



Școala doctorală interdisciplinară
Domeniul de doctorat: Calculatoare și Tehnologia Informației

TEZĂ DE DOCTORAT

3D Analysis of the Normal and Pathological Coronary Morphology

**Analiza 3D a morfologiei coronariene
normale și patologice**

doctorand:

Alexandru DOROBANȚIU

Conducător Doctorat:

Prof. Dr. Ing. Remus BRAD



Focusul acestei teze este procesul de segmentare automată a liniei centrale coronariene din angiografia tomografiei computerizate cardiace (CCTA). Aceasta aparține domeniului imagisticii medicale, care, la momentul în care am ales tema, era un subiect important în comunitatea științifică de imagistică medicală [1].

Lucrarea este împărțită în trei capitole. **Capitolul 1** este axat pe extragerea informațiilor din imagini și îmbunătățirea compresiei de date a imaginilor. Odată cu creșterea numărului de dispozitive de achiziție a imaginilor, inclusiv a camerelor și a instrumentelor de imagistică medicală, crește și cantitatea de informații necesară pentru stocarea pe termen lung. În acest capitol oferim o descriere detaliată a software-ului de compresie fără pierderi de ultimă generație PAQ8PX, cu aplicare pe compresia imaginilor în tonuri de gri. Propunem un nou algoritm de învățare continuă pentru prezicerea probabilității biților dintr-un flux de date. Apoi, descriem integrarea algoritmului în modelul de imagine al PAQ8PX. Pentru a demonstra îmbunătățirile, testăm noul software pe trei benchmark-uri publice. Rezultatele experimentale arată scoruri mai bune pentru toate seturile de date.

Detectarea muchiilor obiectelor dintr-o imagine are un rol important în multe sisteme de viziune computerizată. Prin urmare, în **capitolul 2**, propunem un nou algoritm independent de domeniu pentru predicția probabilităților, care este bazat pe informațiile contextuale disponibile, și apoi aplicăm algoritmul pentru estimarea probabilității pixelilor de a aparține unei margini. Sunt folosite valorile pixelilor înconjurători ca și contexte locale. Următorul pas descris este testarea diferitelor transformări de imagine ca și straturi de intrare, cum ar fi detectorul de contur Canny. Propunem două arhitecturi diferite, una cu un singur strat și una multistrat, care abordează problema scalării prin crearea de rezultate secundare redimensionate, și apoi combinarea acestora printr-un strat de regresie logistică. Am testat abordarea noastră pe setul de date BSDS500 de detectare a conturilor cu rezultate optimiste.

În **capitolul 3**, accentul este pus pe segmentarea de imagini medicale, unde modelul coronarian de tip mesh, obținut din reconstrucția tridimensională folosind secvența de imagini produse prin tomografie computerizată (CT) poate fi folosit mai departe pentru a obține informații utile de diagnostic, cum ar fi extragerea proiecției lumenului (dezvoltare plană de-a lungul unei artere). Propunem o extracție automată a liniei centrale coronariene din angiografia tomografiei computerizate cardiace, prin crearea unei versiuni 3D a arhitecturii U-Net, antrenată cu o funcție de pierdere nouă și cu patch-uri augmentate. Am obținut rezultate promițătoare în ceea ce privește acuratețea (între 90-95%) și suprapunerea (între 90-94%) cu diferite configurații de antrenament ale rețelei pe datele din benchmark-ul Rotterdam de extragere a liniei centrale a arterelor coronariene. De asemenea, am demonstrat capacitatea rețelei propuse de a învăța, în ciuda debalansării imense între cele două clase și a adnotării rare prezente în datele de antrenament.

Capitolul 1

În acest capitol descriem metoda de compresie de imagini de ultimă generație numită PAQ8PX, și introducem un nou algoritm pentru învățarea automată continuă. Am adaptat implementarea metodei propuse, integrând-o cu PAQ8PX, ceea ce a dus la un model mai bun de predicție pentru imagini în tonuri de gri. Am testat implementarea și am obținut îmbunătățiri pentru patru seturi de date din trei benchmark-uri diferite. Cercetarea a fost publicată în [2].

PAQ este o serie de programe experimentale de compresie fără pierderi a datelor, care vizează cel mai bun raport de compresie în defavoarea dimensiunii resurselor de calcul și fără păstrarea compatibilității cu versiunile mai vechi. Comprimarea unui fișier trece prin 4 etape

principale: preprocesare, predicția modelului, combinarea contextuală, și rafinarea probabilităților. PAQ include modelarea contextului pentru multe tipuri de date, inclusiv imagini. Contextele sunt definite folosind modelare specifică, care include modelarea directă, modelarea indirectă și modelarea regresiei liniare. Predicțiile de probabilitate din modelele contextuale sunt convertite într-o singură predicție folosind combinarea contextuală, care este un caz particular al unei rețele neuronale de tipul Gated Linear Network. Rezultatul combinării este apoi rafinat printr-o rețea de funcții de transfer adaptive.

Metoda propusă pentru îmbunătățirea metodei de predicție a probabilității este inspirată din Ensemble learning, iar ideea din spatele algoritmului este de a codifica probabilitățile într-o structură asemănătoare unei memorii. Pentru toleranță la zgomot, nu toate intrările ar trebui să găsească o potrivire în memorie.

În scopul modelării contextelor pentru prezicerea fiecărui bit individual al pixelilor, alegem o metodă simplă: folosim raze în patru direcții cu diferite lungimi și derivatele cuantizate de-a lungul acestora. Obținem o valoare din memorie pentru fiecare context. Facem apoi media tuturor valorilor obținute, apoi convertim media într-o probabilitate folosind funcția sigmoidă. Pentru actualizarea valorilor din memorie, folosim reinforcement learning. Deoarece nu cunoaștem adevărata valoare a probabilității ca un bit să fie 0 sau 1 într-un context dat, nu putem folosi învățarea supervizată (supervised learning). Propagăm înapoi rezultatul binar prin rețea și încercăm să minimizăm pierderea logistică cumulativă într-o manieră continuă. Pierderea pătratică poate fi, de asemenea, utilizată, dar încercăm să minimizăm spațiul de codare irosit. Pentru a transmite informațiile de combinare nodurilor ansamblului, propunem o funcție dublă de minimizare a obiectivului: în ceea ce privește ieșirea rețelei - eroarea globală, și în ceea ce privește ieșirea nodurilor individuale (predicții reziduale) - eroarea locală. Toate valorile corespunzătoare din memorie sunt apoi actualizate prin scăderea erorii lor locale și globale. În loc să actualizăm ponderile combinării, actualizăm direct valorile care contribuie la medie. Nu avem niciun strat adițional care să separe ponderile de context de probabilitățile de intrare, făcând diferită metoda propusă de algoritmul de combinare contextuală.

Algoritmul nu face nici o presupunere asupra unui mod de a structura memoria. Nu restricționăm accesul la memorie la o schemă precisă, ci sugerăm trei abordări: căutare simplă, căutare etichetată și căutare locală.

Am implementat un program pentru evaluarea îmbunătățirilor de compresie a datelor pentru pixeli individuali pe mecanismul de predicție PAQ8PX. Programul primește ca intrare două matrici provenind din două versiuni diferite de predictor, matrici care reprezintă costul de codare pentru fiecare bit al fluxului de intrare. Costul pentru fiecare grup de 8 biți consecutivi (sau 24 de biți pentru imaginile color) poate fi convertit la o singură valoare și limitat la intervalul 0-255 pentru a crea o matrice de pierdere la nivel de pixel pentru o versiune a predictorului. Funcția de conversie poate fi ajustată pentru a sublinia diferitele aspecte ale predicției și, astfel, ajută la estimarea câștigurilor potențiale în cazul în care se schimbă parametrii modelului. Programul furnizează și alte informații utile, cum ar fi de câte ori un predictor a avut o pierdere mai bună și de câte ori un predictor a avut o pierdere mai bună pe o anumită poziție a bitului în pixel.

Pentru a testa eficiența algoritmului, am aplicat versiunea extinsă a algoritmului PAQ8PX cu memoria contextuală pe patru seturi de test. Prezentăm rezultatele pentru toate imaginile din seturile de date pentru a demonstra că nu am ajustat algoritmul doar la câteva imagini selectate. Imaginile sunt comprimate individual (nu într-o arhivă solidă) pentru a preveni reutilizarea corelațiilor.

Contribuția principală a acestui capitol este un algoritm agnostic de aplicație pentru prezicerea probabilităților pe baza informațiilor contextuale disponibile, cu învățarea realizată într-o manieră continuă. Utilitatea algoritmului este demonstrată prin integrarea acestuia cu algoritmul PAQ8PX și testarea acestuia pe mai multe benchmark-uri de compresie de imagini. Rezultatele arată o îmbunătățire generală a raportului de compresie pentru toate seturile de date fără a avea caracteristici particularizate datelor. O diferență importantă față de algoritmi existenți de Ensemble learning este că, în algoritmul nostru, presupunem că diferite contexte se

aplică împreună și predicția finală beneficiază de sinergia predicțiilor reziduale, spre deosebire de ansamblurile care pleacă de la ipoteza independenței constituenților.

Capitolul 2

În acest capitol generalizăm algoritmul pentru învățarea automată descris în capitolul anterior, care stă la bază pentru diferite aplicații. Oferim o descriere a algoritmului și subliniem diferențele implementărilor testate. Pentru a dovedi eficacitatea, am aplicat și testat algoritmul pe un benchmark de detectare a conturului obiectelor din imagini. Până în prezent, algoritmul a fost aplicat doar pentru imagini 2D. Cercetarea a fost publicată în [3].

Detectarea conturului imaginilor a fost subiect de cercetare de mai mulți ani, cu lucrări publicate încă din 1975 [4]. De atunci, un număr mare de tehnici au fost abordate, vizând diferite aspecte ale detectării marginilor obiectelor, cum ar fi contururile închise, percepția asemănătoare cu a omului, sau detectarea rapidă. Berkeley Computer Vision Group oferă un benchmark public pentru detectarea conturului și segmentarea imaginilor. Performanța este evaluată prin măsurarea preciziei și a recall-ului și combinarea acestora prin media armonică într-un scor F1 (măsură F).

Modelarea contextuală descrie modul în care informațiile de context sunt structurate și menținute. În funcție de problemă, sunt utile diferite tipuri de contexte discriminatoare. Contextul local se referă la aspectul datelor examinate, spre deosebire de valoarea contextului, care reprezintă valoarea numerică a aceluia context. Valoarea contextului va fi utilizată pentru accesarea structurii memoriei. Memoria ia o valoare de context ca intrare și scoate o valoare utilizată pentru calcularea probabilității de ieșire a apartenenței la o anumită clasă.

Atunci când se discută despre detectarea conturului și se descrie dacă un pixel aparține sau nu unei margini, unul din cele mai importante aspecte care trebuie luate în considerare este cel al pixelilor vecini. Putem defini contexte ca raze, pornind de la pixelul focalizat și mergând în linie dreaptă într-o direcție dată. Razele sunt definite prin direcție și lungime.

Extindem tehnica de predicție introdusă în capitolul 1 cu o funcție de transfer adaptivă pentru a rafina rezultatul. Funcția de transfer adaptivă, numită și estimarea secundară a simbolului, este utilizată pentru a ajusta fin o probabilitate și funcționează în felul următor: se selectează un set de puncte de interpolare în funcție de o valoare contextuală, se găsesc cele două puncte index între care cade valoarea de intrare, apoi se emite probabilitatea ca medie ponderată a celor două valori ale celor două puncte, unde ponderea este selectată de distanța valorii de intrare față de cele două puncte.

Pentru cazul detectării muchiilor, putem utiliza atât Reinforcement Learning, folosind clasa adevărată ca valoare de zero sau unu, cât și învățarea supervizată, folosind probabilitatea ca pixelul să fie selectat ca parte a clasei adevărate dintr-o marcă a unui utilizator din mai mulți posibili. Eroarea globală este calculată folosind probabilitatea finală care a fost rafinată de funcția de transfer adaptivă. Acest lucru nu este obligatoriu, dar testele noastre arată rezultate mai bune atunci când este utilizată probabilitatea rafinată. Acest lucru poate fi văzut ca permiterea modelului să învețe ceva care poate fi corect.

Ca o optimizare pentru toate cele trei tipuri de memorie, dacă funcția de transfer adaptivă este proiectată să extindă spațiul probabilității înainte de cuantizare, cei doi pași care restrâng spațiul (după aplicarea mediei) și apoi care îl extind (înainte de cuantizare) se anulează reciproc, ceea ce înseamnă că cele două operații care sunt destul de costisitoare computațional pot fi omise, iar valoarea de intrare este cuantizată direct. Faza de antrenament a memoriei duce la dependențe de date între rulări succesive. Dependențele provin din accesarea acelorași locații de memorie și, de asemenea, din accesarea funcției de transfer. Cu toate acestea, există loc de îmbunătățire. De exemplu, calculul valorilor contextelor poate fi complet paralelizat. În funcție de tipul de memorie utilizat, locațiile din memorie pot fi, de asemenea, calculate în paralel. În faza de testare, nu mai există dependențe de citire după scriere, astfel încât întregul proces poate fi rulat în paralel.

Am aplicat algoritmul de memorie contextuală pentru un program detector de contur. Când se analizează imagini bidimensionale, imaginile color au mai multe straturi descrise de spațiul culorilor RGB. Prin urmare, trei straturi sunt intrare pentru algoritm. Aceste straturi sunt preprocesate folosind un lanț de preprocesoare. Am folosit un filtru Gauss pentru eliminarea zgomotului din imaginile de intrare. Aceasta ia straturile RGB originale ca intrare și produce o imagine cu trei straturi. Am folosit un filtru de dimensiune 5 și o valoare sigma de 1,4.

Deoarece algoritmul nu face nicio presupunere a datelor din spatele contextelor, poate fi benefic să includem transformări ale straturilor de culoare. Rezultatul acestor transformări devine, la fel ca și celelalte straturi, intrare a algoritmului. Am folosit filtrul Sobel, algoritmul de detectare a muchiilor Canny, și un algoritm de detectare a muchiilor Kirsch ca și intrare, algoritmi care folosesc canale de culoare și produc un alt strat. Acest lucru face ca imaginea de intrare finală să aibă trei sau mai multe straturi. Având un strat Canny sau orice detector de contur ca intrare este echivalent cu adăugarea în model de cunoștințe specifice domeniului.

Arhitectura cu un singur strat ia o imagine preprocesată ca intrare și folosește un set dat de canale de culoare pentru a calcula o imagine de ieșire care constă dintr-o singură imagine în tonuri de gri în care pixelii reprezintă probabilitatea ca poziția din imaginea originală să aparțină unui contur. Arhitectura cu un singur strat combinată cu razele simple ca și contexte nu ia în considerare informațiile despre conturul păstrat la fel la diferite niveluri de scalare. Pentru a rezolva problema scalării, arhitectura multistrat se comportă în acest fel: se ia imaginea originală, se aplică procesarea, se obține o ieșire; apoi se ia imaginea originală, se redimensionează, se aplică procesarea, se adăugă rezultatul obținut anterior ca strat, și se folosește ca intrare pentru algoritm. Dacă se decide separarea memoriei utilizate de algoritm la diferite nivele de redimensionare, avem o arhitectură multistrat. Fiecare strat este antrenat separat începând de la cea mai mare imagine și mergând spre imagini de dimensiune mică. Un strat al algoritmului poate avea o configurație diferită de celelalte straturi, cu diverse opțiuni care includ lungimea celei mai lungi raze, preprocesarea efectuată, dimensiunea memoriei și altele. Rezultatul arhitecturii multistrat este un set de imagini în tonuri de gri de diferite dimensiuni, care se numesc ieșiri reziduale. Aceste ieșiri reziduale sunt apoi combinate (amestecate) folosind un strat de regresie logistică pentru a forma o singură imagine. Înainte de combinare, imaginile sunt redimensionate pentru a avea toate aceași dimensiune. Ponderile stratului de regresie logistică sunt de asemenea calculate folosind același set de antrenament.

Înainte de a calcula scorul pentru benchmark, imaginile ieșire sunt supuse unei tehnici de suprimare non-maxime și, ulterior, unei subțieri de margine. Benchmark-ul oferă un instrument de evaluare care are o căutare automată în spațiul pragului de binarizare, astfel încât utilizatorul să poată lăsa imagini în tonuri de gri în loc să facă el însuși binarizarea.

Rezultatele experimentale arată că învățarea este orientată spre a nu-și asuma niciun risc, deoarece scorul F1 mai bun este atins în setările de prag de binarizare scăzut. Pentru a obține un scor F1 mai bun, trebuie luată în considerare echilibrarea procentului de exemple aparținând unei clase atunci când se aplică funcția de pierdere.

Prezentăm încă o comparație analitică folosind măsura Cross-Entropy. Dacă avem două distribuții de probabilitate, putem măsura numărul de biți necesari pentru a identifica un eveniment extras dintr-o mulțime dacă o schemă de codificare este utilizată cu o distribuție de probabilitate diferită de distribuția adevărată a mulțimii. Deoarece intensitățile pixelilor din imaginile rezultate pot fi modelate ca probabilitate ca un pixel să aparțină unei margini, putem măsura entropia încrucișată pentru imaginile de ieșire. Arătăm o comparație cu algoritmul Canny pentru primele 50 de imagini din setul de testare al benchmark-ului.

Contribuția principală a acestui capitol este un algoritm agnostic de aplicație pentru predicția probabilității că un pixel aparține unei clase sau alteia, predicție făcută pe baza informațiilor contextuale disponibile, unde învățarea se poate face într-o singură trecere peste datele de antrenament. Mai mult decât atât, nu se impune nicio constrângere asupra modului de alegere sau modelare a contextului. De asemenea, este posibilă integrarea algoritmului în structuri mai mari de învățare și predicție, având cerințe scăzute referitoare la informațiile necesare pentru antrenament.

Capitolul 3

În acest capitol este propus un algoritm de învățare automată pentru extragerea liniei centrale a vaselor de sânge coronariene, care utilizează o adaptare 3D a arhitecturii U-NET care generează probabilitatea ca fiecare voxel (volum pixel) al unui volum complet să fie parte a liniei centrale a unui vas de sânge. De asemenea, pe baza unei revizuirii extinse a literaturii actuale, a fost propusă o funcție de pierdere adaptată pentru a gestiona adnotările rare, ceea ce înseamnă că nu toate liniile centrale sunt marcate în setul de date, și al dezechilibrului de clasă, deoarece foarte puțini voxelii dintr-un întreg volum aparțin unei linii centrale. Cercetarea a fost publicată în [1] și [5].

Liniile centrale ale unui vas de sânge pot fi extrase fie printr-o segmentare și apoi un algoritm de subțiere, fie prin urmărire directă. Liniile centrale extrase pot servi drept intrare pentru algoritmi de urmărire pentru segmentarea întregului arbore coronarian.

Ideea pentru crearea unei rețele 3D U-Net a fost construită pe observarea rezultatelor rețelelor U-Net convoluționale pentru segmentarea imaginilor biomedicale [6], care permite unei rețele neuronale complet convoluționale să ofere o bună segmentare chiar și atunci când este antrenată cu un mic set de date pentru antrenament.

Structura rețelei de acest tip urmează doi pași importanți, similari cu o rețea de tipul Autoencoder. Primul pas este contracția, unde sunt aplicate pe datele de intrare runde consecutive de două convoluții și apoi o operație max pooling pentru a reduce dimensiunea ieșirii. Pentru fiecare rundă, numărul de filtre pentru convoluție este dublat iar dimensiunea ieșirii față de intrare este înjumătățită. Stratul inferior va avea cele mai multe filtre, dar va fi și cel mai mic ca dimensiune. Scopul său este de a învăța o reprezentare codificată a ceea ce trebuie segmentat. Convoluția cu nuclee de $3 \times 3 \times 3$ înseamnă că se va pierde un pixel din toate marginile volumelor de intrare. Pentru a elimina această problemă, s-a folosit procesul de padding. Al doilea pas este extinderea. Începând cu stratul inferior, se aplică runde succesive de upsampling, concatenare și două convoluții (de asemenea, cu padding). Procesul de upsampling redimensionează vectorul de caracteristici. Împreună cu informațiile provenite prin concatenarea intrării de aceeași dimensiune din pasul contracției, cele două convoluții pot reconstrui imaginea în dimensiunea sa originală. După ultima expansiune, se aplică o altă convoluție cu numărul de nuclee egal cu numărul de caracteristici de extras. Deoarece extracția liniei centrale vizează segmentarea binară, a fost necesar doar un ultim strat de neuroni cu funcție de activare sigmoidă. Funcția de pierdere este aleasă astfel încât segmentarea să se comporte ca o funcție de clasificare la nivel de voxel.

Cantitatea limitată de memorie video face imposibilă utilizarea ca intrare pentru o rețea neurală a unui întreg volum CT. În imaginile 2D, dimensiunea memoriei nu este o problemă, chiar și atunci când se lucrează cu loturi mari. Prima abordare a fost de a redimensiona volumele (și volumul cu valoarea de adevăr) la un volum care ar fi suficient de mic pentru a se potrivi în memoria plăcii video. Cu toate acestea, pierderea de precizie la redimensionarea la mărimea originală a făcut rezultatul inutil pentru segmentarea liniilor centrale care sunt prin natura lor subțiri. Antrenarea modelului folosind redimensionarea a ridicat și probleme cu setul de date care era prea mic. A doua abordare propusă a fost de a împărți volumele în patch-uri, tăind doar părți mici din volum (și valoare de adevăr) pentru intrarea în rețea. Această metodă nu are niciunul dintre dezavantajele celei propuse anterior, dar introduce una diferită: lipsa întregului context. Prin urmare, se ajunge la un compromis între creșterea dimensiunii patch-urilor, astfel încât informațiile contextuale disponibile pentru model să fie suficiente pentru o predicție bună, și dimensiunea modelului, astfel încât modelul să fie suficient de mare pentru a capta corelațiile cu privire la locul unde se află poziționată linia centrală și că aceasta este continuă și poate traversa patch-urile. Pentru a genera segmentarea completă, soluția utilizată a fost împărțirea intrării în blocuri de forma patch-ului, trecerea fiecărui bloc prin model și recombinarea predicțiilor într-un volum 3D complet.

Pentru ajustarea treptată a ponderilor modelului în timpul antrenamentului este necesară o funcție de pierdere care ar trebui aleasă pentru a cuantifica cât și în ce direcție se ajustează parametrii, astfel încât, la următoarea iterație, ieșirile să fie mai aproape de obiectiv. Niciuna dintre numeroasele funcții de pierdere propuse în literatura științifică nu se potrivește cu specificul setului nostru de date. Este necesară o funcție care să funcționeze cu linii centrale rar adnotate, ceea ce înseamnă că exemplele ar putea fi contradictorii și, în același timp, cu un imens dezechilibru de clasă. Introducem o funcție de pierdere care este o combinație între pierderea focală și o pierdere prin suprapunere, combinând astfel o funcție de pierdere locală cu una globală.

Pentru a testa implementarea rețelei neuronale propuse, folosim setul de date din cadrul benchmark-ului Rotterdam [7]. Acest benchmark are ca scop evaluarea algoritmilor pentru extragerea liniei centrale din datele angio CT. Setul de date este împărțit într-un set de antrenament și un set de testare. Setul de antrenament este format din 8 imagini volumetrice, iar setul de testare este format din 24 de imagini volumetrice. Fiecare volum conține patru linii centrale, fiecare conținând încă patru puncte de referință. Imaginile setului de antrenament conțin în plus valoarea de adevăr a segmentării sub forma unui fișier de referință pentru vasele de sânge țintă. Acest lucru îl face dificil pentru algoritmi de învățare supervizați, deoarece multe dintre vasele de sânge nu sunt raportate ca parte a segmentării. Cele 8 volume de intrare segmentate au fost împărțite într-un set de antrenament cu șapte volume și un set de validare cu volumul rămas, lăsând un volum în afară, de aceea a fost utilizată validarea holdout. Deși numărul este mic, utilizarea patch-urilor augmentate ajută la reducerea problemei unui set de date foarte mic.

Modelele au fost antrenate și testate pe un NVIDIA GeForce GTX 1060 cu 6 GB VRAM. Timpul de antrenare a fost măsurat la aproximativ 8 ore pe placa video menționată anterior. Timpul de predicție și procesare pentru o intrare de dimensiunea patch-ului variază între 400ms și 600ms, în funcție de dimensiunea patch-ului și dimensiunea modelului. Pentru un volum complet, acest timp este înmulțit cu numărul de patch-uri care compun un volum. La aproximativ 80 de patch-uri și 500ms pe patch, ieșirea completă este calculată în aproximativ 40 de secunde. Combinarea patch-urilor și post-procesarea pentru salvarea rezultatului sunt neglijabile ca și timp. Timpul de antrenament face ca explorarea spațiului hiperparametrilor modelului să fie o operație costisitoare, și se pot realiza configurații mai bune prin integrarea algoritmului în platforme de explorare automată a spațiului hiperparametrilor. Spațiul de căutare trebuie definit mai rigid, iar parametrii propuși aici sunt un început bun.

Urmărind formatul fișierului de referință, am creat un instrument pentru a genera o mască volumetrică pentru fiecare imagine de intrare a setului de antrenament, care servește drept volum de adevăr în algoritmul de învățare supervizat. Instrumentul poate fi parametrizat pentru a specifica lățimea în voxeli a liniei centrale.

Antrenarea rețelei cu arhitectura propusă a fost realizată cu diferite configurații pentru următorii parametri: dimensiunea patch-ului de intrare, reducerea numărului de neuroni în nuclee, dimensiunea lotului, folosirea sau nu a patch-urilor fără nici un singur voxel marcat ca fiind linie centrală. Constrângerea dură aplicată parametrilor este dată de cantitatea limitată de memorie a plăcii video. Un exemplu de constrângere: dacă dimensiunea patch-ului de intrare a fost mărită, dimensiunea lotului sau a numărului de nuclee convoluționale din straturi trebuie redus.

Platforma Batchgenerators [8] a fost utilizată deoarece a furnizat o gamă largă de transformări și include augmentarea spațială, potrivită pentru datele de intrare 3D. Patch-urile au fost augmentate cu transformări spațiale, transformări de culoare, transformări de zgomot. Doar transformările spațiale sunt aplicate, cu aceiași parametri, volumului de adevăr. Fără augmentări, setul de date a fost prea mic pentru ca antrenamentul să fie convergent.

Noțiunea clasică a epocii, în care epoca înseamnă o trecere peste întregul set de date de antrenament, nu se va mai aplica atunci când se lucrează cu patch-uri tăiate aleatoriu. Termenul epocă este definit aici prin înmulțirea numărului de volume din setul de antrenament cu cea mai mare dimensiune a volumului de intrare și apoi împărțită pe fiecare axă cu dimensiunea

corespunzătoare a patch-ului, și rezultatul împărțit la dimensiunea lotului. Rezultatele sunt raportate pentru diferite epoci, în funcție de diferitele dimensiuni ale patch-urilor de intrare. Pentru setul de date de antrenament, valorile pentru o epocă sunt calculate ca medie pe toate patch-urile de intrare. Pentru setul de date de validare, valorile pentru o epocă sunt calculate prin asamblarea întregului volum din patch-urile de ieșire și compararea acestuia cu întregul volum de adevăr. Rezultatele sunt prezentate sub forma unei perechi de precizie binară și suprapunere. Este prezentată și validarea vizuală pentru volumele de ieșire.

Contribuția principală a acestui capitol este implementarea unui U-Net 3D, adecvat pentru segmentarea liniei centrale a arterelor coronariene. Arhitectura are la bază o rețea neuronală U-Net 2D, concepută inițial pentru segmentarea vaselor retinei, și dezvoltarea unei versiuni 3D capabilă să segmenteze cu precizie de voxel. Am creat o nouă funcție de pierdere concepută pentru a combate două probleme simultane, de obicei legate de segmentarea medicală volumetrică: dezechilibrul imens de clasă și adnotare rară. Fără această funcție de pierdere, convergența antrenării modelului nu a fost garantată chiar și atunci când se utilizează intrări augmentate. Am demonstrat convergența antrenamentului utilizând o a doua funcție de pierdere, începând cu o rețea preantrenată cu funcția de pierdere propusă. Fără pregătire, antrenamentul direct cu cea de-a doua funcție de pierdere nu ar converge.



Introduction.....	22
1. The objectives of the thesis	22
2. Thesis structure and content.....	23
1. Lossless image compression with contextual memory	26
1.1 Overview.....	26
1.2. Related work	26
1.3. PAQ8PX algorithm for lossless image compression in detail.....	28
1.3.1. Description.....	28
1.3.2. General aspects	29
1.3.3. Modeling	29
1.3.4. Image compression.....	30
1.3.4.1. Direct modeling.....	30
1.3.4.2. Indirect modeling	30
1.3.4.3. Least squares modeling	31
1.3.4.4. Correlations	31
1.3.4.5. Grayscale 8bpp.....	33
1.3.5. Context mixing.....	33
1.3.6. Adaptive probability maps	34
1.3.7. Other considerations.....	34
1.4. The proposed method – contextual memory	35
1.4.1. Context modeling	36
1.4.2. Description of the contextual prediction	37
1.4.2.1. Model Prediction.....	37
1.4.2.2. Interpretation of values.....	37
1.4.2.3. Updating the model	38
1.4.3. Memory implementation and variations.....	39
1.5. Quantifying the error.....	41
1.6. Experimental Results	42
1.6.1. PAQ8PX Contextual Memory implementation details	42
1.6.2. Evaluation on the benchmarks	42
1.6.3. Discussion on the results	45
1.7. Conclusions.....	46
2. A novel contextual memory algorithm for edge detection.....	48
2.1 Overview.....	48
2.2 Related work	48
2.3 Berkeley Edge Detection Benchmark	49
2.4 Basis for the contextual memory and the processing pipeline	50
2.4.1 Logistic Regression.....	50
2.4.2 Ensemble learning.....	51
2.4.3 Context modeling.....	53
2.3.3 Resources and hashing as a solution	53

2.4 An original method for contextual prediction	54
2.4.1 Model Prediction	55
2.4.2 Updating the proposed model	57
2.4.3 Implementation details	59
2.5 Results	60
2.5.1 Inputs, preprocessing and processing architecture	60
2.5.2 Results on Berkeley Edge Detection Benchmark	61
2.6 Conclusions	73
3. Coronary centerline extraction from CCTA using 3D-UNet	76
3.1 Overview	76
3.2 Related Work	77
3.2.1. Rule based centerline extraction	78
3.2.2. Machine learning based centerline extraction	78
3.3. Proposed Method	79
3.3.1. Neural Network Architecture	79
3.3.2. Resize or patches	81
3.3.3. Loss functions	83
3.3.3.1. Local loss	83
3.3.3.2. Global loss	83
3.3.3.3. Combined loss functions	84
3.3.3.4. Proposed loss function	84
3.3.4. Generating the full output	85
3.4. Experimental Setup	86
3.4.1. Coronary dataset	86
3.4.2. Visualization	87
3.4.3. Execution setup	88
3.4.4. Training the network	88
3.5. Results	90
3.5.1 Network parameters	99
3.5.2 3D U-NET Python dependencies	103
3.6. Discussion	104
3.7. Conclusions	106
4. Final conclusions	108
4.1 Conclusions	108
4.2 Personal Contributions	109
4.2.1 Lossless image compression	109
4.2.2 Image segmentation with edge detection	110
4.2.3 Automated coronary centerline extraction from CCTA	110
4.3 Dissemination of the research results	110
References	113

LISTĂ FIGURI

Figure 1 PAQ8 Image compression diagram	29
Figure 2 Causal pixel neighborhood.....	32
Figure 3 Block scheme of the proposed prediction using the contextual memory method.....	35
Figure 4 Contexts as rays.....	36
Figure 5 Block scheme for the proposed update algorithm	38
Figure 6 Quantified error for image "8068" from BSD500 [38] with (a) original image, (b) pixelwise coding cost, (c) high contrast centered difference of coding costs, (d) computed disagreement.	42
Figure 7 (a) Example BSD500 test image "2018" with (b) ground truth	50
Figure 8 (a) Position of pixels as rays of length 5; (b) ray length 3 (green), length 4 (blue), length 5 (orange)	53
Figure 9 The prediction scheme for each pixel of one channel of a layer	55
Figure 10 (a) uniform distribution of points; (b) non-uniform distribution of points	56
Figure 11 Block scheme for the proposed update algorithm	58
Figure 12 Contextual memory single layer processing architecture	61
Figure 13 Contextual memory multilayer processing architecture	61
Figure 14 (a) sample image from the benchmark along with (b) the ground truth	62
Figure 15 Output example after NMS and thinning	67
Figure 16 Output of the proposed method on the training image "197017", (a) original image, (b) proposed output, (c) ground truth.	68
Figure 17 Output of the proposed method on the test image "51084", (a) original image, (b) proposed output, (c) ground truth.	69
Figure 18 Output of the individual layers on the test image "51084", (a) first layer (b) second layer (c) third layer (d) fourth layer.	70
Figure 19 F1 Score plotted against threshold.....	71
Figure 20 Cross Entropy for the first 50 images of the dataset (lower is better)	72
Figure 21 Precision in respect to the threshold (bigger is better)	73
Figure 22 Result of centerlines (left) to vessel segmentation (right) using sphere fitting, from [78].	77
Figure 23 Example vessel segmentation from extracted centerline using set evolution active contour, from [81].....	78
Figure 24 The original U-NET architecture, from [6]	79
Figure 25 Example Retina Vessel Segmentation using a 2D U-Net Architecture	80
Figure 26 The proposed 3D U-Net architecture	80
Figure 27 Downscaled and upscaled centerline ground truth (80x80x64) plotted in 3D to highlight the difficulty of using it for training	82
Figure 28 2D slice of an augmented patch with ground truth next to it.....	82
Figure 29 Output volume composed from patches with visible stitching.....	85
Figure 30 3D Slicer Volume Rendering with CT-Coronary Arteries preset	87
Figure 31 (a) Training 128x128x96, batch size 1, reduction 1, only patches with ground truth (b) Validation .	91
Figure 32 (a) Training 128x128x128, batch size 2, reduction 2, only patches with ground truth (b) Validation	91
Figure 33 (a) Training 256x256x128, batch size 1, reduction 2, only patches with ground truth (b) Validation	91
Figure 34 (a) Training 256x256x128, batch size 1, reduction 2 (b) Validation.....	92
Figure 35 (a) Training 320x320x64, batch size 1, reduction 4 (b) Validation	92
Figure 36 (a) Training 384x384x48, batch size 1, reduction 4 (b) Validation	92
Figure 37 Patch size 128x128x96 rendering of ground truth (red) and centerline segmentation (transparent yellow)	93
Figure 38 Patch size 128x128x96 rendering of ground truth (red) and thinned centerline segmentation (green).....	93
Figure 39 Patch size 128x128x128 rendering of ground truth (red) and centerline segmentation (transparent yellow)	94
Figure 40 Patch size 128x128x128 rendering of ground truth (red) and thinned centerline segmentation (green).....	94
Figure 41 Patch size 256x256x128, training with only patches with ground truth, rendering of ground truth (red) and centerline segmentation (transparent yellow)	95

Figure 42 Patch size 256x256x128, training with only patches with ground truth, rendering of ground truth (red) and thinned centerline segmentation (green) 95

Figure 43 Patch size 256x256x128 rendering of ground truth (red) and centerline segmentation (transparent yellow) 96

Figure 44 Patch size 256x256x128 rendering of ground truth (red) and thinned centerline segmentation (green)..... 96

Figure 45 Patch size 320x320x64 rendering of ground truth (red) and centerline segmentation (transparent yellow) 97

Figure 46 Patch size 320x320x64 rendering of ground truth (red) and thinned centerline segmentation (green)..... 97

Figure 47 Patch size 384x384x48 rendering of ground truth (red) and centerline segmentation (transparent yellow) 98

Figure 48 Patch size 384x384x48 rendering of ground truth (red) and thinned centerline segmentation (green)..... 98

Figure 49 Loss function switch (a) Patch size 128x128x128 (b) Patch size 256x256x128 105



LISTĂ TABELE



Table 1 Waterloo gray test set 1.....	43
Table 2 Waterloo gray test set 2.....	44
Table 3 Imagecompression.info 8bpp gray new test images.....	44
Table 4 Squeezechart 8bpp grayscale	45
Table 5 Compression running time comparison on image lena2 (expressed in seconds)	45
Table 6 Results for the single layer contextual memory edge detector	62
Table 7 Results for the multilayer architecture of the contextual memory edge detector	66
Table 8 Comparative results of the contextual memory edge detector	71
Table 9 The size and resolution of each CTA volume from the training dataset [84].....	86
Table 10 The overlap and accuracy results for the selected hyperparameters	90
Table 11 Network parameters for Input Size 128x128x96, model reduction 1.....	99
Table 12 Network parameters for Input Size 128x128x128, model reduction 2.....	100
Table 13 Network parameters for Input Size 256x256x128, model reduction 4.....	101
Table 14 Network parameters for Input Size 320x320x64, model reduction 4.....	102
Table 15 Network parameters for Input Size 384x384x48, model reduction 4.....	103





1. Ogorean, V.; Dorobanțiu, A.; Brad, R. Deep Learning Architectures and Techniques for Multi-organ Segmentation. *IJACSA* **2021**, *12*, doi: 10.14569/IJACSA.2021.0120104.
2. Dorobanțiu, A.; Brad, R. Improving Lossless Image Compression with Contextual Memory. *Applied Sciences* **2019**, *9*, 2681, doi: 10.3390/app9132681.
3. Dorobanțiu, A.; Brad, R. A novel contextual memory algorithm for edge detection. *Pattern Analysis and Applications* **2019**, doi: 10.1007/s10044-019-00808-0.
4. Fram, J.R.; Deutsch, E.S. On the quantitative evaluation of edge detection schemes and their comparison with human performance. *IEEE Trans Comput* **1975**, *C-24:6*, doi: 10.1109/T-C.1975.224274.
5. Dorobanțiu, A.; Ogorean, V.; Brad, R. Coronary Centerline Extraction from CCTA Using 3D-UNet. *Future Internet* **2021**, *13*, 101, doi: 10.3390/fi13040101.
6. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*; Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, 2015; Vol. 9351, pp. 234–241 ISBN 978-3-319-24573-7.
7. Rotterdam Coronary Artery Algorithm Evaluation Framework Available online: <http://coronary.bigr.nl/centerlines/> (accessed on Jun 11, 2020).
8. MIC-DKFZ/batchgenerators Available online: <https://github.com/MIC-DKFZ/batchgenerators> (accessed on Aug 29, 2020).
9. Precup, S.-A.; Gellert, A.; Dorobanțiu, A.; Zamfirescu, C.-B. Assembly Process Modeling Through Long Short-Term Memory. In *Recent Challenges in Intelligent Information and Database Systems*; Communications in Computer and Information Science; Springer Singapore: Singapore, 2021; Vol. 1371, pp. 28–39 ISBN 9789811616846.
10. BrainIt: Brain revealed | Brain revealed: innovative Technologies in Neurosurgery Study.
11. Chen, D.; Li, Y.; Zhang, H.; Gao, W. Invertible update-then-predict integer lifting wavelet for lossless image compression. *EURASIP Journal on Advances in Signal Processing* **2017**, *2017*, *8*, doi: 10.1186/s13634-016-0443-y.
12. Khan, A.; Khan, A.; Khan, M.; Uzair, M. Lossless image compression: application of Bi-level Burrows Wheeler Compression Algorithm (BBWCA) to 2-D data. *Multimedia Tools and Applications* **2017**, *76*, 12391–12416, doi: 10.1007/s11042-016-3629-2.
13. Feng, W.; Hu, C.; Wang, Y.; Zhang, J.; Yan, H. A Novel Hierarchical Coding Progressive Transmission Method for WMSN Wildlife Images. *Sensors* **2019**, *19*, 946, doi: 10.3390/s19040946.
14. Schiopu, I.; Munteanu, A. Residual-error prediction based on deep learning for lossless image compression. *Electronics Letters* **2018**, *54*, 1032–1034, doi: 10.1049/el.2018.0889.
15. Hosseini, S.M.; Naghsh-Nilchi, A.-R. Medical ultrasound image compression using contextual vector quantization. *Computers in Biology and Medicine* **2012**, *42*, 743–750, doi: 10.1016/j.compbiomed.2012.04.006.
16. Eben Sophia, P.; Anitha, J. Contextual Medical Image Compression using Normalized Wavelet-Transform Coefficients and Prediction. *IETE Journal of Research* **2017**, *63*, 671–683, doi: 10.1080/03772063.2017.1309998.
17. Borusyak, A.V.; Vasin, Yu.G. Development of an algorithm for adaptive compression of indexed images using contextual simulation. *Pattern Recognition and Image Analysis* **2016**, *26*, 4–8, doi: 10.1134/S1054661816010041.

18. Strutz, T. Context-Based Predictor Blending for Lossless Color Image Compression. *IEEE Transactions on Circuits and Systems for Video Technology* **2016**, *26*, 687–695, doi: 10.1109/TCSVT.2015.2416611.
19. Knezovic, J.; Kovac, M.; Mlinaric, H. Classification and Blending Prediction for Lossless Image Compression. In Proceedings of the MELECON 2006 - 2006 IEEE Mediterranean Electrotechnical Conference; IEEE: Benalmadena, Spain, 2006; pp. 486–489, doi: 10.1109/MELCON.2006.1653144.
20. Strizic, L.; Knezovic, J. Optimization of lossless image compression method for GPGPU. In Proceedings of the 2016 18th Mediterranean Electrotechnical Conference (MELECON); IEEE: Lemesos, Cyprus, 2016; pp. 1–6, doi: 10.1109/MELCON.2016.7495398.
21. Weinlich, A.; Amon, P.; Hutter, A.; Kaup, A. Probability Distribution Estimation for Autoregressive Pixel-Predictive Image Coding. *IEEE Transactions on Image Processing* **2016**, *25*, 1382–1395, doi: 10.1109/TIP.2016.2522339.
22. Biadgie, Y.; Kim, M.; Sohn, K.-A. Multi-resolution Lossless Image Compression for Progressive Transmission and Multiple Decoding Using an Enhanced Edge Adaptive Hierarchical Interpolation. *KSII Transactions on Internet and Information Systems* **2017**, *11*, 6017–6037, doi: 10.3837/tiis.2017.12.018.
23. Biadgie, Y. Edge Adaptive Hierarchical Interpolation for Lossless and Progressive Image Transmission. *KSII Transactions on Internet and Information Systems* **2011**, *5*, 2068–2086, doi: 10.3837/tiis.2011.11.011.
24. Song, X.; Huang, Q.; Chang, S.; He, J.; Wang, H. Lossless medical image compression using geometry-adaptive partitioning and least square-based prediction. *Medical & Biological Engineering & Computing* **2018**, *56*, 957–966, doi: 10.1007/s11517-017-1741-8.
25. Lucas, L.F.R.; Rodrigues, N.M.M.; da Silva Cruz, L.A.; de Faria, S.M.M. Lossless Compression of Medical Images Using 3-D Predictors. *IEEE Transactions on Medical Imaging* **2017**, *36*, 2250–2260, doi: 10.1109/TMI.2017.2714640.
26. Shen, H.; Jiang, Z.; Pan, W. Efficient Lossless Compression of Multitemporal Hyperspectral Image Data. *Journal of Imaging* **2018**, *4*, 142, doi: 10.3390/jimaging4120142.
27. Consultative Committee for Space Data Systems CCSDS RECOMMENDED STANDARD FOR IMAGE DATA COMPRESSION 2017.
28. Knoll, B.; Freitas, N. de A Machine Learning Perspective on Predictive Coding with PAQ8. In Proceedings of the 2012 Data Compression Conference; IEEE: Snowbird, UT, USA, 2012; pp. 377–386, doi: 10.1109/DCC.2012.44.
29. Mahoney, M.V. Adaptive Weighing of Context Models for Lossless Data Compression. 6.
30. Data Compression Explained Available online: http://mattmahoney.net/dc/dce.html#Section_43 (accessed on May 11, 2019).
31. paq8px thread Available online: <https://encode.ru/threads/342-paq8px> (accessed on May 11, 2019).
32. Chartier, M. *MCM file compressor*. <https://github.com/mathieuchartier/mcm>; 2019;
33. Veness, J.; Lattimore, T.; Bhoopchand, A.; Grabska-Barwinska, A.; Mattern, C.; Toth, P. Online Learning with Gated Linear Networks. *arXiv:1712.01897 [cs, math]* **2017**.
34. Mattern, C. Mixing Strategies in Data Compression. In Proceedings of the 2012 Data Compression Conference; IEEE: Snowbird, UT, USA, 2012; pp. 337–346, doi: 10.1109/DCC.2012.40.
35. Mattern, C. Linear and Geometric Mixtures - Analysis. In Proceedings of the 2013 Data Compression Conference; IEEE: Snowbird, UT, 2013; pp. 301–310, doi: 10.1109/DCC.2013.38.
36. Mattern, C. On Statistical Data Compression. PhD Thesis, Technische Universität Ilmenau, Germany, 2016.

37. Fowler–Noll–Vo hash functions Available online: <http://www.isthe.com/chongo/tech/comp/fnv/index.html> (accessed on May 11, 2019).
38. Arbelaez, P.; Maire, M.; Fowlkes, C.; Malik, J. Contour detection and hierarchical image segmentation. *IEEE TPAMI* **2011**, *33*, doi: 10.1109/TPAMI.2010.161.
39. Alexandru Dorobanțiu - GitHub Available online: <https://github.com/AlexDorobantiu> (accessed on May 11, 2019).
40. Dorobanțiu, A. Paq8px167ContextualMemory Available online: <https://github.com/AlexDorobantiu/Paq8px167ContextualMemory> (accessed on May 13, 2019).
41. Image Repository of the University of Waterloo Available online: <http://links.uwaterloo.ca/Repository.html> (accessed on May 11, 2019).
42. S. Garg. The New Test Images - Image Compression Benchmark Available online: http://imagecompression.info/test_images/ (accessed on May 11, 2019).
43. Squeeze Chart • Lossless Data Compression Benchmarks Available online: <http://www.squeezechart.com/> (accessed on May 11, 2019).
44. Dorobanțiu, A. *Compute Bits Per Pixel for compressed images.* <https://github.com/AlexDorobantiu/BppEvaluator>; 2019;
45. 7-cpu Available online: <https://www.7-cpu.com/utills.html> (accessed on Jun 13, 2019).
46. Mahoney, M. The ZPAQ Open Standard Format for Highly Compressed Data - Level 2. 23.
47. Aiazzi, B.; Alparone, L.; Baronti, S. Context modeling for near-lossless image coding. *IEEE Signal Processing Letters* **2002**, *9*, 77–80, doi: 10.1109/97.995822.
48. Gaonkar, B.; Hovda, D.; Martin, N. Deep learning in the small sample size setting: cascaded feed forward neural networks for medical image segmentation. *Proc SPIE* **2016**, 9785.
49. Milletari F, Navab N, Ahmadi S (2016) V-Net: fully convolutional neural networks for volumetric medical image segmentation. In: IEEE fourth international conference on 3D vision, pp 565–571.
50. Dollár, P.; Zitnick, L.C. Fast edge detection using structured forests. *IEEE Trans Pattern Anal Mach Intell* **2015**, *37*, doi: 10.1109/TPAMI.2014.2377715.
51. Xie, S.; Tu, Z. Holistically-nested edge detection. *Proc IEEE Int J Comput Vis* **2017**, *125*, doi: 10.1007/s11263-017-1004-z.
52. Liu Y, Lew MS (2016) Learning relaxed deep supervision for better edge detection. In: IEEE conference on computer vision and pattern recognition (CVPR), pp 231–240.
53. Liu Y, Cheng MM et al (2019) Richer convolutional features for edge detection. In: IEEE transactions on pattern analysis and machine intelligence; <http://mftp.mmcheng.net/Papers/19PamiEdge.pdf>. Accessed 05 Nov 2018.
54. Fu F, Wang C et al (2018) An improved adaptive edge detection algorithm based on Canny. In: Proceedings of SPIE 10827 icOPEN. Accessed 24 Jul 2018.
55. Guadaa, C.; Edwin, Z. A novel edge detection algorithm based on a hierarchical graph-partition approach. *J Intell Fuzzy Syst* **2018**, *34*, doi: 10.3233/JIFS-171218.
56. Minka T (2017) A comparison of numerical optimizers for logistic regression. <https://tminka.github.io/papers/logreg/minka-logreg.pdf>. Accessed 05 Feb 2017.
57. Naftaly, U.; Intrator, N.; Horn, D. Optimal ensemble averaging of neural networks. *Netw Comput Neural Syst* **1999**, *8*.
58. Liu, Y.; Yao, X. Ensemble learning via negative correlation. *Neural Netw* **1999**, *12*, doi: 10.1016/S0893-6080(99)00073-8.
59. Long, P.M.; Servedio, R.A. Random classification noise defeats all convex potential boosters. *Mach Learn* **2010**, *78*, doi: 10.1007/s10994-009-5165-z.
60. <http://www.burtleburtle.net/bob/hash/doobs.html>. Accessed 05 Feb 2017.
61. <http://www.isthe.com/chongo/tech/comp/fnv/index.html>. Accessed 05 Feb 2017.
62. <http://www.byronknoll.com/cmix.html>. Accessed 05 Nov 2018.

63. Dorobanțiu, A. *Contextual Memory Edge Detection* <https://github.com/AlexDorobantiu/ContextualMemoryEdgeDetection>; 2018;
64. Dollar, P. Structured Edge Detection Toolbox <https://github.com/pdollar/edges> Available online: <https://github.com/pdollar/edges> (accessed on Sep 27, 2021).
65. Wang Y, Zhao X et al (2018) Deep crisp boundaries. From boundaries to higher-level tasks. arXiv preprint arXiv:1801.02439.
66. Xu D, Ouyang W et al (2017) Learning deep structured multi-scale features using attention-gated CRFs for contour prediction. In: Advances in neural information processing system, pp 3961–3970.
67. Yu Z, Feng C et al (2017) CASENet: deep category-aware semantic edge detection. In: IEEE conference on computer vision and pattern recognition (CVPR), pp 21–26.
68. Yang J, Price B et al (2016) Object contour detection with a fully convolutional encoder-decoder network. In: IEEE conference on computer vision and pattern recognition (CVPR), pp 193–202.
69. Galal, H.; Rashid, T.; Alghonaimy, W.; Kamal, D. Detection of positively remodeled coronary artery lesions by multislice CT and its impact on cardiovascular future events. *Egypt Heart J* **2019**, *71*, 26, doi: 10.1186/s43044-019-0029-8.
70. Pantos, I.; Katritsis, D. Fractional Flow Reserve Derived from Coronary Imaging and Computational Fluid Dynamics. *Interventional Cardiology Review* **2014**, *9*, 145, doi: 10.15420/icr.2014.9.3.145.
71. Zhao, Y.; Ping, J.; Yu, X.; Wu, R.; Sun, C.; Zhang, M. Fractional flow reserve-based 4D hemodynamic simulation of time-resolved blood flow in left anterior descending coronary artery. *Clinical Biomechanics* **2019**, *70*, 164–169, doi: 10.1016/j.clinbiomech.2019.09.003.
72. Williams, L.H.; Drew, T. What do we know about volumetric medical image interpretation?: a review of the basic science and medical image perception literatures. *Cogn. Research* **2019**, *4*, 21, doi: 10.1186/s41235-019-0171-6.
73. Zhao, F.; Chen, Y.; Hou, Y.; He, X. Segmentation of blood vessels using rule-based and machine-learning-based methods: a review. *Multimedia Systems* **2019**, *25*, 109–118, doi: 10.1007/s00530-017-0580-7.
74. Chen, X.; Fu, Y.; Lin, J.; Ji, Y.; Fang, Y.; Wu, J. Coronary Artery Disease Detection by Machine Learning with Coronary Bifurcation Features. *Applied Sciences* **2020**, *10*, 7656, doi: 10.3390/app10217656.
75. Danilov, A.; Pryamonosov, R.; Yurova, A. Image Segmentation for Cardiovascular Biomedical Applications at Different Scales. *Computation* **2016**, *4*, 35, doi: 10.3390/computation4030035.
76. Guo, Z.; Bai, J.; Lu, Y.; Wang, X.; Cao, K.; Song, Q.; Sonka, M.; Yin, Y. DeepCenterline: a Multi-task Fully Convolutional Network for Centerline Extraction. *arXiv:1903.10481 [cs]* **2019**.
77. Bates, R.; Irving, B.; Markelc, B.; Kaeppler, J.; Muschel, R.; Grau, V.; Schnabel, J.A. Extracting 3D Vascular Structures from Microscopy Images using Convolutional Recurrent Networks. *arXiv:1705.09597 [cs]* **2017**.
78. Han, D.; Shim, H.; Jeon, B.; Jang, Y.; Hong, Y.; Jung, S.; Ha, S.; Chang, H.-J. Automatic Coronary Artery Segmentation Using Active Search for Branches and Seemingly Disconnected Vessel Segments from Coronary CT Angiography. *PLoS ONE* **2016**, *11*, e0156837, doi: 10.1371/journal.pone.0156837.
79. Liu, L.; Xu, J.; Liu, Z. Automatic segmentation of coronary lumen based on minimum path and image fusion from cardiac computed tomography images. *Cluster Comput* **2019**, *22*, 1559–1568, doi: 10.1007/s10586-018-2548-6.
80. Kong, B.; Wang, X.; Bai, J.; Lu, Y.; Gao, F.; Cao, K.; Xia, J.; Song, Q.; Yin, Y. Learning tree-structured representation for 3D coronary artery segmentation. *Computerized Medical Imaging and Graphics* **2020**, *80*, 101688, doi: 10.1016/j.compmedimag.2019.101688.

81. Lv, T.; Yang, G.; Zhang, Y.; Yang, J.; Chen, Y.; Shu, H.; Luo, L. Vessel segmentation using centerline constrained level set method. *Multimed Tools Appl* **2019**, *78*, 17051–17075, doi: 10.1007/s11042-018-7087-x.
82. Gao, Z.; Liu, X.; Qi, S.; Wu, W.; Hau, W.K.; Zhang, H. Automatic segmentation of coronary tree in CT angiography images. *Int J Adapt Control Signal Process* **2019**, *33*, 1239–1247, doi: 10.1002/acs.2762.
83. Sheng, X.; Fan, T.; Jin, X.; Jin, J.; Chen, Z.; Zheng, G.; Lu, M.; Zhu, Z. Extraction Method of Coronary Artery Blood Vessel Centerline in CT Coronary Angiography. *IEEE Access* **2019**, *7*, 170690–170702, doi: 10.1109/ACCESS.2019.2955710.
84. Cui, H.; Xia, Y. Automatic Coronary Centerline Extraction Using Gradient Vector Flow Field and Fast Marching Method From CT Images. *IEEE Access* **2018**, *6*, 41816–41826, doi: 10.1109/ACCESS.2018.2859786.
85. Guo, Z.; Bai, J.; Lu, Y.; Wang, X.; Cao, K.; Song, Q.; Sonka, M.; Yin, Y. DeepCenterline: A Multi-task Fully Convolutional Network for Centerline Extraction. In *Information Processing in Medical Imaging*; Chung, A.C.S., Gee, J.C., Yushkevich, P.A., Bao, S., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, 2019; Vol. 11492, pp. 441–453 ISBN 978-3-030-20350-4.
86. Wolterink, J.M.; van Hamersvelt, R.W.; Viergever, M.A.; Leiner, T.; Išgum, I. Coronary artery centerline extraction in cardiac CT angiography using a CNN-based orientation classifier. *Medical Image Analysis* **2019**, *51*, 46–60, doi: 10.1016/j.media.2018.10.005.
87. Sainath, T.N.; Vinyals, O.; Senior, A.; Sak, H. Convolutional, Long Short-Term Memory, fully connected Deep Neural Networks. In Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP); IEEE: South Brisbane, Queensland, Australia, 2015; pp. 4580–4584, doi: 10.1109/ICASSP.2015.7178838.
88. Salahuddin, Z.; Lenga, M.; Nickisch, H. Multi-Resolution 3D Convolutional Neural Networks for Automatic Coronary Centerline Extraction in Cardiac CT Angiography Scans. *arXiv:2010.00925 [cs, eess]* **2020**.
89. He, J.; Pan, C.; Yang, C.; Zhang, M.; Wang, Y.; Zhou, X.; Yu, Y. Learning Hybrid Representations for Automatic 3D Vessel Centerline Extraction. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*; Martel, A.L., Abolmaesumi, P., Stoyanov, D., Mateus, D., Zuluaga, M.A., Zhou, S.K., Racoceanu, D., Joskowicz, L., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, 2020; Vol. 12266, pp. 24–34 ISBN 978-3-030-59724-5.
90. Irving, B.; Hutton, C.; Dennis, A.; Vikal, S.; Mavar, M.; Kelly, M.; Brady, J.M. Deep Quantitative Liver Segmentation and Vessel Exclusion to Assist in Liver Assessment. In *Medical Image Understanding and Analysis*; Valdés Hernández, M., González-Castro, V., Eds.; Communications in Computer and Information Science; Springer International Publishing: Cham, 2017; Vol. 723, pp. 663–673 ISBN 978-3-319-60963-8.
91. Jin, Q.; Meng, Z.; Pham, T.D.; Chen, Q.; Wei, L.; Su, R. DUNet: A deformable network for retinal vessel segmentation. *Knowledge-Based Systems* **2019**, *178*, 149–162, doi: 10.1016/j.knosys.2019.04.025.
92. Zhuang, J. LadderNet: Multi-path networks based on U-Net for medical image segmentation. *arXiv:1810.07810 [cs, eess]* **2019**.
93. Guo, C.; Szemenyei, M.; Yi, Y.; Wang, W.; Chen, B.; Fan, C. SA-UNet: Spatial Attention U-Net for Retinal Vessel Segmentation. *arXiv:2004.03696 [cs, eess]* **2020**.
94. Çiçek, Ö.; Abdulkadir, A.; Lienkamp, S.S.; Brox, T.; Ronneberger, O. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. *arXiv:1606.06650 [cs]* **2016**.
95. Huang, C.; Han, H.; Yao, Q.; Zhu, S.; Zhou, S.K. 3D U2-Net: A 3D Universal U-Net for Multi-domain Medical Image Segmentation. In Proceedings of the Medical Image Computing and Computer Assisted Intervention – MICCAI 2019; Shen, D., Liu, T.,

- Peters, T.M., Staib, L.H., Essert, C., Zhou, S., Yap, P.-T., Khan, A., Eds.; Springer International Publishing: Cham, 2019; Vol. 11765, pp. 291–299, doi: 10.1007/978-3-030-32245-8_33.
96. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal Loss for Dense Object Detection. *arXiv:1708.02002 [cs]* **2018**.
 97. Cui, Y.; Jia, M.; Lin, T.-Y.; Song, Y.; Belongie, S. Class-Balanced Loss Based on Effective Number of Samples. *arXiv:1901.05555 [cs]* **2019**.
 98. Schaap, M.; Metz, C.T.; van Walsum, T.; van der Giessen, A.G.; Weustink, A.C.; Mollet, N.R.; Bauer, C.; Bogunović, H.; Castro, C.; Deng, X. Standardized evaluation methodology and reference database for evaluating coronary artery centerline extraction algorithms. *Medical Image Analysis* **2009**, *13*, 701–714, doi: 10.1016/j.media.2009.06.003.
 99. Rotterdam Coronary Artery Challenge Categories Available online: <http://coronary.bigr.nl/centerlines/about.php> (accessed on Jun 11, 2020).
 100. 3D Slicer Available online: <https://www.slicer.org/> (accessed on Jun 11, 2020).
 101. GitHub Slicer Available online: <https://github.com/Slicer/Slicer> (accessed on Jun 11, 2020).
 102. Slicer/SlicerJupyter Available online: <https://github.com/Slicer/SlicerJupyter> (accessed on Jun 16, 2020).
 103. Jonker, P.P. Morphological Operations on 3D and 4D Images: From Shape Primitive Detection to Skeletonization. In *Discrete Geometry for Computer Imagery*; Borgefors, G., Nyström, I., di Baja, G.S., Eds.; Lecture Notes in Computer Science; Springer Berlin Heidelberg: Berlin, Heidelberg, 2000; Vol. 1953, pp. 371–391 ISBN 978-3-540-41396-7.
 104. Frid-Adar, M.; Diamant, I.; Klang, E.; Amitai, M.; Goldberger, J.; Greenspan, H. GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification. *Neurocomputing* **2018**, *321*, 321–331, doi: 10.1016/j.neucom.2018.09.013.
 105. AlexDorobantiu/CoronaryCenterlineUnet Available online: <https://github.com/AlexDorobantiu/CoronaryCenterlineUnet> (accessed on Aug 29, 2020).
 106. Iglovikov, V.; Shvets, A. TeraNet: U-Net with VGG11 Encoder Pre-Trained on ImageNet for Image Segmentation. *arXiv:1801.05746 [cs]* **2018**.
 107. Manzo, M.; Pellino, S. Bucket of Deep Transfer Learning Features and Classification Models for Melanoma Detection. *J. Imaging* **2020**, *6*, 129, doi: 10.3390/jimaging6120129.
 108. WHO - Cardiovascular diseases (CVDs) fact sheet Available online: [https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-\(cvds\)](https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)) (accessed on Sep 24, 2020).
-