

# Asistent medical

## Detecția cancerului de sân

– MIRPR report –

### **Membrii echipei**

**Munteanu Bianca-Ștefania**, Informatică-română, 235  
bianca.stefania.munteanu@stud.ubbcluj.ro

**Murariu Alexandra**, Informatică-engleză, 935  
alexandra.murariu1@stud.ubbcluj.ro

**Nichitean Mărioara**, Informatică-română, 235  
marioara.nichitean@stud.ubbcluj.ro

**Pitac Luminița-Gabriela**, Informatică-română, 235  
luminita.pitac@stud.ubbcluj.ro

## Abstract

Ideea centrală a acestui proiect este diagnosticarea computerizată a cancerului de sân. Acesta reprezintă unul dintre factorii majori ai decesului în cazul femeilor, 1 din 39 (în jur de 2.5%) dintre acestea pierzând lupta cu viața anual, la nivel global, din aceasta cauză. Conform American Cancer Society, este al doilea cel mai letal tip de cancer la persoanele de sex feminin, fiind precedat doar de cel de plămâni.

Diagnosticarea timpurie este crucială în creșterea șanselor de supraviețuire. În ultimii ani, rata de incidență a crescut cu 0.5% pe an, 1 din 8 femei fiind supuse unui risc crescut de a dezvolta o astfel de tumoră pe parcursul vieții. În ciuda progreselor tehnologice, există încă dificultăți în identificarea, caracterizarea și monitorizarea precisă a tumorilor maligne.

Ne-am propus să rezolvăm această problemă cu ajutorul algoritmilor inteligenți din sfera rețelelor neuronale. Rezolvarea problemei a implicat 3 etape importante: augmentare, segmentare și clasificare. Experimentul a fost realizat folosind un set de date care conține imagini medicale cu ultrasunete, provenind de la aproximativ 600 de pacienți. Rezultatele experimentului se apropie de obiectivul stabilit de echipa noastră. Acuratetea finală obținută variaza între 75% - 86%.

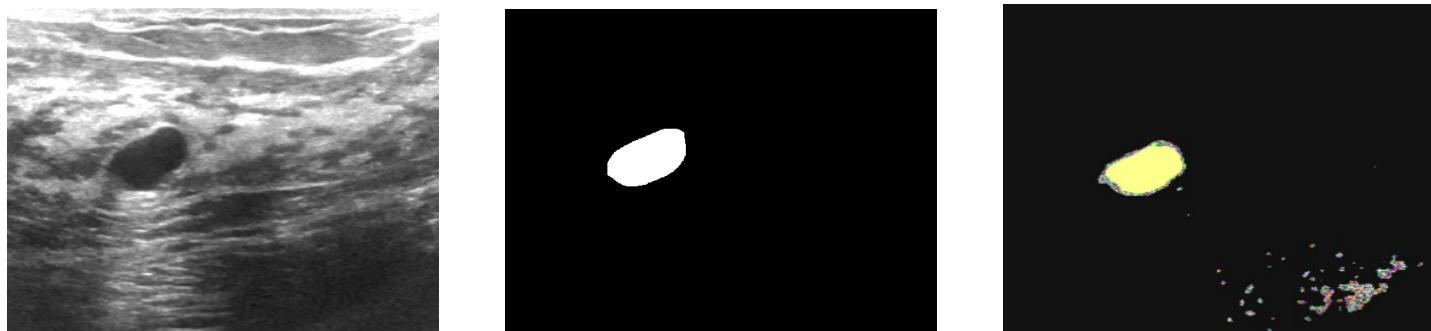


Figure 1: Segmentare imagine benigna

# Contents

1	Problema științifică - Definirea problemei	1
2	Rezultate anterioare - State of the Art	2
3	Descrierea pașilor de rezolvare	6
4	Aplicație (Studiu de caz)	19
4.1	Descrierea aplicației și funcționalitățile de bază	19
4.2	Mindmap	20
4.3	Designul aplicației	20
4.4	Implementare	20
4.5	Set de date	21
4.6	Rezultate	21
5	Concluzii	25

# List of Figures

1	Segmentare imagine benigna . . . . .	
2.1	Citatii ale dataset-ului . . . . .	2
2.2	Rezultate anterioare - clasificare . . . . .	4
2.3	Rezultate anterioare - object detection . . . . .	5
2.4	Rezultate anterioare - segmentare . . . . .	5
3.1	Filtru gamma . . . . .	7
3.2	Convolutie . . . . .	8
3.3	Maxpool 2D . . . . .	10
3.4	Flattening . . . . .	10
3.5	(a) benign (b) malign (c) normal . . . . .	12
3.6	(a) benign (b) malign (c) normal . . . . .	13
3.7	(a) benign (b) malign (c) normal . . . . .	14
3.8	U-Net . . . . .	17
3.9	(a) fără augmentare (b) cu augmentare . . . . .	18
4.1	Flow-ul aplicației . . . . .	19
4.2	Mindmap . . . . .	20
4.3	Rezultate segmentare . . . . .	22
4.4	Rezultate clasificare . . . . .	23
4.5	Inferenta segmentare . . . . .	23
4.6	Inferenta clasificare . . . . .	23
4.7	Merged Segmentation and Classification models . . . . .	24
4.8	Comparatie acuratete . . . . .	24
5.1	Clasificare pe date augmentate . . . . .	25
5.2	Clasificare pe date neaugmentate . . . . .	26

# Chapter 1

## Problema științifică - Definirea problemei

### *Descriere formală:*

Algoritmul primește ca și input o imagine medicală cu ultrasunete de la nivelul sanului. În prima fază, aplicația localizează tumora, dacă aceasta există. În a doua fază, încadrează țesutul în una din cele 3 categorii: sanatos, malign, benign.

### *Descriere plastică:*

Algoritmul este compus din 3 etape importante: augmentarea datelor, segmentare și clasificare. Pasul de augmentare constă în generarea de noi imagini pentru a echilibra discrepanțele din setul de date ales. Pentru segmentare și clasificare folosim arhitectura UNet, un CNN popular în domeniul medical pentru detectarea și diagnosticarea diferitelor afecțiuni.

### *Provocări:*

Ne dorim ca algoritmul să fie cât se poate de performant având în vedere problema delicată pe care trebuie să o abordeze. Așadar, avem nevoie de un set mare și complex de date pentru a obține o antrenare corectă și coerentă a algoritmului. Luând în considerare datele publice disponibile în acest moment, am concluzionat că ne confruntăm cu o problemă dificilă a cărei rezolvare este amplă și complicată.

Totodată, învățarea algoritmului necesită o capacitate mare de procesare care este greu de atins pe un CPU normal. Astfel, nevoia de un GPU sau de o platformă auxiliară care să pună la dispoziție un accelerator este inevitabilă.

## Chapter 2

# Rezultate anterioare - State of the Art

Imaginile ultrasound sunt folosite din ce in ce mai des, iar radiologii petrec un timp foarte indelungat pentru a examina volume mari din aceste imagini. Astfel, a devenit o problema majora in multe tari pentru ca duce la cresterea costurilor medicale si inrautatestea calitatea serviciilor medicale. Studiile asupra tehnologiilor deep learning din ultimul timp a adus numeroase beneficii imagisticii medicale. Au fost aplicate pentru imagini radiologice, de exemplu, pentru a detecta tuberculoza in radiografiile de piept, nodulii pulmonari sau tumora craniana din MRI. Datasetul folosit de noi a fost citat de 325 ori. Mai jos se poate observa distributia articolelor pe parcursul ultimilor ani, in care a fost citat.

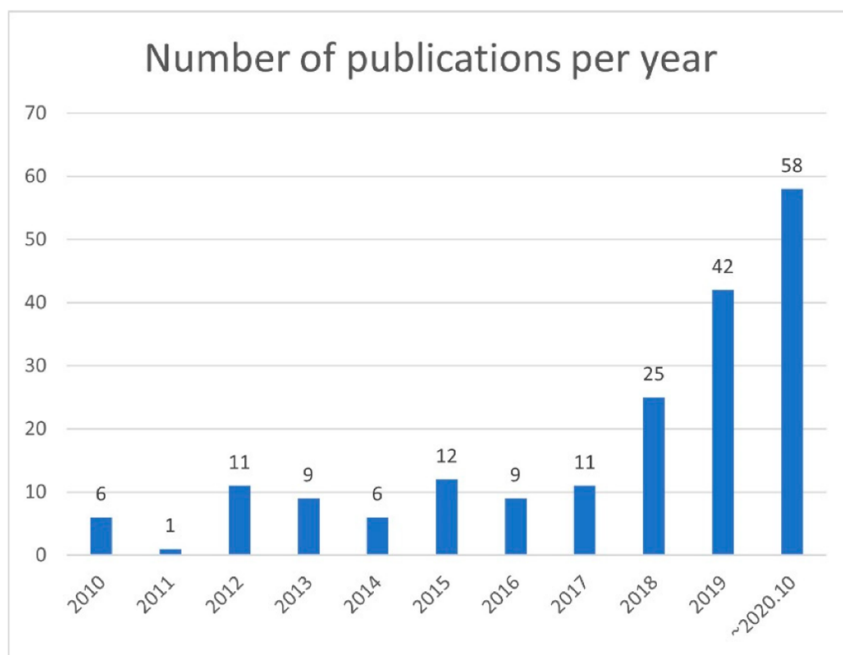


Figure 2.1: Citatii ale dataset-ului

Conform [6], în ultimii ani, CNN-urile au revoluționat cercetarea în domeniul medical pentru detectarea și diagnosticarea cancerului. Aceste metode utilizează diferite tipuri de algoritmi, arhitecturi CNN și proceduri de învățare pentru clasificarea cancerului și au obținut performanțe remarcabile. Unii cercetători utilizează CNN-uri pentru a clasifica rezultatele RMN cu un mecanism de viziune computerizată, în timp ce alții utilizează diferite combinații de imagini mpMRI prin stivuirea fiecărei modalități ca un canal 2D de imagini RGB. Mai mult, pot fi proiectate CNN-uri 3D care utilizează parametri specifici bazați pe RMN, cum ar fi coeficientul de difuzie aparentă, high b-value și constanta de transfer de volum (Ktrans) a modalităților.

Conform [9], cercetarea în clasificarea cancerului mamar pe baza imaginii histologice folosind CNN are o acuratețe de 98%. Folosind mamografia, unele studii în care se aplică rețeaua neuronală convoluțională în clasificarea tumorilor au atins o acuratețe de 97%. Între timp, o abordare îmbunătățită a învățării profunde bazată pe imaginea cu ultrasunete a atins 81,5% acuratețe prin clasificarea binară.

În aceasta secțiune voi prezenta câteva lucrări similare în care s-au folosit tehnici de augmentare în antrenarea modelelor de clasificare a imaginilor medicale. Utilizarea modelelor preantrenate ajută la accelerarea procesului de adaptare a rețelelor, conducând la o rezolvare mai rapidă a problemelor.

În lucrarea [8], autorii s-au folosit de rețelele AlexNet pentru a realiza clasificarea binară, pe care le au modificat introducând mașini cu suport vectorial pe ultimul strat. De asemenea, au folosit Threshold, pentru a automatiza procesul de segmentare. Au realizat cu succes clasificarea tumorilor benigne și maligne, lucrând pe seturile de date DDSM și CBIS-DDSM, obținând o acuratețe de 87.2%.

Levy și Jain [1] au comparat arhitecturile AlexNet, GoogleNet și un CNN simplu, la care au adăugat tehnici de învățare prin transfer, normalizarea batch-urilor, preprocesare și augmentare. Cel mai bun model a obținut 0.934 pentru rapel și 0.924 pentru precizie.

Jung et al. [5] au propus rețeaua RetinaNet, cu ponderile preantrenate în GURO, pentru a demonstra că modelele preantrenate obțin performanțe similare cu cele antrenate direct pe setul de date INbreast. Modelul lor a obținut în medie 0.34 fals pozitive.

În [4], s-a folosit o rețea adversarială pentru detectarea tumorilor. Studiul s-a realizat pe o rețea convoluțională (FCN) pentru a modela funcția, fiind urmat de Conditional Random Fields (CRF) pentru învățarea structurată. De asemenea, s-a folosit și GAN pentru a controla overfitting-ul care putea apărea din cauza faptului că autorii aveau la dispoziție un număr mic de date. Aceștia au obținut performanța state-of-the-art pe 2 seturi de date INbreast și DDSM-BCRP.

Într-o altă lucrare, Bakkour și Afdel [3] au propus o nouă tehnică discriminativă pentru învățarea supervizată, utilizând stratul Softmax drept clasificator. Rețeaua a fost imunată utilizând piramide Gaussiene pentru a accentua regiunile de interes. Rezultatele pe DDSM și BCDR au relevat o acuratețe de 97.28%.

Un studiu recent [7] propune combinarea arhitecturii CNN cu augmentarea datelor pentru detectarea distor-

siunilor. S-au aplicat mai multe transformari asupra imaginilor cu mamografii, atat din perspectiva MLO cat si CC. S-a obtinut o acuratete de 93.75%, in urma aplicarii modelului pe seturile de date MIAS, DDSM+CBIS si INBreast.

In continuare, vom prezenta rezultate pe imaginile ultrasound, obtinute pana in acest moment, ilustrate in articolul [10]

Table 1. Deep learning models for image classification.				
Purpose (Type of Image)	Model	Number of Training Set Images	Number of Test Set Images	Result
Breast lesions (B-mode images)	GoogLeNet	6579	829	Sensitivity: 86% Specificity: 96% Accuracy: 90% AUC: 0.9
Breast lesions (B-mode image)	GoogLeNet Inception v2	937	120	Sensitivity: 95.8% Specificity: 87.5% Accuracy: 92.5% AUC: 0.913
Breast lesions via CAD (B-mode image)	Koios DS	Over 400,000	900	AUC without CAD: 0.83 AUC with CAD: 0.87
Breast lesions (SWE image)	PGBM and RBM	227	Five-fold cross-validation	Sensitivity: 88.6% Specificity: 97.1% Accuracy: 93.4% AUC: 0.947
Breast lesions (SWE image)	DenseNet 169	304	73	Sensitivity: 85.7% Specificity: 78.9% AUC: 0.898

Figure 2.2: Rezultate anterioare - clasificare



**Table 2.** Deep learning models for object detection.

Purpose (Type of image)	Model	Number of Training Set Images	Number of Test Set Images	Result
Object detection of breast lesions (B-mode image)	SSD300	860	183	Precision rate: 96.89% Recall rate: 67.23% F1 score: 79.38%
Object detection of breast lesions by CAD (ABUS image)	QVCAD	Over 20,000	185	AUC without CAD: 0.828 AUC with CAD: 0.848
Object detection of breast lesions by CAD (ABUS image)	QVCAD	Over 20,000	1485	AUC without CAD: 0.88 AUC with CAD: 0.91 Sensitivity without CAD: 67% Sensitivity with CAD: 88%
Object detection of breast lesions by CAD (ABUS image)	QVCAD	Over 20,000	1000	AUC without CAD: 0.747 AUC with CAD: 0.784

Figure 2.3: Rezultate anterioare - object detection

**Table 3.** Deep learning models for segmentation.

Purpose (Type of image)	Model	Number of Training Set Images	Number of Test Set Images	Result
Segmentation of breast lesions (B-mode image)	RDAU-NET	857	205	Precision rate: 88.58% Recall rate: 83.19% F1 score: 84.78
Segmentation of breast lesions (B-mode image)	Combining DFCN with a PBAC model	400	170	Dice similarity coefficient: 88.97% Hausdorff distance: 35.54 pixels Mean absolute deviation: 7.67 pixels
Segmentation of breast lesions (B-mode image)	Multi U-net algorithm	372	61	Mean Dice coefficient: 0.82 True positive fraction: 0.84 False positive fraction: 0.01

Figure 2.4: Rezultate anterioare - segmentare

## Chapter 3

# Descrierea pașilor de rezolvare

Am citit matricea de pixeli în format RGB și am aplicat diferite tehnici în scopul obținerii de informații cât mai relevante. La fiecare pas, am testat diverse filtre, am afișat imaginile rezultat și am continuat procesul cu rezultatul pe care l-am considerat a fi cel mai adecvat pentru problema noastră. În continuare, vom atașa imaginile cu filtrările parțiale aplicate pe câte o imagine din fiecare categorie (țesut benign/malign/sănătos), urmând a detalia doar teoria din spatele filtrelor cu care am ales să continuăm analiza.

### Pasul 1 - Corecția Gamma:

Operație neliniară care definește relația dintre valoarea numerică a unui pixel și luminozitatea reală a acestuia. Cu ajutorul acestei corecții, aducem nuanțele captate de dispozitivele care au efectuat imaginea, cât mai aproape de nuanțele ce pot fi surprinse de ochiul real.

$$s = c * r^{\gamma} \quad (3.1)$$

unde:

r=pixeli de intrare

s=pixeli de ieșire

c=constanta (factor de scalare)

$\gamma$ =principalul exponent responsabil de modificarea pragului de luminozitate a imaginii

Caz 1 [Fig 4.1.2]:  $\gamma < 1$

Codificarea gamma - avem o gamă restrânsă de pixeli întunecați în imaginea originală și dorim să extindem intervalul valorilor de ieșire. Efectul unei astfel de curbe constă în accentuarea zonelor luminoase ale unei imagini.

Caz 2: [Fig 4.1.3]:  $\gamma > 1$

Decodificarea gamma - avem o gamă largă de pixeli întunecați în imaginea originală și dorim să restrângem intervalul valorilor de ieșire. Efectul unei astfel de curbe constă în atenuarea zonelor luminoase ale unei imagini.

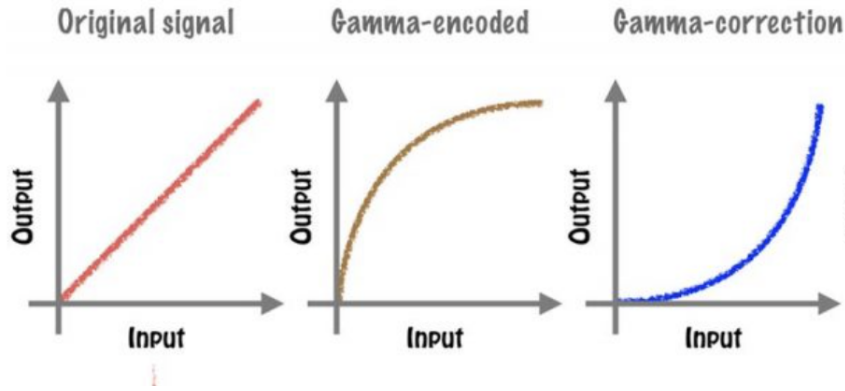


Figure 3.1: Filtru gamma

În cazul nostru, am ales  $\gamma=2$  deoarece ne interesează să atenuăm luminozitatea pixelilor și să punem accentul pe zona mai întunecată din imagine, care poate fi suspectă de tumoră.

## Pasul 2 - Convoluția:

Matematic, convoluția este o operație care combină două semnale și emite un al treilea semnal. Presupunând că avem două funcții,  $f(t)$  și  $g(t)$ , definim convoluția drept integrala ce exprimă suprapunerea unei funcții  $g$ , peste funcția  $f$ .

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau$$

În procesarea imaginilor, convoluția este procesul în care fiecare element al imaginii este adăugat vecinilor săi locali și apoi este ponderat de nucleu. Un nucleu este o matrice mică, cu număr impar de linii/coloane în care fiecare celulă are un număr și un punct de ancorare (folosit pentru a cunoaște poziția nucleului în raport cu imaginea). Fiecare pixel din matricea inițială este înmulțit cu elementul corespunzător din nucleu și apoi adăugat la sumă. La final, matricea rezultat este compusă din aceste sume.

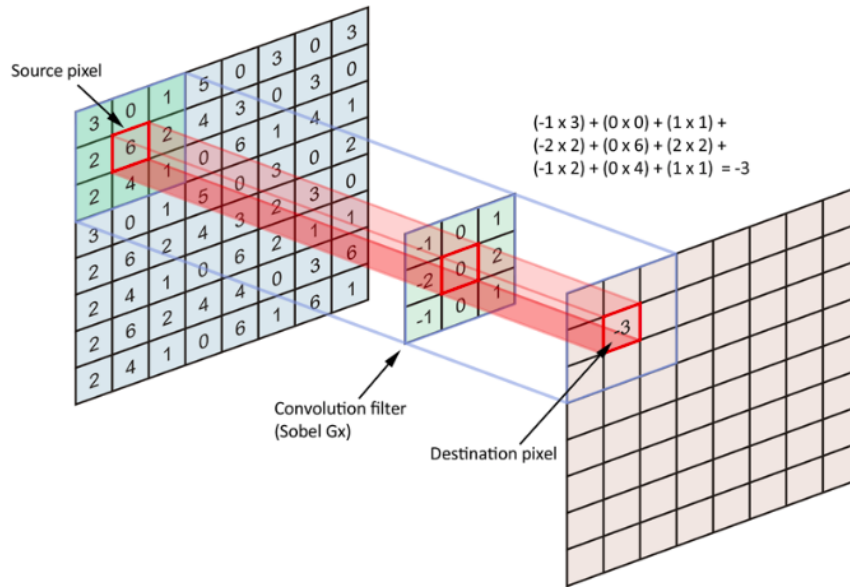


Figure 3.2: Convolutie

Convoluția este foarte importantă în prelucrarea imaginilor, deoarece poate fi utilizată pentru estompare, ascuțire, detectarea marginilor și reducerea zgomotului.

În aplicația noastră, am scris o funcție ce calculează convoluția pentru o matrice inițială și un nucleu dat, funcție pe care o vom apela la pasul următor.

### Pasul 3 - Gaussian Blur:

Filtrul Gaussian este folosit pentru reducerea zgomotului și a detaliilor dintr-o imagine. Acesta utilizează o funcție gaussiană pentru a calcula transformarea fiecărui pixel din imagine. Astfel, utilizarea acestui filtru, reprezintă din punct de vedere al semanticii efectului, aplicarea operației de convoluție asupra imaginii cu o funcție gaussiană.

Formula Gaussiană pentru spațiul bidimensional:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (3.2)$$

unde:

$x$  = distanța de la origine pe axa  $Ox$

$y$  = distanța de la origine pe axa  $Oy$

$\sigma$  = deviația standard

De valoarea deviației depinde lățimea imaginii rezultate în urma aplicării filtrului. Cu cât varianța este mai mică, cu atât pixelii sunt mai concentrați într-o anumită zonă. Un nucleu gaussian conține în centrul său valoarea maximă, iar pe măsură ce ne depărtăm de centru, conține valori mai mici, dar poziționate simetric.

În aplicația noastră am folosit următorul nucleu gaussian:

$$A_{5 \times 5} = \begin{pmatrix} 1/256 & 4/256 & 6/256 & 4/256 & 1/256 \\ 4/256 & 16/256 & 24/256 & 16/256 & 4/256 \\ 6/256 & 24/256 & 36/256 & 24/256 & 6/256 \\ 4/256 & 16/256 & 24/256 & 16/256 & 4/256 \\ 1/256 & 4/256 & 6/256 & 4/256 & 1/256 \end{pmatrix}$$

#### Pasul 4 - Binarizarea

În acest pas, imaginile sunt normalizate în 0 - 1, și sunt păstrați doar pixelii care au o valoare care depășește un anumit threshold. Dacă pixelii depășesc valoarea respectivă, aceștia vor primi valoarea 1, altfel vor deveni albi (0). În codul nostru, noi am ales ca threshold valoarea 0.45.

#### Pasul 5 - Sauvola Thresholding

Sauvola Thresholding o extensie (se poate considera o îmbunătățire) a algoritmului lui Niblack<sup>1</sup>. Aceasta tehnică este utilizată pentru imagini în care fundalul nu este unul uniform.

$$T = m * (1 + k * ((stdN/R) - 1)) \quad (3.3)$$

unde:

m=media vecinilor

k=constantă în intervalul [0.2, 0.5] (implicit 0.5)

R=intervalul dinamic al deviației standard (implicit 128)

Această tehnică în loc să calculeze un singur threshold global pentru întreaga imagine, calculează mai multe threshold-uri pentru fiecare pixel, luându-se în considerare media și abaterea standard a vecinătății locale (definite de o fereastră centrată în jurul pixelului).

<sup>1</sup>tehnică ce a încercat să rezolve problema contrastului scăzut prin decrementarea valorii threshold-ului

### Algoritmul de învățare:

Pentru clasificare, am folosit rețele neuronale convoluționale (CNN). Am ales CNN, deoarece aveam nevoie de o imagine de ansamblu cât mai apropiată de cea percepută de om, întrucât ochiul uman, în momentul în care analizează o imagine, o împarte automat în altele de dimensiune mai mică, cele din urmă fiind cele analizate. Prin asamblarea acestor subimagini, procesăm și interpretăm imaginea într-un mod eficient. În CNN-ul folosit am aplicat 3 pași importanți : Convolution, Max Pooling și Flattening.

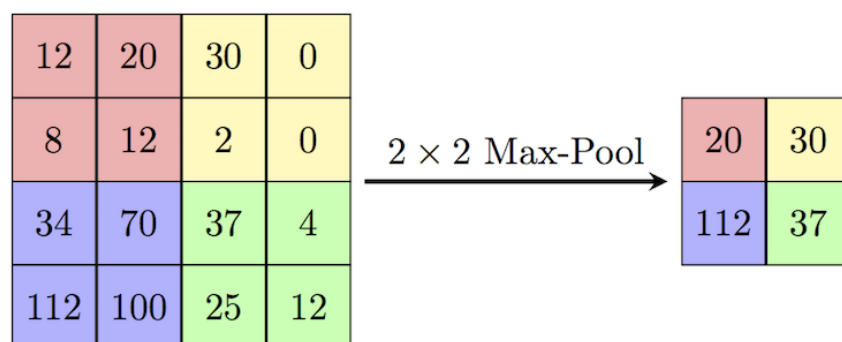


Figure 3.3: Maxpool 2D

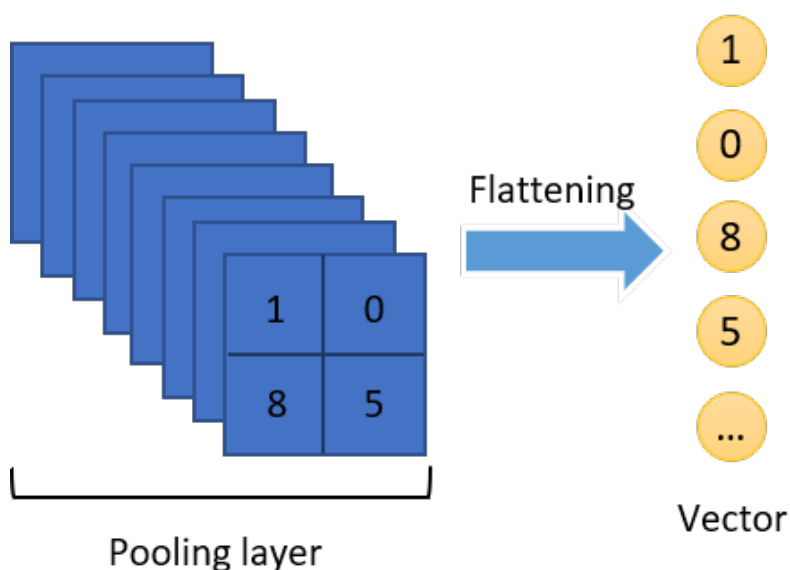


Figure 3.4: Flattening

În implementare am utilizat biblioteca keras și am creat un model secvențial la care am adăugat straturi rând pe rând. Primul strat este reprezentat de un kernel de convoluție 2D care creează 16 ferestre de dimensiune 3x3, cu activare ReLU(elimină componenta liniară creată de umbrele dintr-o imagine). Apoi am efectuat max pooling pentru a extrage cei mai importanți pixeli necesari clasificării. Regularizarea dropout este folosită pentru a preveni overfitting pe setul de antrenament. Aceste 3 straturi se repetă de 3 ori, cu diferiți parametri pentru kernel și

dropout.

La final, imaginea este liniarizată (datele se convertesc într-un tablou unidimensional) și se folosește funcția de activare softmax (întrucât este necesară prezența mai multor clase). Am împărțit imaginile în set de train/validation, iar mai apoi am testat acuratețea și loss-ul modelului pe date de antrenament, respectiv de test.

Pentru cele aproximativ 800 poze am folosit 15 epoci cu batch size 32, o alegere clasică în seturile de date de dimensiuni mici. În fiecare epocă, avem 18 pași de antrenament și 3 pași de validare.

În funcție de dimensiunile alese pentru kernelul de convoluție, batch size și numărul de epoci, am obținut valori pentru setul de antrenament între 94-99%, iar pentru setul de test am obținut acuratețe de 65-71%.

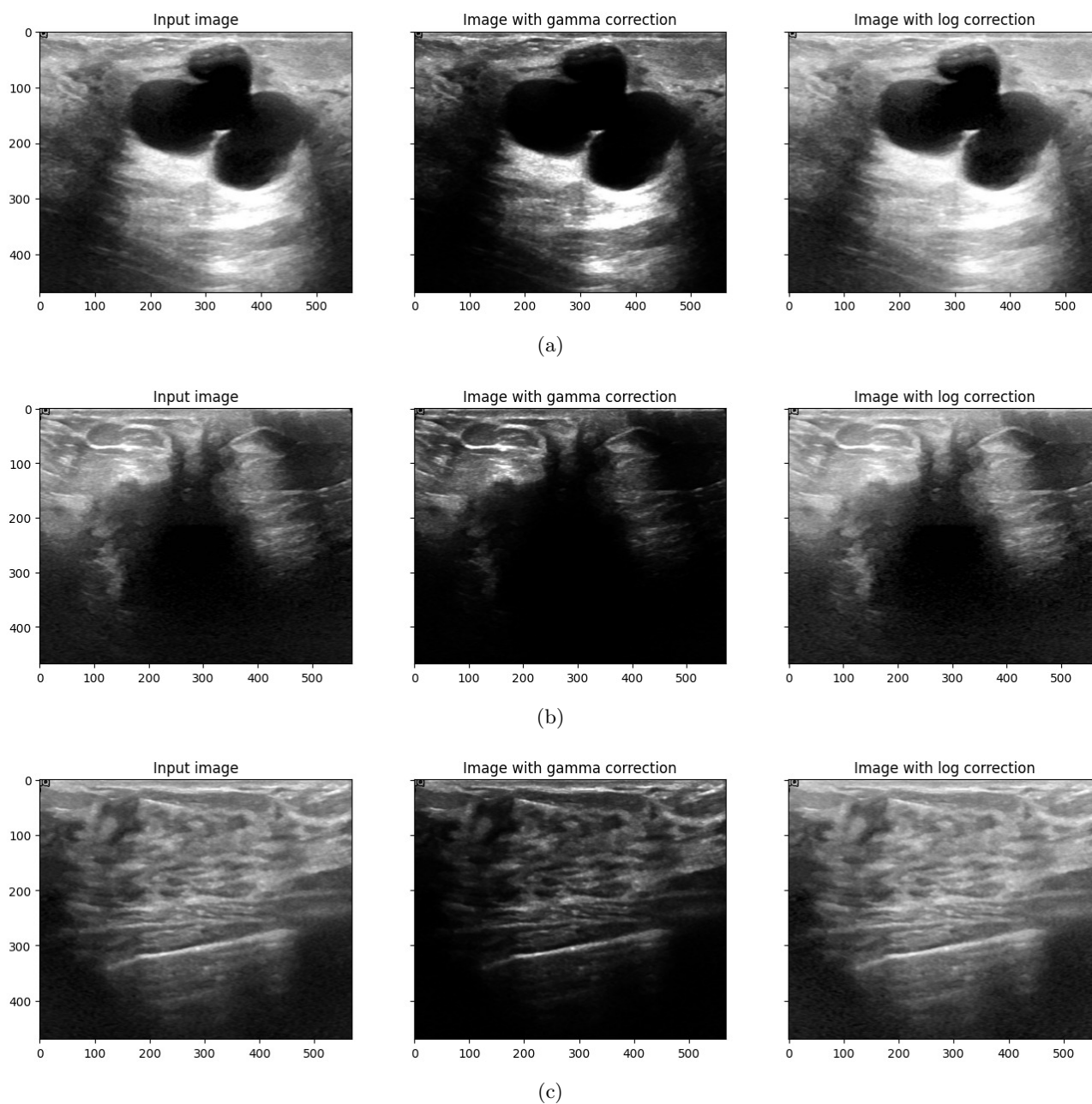
**Pasul 1 - Corecția Gamma/Corecția Logaritmică**

Figure 3.5: (a) benign (b) malign (c) normal



**Pasul 2 - Edges by Canny / Box Blur (netezire-mediare uniformă) / Gaussian Blur (mediere ponderată)**

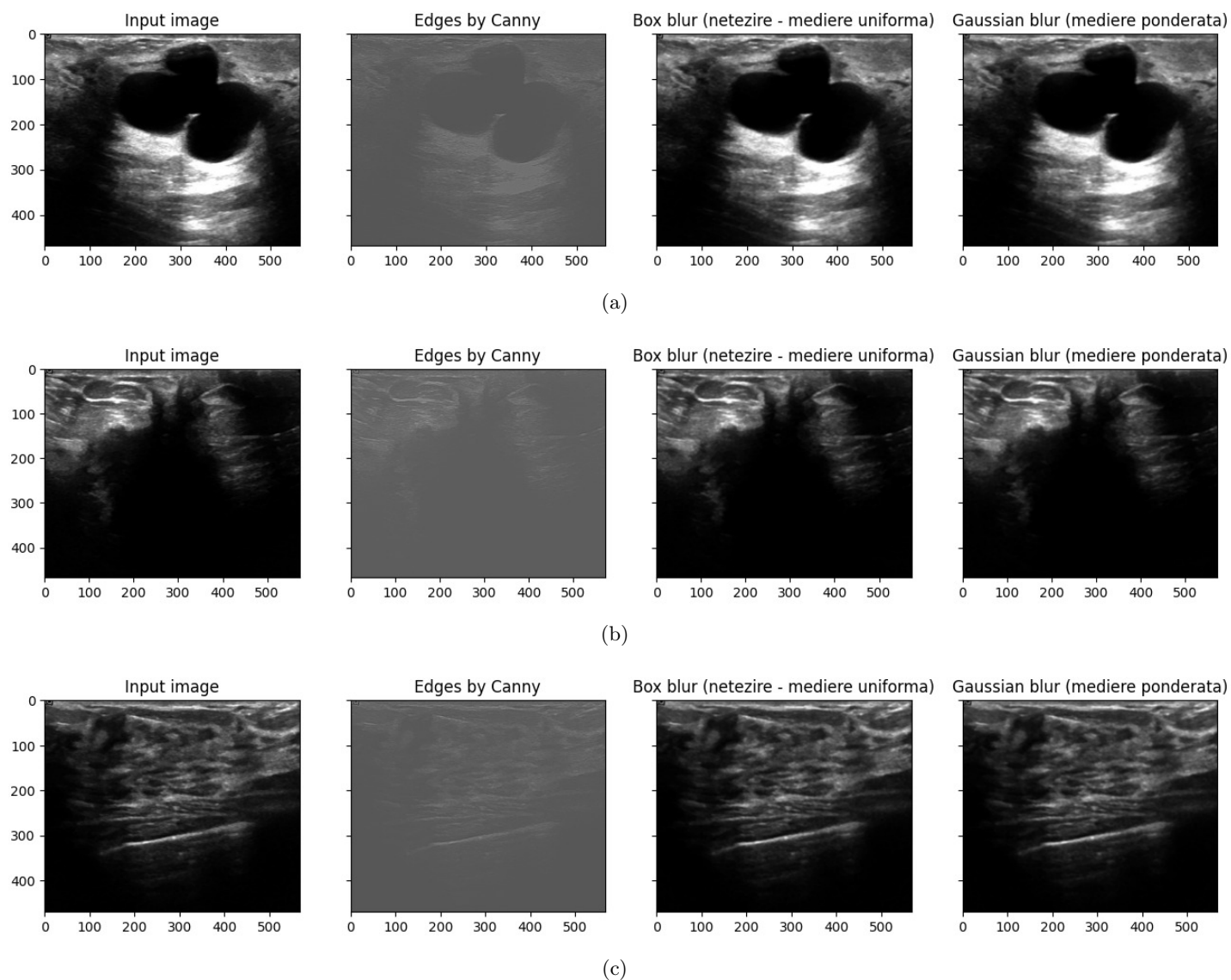
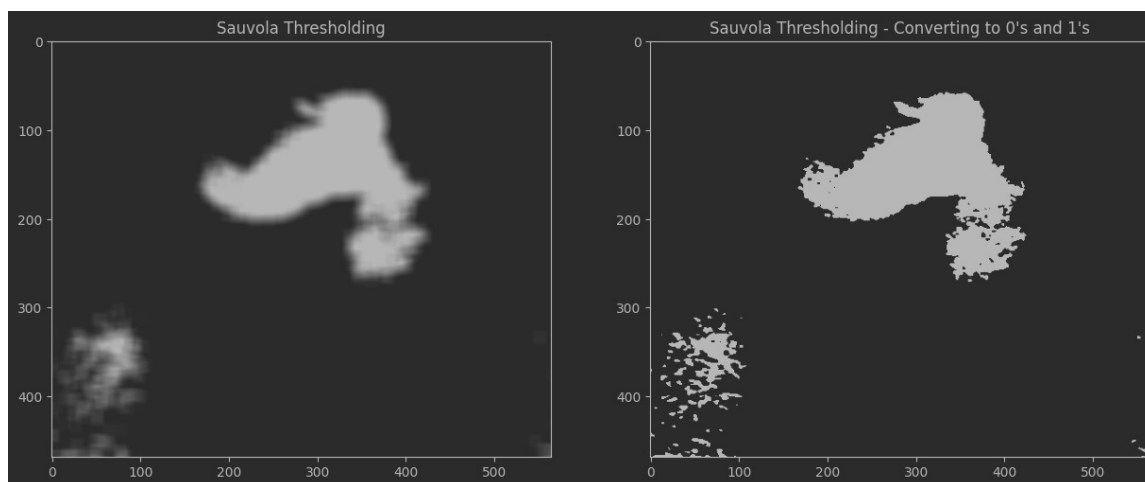
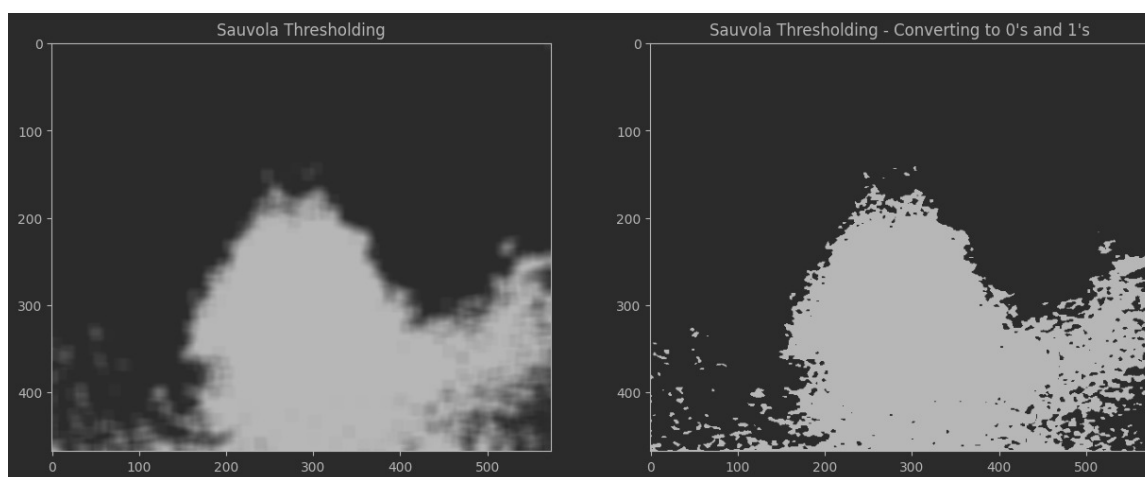


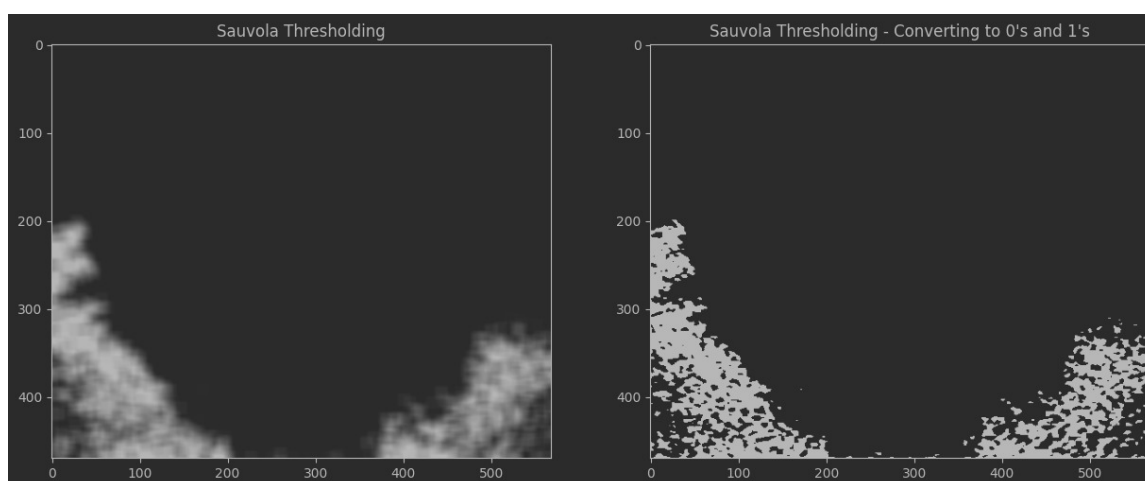
Figure 3.6: (a) benign (b) malign (c) normal

**Pasul 5 - Sauvola Tresholding**

(a)



(b)



(c)

Figure 3.7: (a) benign (b) malign (c) normal

## Augmentarea

Setul de date pe care îl folosim este breast-ultrasound-images-dataset: Dataset-BUSI-with-GT. În acest set de date, imaginile sunt împărțite în 3 categorii (numele folder-ului corespunde categoriei) : benign, malignant și normal.

Categoria "benign" conține 891 de imagini, dintre care aproximativ 446 imagini ultrasound și 445 maști.

Categoria "malignant" conține 421 de imagini, dintre care aproximativ 211 imagini ultrasound și 210 maști.

Categoria "normal" conține 266 de imagini, dintre care aproximativ 133 imagini ultrasound și 133 maști.

După cum am precizat anterior, în urma aplicării algoritmului folosit, pentru setul de antrenament am obținut acuratețea între 94-99%, iar pentru setul de test între 65-71%. Analizând aceste rezultate și discrepanța dintre rezultatul obținut pe setul de antrenament și cel de pe setul de test, am concluzionat că diferențele, în ceea ce privește numărul de imagini, dintre categoriile din setul de date, au un impact major asupra output-ului metodei noastre.

Am luat decizia că este necesară folosirea unui algoritm pentru augmentarea datelor, propunându-ne inițial analiza comportamentului algoritmului pe un set care conține 500 de imagini ultrasound (cu maști corespondente) pentru fiecare categorie (malignant, benign și normal).

Augmentarea reprezintă una dintre principalele metode folosite în procesul de creare de noi eșantioane de input-uri prin manipularea datelor originale existente. Există două situații frecvente în care se poate lua în considerare folosirea augmentării. În cazul primei situații, avem problema existenței unui set de date dezechilibrat, iar în cazul celei de-a doua situații avem problema existenței unui set de date prea mic. După cum se observă în input-ul algoritmului nostru, ne aflăm în ambele situații.

De obicei, augmentare se rezumă la o abordare în care se întoarce, se rotește sau se modifică aleatoriu nuanța, saturația, luminozitatea și contrastul unei imagini. Procedura este simplă și poate fi realizată fără mult efort. Dezavantajul utilizării acestor tehnici este că nu introduce noi date sintetice în model, ci doar include aceleași mostre într-o stare diferită. Așadar, modelul cunoaște deja aceste eșantioane, impactul asupra rezultatului fiind unul limitat.

Generarea de noi date realiste este o sarcină dificilă care include învățarea pentru a imita distribuția originală a setului nostru de date. Pentru asta există diferite tehnici de augmentare, cea folosită de noi fiind GAN (Rețele Adversiale Generative). Am ales această tehnică, deoarece astfel putem genera eșantioane realiste de date care pot îmbunătăți performanța modelului. GAN este compusă din două părți importante: Generator și Discriminator.

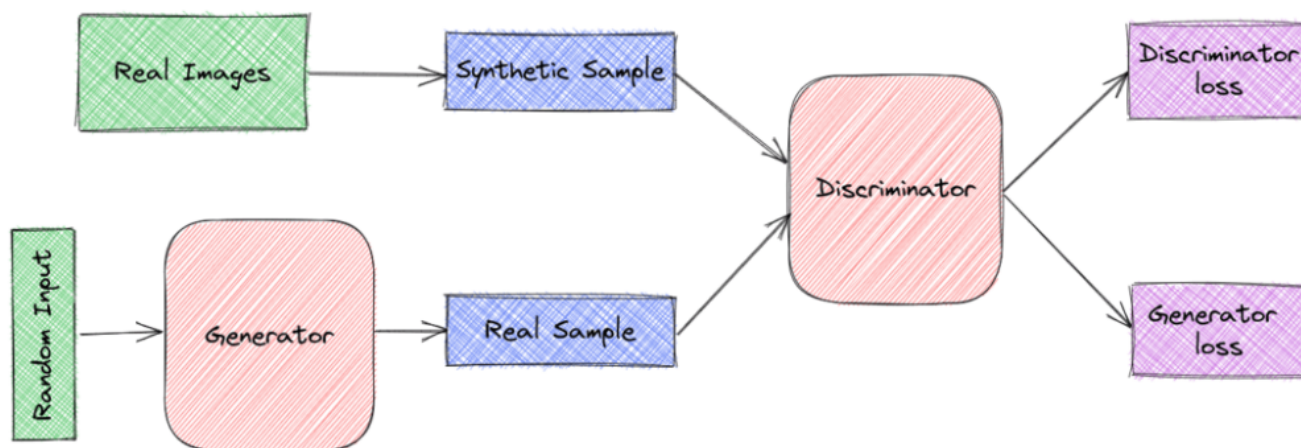
Generatorul este un CNN care învață să creeze noi date plauzibile. Acesta primește ca intrare un vector aleato-

riu de lungime fixă și învață să producă eşantioane care imită distribuția setului de date original. Eşantioanele generate devin apoi exemple negative pentru discriminator.

Discriminatorul este un CNN care învață să distingă datele sintetice ale generatorului de datele reale. Acesta primește ca intrare un eşantion și îl clasifică ca fiind "real" (provine din setul de date original) sau "fals" (provine de la generator). Discriminatorul penalizează generatorul pentru producerea de eşantioane neplauzibile.

Astfel, discriminatorul și generatorul se află într-un "joc" format din doi participanți, în care generatorul încearcă să ducă în eroare discriminatorul (să clasifice eşantioanele false ca fiind reale). Drept rezultat al augmentării, am obținut o acuratețe de 98% pe setul de antrenament și 86% pe setul de testare. Așadar, rezultatul este unul satisfăcător.

### Augmentare



### Segmentarea folosind UNet

U-Net este o rețea neuronală de convoluție care a fost implementată pentru a fi folosită în segmentarea imaginilor biomedicale. Ea constă într-o rețea plină cu o arhitectură modificată astfel încât să funcționeze cu mai puține imagini în setul de antrenament, în timp ce rezultatele să fie destul de precise. Pentru dataset-ul pe care am implementat algoritmul de U-Net, acuratețea medie obținută cu această rețea de către alți cercetători a fost de 90%.

Pe lângă faptul că clasifică imaginea dată ca input, U-Net este capabil și să localizeze anomalia. Noi am folosit U-Net pentru a crea măști ale tumorilor care fac parte din imaginile ultrasound, având drept valori "adevarate" de referință măști deja create în setul de date. Am folosit rețeaua următoare, deoarece cei care au folosit acest dataset au obținut o acuratețe foarte mare cu aceasta. Diagrama straturilor rețelei este prezentată mai jos:

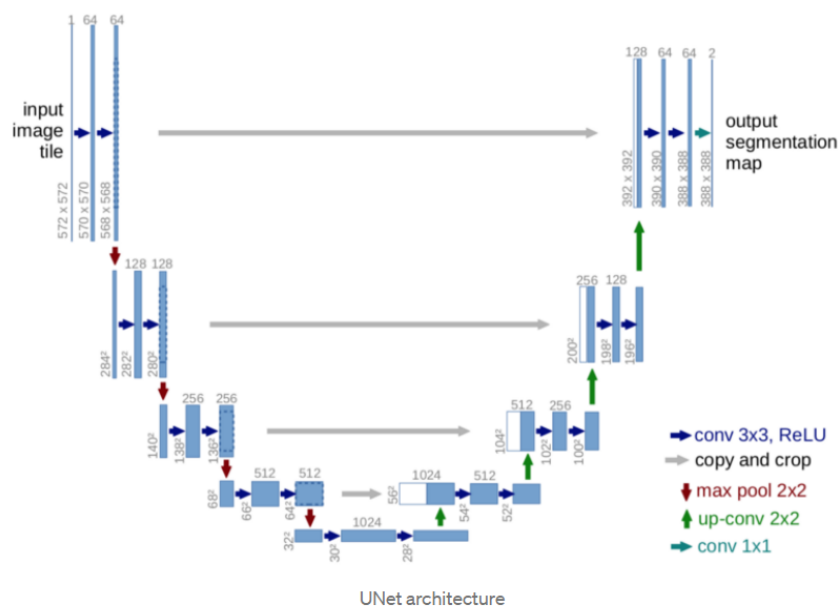


Figure 3.8: U-Net

Rețeaua conține 23 straturi de convoluție. Prima parte (cea din stânga, cu un trend de coborâre) este numită partea de contractare, în care are loc procesul general de convoluție. Fiecare pas este compus dintr-o convoluție 3x3 fără padding, ReLU și un 2x2 maxpooling. Partea din dreapta se numește partea de expansiune ("up-convolution"), care la început copiază feature map-ul din pasul de contractare corespundent, îl croșează, apoi urmează straturi 2D de convoluție urmate de ReLU. Practic, în prima parte imaginea se "mărește" (numărul de feature-uri se dublează la fiecare pas), iar în a doua parte ea se "micșorează" la loc (numărul de feature-uri se contractează). La toți acești pași am adăugat și dropout, pentru a nu avea bias pe setul de antrenament.

Aceasta nu este o rețea neuronală secvențială, întrucât, în pașii de expansiune, se concatenează cu feature map-ul din pasul de contractare corespundent, pentru a nu pierde pixelii de pe margine.

În urma testelor am obținut acuratețe de 94% pe setul de date originale + augmentate, și tot 94% pentru când am antrenat doar cu setul original. În schimb, am observat ca măștile obținute folosind doar setul de date original sunt mult mai precise decât cele folosind și imagini augmentate. În figurile de mai sus se poate vedea diferența prezentă în majoritatea segmentărilor făcute. De asemenea, pentru segmentarea realizată pe imagini augmentate am obținut un dice de 0.8911, iar pentru cea pe imagini neaugmentate 0.9015. Fiind o zona foarte mica din imagine, tumora identificata pe o arie mai restransa nu contribuie la o schimbare radicala a valorii dice, de aceea cele doua valori obtinute sunt apropiate si nu reflecta faptul ca segmentarea augmentata este mai putin precisa. Pe prima linie din imagine se află măștile prezise de algoritmul, iar pe a doua linie sunt măștile care trebuiau prezise.

## Segmentarea folosind U-Net

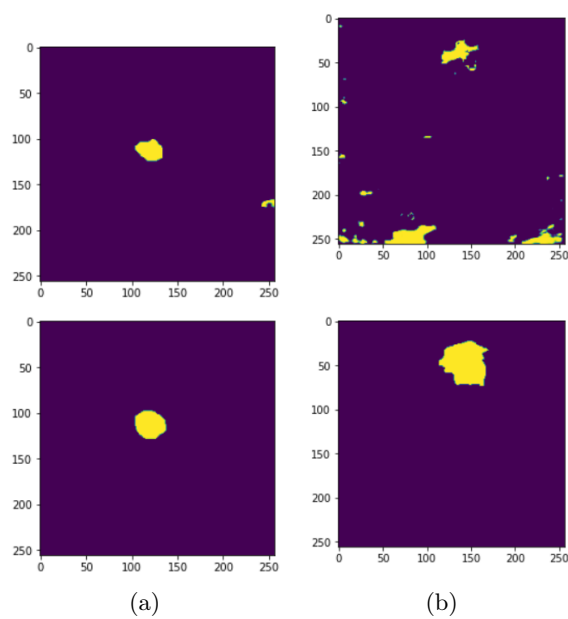


Figure 3.9: (a) fără augmentare (b) cu augmentare

## Chapter 4

# Aplicație (Studiu de caz)

### 4.1 Descrierea aplicației și funcționalitățile de bază

**Funcționalități:** Aplicația are capacitatea de a returna utilizatorului segmentarea zonei care prezintă anomalii (pe baza unei radiografii introduse) și clasificarea tumorii (pe baza unui document care conține segmentarea)

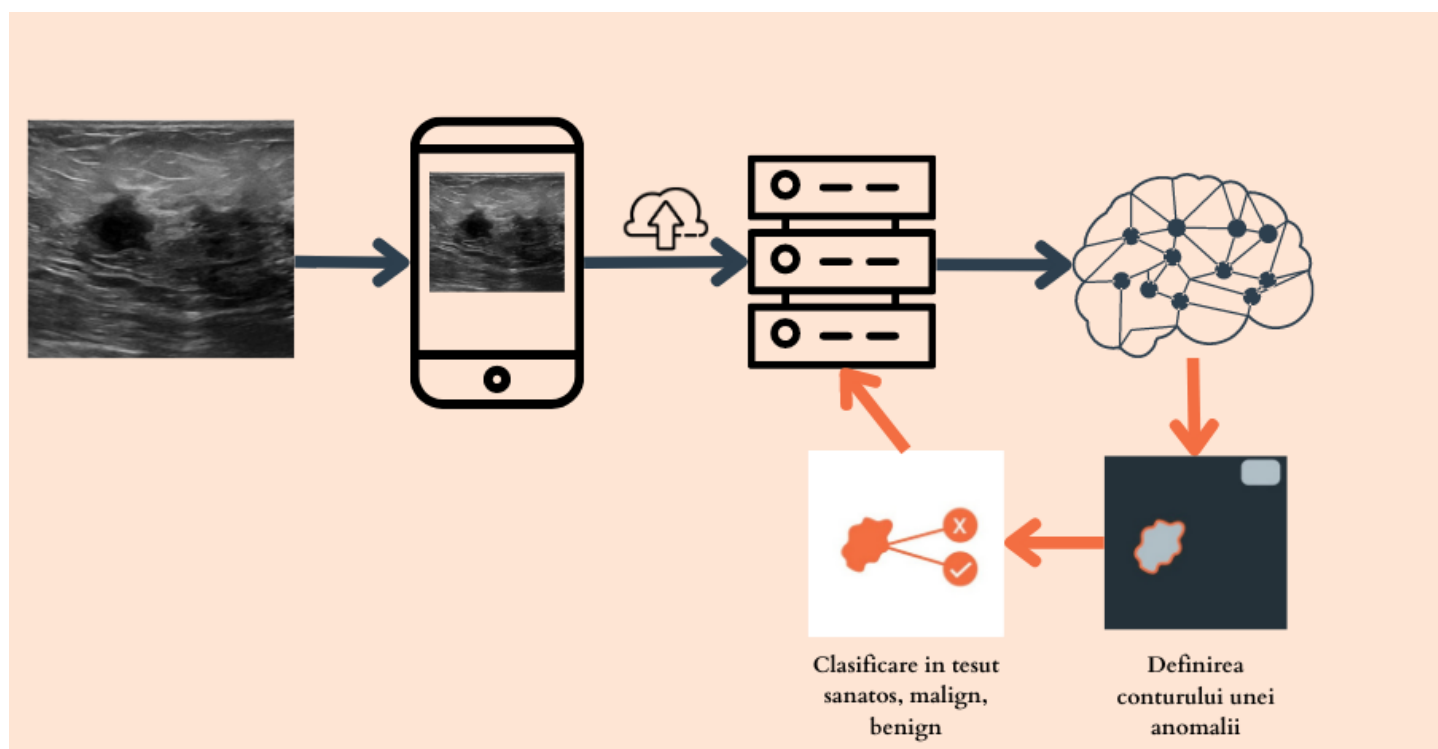


Figure 4.1: Flow-ul aplicației

## 4.2 Mindmap

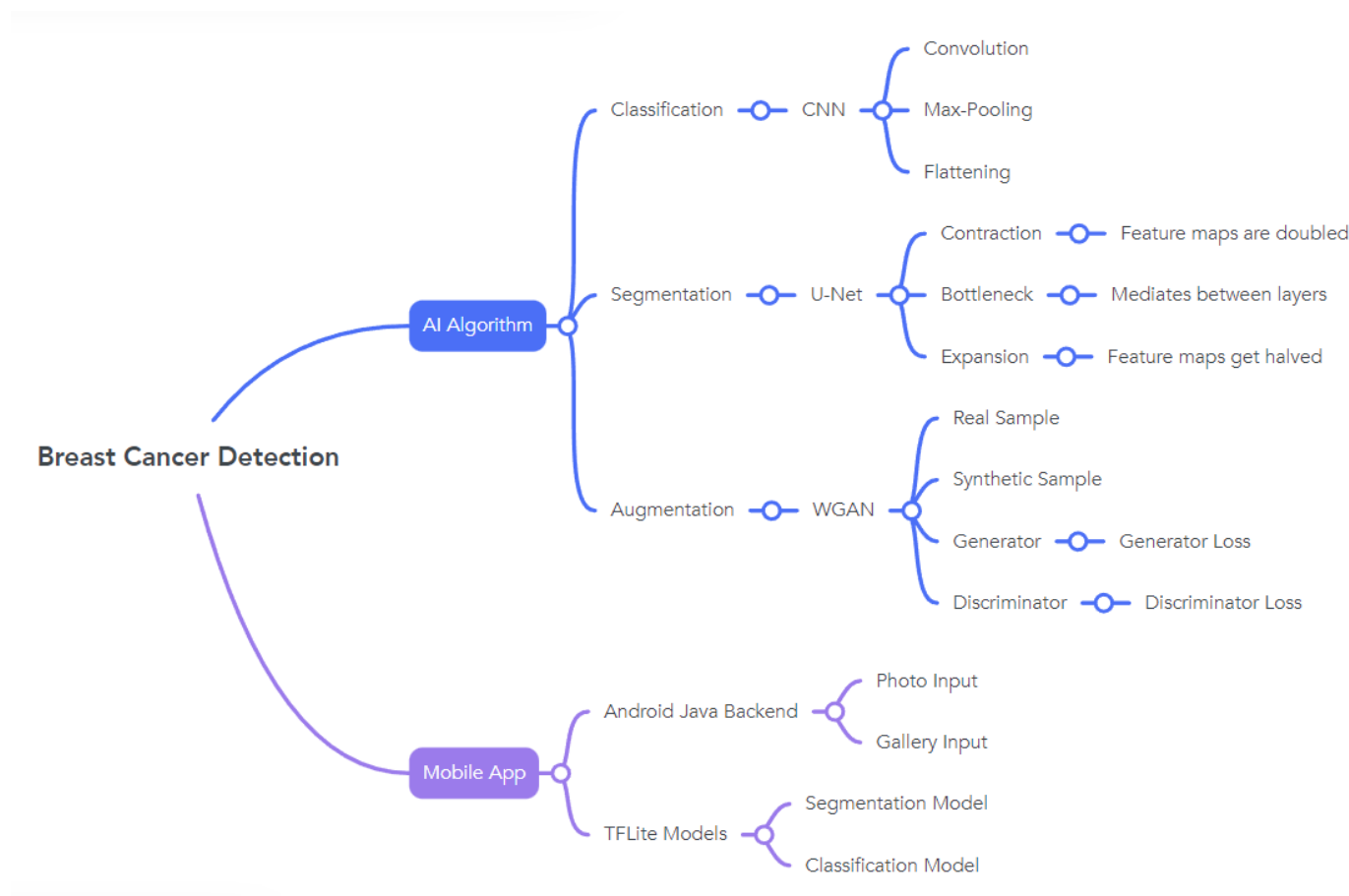


Figure 4.2: Mindmap

## 4.3 Designul aplicației

### Caz de utilizare

Utilizatorul are la dispoziție două butoane: unul care îi permite să facă o fotografie folosind camera telefonului și unul prin care poate încărca o fotografie deja existentă în galeria proprie. Această fotografie trebuie să corespundă unei imagini medicale cu ultrasunet. Ulterior, va putea vizualiza regiunile în care apar anomalii, precum și clasificarea țesutului (normal, malign, benign).

## 4.4 Implementare

Pentru acest experiment am implementat o aplicație disponibilă pe telefoanele mobile cu sistem de operare Android. Pentru dezvoltarea aplicației, am utilizat ca mediu de dezvoltare Android Studio, iar ca limbaj de programare am



ales Java. Pentru partea de interfata cu utilizatorul am dezvoltat elementele grafice cu ajutorul XML. Pentru integrarea modelelor inteligente am folosit libraria tensorflow lite.

## 4.5 Set de date

Datele colectate la momentul initial [2] includ imagini cu ultrasunete mamare la femei cu varste cuprinse intre 25 si 75 de ani. Aceste date au fost colectate in 2018. Numarul pacientilor este de 600, reprezentand persoane de sex feminin. Setul de date este format din 780 de imagini cu o dimensiune medie a imaginii de 500\*500 pixeli. Imaginile sunt in format PNG. Imaginile sunt clasificate in trei clase (normale, benigne si maligne), distribuite in directoarele corespunzatoare numelui.

## 4.6 Rezultate

Am reprezentat rezultatele obtinute cu ajutorul a doua tabele. Primul tabel corespunde rezultatelor obtinute pentru etapa de segmentare, iar cel de-al doilea tabel corespunde etapei de clasificare.

Pentru segmentare am ales sa folosim ca metrica de evaluare coeficientul Dice, avand in vedere necesitatea evaluarii nivelului de similaritate dintre masca rezultata in urma aplicarii algoritmului nostru si masca reala pe care o gasim in setul initial de date. Valoarea coeficientului Dice este in intervalul  $(0, 1)$ . Cu cat aceasta valoare se apropie mai mult de 1, cu atat similaritatea dintre cele doua imagini este mai mare. Coeficientul Dice reprezintă, în cazul tabelelor de mai jos, media pe tot setul de date.


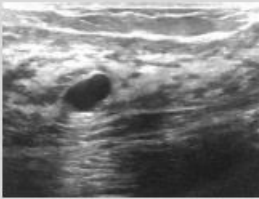


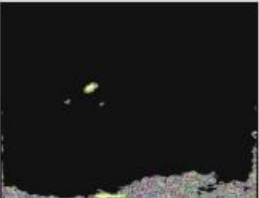

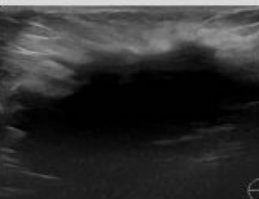
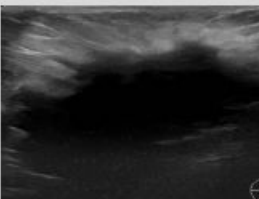



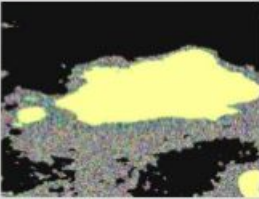
Segmentare	Augumentat	Neaugumentat
Dice coefficient	0.8911	0.9015
Ultrasound benign		
Masca reala benign		
Masca prezisa benign		
Ultrasound malign		
Masca reala malign		
Masca prezisa malign		

Figure 4.3: Rezultate segmentare

Pentru clasificare am folosit ca metrica de evaluare acuratetea. Acuratetea are valori cuprinse in intervalul [0%, 100%]. Cu cat valoarea aceasta este mai aproape de 100%, cu atat algoritmul de clasificare este mai performant.

Clasificare(masti reale)	Augmentat	Neaugmentat
Acuratete	86%	65%

Figure 4.4: Rezultate clasificare

De asemenea, am calculat timpul de inferenta, atat pentru segmentare, cat si pentru clasificare. Acesta reprezinta timpul necesar algoritmului pentru a aplica modelul rețelei neuronale antrenate, pe noi date de intrare.

Timp inferență – segmentare (secunde)	Model neaugmentat	Model augmentat
Imagini augmentate	0.17128601074	0.2148986577987671
Imagini neaugmentate	0.18496899604	0.17141962051391602

Figure 4.5: Inferenta segmentare

Timp inferență – clasificare (secunde)	Model neaugmentat	Model augmentat
Măști augmentate	0.00617516836	0.00686089198
Măști neaugmentate	0.0105894407	0.00779174169

Figure 4.6: Inferenta clasificare

Pentru a calcula acuratetea finala a algoritmului, am unit modelul de segmentare cu cel de clasificare. Mai jos se poate observa reprezentarea modelului final pe straturi, unde 'Functional' este modelul de segmentare si 'Sequential' este modelul de clasificare.

Model: "model\_1"

Layer (type)	Output Shape	Param #	Connected to
input_105 (InputLayer)	[(None, 256, 256, 1)]	0	
model (Functional)	(None, 256, 256, 1)	31401345	input_105[0][0] resizing_62[0][0]
resizing_62 (Resizing)	(None, 400, 400, 1)	0	model[96][0]
concatenate_48 (Concatenate)	(None, 400, 400, 2)	0	resizing_62[0][0] model[97][0]
reshape_31 (Reshape)	(None, 2, 400, 400)	0	concatenate_48[0][0]
resizing_63 (Resizing)	(None, 3, 400, 400)	0	reshape_31[0][0]
permute_6 (Permute)	(None, 400, 400, 3)	0	resizing_63[0][0]
sequential (Sequential)	(None, 3)	75523107	permute_6[0][0]
Total params: 106,924,452			
Trainable params: 106,912,676			
Non-trainable params: 11,776			

Figure 4.7: Merged Segmentation and Classification models

Segmentare/ Clasificare	Augmentat		Neaugmentat	
	<u>Train</u>	Test	<u>Train</u>	Test
Augmentat	99.4%	71.8%	51.5%	50%
Neaugmentat	57.2%	68.7%	60.3%	43.7%

Figure 4.8: Comparatie acuratete

## Chapter 5

# Concluzii

În urma experimentului realizat, ne-am apropiat de rezultatele stabilite ca obiectiv.

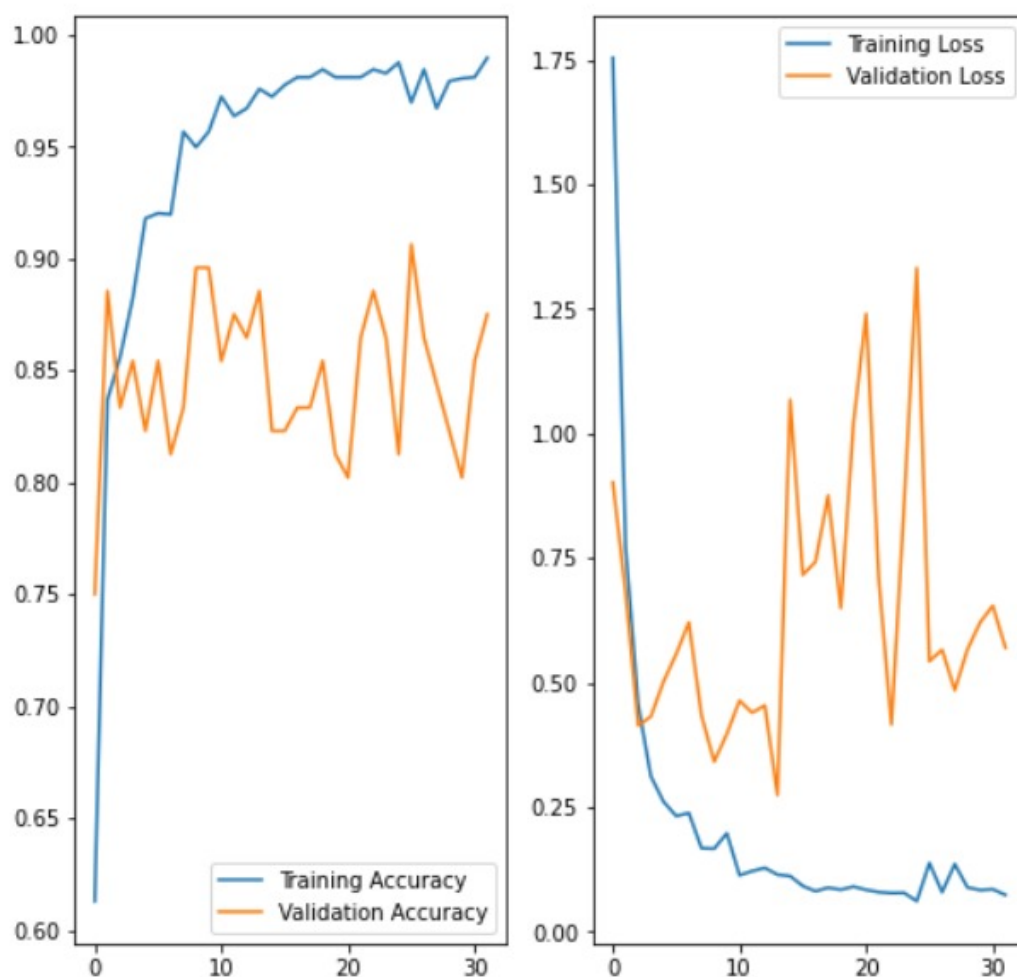


Figure 5.1: Clasificare pe date augmentate

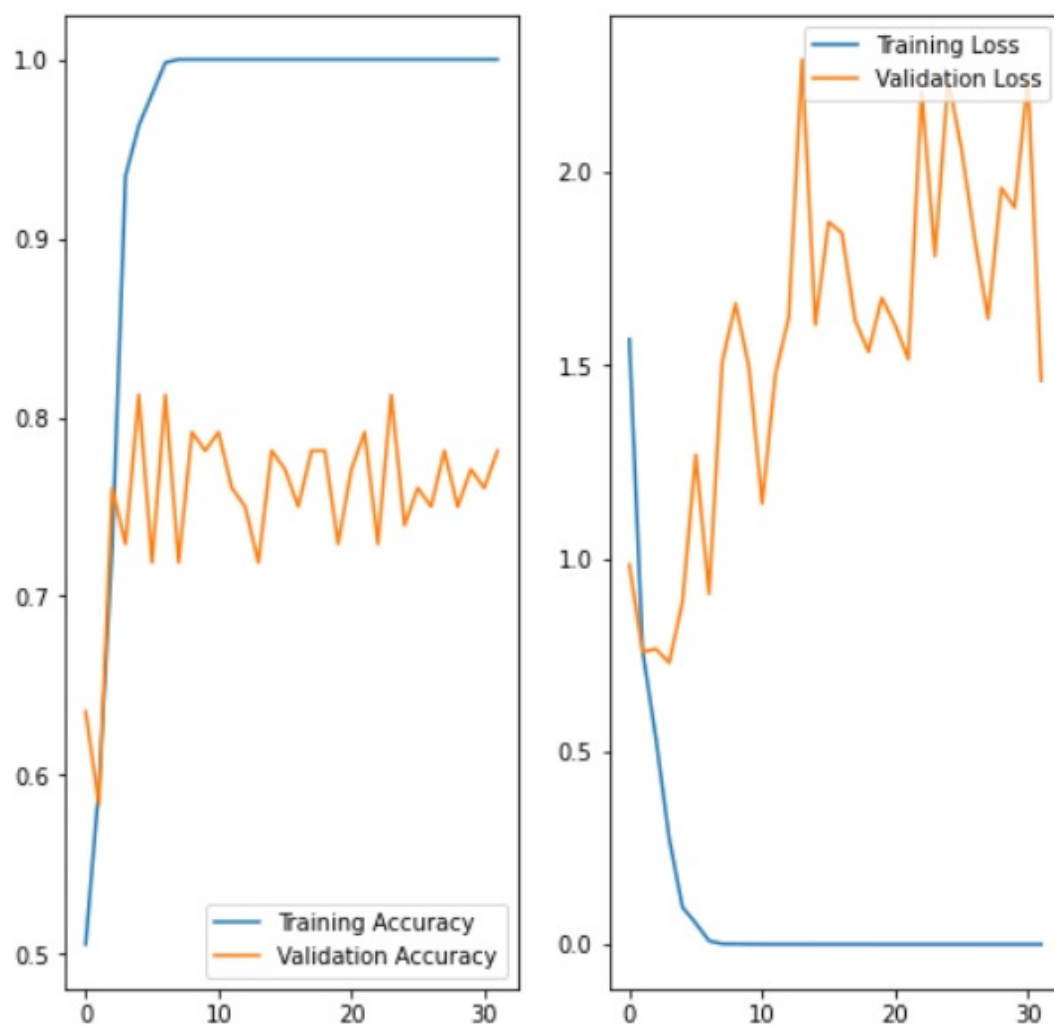


Figure 5.2: Clasificare pe date neaugmentate

# References

Breast mass classification from mammograms using deep convolutional neural networks. in 30th conference on neural information processing systems (nips 2016).

Khaled H Fahmy A Al-Dhabyani W, Gomaa M. Dataset of breast ultrasound images, 2020.

K Bakkour, I. I. Afdel. Breast tumor classification based on deep convolutional neural networks. In *Advanced Technologies for Signal and Image Processing (ATSIP)*, pages 1–6. IEEE, 2017.

Liu Z. Hannun A. Hang, W. Glimpsenet: Attentional methods for full-image mammogram diagnosis. Technical Report Stanford AI Lab Internal Report, Stanford University, 2017.

Hwejin Jung, Bumsoo Kim, Inyeop Lee, Minhwan Yoo, Junhyun Lee, Sooyoun Ham, Okhee Woo, and Jaewoo Kang. Detection of masses in mammograms using a one-stage object detector based on a deep convolutional neural network. *PloS one*, 13(9):e0203355, 2018.

Wenya Linda Bi MD, Ahmed Hosny MS, Matthew B. Schabath PhD, Maryellen L. Giger PhD, Nicolai J. Birkbak PhD, Alireza Mehrtash MSc, Tavis Allison BS, Omar Arnaout MD, Christopher Abbosh MD, Ian F. Dunn MD, Raymond H. Mak MD, Rulla M. Tamimi PhD, Clare M. Tempany MD, Charles Swanton MD PhD, Udo Hoffmann MD, Lawrence H. Schwartz MD, Robert J. Gillies MD, Raymond Y. Huang MD PhD, and Hugo J. W. L. Aerts PhD. Artificial intelligence in cancer imaging: Clinical challenges and applications. *CA A Cancer J Clin*, 69(2):127–157, 2019.

Olaide N. Oyelade and Absalom E. Ezugwu. A deep learning model using data augmentation for detection of architectural distortion in whole and patches of images. *Biomedical Signal Processing and Control*, 65:102366, 2021.

Dina A Ragab, Maha Sharkas, Stephen Marshall, and Jinchang Ren. Breast cancer detection using deep convolutional neural networks and support vector machines. *peerj* 7: e6201, 2019.

R. Roslidar, K. Saddami, F. Arni, M. Syukri, and K. Munadi. A study of fine-tuning cnn models based on

thermal imaging for breast cancer classification. In *International Conference on Cybernetics and Computational Intelligence*, pages 77–81. IEEE, 2019.

Fujioka T, Mori M, Kubota K, Oyama J, Yamaga E, Yashima Y, Katsuta L, Nomura K, Nara M, Oda G, Nakagawa T, Kitazume Y, and Tateishi U. The utility of deep learning in breast ultrasonic imaging: A review. *Diagnostics (Basel)*, 1005:10–12, 2020.