Detecția și recunoașterea obiectelor

Lemny Erich
Universitatea Tehnica Gh. Asachi
Facultatea de Automatica si Calculatoare
Iasi, Romania
erich.lemny@student.tuiasi.ro

Popescu Vlad
Universitatea Tehnica Gh. Asachi
Facultatea de Automatica si Calculatoare
Iasi, Romania
vlad.popescu@student.tuiasi.ro

Otilia Zvorișteanu - îndrumătoare
Universitatea Tehnica Gh. Asachi
Facultatea de Automatica si Calculatoare
Iasi, Romania
otilia.zvoristeanu@academic.tuiasi.ro

***Abstract*—Domeniul vederii artificiale este în continuă dezvoltare, se caută și dezvoltă noi soluții cu anumite specificatii să meargă in timp-real, să poată deduce contextul situației din imagine, să filtreze informații din orice tip de zgomot/distorsiuni etc.**

***Index Terms*—Neural Networks, Object Detection, Image Segmentation**

I. Rezumat

Acest proiect are drept scop analiza și implementarea soluțiilor ce țin de folosirea vederii artificiale cu scopul de a detecta și recunoaște obiecte.

II. State-of-the-art (5 articole)

A. Relation Networks for Object Detection, Han Hu et al.

La ora actuală, diversele sisteme de detecție a obiectelor în imagini se bazeazaă pe detecția obiectelor individual, fără folosirea a oricăror relații între ele. Această lucrare își propune să le folosească, procesând concomitent mai multe obiecte și încercând să stabilească relațiile dintre ele.

Deși se înțelege faptul că informația contextuală din imagini ajută la detecția obiectelor, nu sunt depuse eforturi semnificative pentru a o expluata. O cauză menționată în lucrare este faptul că relațiile obiect-obiect sunt greu de modelat, rețelele neuronale utilizând structuri neuronale simple.

Autorii au folosit ”module de atenție” (attention modules) din procesarea limbajului natural. Un modul de acest gen poate afecta un element individual prin agregarea informației dintr-un set de elemente. Această agregare se produce automat.

În urma mai multor experimente prin utilizarea diferitor rețele neuronale s-a ajuns la o îmbunătățire maximă de +3 mAP (de la 32.2 până la 35.2).

B. EfficientDet: Scalable and Efficient Object Detection, Mingxing Tan et al.

Eficiența modelelor devine tot mai importantă în domeniul vederii artificiale. Mărimea considerabilă a modelelor și cantitatea considerabilă de resurse computaționale necesare împiedică adoptarea la scară largă a celor mai noi rețele neuronale pentru detecția obiectelor.

În acest articol se propun două optimizări cheie pentru eficientizarea utilizării resurselor hardware. Prima este utilizarea a unui tip de rețele neuronale specializat, denumit BiFPN, iar al doilea este o metodă compusă de scalare (compound scaling method) care modifică rezoluția, adâncimea și lățimea tuturor tipurilor de rețele concomitent.

Autorii au putut micșora mărimea detectorului de obiecte de 4x - 9x ori mai mic și numărul de operații în virgulă mobilă de 13x - 42x ori decât detectoarele precedente.

C. Object Recognition in Very Low Resolution Images Using Deep Collaborative Learning, Jeongin Seo et al.

Deși rețelele neuronale actuale au o performanță remarcabilă, ele presupun o mărime a obiectului adecvată și o oarecare rezoluție bună a imaginii. Acest articol își propune o modalitate de recunoaștere a obiectelor în imagini de rezoluție mică prin colaborarea a două rețele neuronale: una care se ocupă cu îmbunătățirea imaginilor și alta care recunoaște obiectul din imagine.

Pentru antrenearea rețelei corespunzătoare de îmbunătățirea imaginilor s-a folosit cealaltă rețea utilizând imagini de rezoluție ridicată. Apoi rețeaua de detecție a imaginilor s-a utilizat de rezultatele obținute de prima rețea pentru detecția obiectelor în imagini de rezoluție mică. Ca exemplu, la ILSVRC se utilizau imagini cu rezoluția medie 482 * 415 pixeli. Se propune în articol detecția obiectelor în imagini de 16 * 16 pixeli și mai puțin.

D. Scene Semantic Recognition Based on Modified Fuzzy C-Mean and Maximum Entropy Using Object-to-Object Relations, Ahmad Jalal et al.

Recunoașterea semantică a scenei (SSR) se utilizează pe larg în self driving, navigarea în scop turistic, trafic inteligent etc. Deși s-a făcut un progres considerabil în acest domeniu, încă rămân unele provocări nerezolvate, ca fundalul dinamic, schimbarea iluminări etc. Autorii acestui articol prezintă o modalitate nouă pentru realizarea acestui recunoașterii semantice a scenei.

Întâi se elimină zgomotul și se face medierea imaginii. Apoi se integrează Fuzzy C-Means cu super-pixeli și Random Forest face segmentarea obiectelor. După aceste segmente se utilizează pentru extragerea a unui Bag-of-Features (colecție de date variate). O rețea neuronală recunoaște multiplele obiecte din imagine, și în final se face modelul Maximum Entropy pentru a da imaginilor etichete.

E. Object Detection in 20 Years: A Survey, Zhengxia Zou et al.

În acest articol autorii prezintă evoluția materiei de detectare a obiectelor din anii '90 ai secolului trecut până în 2022. În particular, ei prezintă în detalii evoluția pe plan tehnologic, exploră tehnologiile cheie și starea domeniului la ora actuală și analizează metodele de mărire a vitezei de detecție a obiectelor în imagini.

Progresul acestui domeniu se consideră împărțit în două perioade: până în 2014 era detecția tradițională a obiectelor, iar după deja detecția obiectelor utilizând deep learning.

Anumite evenimente cheie în acest domeniu sunt:

Detecorul Viola Jones din 2001, care rula pe un Pentium 3 de 700 MHz și se folosea la detecția facială.

Detecorul HOG din 2005, care pentru detectarea obiectelor de mărimi diferite schimba rezoluția imaginii de intrare dar ținea mărimea ferestrei de detecție neschimbată.

RCNN (Regional Based Convolutional Network) a fost propus pentru prima dată în 2014, el contribuind considerabil la dezvoltările ulterioare în domeniul detectării obiectelor prin rețele neuronale.

III. Related work (2 articole)

Următoarele articole descriu solutii comerciale ale algoritmilor actuali in domeniul detectiei, recunoasterii si clasificarii de obiecte si tipare (en. Object detection). O companie care utilizează algoritmi similari este viso.ai, ce propune Viso Suite, o platformă pentru dezvoltarea aplicațiilor pentru recunoașterea obiectelor. Propune citirea automată a instumentelor analogice, detecția numerelor de înmatriculare de pe mașini etc.

A. Art 1 - YOLO

YOLOv4 (You Only Look Once) de Alexey Bochkovskiy, Chien-Yao Wang și Hong-Yuan Mark Liao este un model destinat uzului în sisteme de producție. Mai întâi, i se aduce o imagine de H x W (H - înălțimea, W - lungimea). Apoi, se extrag trăsăturile folosind o serie de convoluții folosind rețele precum VGG16, Darknet53, ResNet50 denumite drept "backbone" (tr. coloana). Pe urmă, se folosește un "gât" să extragă trăsăturile de la diferite scale folosindu-se de Feature Pyramid Network (FPN), Path Aggregation Network (PAN), La final, se folosesc detectoare de scene (single-stage) ce categorizează, aproximează și prezic informațiile finale, fiind denumite "capul".

Yolo version	Underlying Dataset	Processing Speed – frames per second (fps)	Performance Measures (mAP)
Yolo(v1)	PASCAL VOC 2007+2012	45	63.4
Fast Yolo	PASCAL VOC 2007+2012	155	52.7
Yolo(v2)	PASCAL VOC 2007+2012	40	78.6
Yolo - lite	PASCAL VOC 2007+2012	21	33.77
Yolo(v3)	MS COCO	20	57.9
Yolo(v4)	MS COCO	33	65.7

Fig. 1. Comparația dintre diverse versiuni ale modelului YOLO)[1]

B. Art 2 - R-CNN

Faster R-CNN (Region-Based Convolutional Neural Network) este un model compus din 2 module principale: primul este o rețea convoluțională ce procesează regiuni propuse, numit RPN, iar al doilea este un detector Fast R-CNN. Sistemul funcționează ca o singură rețea, unitară, unde RPN detectează anumite zone de interes iar Fast R-CNN le analizează.

method	proposals	training data	COCO val		COCO test-dev	
			mAP@.5	mAP@[.5, .95]	mAP@.5	mAP@[.5, .95]
Fast R-CNN [2]	SS, 2000	COCO train	-	-	35.9	19.7
Fast R-CNN [impl. in this paper]	SS, 2000	COCO train	38.6	18.9	39.3	19.3
Faster R-CNN	RPN, 300	COCO train	41.5	21.2	42.1	21.5
Faster R-CNN	RPN, 300	COCO trainval	-	-	**42.7**	**21.9**

Fig. 2. Comparația dintre diverse versiuni ale modelelor CNN)[2]

C. Diferente si detalii ale solutiilor

Features	R-CNN and its successors	YOLO and its successors
Region proposals	Region proposals (or ROI) are generated using Selective search algorithm	Region proposals are generated by a single convolutional network.
Feature extraction (Backbone network)	The backbone network is a heavyweight and time consuming.	The backbone network is a lightweight and faster feature extractor.
Number of stages and their role	First stage extracts region proposals. Second stage extracts feature vectors, thereafter detections.	A single stage network predicts the bounding boxes offsets, confidence score and class conditional probabilities
Speed and accuracy	Higher accuracy and low speed.	Faster detection and accuracy nearer to two stage object detectors.
Computational cost	They require powerful resources for computation and are computationally expensive.	Not necessary for powerful resources for computation and are less expensive.
Performance	They are efficient for detecting smaller and larger objects.	They have mostly shown poor performance for detecting smaller objects and have been efficient for larger objects.

Fig. 3. comparatia functionalitatilor ale algoritmilor de single-stage (precum YOLO) si cei de double-stage (precum R-CNN si succesorii) (in engleza)

IV. Prezentarea implementării finale a soluției tehnice

Noi am folosit modelul YOLOv8 distribuit de Ultralytics si un dataset făcut manual, alcătuit din alte date de pe Roboflow și imagini achiziționate de către noi, unde era nevoie, pentru a face detecția diverselor obiecte dintr-o poză. Modelul este antrenat pe 20 epoci, iar la o nouă imagine oferita, face o "prezicere", afisand care clase anume de obiecte le-a detectat si gradul de "încredere" pe care îl are pentru fiecare în parte.

Concret, odată oferită o poză spre analiză, se crează o nouă poză într-un timp foarte scurt, având căsuțele de delimitare ale fiecărui obiect identificat etichetate corespunzător.

Programul a fost realizat cu ajutorul mediului Google Colab, ce foloseste limbajul Python, care ruleaza pe un T4. De asemenea, este integrat cu Google Drive pentru a facilita importarea datasetului și a pozelor dorite pentru recunoaștere.

Acest dataset are următoarele componente: - imaginile pe care modelul le folosește la antrenare; - date despre fiecare imagine într-un fisier text: ID-ul clasei din care face parte, coordonatele bounding box-ului de unde se află obiectul; - un fisier YAML ce contine informații despre proiect și împărțirea pozelor în cele 3 foldere necesare antrenarii modelului (test, train, valid);

De asemenea, având propriul dataset, este foarte ușor să adăugam sau să eliminăm categorii noi de obiecte, fără a fi nevoie de modificări majore în cod.

Datasetul alcătuit de noi conține la momentul de față 3 categorii de obiecte care pot fi identificate: zaruri alfabetice (din jocul Boggle), cu fiecare fată posibilă din fiecare variantă lingvistică, mobilă de interior (scaun, canapea, masa) si markere (negru, albastru).

Pentru a creea un dataset cât se poate de potrivit pentru a putea antrena, am folosit si imagini editate pentru a ne asigura că modelul este capabil să detecteze obiectele în orice poziție din realitate (imagini rotite, oglindite, din unghiuri si perspective diferite, având culori sau contraste diferite etc).

Pe viitor vor fi adăugate mai multe categorii de obiecte în măsură în care vom găsi/achiziționa imagini mai variate la diverse calități pentru a putea îmbunătății acuratețea modelului.

V. Evoluția soluției de la stadiul intermediar la stadiul final

Modelul YOLO v8 ne oferă informații despre timpul de pe durata antrenării, cât timp a durat să efectueze o prezicere, gradul de încredere (confidence) pe care l-a avut atunci când a detectat fiecare obiect.

Desi YOLO v8 este un model de detecție foarte rapid, eficient si developer-friendly, rezultalele inițiale, deși sunt un inceput bun pentru o soluție out-of-the-box, nu sunt neapărat corecte din punct de vedere al calitatii prezicerilor. Conform analizelor noastre, cu cât modelul se antrenează cât mai puțin, cu atât are tendința de a face mai multe preziceri incorecte.

Pentru a ne asigura că modelul face o recunoastere corectă a tuturor elementelor și pentru a testa eficacitatea acestuia, trebuia să găsim un număr de epoci suficient de mic astfel încât durata de antrenare a modelului să nu fie prea mare, iar rezultatele să fie cât mai apropriate de realitate. La inceput, am luat fiecare categorie de elemente și am căutat un număr de epoci minim necesar pentru care modelul oferea rezultatele dorite. Având acest număr drept bază, a trebuit să vedem de încă câte epoci ar mai fi nevoie pentru ca algoritmul să funcționeze corect pentru o scenă ce conține mai multe (sau chiar toate) categoriile de obiecte pe care ni le dorim să le identificăm.

Nr. epoci	Tip obiect			
	Mobilă	Zar	Marker	Toate
10-12	Incorect	Incorect	Incorect	Incorect
14	Parțial Corect	Parțial Corect	Corect	Parțial Corect
16	Incorect	Corect	Corect	Parțial Corect
18	Parțial Corect	Parțial Corect	Corect	Parțial Corect
20	Corect	Corect	Corect	Corect

Fig. 4. Monitorizarea rezultatelor

De asemenea, ne-am dorit să măsurăm și perioada de timp în care s-a antrenat modelul, aceasta fiind strâns legată de numărul de epoci folosite.

Timpul de prezicere nu a fost măsurat deoarece acesta era mereu cuprins în intervalul de 13-16 ms, indiferent de numărul de epoci în care a fost lăsat modelul să se antreneze sau tipul de detecție (a unei singure categorii de obiecte sau a mai multor).

Nr. epoci	Timp durata antrenament pentru tipul de obiect			
	Mobilă	Zar	Marker	Toate
10-12	2.8 – 3.6 m	6,12 - 7.6 m	~1 m	22.8 m
14	4.1 m	7.9 m	6.7 m	26.4 m
16	4.8 m	9.18 m	7.8 m	30.6 m
18	5.54 m	10.5 m	9.6 m	35.3 m
20	6.14 m	11.7 m	10.08 m	39.12 m

Fig. 5. Monitorizarea timpilor de antrenare

Singurele metrici pe care nu le-am putut monitoriza au fost cele ale resurselor folosite, deoarece varianta gratis de Google Colab are acces limitat și nu permite modificarea parametrilor mașinii pe care rulează.

VI. Concluzii

Modelele de vedere artificială dovedesc, în continuare, că sunt pe drumul cel bun, iar îmbunătățirile continue din acest domeniu ajută la optimizarea timpilor de antrenare, prezicere.

Pe viitor, ne dorim să putem creea un model mai optimizat, cu un dataset mai divers și, eventual, cu funcționalități suplimentare. Am vrea să testăm și performanța modelului la diverși factori externi ce ar ingreuna recunoașterea obiectelor (zgomot, rezoluție prea mică, imagini distorsionate etc).

References

[1] Tausif Diwan 1 & G. Anirudh 2 & Jitendra V. Tembhurne, "Object detection using YOLO: challenges, architectural successors, datasets and applications", Springer Science, 8 August 2022.

[2] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", arxiv 6 Jan 2016.

[3] Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, Yichen Wei "Relation Networks for Object Detection", arXiv:1711.11575v2, 14 Jun 2018

[4] Mingxing Tan, Ruoming Pang, Quoc V. Le "EfficientDet: Scalable and Efficient Object Detection", arXiv:1911.09070v7, 27 Jul 2020

[5] J. Seo and H. Park, "Object Recognition in Very Low Resolution Images Using Deep Collaborative Learning," in IEEE Access, vol. 7, pp. 134071-134082, 2019, doi: 10.1109/ACCESS.2019.2941005.

[6] Ahmad Jalal, Abrar Ahmed, Adnan Ahmed Rafique, Kibum Kim "Scene Semantic Recognition Based on Modified Fuzzy C-Mean and Maximum Entropy Using Object-to-Object Relations", Digital Object Identifier 10.1109/ACCESS.2021.3058986, February 19, 2021

[7] Zhengxia Zou, Keyan Chen, Zhenwei Shi, Yuhong Guo, Jieping Ye "Object Detection in 20 Years: A Survey" DOI: 10.1109/JPROC.2023.3238524, 27 January 2023

[8] https://viso.ai/deep-learning/yolov3-overview/

[9] https://docs.ultralytics.com/yolo-a-brief-history

[10] https://universe.roboflow.com/roboflow-100/furniture-ngpea

[11] https://universe.roboflow.com/joseph-nelson/boggle-boards

[12] https://universe.roboflow.com/marker-ijzui/marker-colors