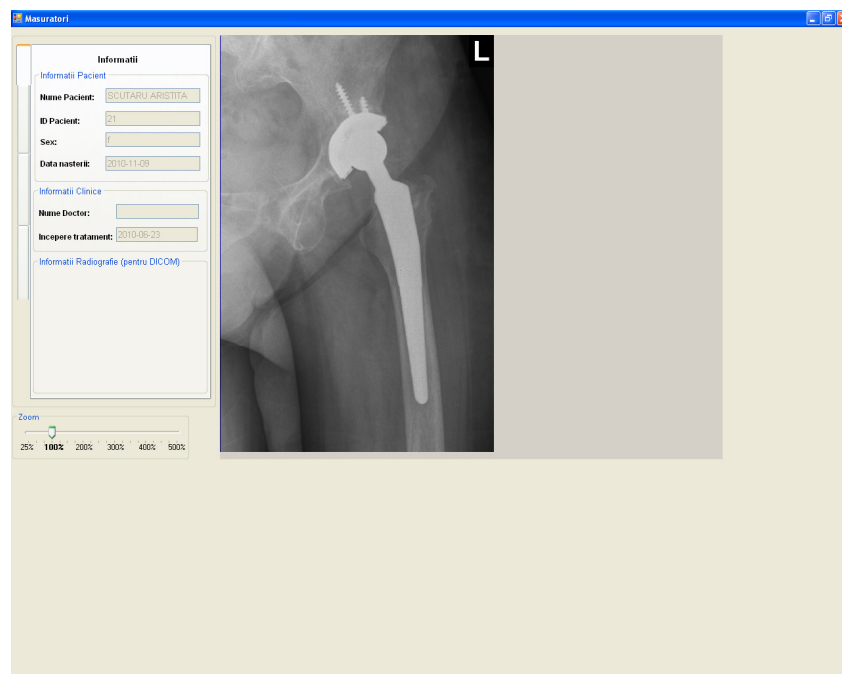
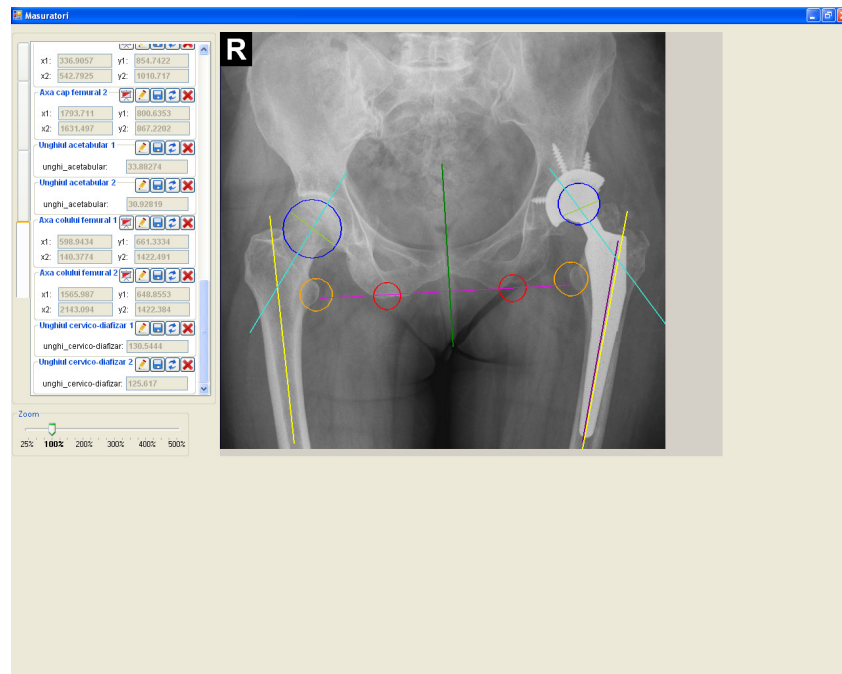


Figura 52. Masuratori efectuate pe radiografii cu ajutorul aplicatiei de analiza

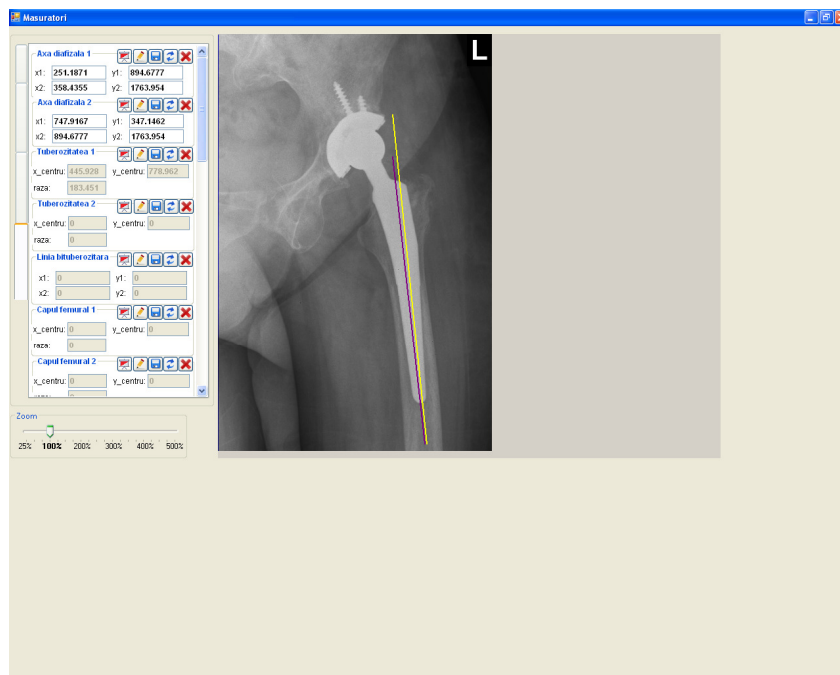
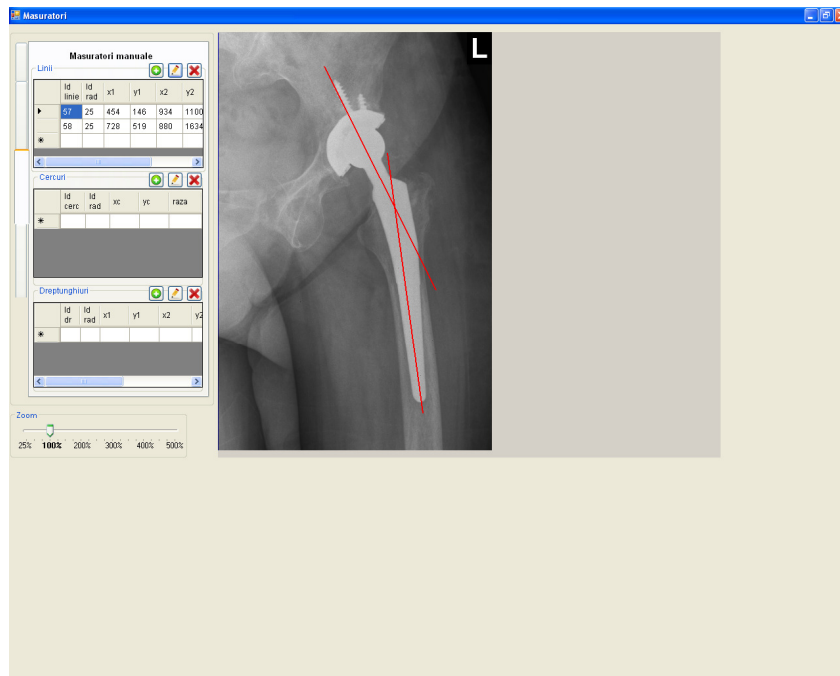


Figura 52. Masuratori efectuate pe radiografii cu ajutorul aplicatiei de analiza

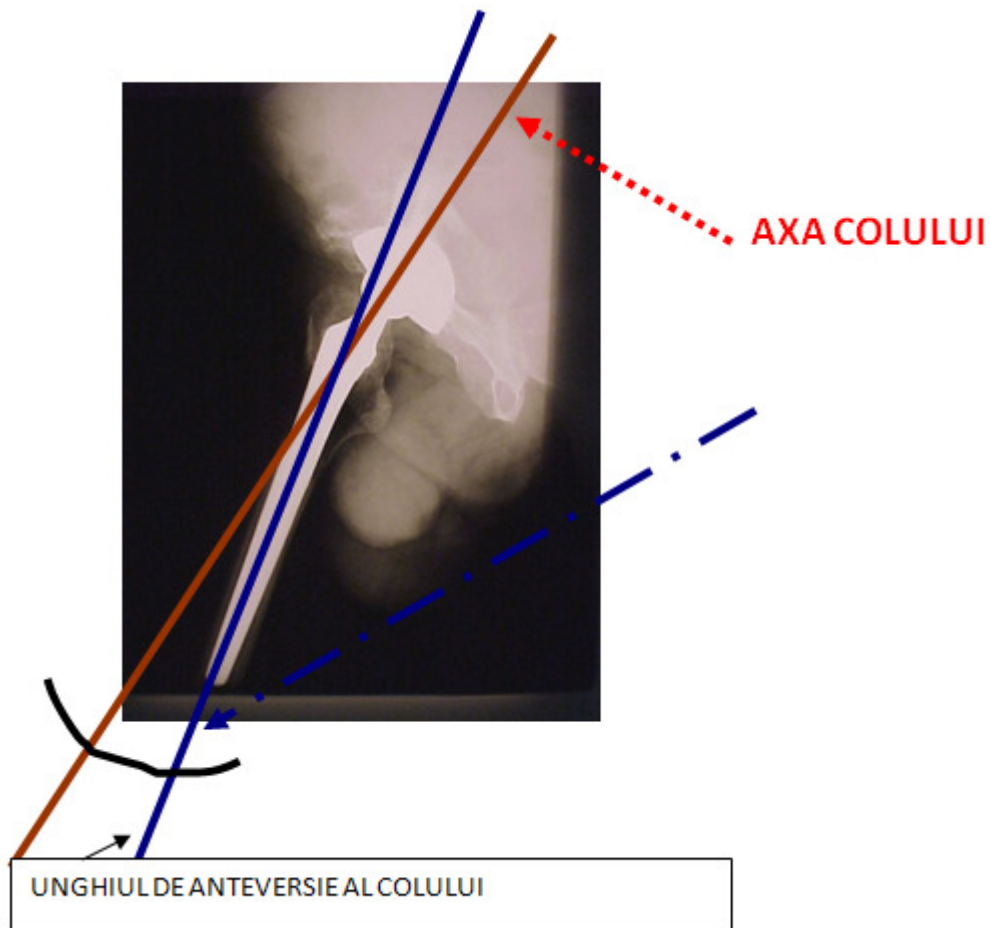


Figura 53.

Planning-ul permite chirurgului masurarea urmatoarelor parametri:

- marimea componentelor
- anticiparea profunzimii patrunderii componentei femurale in canal
- diferenta de lungime dintre membrele inferioare (ce trebuie corectata)
- nivelul optim de realizare a osteotomiei femurale
- anticiparea pozitiei componentei acetabulare
- stocul osos acetabular si femural

Scopul planning-ului preoperator este acela de a restaura:

a) centrul de rotatie a soldului in pozitie anatomica

b) offset-ul femural normal

c) egalitatea de lungime a membrelor inferioare

Designul individual, unic al fiecarui implant, bazat pe un model 3D precis al femurului, va face posibila obtinerea unei transe de osteotomie de mare precizie la nivelul jonctiunii cap- col femural. Modelul stemului distal de transa va fi creat de fiecare imagine CT. Acesta va avea ca scop obtinerea unor puncte de contact endocortical in urmatoarele zone „tinta” ale femurului:

- proximal-medial
- proximal- anterior
- lateral
- proximal posterior, imediat distal de transa de rezectie.

Se poate obtine astfel o arie de contact mai mare de 50-60% spre deosebire de 10-20% cat se obtine in cazul protezelor standard necimentate.

Partea distala a stemului va trebui indreptata in jos pentru a preveni contactul cu osul cortical.

Designul individual al colului femural va permite chirurgului sa optimizeze geometria extramedulara spre deosebire de protezele standard. Bazandu-ne pe studii biomecanice cunoscute si principii de functionare, putem imbunatatii cateva elemente:

- optimizarea offset-ului medial si anterior
- corectarea lungimii membrului
- imbunatatirea anteversiei

Analiza computerizata a radiografiilor studiate a confirmat faptul ca positionarea initiala corecta a protezei este esentiala pentru obtinerea unor rezultate foarte bune.

Radiografiile digitale sunt in mod evident mai utile decat cele clasice, iar un program complex de analizare a imaginii poate detecta fenomenul de loosening (figura 54).

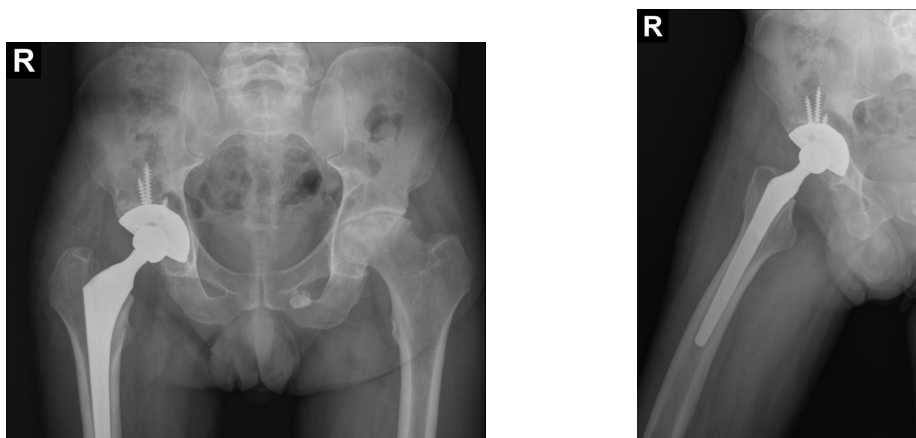


Figura 54.

9. CONCLUZII SI DIRECTII VIITOARE DE DEZVOLTARE

Sistemul de prelucrare si analiza a imaginilor radiografice a fost testat pe mai multe imagini radiografice in format DICOM si BITMAP.

In 80% dintre cazuri au fost determinati automat toti parametrii importanti in artroplastie. Partile din oase care nu au fost intotdeauna determinate automat sunt urmatoarele:

- Trohanterele mici. Acestea ocupa o parte foarte mica din imagine, de aceea sunt greu de determinat.
- Capetele femurale. Acestea nu se determina cu precizie de 100% din cauza frontierei vagi care desparte bazinul de os.

O varianta mai exacta in determinarea parametrilor ar fi implementarea transformatei Hough pentru cercuri fara limitarea la gasirea primului cerc care se incadreaza in reguli, ci cautarea tuturor cercurilor, apoi calcularea cercului final ca media dintre aceste cercuri gasite. Aceasta varianta da rezultate mai bune, dar necesita un timp mult mai mare pentru procesarea imaginilor.

In concluzie, exista 2 directii de cercetare viitoare care ar conduce la imbunatatirea performantelor aplicatiei: marirea preciziei masuratorilor si construirea unor algoritmi mai rapizi de extragere a conturilor si a anumitor forme din imagini.

Organizarea pacientilor si a radiografiilor intr-o baza de date permite gestiunea inter- si intra-institutionala. Pornind de la acest proiect, se poate crea o retea a tuturor spitalelor locale (de exemplu a tuturor spitalelor din Bucuresti) care sa tina o evidenta completa si unitara a tuturor pacientilor. Aceasta aplicatie ar simplifica foarte mult munca personalului din spitale, datorita existentei unei baze de date globale, cu datele si istoricul medical al fiecarui pacient.

Pastrarea radiografiilor si a masuratorilor lor intr-un format electronic reprezinta un pas inainte in ce priveste organizarea sistemului medical din tara noastra. Posibilitatea realizarii de masuratori manuale si automate care sunt salvate intr-o baza de date deschide noi orizonturi in ce priveste gestiunea pacientilor, urmarirea evolutiei acestora inainte si dupa operatie, realizarea de colaborari intre spitale, etc.

Activitatile din etapa urmatoare a proiectului se vor centra pe realizarea de masuratori pe imagini medicale 3D, vizualizarea acestora, precum si determinarea suprafetelor 3D ale oaselor din imagini CT si RMN, in vederea prototiparii protezelor pentru asistarea preoperatorie. O versiune experimentală a sistemului de reconstructie 3D si prototipare a fost deja realizata de partenerul INCDMTM, care a achizitionat in acest scop un echipament de prototipare rapida.

Avand in vedere timpul foarte mare pentru procesarea imaginilor 2D, este de asteptat ca procesarea imaginilor 3D sa dureze si mai mult. Din acest motiv, echipa de la UPB a avansat deja pe directia de cercetare legata de paralelizarea algoritmilor de prelucrare a imaginilor 3D (Algoritmii *Marching Cubes* si *Ray Casting* folosind CUDA si OpenCL).

Sistemului de Prelucrare si Analiza a Imaginilor Radiografice a fost utilizat in cadrul unui studiu, de catre partenerii de la SCUB si UMF, pentru a se obtine Rapoartele privind buna sa functionare.

Studiul a luat in calcul un numar de 22 de pacienti (cu varste intre 42- 65 ani) supusi artroplastiei totale necimentate de sold. La acesti pacienti au fost efectuate 20 de artroplastii (4 bilaterale) pentru coxartroza primara (14 pacienti) si coxartroza posttraumatica (2 pacienti).

A fost aplicat acelasi protocol efectuand radiografiile standard antero-posterioare si laterale la 3, 6 si 12 luni postoperator. Studiul radiologic a fost facut prin compararea soldului opus care putea fi normal, cu semne de artroza sau supus unei alte artroplastii.

Diferentele dintre axul protezei si axele anatomice calculate automat de aplicatie si manual de specialistii in ortopedie au fost in ambele cazuri sub 2% in grupul de studiu, demonstrand atat o pozitionare corecta a implantului, cat si o determinare automata corecta a parametrilor.

Pentru diseminarea rezultatelor obtinute pana la finalul etapei a 3-a, s-au publicat urmatoarele articole:

- International Conference on Signal Processing : Session Image Processing, Catania, Italia, 29-31 Mai 2010, „**Computer Assisted Analysis of Orthopedic Radiographic Images**”, Autori Anca Morar, Florica Moldoveanu, Alin Moldoveanu, Victor Asavei, Alexandru Egner – Indexata ISI
- International Conference on Telecommunications and Informatics : Session Information Science and Applications, Catania, Italia, 29-31 Mai 2010, „**GPGPU for Cheaper 3D MMO Servers**”, Autori Victor Asavei, Alin Moldoveanu, Florica Moldoveanu, Anca Morar, Alexandru Egner – Indexata ISI
- Romanian Review Precision Mechanics, Optics and Mechatronics nr. 38/2010 (ISSN 1584-5982, B+ si indexata BDI EBSCO) : „**Joint Motion Area Related To Prosthesis Component Position In Total Hip Arthroplasty**”, autori Adrian Pacioga, Doru Dumitru Palade, Stanca Comşa
- Buletinul Ştiinţific al UPB, „**Computational Simulation Of Bone - Personalized Hip Prosthesis Assembly**”, autori Adrian Pacioga, Doru Dumitru Palade, Stanca Comşa, lucrare acceptata pentru publicare.

Bibliografie

- [1] Bankman I, *Handbook of Medical Image Processing and Analysis*, Academic Press, 2000
- [2] Feng D D, *Biomedical Information Technology*, Elsevier, 2008
- [3] Chen Y, Ee X, Leow K W, Howe T S, *Automatic Extraction of Femur Contours from Hip X-ray images*, 2000
- [4] Gonzales R C, Woods R E, *Digital Image Processing*, Prentice-Hall, 2002
- [5] Canny J F, *A Computational Approach to Edge Detection*, IEEE Trans. Pattern Analysis and Machine Intelligence, 1986
- [6] Campilho A, Kamel M, *Image Analysis and Recognition*, Springer, International Conference ICIAR 2004 Proceedings, Part I
- [7] Raj K Sinha, *Hip Replacement, Current Trends and Controversies*, 2002.
- [8] Kennon R, *Hip and Knee Surgery: A Patient's Guide to Hip Replacement, Hip Resurfacing, Knee Replacement, and Knee Arthroscopy Book Description*
- [9] Botez P, *Ortopedie*, Bit Publishing House (Iasi), 2001
- [10] Morrey B F, *Joint Replacement Arthroplasty*
- [11] Bernhard Preim, Dirk Bratz, *Visualization in Medicine*, Elsevier, Morgan Kaufmann Publishers, 2007
- [12] Wei Lu, *Hough transforms for shape identification and applications in medical image processing*, 2003
- [13] Pagina SCANIP: Disponibila la <http://www.simpleware.com/software/scanip/iptechdata.php> in 10.10.2009
- [14] Pagina 3D-DOCTOR: Disponibila la <http://www.ablesw.com/3d-doctor/index.html> in 10.10.2009
- [15] Pagina Hip-OpCT: Disponibila la <http://www.hipop.it/hipopct.html> in 20.10.2009
- [16] Pagina oficiala a standardului DICOM. Disponibila la <http://dicom.nema.org> in 20.05.2009
- [17] Pagina oficiala a bibliotecii AForge. Disponibila la <http://www.aforgenet.com/> in 20.12.2009
- [18] Documentatie MySQL. Disponibila la <http://dev.mysql.com/doc/> in 20.12.2009
- [19] Transformata Hough. Disponibil la http://en.wikipedia.org/wiki/Hough_transform in 20.12.2009