SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 2291

KONVOLUCIJSKI MODELI ZA SEMANTIČKU SEGMENTACIJU PRIMJERAKA

Bruno Kovač

Zagreb, lipanj 2020.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 2291

KONVOLUCIJSKI MODELI ZA SEMANTIČKU SEGMENTACIJU PRIMJERAKA

Bruno Kovač

Zagreb, lipanj 2020.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Zagreb, 13. ožujka 2020.

DIPLOMSKI ZADATAK br. 2291

Pristupnik:	**Bruno Kovač (0036490749)**
Studij:	Računarstvo
Profil:	Računarska znanost
Mentor:	prof. dr. sc. Siniša Šegvić

Zadatak: **Konvolucijski modeli za semantičku segmentaciju primjeraka**

Opis zadatka:

Lokaliziranje objekata u slikama predstavlja važan problem računalnog vida s mnogim zanimljivim primjenama. U posljednje vrijeme najbolji rezultati u tom području postižu se dubokim konvolucijskim modelima. Tema ovog rada su postupci koji pored pravokutnog okvira prediktiraju i segmentacijsku masku primjerka objekta. U okviru rada, potrebno je ukratko opisati model Mask R-CNN za seantičku segmentaciju primjeraka. Implementirati i evaluirati podmodel za predlaganje kandidata objekata (RPN). Implementirati model za raspoznavanje kandidata s glavama za klasifikaciju i segmentaciju. Validirati hiperparametre, prikazati i ocijeniti ostvarene rezultate te provesti usporedbu s rezultatima iz literature. Predložiti pravce budućeg razvoja. Radu priložiti detaljne rezultate eksperimeata te izvorni kod razvijenih postupaka uz potrebna objašnjenja i dokumentaciju. Citirati korištenu literaturu i navesti dobivenu pomoć.

Rok za predaju rada: 30. lipnja 2020.

Ovom prilikom htio bih zahvaliti prof. dr. sc. Siniši Šegviću na pomoći i mentorstvu pri izradi svih zadataka počevši od Završnog rada pa sve do Diplomskog rada. Isto tako zahvalio bih i Josipu Šariću na svoj pomoći prilikom izrade ovog rada.

Posebne zahvale moram uputiti i svojim roditeljima na podršci i potpori tijekom cijelog školovanja pa tako i do samog kraja fakultetskog obrazovanja.

SADRŽAJ

1. Uvod

U posljednjih nekoliko godina možemo primijetiti sve veću korištenost raznih modela računalnog vida u svakodnevnom životu. S pojačanim razvojem područja primjene autonomnih robota, vozila i sl. pojavili su se novi zadaci i problemi koje prethodno razvijeni modeli nisu mogli riješiti. Ubrzo je postalo jasno da nam više nije dovoljno samo klasificirati ulaznu sliku u neku od određenih klasa, nego je sve bitnije postalo i točno locirati takav objekt, kao i prilagoditi se na situaciju da na slici može biti više različitih objekata, s čim se noviji modeli moraju moći nositi. Tako se slijedeći te potrebe ubrzano razvijalo područje detekcije i lokalizacije objekata, koje je cijelo vrijeme bilo pod budnim okom javnosti i svaka pogreška mogla je biti kobna.

Upravo zbog toga razvijali su se sve bolji i bolji duboki konvolucijski modeli koji bi čim preciznije lokalizirali objekte. Za potrebe lokalizacije razvijene su konvolucijske neuronske mreže temeljene na regijama - R-CNN [3] (engl. *Region-CNN*). Originalni R-CNN model temeljio se na algoritmu selektivnog pretraživanja koji bi generirao nekoliko tisuća regija interesa. Te regije koje bismo tada grupirali i dodatno propustili kroz konvolucijsku neuronsku mreže te SVM nakon čega bismo dobili klasifikaciju pojedinih područja te korekcije pozicija objekata. Ovaj postupak je bio poprilično spor upravo zbog potrebe ponavljanja postupka klasifikacije i lokalizacije za sve predložene regije. Kao posljedicu tog problema isti autori razvili su sljedeći model - Fast R-CNN. Fast R-CNN riješio je problem brzine prethodnog modela tako što je uvedena konvolucijska neuronska mreža koja bi generirala mape značajki na temelju ulazne slike te bi se tada generiranje regija interesa radilo na tim značajkama što je smanjilo broj unaprijednih prolaza kroz mrežu na samo jedan. Nakon tog prolaza kroz konvolucijsku mrežu provodi se selektivno pretraživanje, RoI Pooling te potpuno povezane slojeve nakon čega kao rezultat dobivamo klasifikaciju i lokaciju objekata.

Prethodna opisana dva modela i dalje su dijelila jedan problem koji je dosta utjecao na performanse i preciznost lokalizacije objekata, a to je upravo navedeni selektivni algoritam za predlaganje regija interesa. Kao rješenje ovog problema stvoren je sljedeći model - Faster R-CNN - koji, kako mu u ime govori, rješava preostali problem br-

zine prethodnih modela te dobivamo još brže vrijeme izvođenja u odnosu na R-CNN i Fast R-CNN. Iako je ovaj model u mnogočemu sličan svom neposrednom prethodniku, ključnu razliku možemo pronaći u zamjeni algoritma selektivnog pretraživanja novom konvolucijskom mrežom za predlaganje regija interesa - RPN (engl. *Region Proposal Network*). Prednost uvođenja ove konvolucijske neuronske mreže je u tome što sada i za predlaganje regija imamo model koji možemo učiti te tako u samom početku dobiti bolje prijedloge regija interesa koje tada samo dodatno modificiramo i klasificiramo u nastavku modela.

Najnovije unaprijeđenje prethodnih modela predstavlja Mask R-CNN. Ovaj model uvelike se temelji na Faster R-CNN-u, a ključna promjena je dodavanje izlaza za semantičku segementaciju koja nam u ovom slučaju može poslužiti i za odvojenu segmentaciju primjeraka iste klase upravo zbog već odvojenih okvira svakog objekta. Konkretno zbog potrebe segmentacije i RoI Pooling je zamijenjen RoI Align-om. Upravo Mask R-CNN je temelj ovog rada te ćemo u nastavku detaljnije proći kroz sve njegove komponente i specifičnosti, a ujedno i kroz nekoliko zanimljivih primjena.

2. Kralježnica modela

Prvi od nekoliko korištenih modela u sklopu Mask R-CNN-a svakako je njegova kralježnica (engl. *backbone*) - konvolucijska neuronska mreža koja se koristi za ekstrakciju korisnih značajki iz ulaznih podataka. Ovaj početni korak ključan je za cijeli nastavak ukupnog modela jer sve daljnje operacije i predviđanja regija interesa, klasifikacija, segmentacija i sl. temeljimo upravo na dobrim informacijama o značajkama slika koje dobivamo od kralježnice modela. Iako konkretna implementacija kralježnice Mask RCNN-a u originalnom radu nigdje nije strogo definirana, za ovu svrhu najčešće se koriste konvolucijski dijelovi dubokih konvolucijskih neuronskih mreža, kao npr. rezidualnih neuronskih mreža ResNet.

2.1. Rezidualne neuronske mreže - ResNet

S pojavom sve kompleksnijih problema klasifikacije i općenito računalnog vida, s vremenom se razvila potreba za sve kompleksnijim i sve dubljim modelima koji bi tako dobili veći kapacitet za prilagodbu podacima koje moraju naučiti. Početno rješenje bilo je jednostavno dodavanje sve većeg broja slojeva, ali ubrzo se pokazalo da to rješenje ne daje rezultate, štoviše neki takvi modeli su davali lošije rezultate od plićih modela. Jedan od ključnih problema je bio problem iščezavajućeg gradijenta (engl. *vanishing gradient*). Nakon određenog vremena učenja, kada je model već dosta naučio, gradijenti postaju sve manji te zbog toga gotovo potpuno nestanu do dolaska unazadanim prolazom do početnih slojeva mreže.

Rješenje ovog velikog problema pojavilo se u ideji uvođenja normalizacije po grupi (engl. *batch normalization*) [9]. Normalizacija po grupi predstavlja dodatan sloj mreže kojeg postavljamo između izlaza pojedinih slojeva i njihovih aktivacija kako bismo ujednačili moguće vrijednosti i riješili problem potencijalno velikog raspona istih. Ovaj sloj uvodi dodatne parametre standardne devijacije i srednje vrijednosti razdiobe koje također učimo u postupku učenja cjelokupnog modela. U konačnici te vrijednosti koristimo i prilikom upotrebe naučenog modela za normalizaciju izlaznih vrijednosti

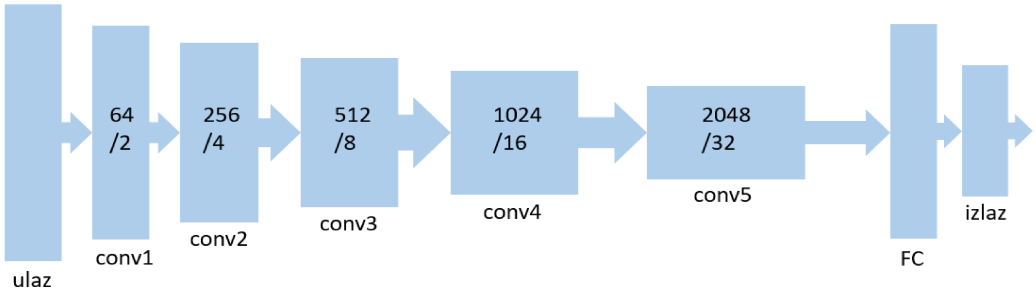
pojedinih slojeva. Normalizacija po grupi uvelika je pomogla kod problema iščezavajućeg gradijenta značajno zagladivši funkciju gubitka [15] što je rezultiralo i boljim te bržim postupkom optimiziranja gubitka modela.

Unatoč uvođenju normalizacije po grupi, dodavanjem sve većeg broja konvolucijskih slojeva te produbljivanjem samih modela uočen je problem stagnacije pa potom i degradacije točnosti i ostalih metrika. Iako bismo s većim brojem slojeva očekivali sve bolje rezultate, došlo je do situacije da i jednostavnije funkcije ne možemo dobro naučiti u slučaju velike dubine modela. Rješenje ovog velikog problema pojavilo se u ideji rezidualnih mreža s preskočnim vezama - ResNet. Ovo rješenje ponudio je Microsoftov istraživački tim (Kaiming He i ostali) 2015. godine u radu *Deep Residual Learning for Image Recognition* [5]. Uveden je koncept rezidualne funkcije i rezidualnih jedinica [6] čijim slaganjem dobivamo veliku dubinu modela. Svaki blok se sastoji od nekoliko konvolucijskih slojeva te prečice od početna do kraja navedenog bloka. Izlaz rezidualne jedinice predstavlja sumu prolaska kroz nekoliko dodatnih slojeva i samog ulaza bloka. Upravo ovaj princip preskočnih veza rješava prethodni problem tako da dodatno zaglađuje funkciju gubitka [12] pa optimizacija bolje napreduju. U prolasku unazad gradijenti mogu preskočiti nekoliko konvolucijskih slojeva te tako s jačim efektom djeluju na ažuriranje težina prethodnih slojeva. Bitno je napomenuti i da za modele od pedeset ili više konvolucijskih slojeva koristimo malo drugačije definirane rezidualne jedinice, točnije jedinice s uskim grlom (engl. *bottleneck*) gdje se korištenjem 1x1 konvolucija smanjuju dimenzije mapa značajki nad kojima će se provodi operacije konvolucije, nakon čega istim postupkom na kraju bloka radimo povrat na ulazne dimenzije.

Ova ideja primjenjiva je na više konvolucijskih modela različitih dubina, a unatoč povećanju dubine ne dolazi do prethodno opisanih negativnih efekata. Ovisno o broju konvolucijskih slojeva naslaganih u rezidualne jedinice dobivamo različita imena ovakvih modela, npr. ResNet-34, ResNet-50, ResNet-101 itd., gdje iz svakog imena možemo iščitati broj blokova. Svaki od ResNet modela sastoji se i od pet većih blokova koje najčešće nazivamo *conv1, conv2, conv3, conv4, conv5*, što će nam kasnije biti bitno. Redom navođenja izlazi ovh blokova su sljedećih veličina: *(*$w/2$*,* $h/2$*), (*$w/4$*,* $h/4$*), (*$w/8$*,* $h/8$*), (*$w/16$*,* $h/16$*), (*$w/32$*,* $h/32$*)*, gdje w označava širinu, a h visinu ulazne slike.

2.2. Korišteni model - ResNet-50

U ovom radu je kao kralježnica modela Mask R-CNN-a korišten duboki rezidualni model Resnet-50. Model ResNet-50 prikazan je na slici 2.1. Za potrebe kralježnice Mask R-CNN-a koristimo samo izlaze *conv2, conv3, conv4* i *conv5* blokova za ekstraciju značajki, dok potpuno povezani i izlazni sloj možemo izbaciti. Kod inicijaliziranja ovog modela ujedno definiramo i korištenje ImageNet inicijalizacije težina te uklanjamo završne potpuno povezane slojeve koji nam nisu potrebni za esktrakciju značajki iz ulaznih slika. Kako se ImageNet skup podataka sastoji od velikog broja slika raspodijeljenih u tisuću različitih klasa, upravo ta velika količina podataka koju modeli učeni na njemu moraju zapamtiti daje odličnu podlogu za daljni rad.



Slika 2.1: ResNet-50

U sklopu ovog rada implementiran je i vlastiti model ResNet-50, no usporedbom s predtreniranim modelom možemo primijetiti da inicijalizacija prethodno naučenim težinama uvelike pomaže i ubrzava naredni rad. Za potrebe implementacije Mask R-CNN-a korisni su nam izlazi ranije navedenih većih blokova sastavljenih od više rezidualnih jedinica. Za učenje našeg skupa podataka koristi se samo ugađanje predtreniranih parametara (engl. *fine-tuning*).

3. FPN - rezolucijska piramida značajki

Sljedeća komponenta koju moramo obraditi u sklopu cjelokupnog modela Mask R-CNN-a je rezolucijska piramida značajki FPN (engl. *Feature Pyramid Network*) [13]. Iako za implementaciju ovog modela, točnije RPN-a, nije nužno koristiti FPN strukturu, često se koristi kao nadogradnja kralježnice zbog mnogih pozitivnih efekata koje nam ona donosi. Specifičnosti ovakve organizacije strukture kralježnice uvelike pomažu u boljim rezultatima raznih problema lokalizacije i segmentacije objekata. Iako se u ovom radu koristi FPN, bitno je napomenuti kako alternativni pristup rezolucijskoj piramidi značajki predstavlja ljestvičasto naduzorkovanje koje je također dosta korišteno [10] [11].

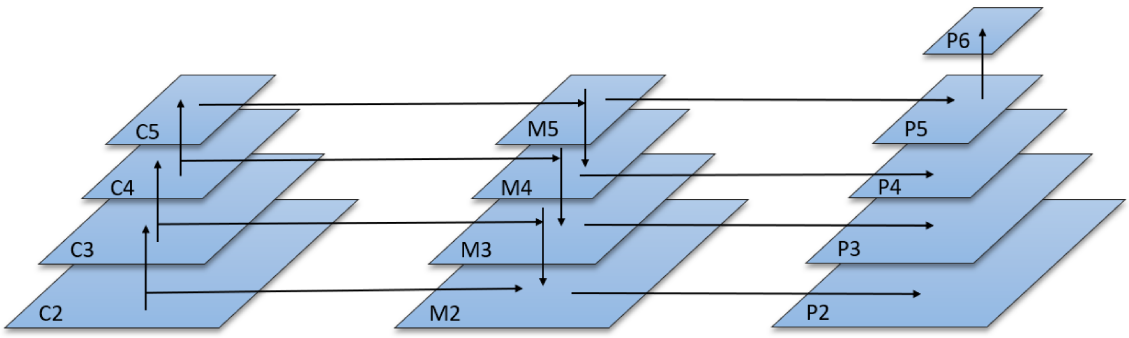
3.1. Opis strukture

Klasifikacijski modeli konvolucijskih neuronskih mreža tipično se temelje samo na jednom, tzv. bottom-up prolazu te se u konačnici generiraju značajke 32 puta umanjene rezolucije. U ovom procesu prolaskom kroz sve više konvolucijskih slojeva u izlaznim aktivacijskim mapama dobivamo sve više informacija o bitnim značajkama ulaznih podataka, ali usput gubimo na prostornim dimenzijama i rezoluciji podataka. Za potrebe klasifikacije slika gubitak dimenzija i razlučivosti nije ni toliko bitan jer u konačnici iz semantički vrlo jakih mapi značajki na vrhu bottom-up prolaza imamo dovoljno informacija za kvalitetno prepoznavanje točne klase prikazanih objekata.

Međutim u slučaju kada nam je cilj i lokalizirati pojedine objetke, a ne samo odrediti kojoj klasi pripadaju, ranije navedeni gubitak prostornih dimenzija i rezolucije nam predstavlja veliki problem. Ukoliko izgubimo precizne podatke o poziciji objekata, do čega neminovno i dolazi, posljedično gubimo i na preciznosti točnog pozicioniranja pojedinih objekata na ulaznoj slici. Naizgled dobra ideja moglo bi biti korištenje mapi značajki s nižih razina piramide u kojima još imamo bolju rezoluciju i očuvane točnije

informacije o prostornom rasporedu objekata. Iako se ovaj pristup na prvu čini kao dobar, analizom dolazimo do zaključka da ranije aktivacijske mape nemaju dovoljno jak semantički značaj te kao takve ne mogu poslužiti za klasifikaciju i lokalizaciju objekata.

Upravo ovome problemu želimo doskočiti uvođenjem rezolucijske piramide dubokih značajki koja uz podatkovni put za poduzorkovanje (engl. *bottom-up*) sadrži i podatkovni put za naduzorkovanje (engl. *top-down*). Na slici 3.1 bottom-up prolaz prikazan je na lijevom dijelu, dok je top-down u sredini. Cilj nam je dodatnim top-down prolaskom još malo semantički obogatiti mape značajki, no ono ključno što dobivamo u tom prolazu mrežom je kombinacija jače semantike i bolje rezolucije čime dobivamo vrlo kvalitetne aktivacijske mape koje sadrže mnogo korisnih informacija o značajkama ulaznih podataka, a ujedno i ne gube na rezoluciji što nam je vrlo bitno za lokalizaciju objekata. U dodatnom top-down prolasku modela počevši od najgrublje razine piramide pa niže lateralnim vezama (prikazane vodoravno) kombiniramo prostorno i rezolucijom kvalitetnije mape značajki iz bottom-up prolaska s onima semantički značajnijim iz bottom-up prolaza (pri čemu su te veze prikazane okomitim prema dolje usmjerenim strelicama). Kako su aktivacijske mape viših razina piramide manjih dimenzija nego one s kojima ih želimo kombinirati, potrebno je napraviti naduzorkovanje kako bismo međusobno prilagodili sve dimenzije. U konačnici nakon top-down prolaza kao rezultat dobivamo semantički i prostorno značajne mape značajki pune korisnih informacija kako za klasifikaciju tako i za lokalizaciju, što možemo iskoristiti za bolje konačne rezultate.



Slika 3.1: FPN - rezolucijska piramida značajki. Lijevo je prikazan bottom-up prolaz kralježnice modela, u sredini top-down dodatni prolazak karakterističan za FPN te desno mape značajki koje ćemo koristiti za potrebe RPN-a.

3.2. Detalji izvedbe

Pogledajmo sada i što konkretno za naš model predstavlja prethodno opisana rezolucijska piramida značajki FPN te kako je ona implementirana u skladu s opisanim potrebama i specifičnostima ovakve organizacije.

Vratimo se za početak na prethodno opisane konvolucijske blokove ResNet modela opisanih u poglavlju 2.1. Izlaze tih većih konvolucijskih blokova tamo smo nazvali *conv2, conv3, conv4* i *conv5* što odgovara oznakama *C2, C3, C4* i *C5* sa slike 3.1. Mape značajki koje dobivamo na izlazima ovih blokova umanjenje su redom za 4, 8, 16 i 32 puta u odnosu na originale dimenzije ulaznih podataka.

Prvi korak koji moramo napraviti za potrebe naduzorkovanja predstavlja prilagodbu broja kanala aktivacijskih mapa kralježnice u svrhu njihovog izjednačavanja. Upravo to možemo lako postići konvolucijama s filtrima veličine 1x1 kojima zadržavamo visinu i širinu mapi značajki, a dubinu, tj. broj kanala, prilagođavamo na jednaki broj, npr. 256. Nazovimo izlaze tih konvolucijskih slojeva jednake dubine oznakama *C2', C3', C4'* i *C5'*.

Sada kada smo pojasnili lateralne veze između bottom-up i top-down prolaza sa slike 3.1, ostaje nam još objasniti dolje usmjerene strelice iz top-down prolaska. Kako smo već ranije opisali, a i jasno je iz grafičkog prikaza i općenito strukture konvolucijskih neuronskih mreža, dimenzije mapa značajki *M2, M3, M4* i *M5* nisu jednake. Zbog toga prilikom prolaska prema dolje moramo uskladiti dimenzije aktivacija razine iznad i onih dobivenih lateralnim vezama iz bottom-up prolaza što se postiže naduzorkovanjem metodom najbližeg susjeda (engl. *nearest-neighbour upsampling*) [13]. Svaka niža razina top-down prolaska tako se dobiva zbrajanjem aktivacijskih mapa razine iznad i lateralnim vezama prilagođenih mapa iz bottom-up prolaska.

Sada imamo definirano gotovo sve što nam treba za potpunu definiciju FPN-a. Jedina razlika koju treba naglasiti je ta da se u konačnici koriste mape značajki *P2, P3, P4* i *P5* koje su dobivene primjenom još jedne konvolucije na odgovarajuće aktivacije *M2, M3, M4* i *M5*. Zbog mogućnosti pojave većih objekata i za potrebe RPN-a bitno je napomenuti da se uvodi i aktivacijska mapa *P6* dobivena primjenom sažimanja maksimumom.

4. RPN - modul za predlaganje područja interesa

Idući ključan dio ukupnog modela Mask R-CNN-a nazivamo modulom za predlaganje regija interesa RPN (engl. *Region Proposal Network*). Ovaj podmodel uveden je kao veliki novitet u Faster R-CNN-u [14], a predstavlja veliku razliku ideje i implementacije prijedloga područja u kojima se nalaze objekti. Upravo ova ideja značajno je ubrzala rad R-CNN modela u odnosu na Fast R-CNN i neposredne prethodnike. Osim toga korištenje modela koji sam ima sposobnost učenja određenih podataka pomoglo je i u preciznost početnih prijedloga okvira objekata koje ćemo u nastavku samo još dodatno modificirati.

4.1. Sidra

Prvi koncept koji moramo obraditi svakako su sidra (engl. *anchors*). Sidra predstavljaju početnu, ali i temeljnu komponentu na kojoj se temelji rad RPN-a. Ona su na početku jednaka za svaku sliku s kojom ćemo raditi, no bitna razlika u odnosu na prethodnike ovog modela je ta da ipak u cjelokupnom postupku učenja za svako sidro također učimo nekoliko korisnih informacija za njihovo poboljšanje. Konačni cilj nam je dobiti okvire koji sadrže pouzdanu informaciju nalazi se unutar njihovih granica objekt ili pokrivaju samo pozadinu. Ujedno želimo saznati i podatke o potrebnom pomaku središta sidra te modifikacijama vezanim uz njihovu visinu i širinu kako bismo na kraju imali bolje prilagođene regije interesa od samih početnih sidara. Upravo to nam pomaže u incijalno boljim prijedlozima regija nego što je to prethodno bio slučaj te samim time i na samom kraju Mask R-CNN-a imamo bolje rezultate.

Iako RPN možemo koristiti i na temelju same jedne rezolucije u kojem slučaju bismo imali samo jednu veličinu sidra, u ovom slučaju podrobnije ćemo obraditi situaciju kada koristimo i rezolucijsku piramidu značajki FPN. Kako kod FPN-a imamo više razina s kojih možemo dohvatiti korisne mape značajki tako možemo i dobiti veću

raznolikost sidara. Ključne karakteristike svakog sidra predstavljaju njegova veličina (engl. *scale*) i omjer visine i širine (engl. *ratio*). U slučaju korištenja FPN-a raznolikost sidara postižemo tako što svaka razina sadrži sidra različite veličine, dok omjere visine i širine jednako primjenjujemo na sva sidra na svim razinama piramidalne strukture. Tako u konačnici na svakom pikselu svake aktivacijske mape određene razine dobivamo nekoliko različitih sidara koja pokrivaju objekte različitih omjera visine i širine, a primjenom takvog postupka na više razina obuhvaćamo i objekte različitih veličina.

Pojasnimo sada prethodno opisano na konkretnom primjeru razina opisanih u poglavlju 3.2. Najnižu razinu piramide koju koristimo nazivamo *P2*. Mape značajki te razine najvećih su prostornih dimenzija i najmanjeg receptivnog polja pa upravo zbog toga sadrže sidra najmanje veličine. Iako su sidra ove razine najmanja, upravo zbog toga ih i imamo najviše. Kako idemo prema razini piramide *P6* mape značajki su sve manjih dimenzija i većeg receptivnog polja te stoga imamo sve manje sidara većih dimenzija. Sve što smo upravo opisali možemo vidjeti i na slici 4.1 koja prikazuje sidra na razinama *P2, P3* i *P4*. Za svaku razinu FPN-a prikazana su sidra svih omjera na pojedinoj lokaciji, u ovom konkretnom slučaju to su omjeri visine i širine od 1:1, 1:2 i 2:1, a usput možemo i primijetiti povećanje veličine sidara kako se krećemo prema vrhu piramidalne strukture.

(a) Razina P2 **(b)** Razina P3 **(c)** Razina P4

Slika 4.1: Sidra na različitim razinama FPN-a. Prikazana su sidra na različitim mapama rezolucijske piramide značajki, redom od niže pa prema višoj razini. Svi korišteni omjeri visine i širine sidra prikazani su za jednu određenu poziciju po svakoj razini FPN-a.

4.2. Pripremanje oznaka

Iako nam se na prvu prethodno opisani model RPN-a možda čini jednostavan, pri kraćoj analizi istinitih (engl. *ground truth*) izlaza modela koje ćemo htjeti naučiti primjećujemo da odmah dolazimo do problema. Za razliku od klasičnih problema strojnog i dubokog učenja gdje uglavnom kao istinite izlaze dobijemo klasu objekta i odmah

s tim možemo krenuti trenirati modele, u ovom slučaju ranije opisanu klasifikaciju u klase objekt/pozadina te pomake za centre i dimezije sidara nigdje ne dobivamo te je potrebno provesti dodatnu obradu istinitih izlaza kako bismo ih prilagodili našim potrebama. Upravo kvalitetna izvedba ovog početnog koraka, bez kojeg ne možemo ni započeti implementaciju modela za predlaganje regija interesa RPN, ključna je za dobar rad ovog modela i uopće postupak optimizacije.

Krenimo za početak od problema klasifikacije područja sidra kojeg RPN mora riješiti. Mreža za predlaganje regija bavi se određivanjem toga pokriva li neko specifično sidro objekt ili pozadinu, dakle radi se o problemu klasifikacije u samo dvije klase. Kako u samim istinitim podacima skupa podataka nećemo direktno dobiti ove podatke, nego uglavnom samo informaciju o okviru i klasi objekta kojeg on pokriva, te podatke moramo sami pripremiti za postupak optimizacije gubitka RPN-a. Za potrebe ovog problema iskoristit ćemo mjeru udjela presjeka dvaju okvira u njihovoj ukupnoj površini - IoU (engl. Intersection over Union). Nakon što izračunamo površinu presjeka okvira i podijelimo ga s ukupnom površinom koju tu okviri pokrivaju dobit ćemo postotak tog udjela, dakle broj u intervalu 0-1 kojeg dalje možemo iskoristiti. Sada kada imamo IoU mjeru između svih RPN sidara i istinitih okvira koji su određeni u skupu podataka, možemo definirati i nekoliko različitih situacija u ovisnosti o toj mjeri presjeka. Odredimo sada donju (npr. 0.3) i gornju (npr. 0.7) granicu kojima ćemo podijeliti IoU raspon vrijednosti na tri intervala [14]. Ukoliko je iznos presjeka ispod ispod donje granice takva sidra ne pokrivaju nijedan objekt i nazivamo ih negativnim. Ako je mjera presjeka između donje i gornje granice takva sidra smatramo neutralnim. Ukoliko je udio presjeka iznad gornje granice za takva sidra smatramo da u sebi sadrže dovoljan dio objekta i nazivamo ih pozitivnim. Kako su nam pozitivna sidra najbitnija, a postoji i mogućnost da zbog specifičnog oblika i veličine objekta nijedno sidro ne pokriva taj objekt s IoU mjerom presjeka većom od gornje granice, u tom slučaju kao pozitivna sidra uzimamo sva ona koja imaju najveći udio presjeka koliki god on bio. Razlog tome je što u svakom slučaju želimo pokriti sve objekte kako ne bismo propustili naučiti korisne informacije o bilo kojem objektu iz skupa podataka.

Pozitivna sidra su nam možda i najzanimljivija za promatranje i daljnji rad, a primjer takvih sidara možemo vidjeti i na slici 4.2. Na toj slici se nalaze četiri istinita objekta (biljka, dva stolca i stol), a upravo na tom primjeru prikazna su sva pozitivna sidra koje će model morati naučiti. Oko svakog objekta možemo primijetiti po nekoliko grupiranih pozitivih sidara što nam je u konačnici i bio cilj postići kako bismo imali više primjera sidara koja RPN tada mora zapamtiti kao primjerke značajki objekata.

Slika 4.2: Pozitivna sidra

Sljedeća komponenta koju moramo obraditi je priprema za regresijski problem određivanja pomaka pojedinih sidara kako bi ona čim preciznije obuhvatila objekt kojeg moraju pokriti. Svaki istiniti okvir, kao i svako sidro mreže za predlaganje regija interesa, definiran je koordinatama gornjeg lijevog (x_1, y_1) i donjeg desnog vrha (x_2, y_2). Kako bismo imali mogućnost učenja prethodno opisanih pomaka centra i dimenzija sidra, potrebno je izračunati te vrijednosti na temelju navedenih koordinata. Taj postupak provodimo sljedećim formulama, pri čemu oznaka a predstavlja sidro RPN-a, a gt istiniti okvir iz izlaza skupa podataka. Konačni skup istinitih izlaza koje ćemo koristiti za učenje RPN-a sastoji se od podataka (dx, dy, dw, dh) koji redom predstavljaju pomak x i y koordinate centra sidra te modifikacije visine i širine istog.

$$a_h = a_{y2} - a_{y1} \qquad gt_h = gt_{y2} - gt_{y1}$$

$$a_w = a_{x2} - a_{x1} \qquad gt_w = gt_{x2} - gt_{x1}$$

$$a_x = \frac{a_{x2} - a_{x1}}{2} \qquad gt_x = \frac{gt_{x2} - gt_{x1}}{2}$$

$$a_y = \frac{a_{y2} - a_{y1}}{2} \qquad gt_y = \frac{gt_{y2} - gt_{y1}}{2}$$

$$dx = \frac{gt_x - a_x}{a_w} \qquad dy = \frac{gt_y - a_y}{a_h}$$

$$dw = \ln(\frac{gt_w}{a_w}) \qquad dh = \ln(\frac{gt_h}{a_h})$$

Isto tako potrebno je napomenuti kako ove pomake računamo samo za pozitivna sidra jer zapravo samo za njih i želimo naučiti kako ih modificirati da čim bolje obuhvate objekt, dok za neutralna i negativna sidra nema potrebe za tim. Također za učenje klasifikacije sidara koristimo samo pozitivna i negativna sidra koja redom učimo da pripadaju klasi objetka i pozadine, dok neutralna sidra koja su na granici obje klase upravo zbog toga zanemarujemo.

Nakon što smo odredili IoU mjeru presjeka svih sidara RPN-a sa svim istinitim okvirima objekata te odredili podjelu sidara na pozitivna, neutralna i negativna nailazimo na novi problem kojim se moramo pozabaviti. Kako u postupku učenja modela koristimo samo negativna i pozitivna sidra, pri čemu negativnih ima značajno više, sami izbor svih takvih sidara ne bi bio pogodan zbog velike nebalansiranosti klasa koja bi dovela do toga da bi RPN dobre rezultate postizao i u slučaju da sva područja koja sidra pokrivaju klasificira kao pozadinu. Upravo se zbog u postupku pripremanja željenih izlaza modela unaprijed određuje manja veličina uzorka (engl. *sample*), npr. 128, koji će sadržavati samo pozitivna i negativna sidra, po mogućnosti u jednakom udjelu. To lako postižemo tako da odaberemo određeni podskup pozitivnih i negativnih sidara, a sve ostale označimo kao neutralne te s takvim oznakama pokrećemo postupak učenja. Iako je očito kako ovim postupkom pokrivamo samo vrlo mali udio svih sidara, u svakoj novoj iteraciji učenja uzorak pozitivnih i negativnih sidara se ponovno bira potpuno slučajno i neovisno o prethodnom pa tako kroz više prolazaka kroz iste podatke ipak prođemo veliki broj sidara i tako naučimo korisne informacije.

Ukoliko želimo koristiti čim veći uzorak što bi nam ubrzalo postupak učenja, a ujedno i ne pokvariti omjer pozitivnih i negativnih sidara, prije postupka učenja mreže za predlaganje regija RPN potrebno je napraviti dobru analizu podataka. Prvi korak sastoji se od prolaska kroz sve istinite podatke te izračuna i grupiranja površina istinitih okvira na temelju čega tada određujemo najprikladnije veličine sidara RPN-a kako bismo dobili čim veći broj onih koja pokrivaju objekt, tj. čim veći broj pozitivnih sidara.

4.3. Funkcije gubitka

Ukupna funkcija gubitka modula za predlaganje regija interesa RPN sastoji se od zbroja dvaju različitih gubitaka, prvog za klasifikaciju područja sidra te drugog za regresijski problem predviđanja pomaka centra, visine i širine sidra. U konačnici u procesu optimizacije cilj nam je minimizirati upravo ukupni iznos ovih gubitaka.

Spomenimo prvo klasifikacijski gubitak sidra. Za minimizaciju pogreške klasifi-

kacije područja sidra u klase objekta ili pozadine koristi se gubitak unakrsne entropije. Kao što smo prethodno i spomenuli, ovu funkciju računamo samo na negativnim i pozitivnim sidrima iz ranije definiranog poduzorka svih sidara. Negativna sidra želimo klasificirati kao pozadinu, dok bi pozitivna morala predstavljati objekte, što definiramo u samim istinitim izlazima u predobradi okvira za svaku pojedinu sliku.

Sljedeća funkcija gubitka koju ćemo koristiti je ona za izračun pogreške koju naš model čini pri predviđanju potrebnih pomaka kako bi se prilagodili istinitim okvirima skupa podataka. Nad četiri regresijska rezultata (dx, dy, dw, dh) potrebno je izračunati glatki L1 gubitak (engl. *smooth L1 loss*). Za razliku od običnog L1 gubitka, specifičnost ove varijante je u tome da oko nule imamo zaglađenje umjesto oštrog prijelaza, od čega i dolazi ime ovog gubitka. Formula kojom računamo iznos ove funkcije pogreške za sve komponente predviđenih pomaka sidara definirana je na sljedeći način:

$$L(gt, a) = \sum_{i \in (dx, dy, dw, dh)} smooth\_L1(gt_i - a_i)$$

$$smooth\_L1 = \begin{cases} 0.5x^2, |x| < 1 \\ |x| - 0.5, |x| \geq 1 \end{cases}$$

Oznakom gt označeni su konkretni izračunati pomaci prema istinitim okvirima koje model mora naučiti, dok oznakom a definiramo skup pomaka sidara koje je naš model za predviđanje regija predvidio.

5. Mask R-CNN

Nakon što smo opisali kralježnicu modela te modul za predlaganje regija interesa vrijeme je za predstavljanje ključnog modula ovog rada, a to je sam Mask R-CNN [7]. Kao rezultat obrade ulaznih podataka, u obliku regija interesa koje je predložio RPN, ovaj model određuje klase pojedinih objekata na slici, okvire istih te uvodi i segmentacijsku masku objekta. Upravo segmentacija pojedinih objekata predstavlja novitet ovog modela u odnosu na neposrednog prethodnika. Ovu promjenu u odnosu na Faster R-CNN, pored još nekoliko manjih koje ćemo detaljnije objasniti, uveo je 2018. godine Facebookov istraživački tim predvođen Kaimingom Heom sa suradnicima. Mask R-CNN predstavlja stanje tehnike (engl. *state-of-the-art*) na području lokalizacije i segmentacije pojedinih primjeraka objekata.

5.1. Obrada prijedloga RPN-a

Prvi korak prije ulaska u modul Mask R-CNN-a predstavlja obradu predikcija predloženih regija RPN-a. Kako kao izlaze modula za predlaganje regija interesa dobivamo klasifikacijsku mjeru i potrebne pomake sidara, za početak je potrebno napraviti obradu tih podataka kako bismo dobili stvarna područja na slici za koje RPN s najvećom sigurnošću tvrdi da sadrže objekte.

Na samom početku se na temelju klasifikacije mjere za klasu objekta svakog sidra određuje broj najboljih prijedloga koje ćemo dalje koristiti. Ovaj broj je definiran kao jedan od hiperparametara modela, a odnosi se na određenu količinu najpouzdanijih sidara po slici. U postupku treniranja ovaj hiperparametar je uglavnom veći nego u fazi korištenja modela upravo zbog toga što kod upotrebe modela želimo dobiti i na efikasnosti, a s manjim brojem prijedloga RPN-a brže ćemo ih obraditi te će i u nastavku modela biti manje podataka za daljnju obradu. Vrijednosti ovog hiperparametra mogu biti primjerice 5000 u postupku učenja te 1000 za vrijeme testiranja modela.

Nakon što smo odredili pojedina sidra koja s najvećom vjerojatnošću, po mišljenju modula za predlaganje regija interesa, pokrivaju objekt, sljedeći korak nam je obraditi

ta sidra na temelju predviđenih pomaka svakog od njih. Kako smo i za definiranje istinitih potrebnih pomaka u odnosu na istinite okvire skupa podataka računali specifične vrijednosti koje ćemo učiti, tako je i u ovom koraku potrebno izračunati koordinate vrhova pomaknutih sidara na temelju sljedećih formula:

$$w = x_2 - x_1 \qquad h = y_2 - y_1$$
$$x_C = x_1 + \frac{w}{2} \qquad y_C = y_1 + \frac{h}{2}$$

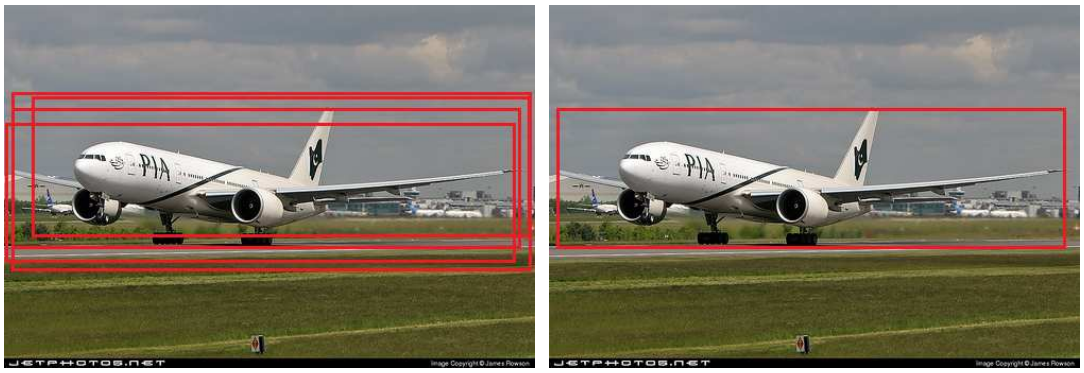
$$x'_C = x_C + dx \cdot w \qquad y'_C = y_C + dy \cdot h$$
$$w' = w \cdot e^{dw} \qquad h' = h \cdot e^{dh}$$

Oznake x_C i y_C predstavljaju centralnu točku sidra RPN-a, a w i h redom visinu i širinu istog. x'_C i y'_C označavaju centar obrađenog prijedloga RPN-a, a w' i h' njegovu visinu i širinu nakon primjene pomaka.

Sada kada imamo izračunate koordinate i dimenzije svih okvira, sljedeće nam ostaje pregledati takve prijedloge područja i izbaciti one nepravilne. Nepravilnim područjima smatramo one koji izlaze izvan dimenzija originalne slike ili imaju vrlo malo površinu gotovo jednaku nuli.

Ovisno o postavkama ranije navedenog hiperparametra za broj najboljih sidara, broj predloženih regija interesa može biti relativno velik. To nije nešto što bismo htjeli u nastavku modela gdje nam prevelika količina područja nužno zahtjeva i velike memorijske potrebe modela. Cijeli postupak klasifikacije, predikcije okvira te segmentacije objekata provodi se za svaku regiju interesa pa tako s njihovim povećanjem imamo više podataka koje je potrebno držati u memoriji u svakom trenutku. Isto tako kako modul za predlaganje regija interesa RPN u postupku učenja postaje sve bolji, te zbog činjenice da je broj istinitih objekata na slici značajno manji od predloženih područja, posljedično se događa situacija da veliki broj regija interesa predstavlja isti objekt što dovodi i do preklapanja dobrog dijela okvira objekata. Za smanjivanje redundancije poslužit ćemo se algoritmom NMS (engl. *Non-Maximum Suppression*). Navedeni algoritam funkcionira na principu odabira predloženog okvira s najvećom klasifikacijskom mjerom te potom uklanjanja svih okvira s mjerom preklapanja IoU većom od definirane granice (npr. 0.7). U slučaju sve sličnijih prijedloga RPN-a po pitanju okvira objekata tako dobivamo dobro filtriranje predloženih regija. Ujedno je i algoritmu NMS moguće zadati maksimalan broj područja koje bismo dobili kao rezultat, a taj broj također definiramo kao hiperparametar sličan onom prethodnom. Ova

maksimalna količina okvira isto je tako određena po pojedinoj slici te različita u postupku učenja i testiranja modela, no taj iznos je i dosta manji od hiperparametra za početni broj prijedloga regija interesa s najvećom klasifikacijskom mjerom. Moguće vrijednosti ovog hiperparametra su primjerice 1000 za postupak učenja te 500 za postupak testiranja. Primjer mogućih rezultata nakon primjene NMS-a prikazan je na slici 5.1.



(a) Prije NMS-a **(b)** Poslije NMS-a

Slika 5.1: Primjena algoritma NMS

Nakon što smo obradili prijedloge modula za predlaganje regija interesa te imamo konkretne koordinate svih okvira ostaje nam još odabrati prikladni izvor značajki za provođenje drugog prolaza obrade. Kako okviri regija interesa mogu biti različitih veličina, tako želimo tu činjenicu i prilagoditi razini s koje ćemo uzimati mape značajki za ekstrakciju korisnih informacija o pokrivenim objektima. Što je područje objekta manje, htjeli bismo ga obraditi na čim finijoj rezoluciji pa odabiremo aktivacije nižih razina piramidalne strukture modela. Što je okvir objekta veći to su nam detalji ipak manje bitni pa možemo koristiti i mape značajki viših razina. Ovdje je bitno napomenuti kako za izvlačenje značajki područja okvira od ranije opisanih razina *P2, P3, P4, P5* i *P6*, razinu *P6* ne koristimo jer je ona dobivena korištenjem sloja sažimanja maksimumom pa je određeni dio informacija izgubljen. Mapiranje pojedinih okvira na određenu razinu izvodi se na temelju njegove veličine, a primjer korištene formule u kojoj na razinu k_0 postavaljamo objekte površine 224 dan je u nastavku:

$$k = \min(\max(\lfloor k_0 + \log_2 \frac{\sqrt{w \cdot h}}{224} \rfloor, 2), 5)$$

U ovom trenutku kada smo već dobili prijedloge RPN-a, obradili ih te filtrirali one najbolje nailazimo na pitanje kako ih iskoristiti u nastavku modela za daljnje probleme

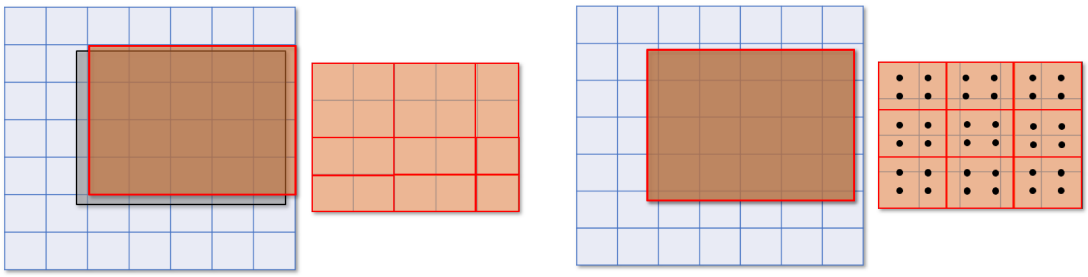
klasifikacije, lokalizacije i segmentacije. Varijabilne dimenzije pojedinih područja okvira predstavljaju nam problem za daljnju obradi značajki pa moramo pronaći dobar način kako se prilagoditi i svesti ih sve na isti oblik neovisno o stvarnim dimenzijama. Upravo ta konačna područja slika koje nam je predložio RPN nazivamo područjima ili regijama interesa - RoI (engl. *Region of Interest*). Fast R-CNN [2] je za ekstrakciju značajki određenog predloženog područja uveo ideju RoI Poolinga, no zbog određenih nedostataka ovog pristupa rad o Mask RCNN-u uvodi novi pristup - RoI Align.

5.1.1. RoI Pooling i RoI Align

Usporedimo sada navedena dva sloja za ekstrakciju značajki. Prvi takav sloj koji se koristio nazivamo RoI Pooling. Kao stariji od spomenuta dva sloja RoI Pooling značajno je jednostavniji za upotrebu, no s time naravno dolazi i do nekih nedostataka koje ćemo opisati u nastavku i pokuštai ih riješiti Roi Alignom.

RoI Pooling sloj se temelji na cjelobrojnoj rešetkastoj podjeli mapa značajki po njezinoj visini i širini. Kao što i znamo predloženi okviri modula za predlaganje regija interesa nisu cijeli brojevi te upravo zbog toga predviđeno područje koje moramo obraditi RoI Pooling slojem gotovo nikada neće točno odgovarati rešetci na koju je podijeljena aktivacijska mapa koju koristimo. Primjer takve situacije možemo primijetiti na slici 5.2 pod a) gdje je crnom bojom prikazan predloženi okvir RPN-a. Ovdje je očito došlo do navedenog problema s kojim se RoI Pooling nosi tako da zaokružuje koordinate vrhova okvira na najbliže cijele brojeve i takvo područje koristi za daljnju obradu. Ovakav okvir zaokruženih koordinata prikazan je crvenom bojom. Upravo pri usporedbi stvarnog okvira crne boje i onog prilagođenog crvene boje možemo uočiti kako smo na lijevim i donjim dijelovima izgubili određeni dio informacija mape značajki, dok smo gore i desno uključili određeni dio podataka koje zapravo prijedlog RPN-a ne pokriva. Kao što je i ranije navedeno, cilj ovog tipa slojeva je svesti sva područja interesa na mape značajki istih dimenzija, tako i RoI Pooling za početak mora podijeliti prethodno odabrana područja na onoliko dijelova kolike izlazne dimenzije želimo dobiti. Ta podjela se također izvodi cjelobrojno kako bismo se prilagodili rešetci aktivacijskih mapa, pri čemu se pazi na čim bolju raspodjelu i sličnost podijeljenih regija. Nakon ovakve podjele na svakom potpodručju provodi se sažimanje maksimumom te kao konačni rezultat dobivamo mapu značajki unaprijed definiranih dimenzija. Na već spomenutom primjeru slike 5.2 možemo primijetiti navedenu podjelu odabranog područja na manja područja, u ovom slučaju željene dimenzije izlaznih značajki su 3x3. Kako je u ovom primjeru predložena regija interesa RPN-a veličine 4x5, a

željeni izlaz 3x3, uzimajući u obzir cjelobrojnu podjelu odabranog područja dobivamo relativno nepravilnu podjelu prikazanu desno pod a) na kojoj tada provodimo postupak sažimanja.



(a) RoI Pooling **(b)** RoI Align

Slika 5.2: Usporedba RoI Pooling i RoI Align slojeva. Na obje slike plavom rešetkom prikazana je mapa značajki, okvir crnog ruba predstavlja predviđeni okvir RPN-a, a crveni okvir stvarno područje koje se obrađuje. Područje interesa izdvojeno je u rešetku crvene boje te su prikazane podjele tog područja u ovisnosti o primjeni RoI Pooling ili Roi Align sloja.

Iz opisa i slikovnog primjera RoI Poolinga možemo uočiti neke nedostatke tog sloja koje ćemo pokušati riješiti RoI Alignom. Za početak prilagodbom predloženog okvira regije interesa RPN-a na cjelobrojne koordinate mape značajki gubimo određene informacije te isto tako potencijalno i ubacujemo određene podatke koji zapravo ne bi trebali biti sadržani unutar prijedloga regije. Isto tako i pri upotrebi sažimanja maksimumom na potpodručjima određeni dio informacija biva izgubljen. Iako nam je ovaj pristup dovoljno dobar za klasifikaciju i lokalizaciju objekata, uvođenjem segmentacije u model Mask R-CNN-a svaka nepreciznost i gubitak podataka može imati negativan utjecaj na preciznost segmentacije primjeraka objekata.

RoI Align rješava navedene probleme tako da izbacuje sva zaokruživanja koordinata i područja te smanjuje gubitak informacija korištenjem sažimanja potpodručja. Na slici 5.2 pod b) primjećujemo kako u ovom slučaju odabrano područje točno odgovara onom definiranom koordinatama okvira RPN-a. Upravo zbog toga ovdje nemamo dodatne ili izgubljene podatke što odmah u startu utječe na kvalitetu ovog pristupa. Kod sljedećeg koraka podjele predložene regije interesa na preddefinirane dimenzije željenog izlaza sada tu podjelu odrađujemo potpuno ravnomjerno i dobivamo jednolika potpodručja koja ne moraju odgovarati cjelobrojnoj rešetci aktivacijske mape značajki. Nakon podjele u svakom potpodručju bilinearnom interpolacijom aktivacija određujemo četiri ravnomjerno raspoređene točke unutar tog područja te u konačnici kao rezultat svakog dobivamo jedan broj sažimanjem maksimumom između te četiri

vrijednosti. U ovom postupku, za razliku od običnog sažimanja maksimumom na aktivacijama mape značajki, gubimo puno manje informacija o pojedinim dijelovima prijedloga mreže za predlaganje regija pa tako zadržavamo i puno korisnije podatke za nastavak modela.

Za potrebe glava za klasifikacijsku i predikciju potrebnih pomaka regija interesa koriste se manje regije na izlazu RoI Align sloja (npr. 7x7), dok se za segmentacijsku glavu ipak koriste nešto veće regije (npr. 14x14). Veće regije se koriste jer su nam ipak potrebniji precizniji podaci mapi značajki kako bismo dobro odradili problem segmentaciji koji je dosta teži od same klasifikacije i lokalizacije. U nastavku ćemo detaljnije opisati kako se koriste ove regije interesa te pobliže pojasniti navedene glave nakon kojih dolazimo do finalnih predikcija Mask R-CNN-a.

5.2. Glava za klasifikaciju

Sve što smo dosad spominjali i opisivali temeljilo se samo na dvjema klasama regija interesa RPN-a - objekt ili pozadina. Kao i kod velike većine drugih problema dubokog učenja razreda objekata naravno može biti puno više, a za njihovu klasifikaciju se brine jedna od grana za obradu regija interesa koju nazivamo glavom za klasifikaciju. Mask R-CNN klasificira objekte unutar opisnih okvira u N+1 klasu pri čemu je N broj razreda objekata u skupu podataka, a jednu dodatnu klasu predstavlja pozadina. Uvođenjem dodatne klase pozadine Mask R-CNN ima mogućnost ispraviti netočne prijedloge modula za predlaganje regija interesa RPN, točnije one prijedloge koji ipak ne predstavljaju područje nijednog objekta.

Nakon što za svaku od maksimalnog broja regija interesa poslije primjene NMS-a imamo ekstrahirane i značajke fiksnih dimenzija nakon primjene RoI Align sloja, dolazimo i do same glave za klasifikaciju. Za početak je potrebno značajke regija interesa preoblikovati u jedan vektor koji tada propuštamo kroz dva potpuno povezana sloja od kojih svaki dijeli jednake parametre za različite prijedloge RPN-a. Navedena dva potpuno povezana sloja zajednička su dijeljena među glavama za klasifikaciju i za predviđanje okvira te koriste za dobivanje dodatnih informacija o područjima od interesa.

Specifičnost same glave za klasifikaciju je dodavanje još jednog, među regijama interesa dijeljenog, potpuno povezanog sloja koji se sastoji od N+1 neurona za svaku klasu zadanog skupa podataka uključujući i pozadinu. Primjenom softmax aktivacije na izlazima neurona dobivamo vjerojatnosnu razdiobu pouzdanost klasifikacija za svaki razred objekata.

5.3. Glava za okvire

Sljedeći jednako važan dio Mask R-CNN modela glava je za predviđanje okvira. Izlazi ove grane modela jednaki su izlazima mreže za predlaganje regija RPN te predstavljaju dodatne potrebne pomake (dx, dy, dw, dh) okvira kako bismo dobili još precizniju lokalizaciju objekata na slici.

Konačne predikcije okvira objekata dobivamo propuštanjem značajki regija interesa kroz dva ranije spomenuta potpuno povezana sloja dijeljena s glavom za klasifikaciju te primjenom još jednog dodatnog potpuno povezanog sloja. Ovaj posljednje navedeni sloj također dijeli parametre između svih predloženih područja interesa RPN-a, a broj izlaznih neurona je četiri puta veći od broja klasa objekata čime dobivamo predviđanje svakog od potrebnih pomaka za svaki razred objekta kojeg regija interesa može sadržavati. Predikcije okvira između različitih klasa su potpuno neovisne te kao takve nemaju nikakvog utjecaja jedne na druge. U konačnici ćemo za finalna predviđanja lokacija objekata koristiti pomake okvira samo određene klase kojoj objekt pripada s najvećom vjerojatnošću.

5.4. Glava za segmentaciju primjeraka

Jedan od najbitnijih dijelova modela Mask R-CNN upravo je glava za segmentaciju primjeraka te kao takva predstavlja i jednu od ključnih razlika i nadogradnji u odnosu na model Faster R-CNN-a koji je bio neposredni prethodnik. Korištenjem semantičke segmentacije cilj nam je postići još precizniju lokalizaciju objekata i unutar njihovih okvira. Kako već ionako rješavamo problem lokalizacije objekata, samim time olakšan nam je i zadatak segmentacije primjeraka u odnosu na češći problem klasifikacije piksela u klase neovisno o granicama pojedinih primjeraka objekata istog razreda.

Kao što smo već ranije i spomenuli, semantička segmentacija teži je problem od klasifikacije i lokalizacije objekata pa stoga mape značajki predloženih regija interesa RPN-a svodimo na značajke nešto većih dimnezija (npr. 14x14) nego za potrebe glava za klasifikaciju i predviđanje okvira. U odnosu na prethodno opisane dvije glave, glava za segmentaciju primjeraka sastoji se od nešto više slojeva, koji i u ovom slučaju dijele parametre između svih predloženih područja interesa. Početni korak nad regijama interesa je provođenje niza od nekoliko konvolucijskih slojeva sa nadopunavanjem kako bismo na kraju dobili značajke jednakih dimenzija kao i prije, ali obogaćene dodatnim semantički značajnim informacijama. Nad izlazom ovog niza konvolucija provodimo operaciju transponirane konvolucije koja dvostruko povećava prostorne dimenzije vi-

sine i širine mapa značajki. Još nam za kraj preostaje provesti ove aktivacijske mape kroz jedan dodan konvolucijski sloj kako bismo postigli sličan efekt kao i kod glave za predviđanje okvira, a to je odvajanje segmentacija objekata neovisno za svaki mogući razred objekata. Opisane rezultate postižemo primjenom konvolucijkog sloja s filterima veličine 1x1 te brojem izlaznih mapi značajki koji odgovara broju klasa koje možemo pronaći u korištenom skupu podataka. Aktivacijska funkcija izlaznog sloja ove glave Mask R-CNN-a je sigmoida. Vezano uz konačnu predikciju segmentacijske maske za svaki okvir, vodimo se istom metodom kao kod glave za previđanje pomaka okvira, a to je konačni odabir maske za onu klasu za koju model s najvećom sigurnošću tvrdi da opisuje lokalizirani objekt.

5.5. Pripremanje oznaka

Vrlo važan korak, kao i kod modula za predlaganje regija interesa RPN, obrada je istintih izlaza skupa podataka te njihova prilagodba za potrebe učenja modela Mask R-CNN-a. U ovom slučaju istinite klase i okvire pojedinih objekata ćemo obrađivati na malo drugačji način nego za RPN, a posebnost nalazimo u obradi i modifikaciji istinitih segmentiranih slika.

Istinite razrede učitavamo mapiranjem istinitih oznaka skupa podataka na brojčanu oznaku klasa, a okvire objekata pripremamo u formatu koordinata (x_1, y_1) i (x_2, y_2) koje označavaju gornji lijevi i donji desni vrh pojedinih okvira. Semantičke segmentacije pojedinih objekata uglavnom su zadane na zajedničkoj slici koju moramo obraditi za svako područje pokriveno istinitim okvirima. Nakon što iz tako definirane istinite segmentacijske slike izrežemo područje svakog okvira, dolazimo do situacije u kojoj segmentirane podatke imamo zadane u raznim dimenzijama što ne možemo direktno iskoristiti u postupku učenja modela. Isto tako potrebno je pripaziti i na činjenicu da se unutar okvira mogu nalaziti i dijelovi drugih objekata pa treba pažljivo izdvojiti piksele željenog objekta od pozadine te drugih objekata. Upravo zbog različitih dimenzija visine i širine segmentiranih podučja, kao što smo već i ranije činili u nekoliko situacija, sve istinite oznake za semantičku segmentaciju primjeraka potrebno je svesti na isti oblik (npr. 28x28) prilagođen izlazu glave za predviđanje segmentacijskih maski. Isto tako neovisno o načinu definiranja segmentacijskih maski u samom skupu podataka, za potrebe učenja modela pozicije piksela koji sadrže objekt postavljamo na vrijednost 1, dok sve ostale piksele postavljamo na vrijednost 0.

Iako podatke o istinitim klasama, okvirima i segmentacijskim maskama učitavamo pri generiranju svake sljedeće grupe slike (engl. *batch*), izravno na tim podacima ne

možemo provoditi postupak učenja modela te ih je potrebno dodatno obraditi. Specifičnost ovog pristupa nalazimo u tome što je, za razliku od pripreme željenih izlaza RPN-a, za potrebe Mask R-CNN-a istinite izlaze moguće generirati tek nakon unaprijednog prolaza. Razlog tome je to što su nam za pripremu podataka za funkcije gubitaka i učenje modela potrebni prijedlozi regija interesa RPN-a te informacije o istinitim i predviđenim podacima svih glava Mask R-CNN-a.

Početni korak predstavlja obrada prijedloga modula za predlaganje regija interesa, pri čemu i u ovom slučaju koristimo ideju podjele na pozitivne i negativne okvire temeljenu na mjeri udjela presjeka u ukupnoj površini IoU. Pozitivnim prijedlozima regija interesa smatramo one kojima je mjera IoU s bilo kojim od istinitih okvira veća od preddefinirane granice (npr. 0.5) te za njih smatramo da predstavljaju objekte, dok su negativni prijedlozi okvira oni kojima je mjera presjeka manja od te granice pa za njih smatramo da predstavljaju pozadinu. Za svaki pozitivni prijedlog RPN-a određujemo istiniti okvir objekta s kojim ima najveću IoU mjeru te radimo povezivanje s odgovarajućom istinitom klasom, okvirom i segmentacijskom maskom. Potrebni pomaci okvira prijedloga izračunati su pomoću istih formula kao u slučaju pripremanja željenih izlaza RPN-a, a prikazuju odnos prijedloga RPN-a te istinitih okvira definiranih za pojedinu ulaznu sliku. Negativne okvire ne povezujemo ni s čim, nego za njih definiramo klasu pozadine kao istinitu, dok okvire i segmentacijsku masku ne moramo definirati jer ih ionako ne želimo naučiti.

Kako i u ovom slučaju imamo sličnu situaciju s kojom smo se susreli kod pripremanja željenih izlaza modela RPN-a, a to je prevladavanje negativnih prijedloga, također određujemo veličinu uzorka nad kojim ćemo provoditi postupak učenja. Bitno je napomenuti da je ovdje veći naglasak stavljen na omjer pozitivnih i negativnih okvira te taj omjer koristimo kao prvi kriterij pri definiranju uzorka, što posljedično znači da uzorak može biti i manji od definirane maksimalne veličine. Kroz postupak učenja modela Mask R-CNN-a s vremenom dobivamo sve više pozitivnih prijedloga RPN-a pa se tako i približavamo maksimalnoj popunjenosti uzorka koji postaju sve veći. Ako primjerice koristimo maksimalnu veličinu uzorka od 100 regija interesa te omjer pozitivnih i negativnih prijedloga 1:3, na samom početku učenja modela mogli bismo imati samo 10 pozitivnih regija zbog čega ćemo odabrati još 30 negativnih pa će u konačnici veličina uzorka biti 40 iako je definirana maksimalna veličina od 100. Kako model postaje sve bolji imat ćemo sve više pozitivnih prijedloga RPN-a, potencijalno i veći broj od udjela omjera u maksimalnoj veličini uzorka, pa u tom slučaju zbog definiranog 1:3 omjera odabiremo 25 pozitivnih i 75 negativnih regija interesa neovisno o njihovom stvarnom ukupnom broju.

5.6. Funkcije gubitka

Pored gubitaka modula za predlaganje regija interesa, cjelokupni model Mask R-CNN-a učimo na temelju još tri gubitka. Ti gubitci su: gubitak razreda objekata, gubitak okvira te gubitak semantičke segmentacijske pojedinih primjeraka objekata.

Za klasifikacijski gubitak koristimo gubitak unakrsne entropije. Ovdje se minimazacija gubitka provodi i na pozitivnih i na negativnim prijedlozima u uzorku, pri čemu je u popis klasa skupa podataka dodana i klasa pozadine te nju također učimo na negativne prijedloge RPN-a.

Sljedeća funkcija gubitka definirana je za dodatne pomake (dx, dy, dw, dh) prijedloga modula za predlaganje regija interesa koje koristimo kako bismo još kvalitetnije precizirali i odredili okvire objekata. U ovom slučaju također ćemo iskoristiti prethodno predstavljeni glatki L1 gubitak iz poglavlja 4.3. Također je bitno napomenuti kako minimizaciju ovog gubitka provodimo samo na pozitivnim prijedlozima modula za predlaganje regija interesa. Isto tako u opisu modela navedeno je da se neovisno predviđaju pomaci okvira za svaku od klasa skupa podataka uključujući i klasu pozadine, a pri izračunu funkcije gubitka vrijednost računamo samo na temelju predviđenih pomaka za istinitu klasu objekta.

Posljednja funkcija gubitka našeg modela definirana je za segmentacijske maske pojedinih primjeraka objekata. Kako smo već ranije pri obradi istinitih semantičkih segmentacija objekata odredili vrijednosti 1 za područje objekta te 0 za područje pozadine, tako sada jednostavno možemo primijeniti gubitak binarne unakrsne entropije. Pri minimizaciji ovako definiranih gubitka također koristimo samo pozitivne prijedloge RPN-a, a ujedno i samo predviđene segmentacijske maske koje odgovaraju istinitoj klasi pokrivenog objekta.

5.7. Konačna obrada izlaza modela

U trenutku kada smo proveli unaprijedni prolaz kroz kralježnicu, FPN, RPN i Mask R-CNN te imamo sve prijedloge modula za predlaganje regija interesa, kao i predviđanja svih glava Mask R-CNN-a, te podatke potrebno je još dodatno obraditi kako bismo dobili konačne predikcije koje se sastoje od klase svakog objekta, njegovog okvira i segmentacijske maske.

Iako na izlazu glave za klasifikaciju direktno dobivamo korisnu informaciju za svaki objekt, glave za okvire i segmentacijske maske daju nam redom pomake prijedloga okvira RPN-a te semantičke segmentacije jednakih fiksnih veličina neovisno o

stvarnim dimenzijama objekata. Od pomaka i segmentacija koji su definirani za svaku moguću klasu, odabiremo upravo one koji pripadaju klasi za koju model s najvećom sigurnošću tvrdi da najbolje opisuje objekt. Problem pomaka prijedloga okvira rješavamo korištenjem jednakih formula kao u poglavlju 5.1. Iako su formule jednake, jedina je razlika u tome što ih u ovom slučaju primjenjujemo na prijedloge regija interesa, za razliku od primjene nad sidrima na izlazu RPN-a opisane u navedenom poglavlju. Nakon izračuna konačnih okvira objekata i same segmentacijske maske veličinom možemo prilagoditi dimenzijama okvira.

U konačnici kada imamo sve predviđene klase, okvire i segmentacijske maske možemo primijetiti da broj ovakvih rezultata gotovo odgovara hiperparametru koji određuju maksimalan broj prijedloga regija interesa RPN-a nakon primjene algoritma NMS-a, izuzevši regije klasificirane kao pozadina. Upravo zbog toga potrebno je na neki način filtrirati sve konačne predikcije Mask R-CNN-a kako bismo odredili što točno prikazati kao rezultat lokalizacije i semantičke segmentacije objekata na ulaznoj slici. Odabir rezultata koje ćemo prikazati obavljamo neovisno za svaku klasu, pri čemu klasu pozadine preskačemo. Takvim prolaskom kroz sve moguće klase objekata odabiremo one predikcije okvira i segmentacija primjeraka za koje klasifikacijska mjera pripadnosti toj klasi iznosi više od nekog definiranog graničnog hiperparametra (npr. 0.7), a sve rezultate s manjom pouzdanošću možemo odbaciti. Nad odabranim okvirima i maskama, zbog mogućnosti grupiranja i preklapanja sličnih predikcija, provodimo algoritam NMS za koji također definiramo granicu mjere udjela presjeka IoU na temelju koje izbacujemo najsličnije prijedloge te maksimalan broj predviđenih okvira po svakoj klasi iz skupa podataka.

6. Rezultati

Nakon što smo opisali sve detalje svih podmodela i cjelokupni model Mask R-CNN-a, samu implementaciju svega opisanog potrebno je i validirati te testirati kako bismo vidjeli kvalitetu i uspješnost modela u konkretnoj primjeni.

Mask R-CNN korišten za dobivanje sljedećih rezultata u potpunosti je samostalno implementiran korištenjem NumPy i TensorFlow programskih paketa. Implementacija je dostupna na GitHub repozitoriju https://github.com/brunokovac/Mask-RCNN. Iako je implementirana i sama kralježnica modela ResNet-50 u konačnici je korišten predtreniran Kerasov model zbog ubrzanja samo postupka učenja cjelokupnog modela.

Poseban naglasak moramo staviti na obradu podataka prije pokreatanja postupka učenja modela. Odabir svih kroz rad spomenutih hiperparametara vrlo je bitan za uspješnost rada Mask R-CNN-a, a poseban naglasak bitno je staviti na odabir veličina i omjera okvira modula za predlaganje regija interesa RPN. Ukoliko loše odaberemo navedene veličine broj pozitivnih okvira će biti manji što pri odabiru prevelikog uzorka za učenje rezultira nebalansiranošću klasa gdje klasa pozadine uvjerljivo prevladava, dok odabir manjeg uzorka u kojem bismo se približili 50/50 omjeru pozitivnih i negativnih okvira usporava cijeli postupak učenja modela. Upravo zbog toga potrebno je provesti analizu svih veličina i omjera istinitig okvira u skupu podataka te potom vidjeti za koje konfiguracije hiperparametara postižemo najkvalitetnije rezultate i najveći broj pozitivnih okvira.

Programska izvedba modela Mask R-CNN-a napravljena je tako da se model može koristiti na različitim skupovima podataka jednostavnim promjena konfiguracija, a dobivene rezultate i moguće primjene prikazat ćemo na dva različita skupa podataka.

6.1. AP metrika

Za potrebe određivanja kvalitete uspješnosti modela za klasifikaciju i lokalizaciju objekata uobičajene metrike poput točnosti, F1-mjere i sl. nisu primjenjive. Zbog toga je potrebno pronaći odgovarajuću metriku koja će nam dati dobru sliku o kvaliteti rezul-

tata lokalizacije pojedinih objekata, uzimajući u obzir i preciznost određivanja pozicije kao i samu točnost klasifikacije istih objekata. Za te potrebe najčešće se koristi metrika prosječne preciznosti - AP (engl. *Average Precision*).

U početku definiranja AP metrike potrebno je spomenuti kako ovu metriku određujemo u odnosu na odabranu granicu IoU mjere presjeka koju smo već prethodno deifinirali i pojasnili pa tako imamo npr. AP_{50} (IoU granica je postavljena na 0.5), AP_{75} (IoU granica je postavljena na 0.75) itd. Za određivanje AP metrike također je potrebno definirati i što u ovom slučaju znače pojmovi točnih pozitiva TP (engl. *True Positive*), lažnih pozitiva (engl. *False Positive*) i lažnih negativa (engl. *False Negative*). Točnim pozitivima smatramo sve one okvire klasificirane u točnu klasu objekata koji za odgovarajući istiniti okvir imaju najveću mjeru presjeka IoU, koja je ujedno i veća od preddefinirane granice iz samog naziva konkretne AP metrike. Lažne pozitive predstavljaju svi okviri s IoU vrijednošću u odnosu na istinite okvire manjom od definirane granice, ali i svi oni okviri istog objekta za koji je već s većom klasifikacijskom mjerom pronađen točan pozitiv. Lažne negative definiramo za sve istinite okvire za koje naš model nije predvidio niti jedan okvir, kao i za one s kojima postoji okvir veće mjere presjeka od definirane granice, no netočno predviđene klase obuhvaćenog objekta. Nakon što smo definirali pojmove TP, FP i FN lako možemo izračunati i mjere preciznosti i odziva koje ćemo dalje koristiti.

Nakon što silazno sortiramo sva predviđanja modela za lokacije i klase objekata na temelju njihove klasifikacijske mjere, možemo prolaskom kroz sve takve predikcije izračunati vrijednosti preciznosti i odziva. Ukoliko odlučimo nacrtati dijagram odnosa preciznosti u ovisnosti o odzivu, dobit ćemo graf na kojem možemo uočiti zig-zag uzorak kretanja vrijednosti. Sa svakim lažnim pozitivom graf pada da bi onda opet sa svakim novim točnim pozitivom porastao što u konačnici rezultira navedenim uzorkom. Vrijednost AP metrike računamo kao površinu ispod takve krivulje.

Za potrebe računanja metrike prosječne preciznosti za segmentacijske maske objekata u početku računamo mjeru presjeka IoU temeljenu na samim pikselima istinite maske i one koje ju model predvidio. Za ovakvu definiciju IoU mjere također određujemo granicu na temelju koje ćemo određivati točne i lažne pozitive, a nastavak izračuna preciznosti, odziva i površine ispod krivulje potpuno je isti kao i za okvire.

6.2. PASCAL VOC-2012

Prvi skup podataka na kojemu je testirana implementacija modela Mask R-CNN predstavlja *PASCAL Visual Object Classes* [1] iz 2012. godine, u nastavku teksta spominjan

pod nazivom VOC-2012. Skup podataka VOC-2012 sastoji se od velikog broja slika prosječnih dimenzija oko 500x375, kako položene tako i uspravne orijentacije. Objekti koji se nalaze na tim slikama dolaze u velikom rasponu veličina i okruženja što pridonosi raznolikosti ovog skupa podataka. Sve objekte koje nalazimo na tim slikama možemo podijeliti u dvadeset klasa: avion, bicikl, ptica, brod, boca, autobus, automobil, mačka, stolac, krava, stol, pas, konj, motocikl, osoba, biljka, ovca, kauč, vlak i TV/monitor. Podaci su podijeljeni u nekoliko skupina kao što su *Action, Layout, Main* i *Segmentation* pri čemu je najinteresantnija posljednje navedena skupina te ćemo postupak učenja i testiranja modela temeljiti upravo na njoj.

Podskup *Segmentation* skupa podataka VOC-2012 sastoji se od 1464 slike za učenje te 1449 slika za validaciju, a skup za testiranje koji uključuje i istinite podatke klasifikacije, okvira te semantičke segmentacije nije javno dostupan. Za svaku od navedenih slika klase i okviri lokacije objekata zadani su u datoteka xml formata, dok je semantička segmentacija zadana u obliku novih slika crne pozadine te istaknutih segmentacijskih maski objekata. Kod odabira istinitih segmentiranih slika na izboru nam stoje dvije mogućnosti, *SegmentationClass* te *SegmentationObject* skupovi slika od kojih za učenje segmentacije pojedinih primjeraka objekata odabiremo drugu opciju.

Nakon upoznvanja s podacima proveden je i postupak učenja Mask R-CNN-a nad ovim skupom podataka. Učenje je zaustavljeno nakon što smo deset puta u nizu dobili lošiji rezultat AP_{50} metrike na validacijskom skupu podataka te smo tako nakon 330 epoha, s početnom stopom učenja od 0.004 nad veličinom mini-grupe od 12 slika, dobili konačne rezultate prikazane u tablici 6.1. Od ostalih hiperparametara valja spomenuti veličinu uzorka sidara od 100, pri čemu su definirane granice udjela presjeka IoU redom 0.3 i 0.7 za podjelu na intervale negativnih, neutralnih i pozitivnih sidara. Površine sidara definare su veličinom stranice kvadratnog oblika sidra te iznose 95, 140, 210, 280, 350 redom po razinama FPN-a, dok su korišteni omjeri visine i širine 1:1, 1:2 i 2:1. Veličina uzorka za učenje Mask R-CNN-a iznosi 200 s omjerom pozitivnih i negativnih prijedloga RPN-a 1:3.

	AP_{50} za okvire (%)	AP_{50} za segmentacijske maske (%)
Skup za učenje	97.48	69.63
Skup za validaciju	51.11	39.39

Tablica 6.1: Metrike na PASCAL VOC-2012 skupu podataka

Iz tablice možemo primijetiti kako smo na skupu za učenje postigli jako dobre rezultate po pitanju AP_{50} mjere prosječne preciznosti za okvire, dok po iznosu iste

metrike za segmentacijske maske objekata možemo zaključiti kako je problem označavanja konkretnih piksela objekata ipak podosta teži te su stoga i rezultati nešto slabiji. Razlike između rezultata na skupu za učenje i validaciju nešto su veće nego što smo navikli u mnogim problemima klasifikacije slika, no bitno je uzeti u obzir težinu ovog problema te teže postizanje ujednačenosti ovih dvaju skupova podataka zbog mnogih mogućih oblika, dimenzija, veličina i ostalih karakteristika objekata.

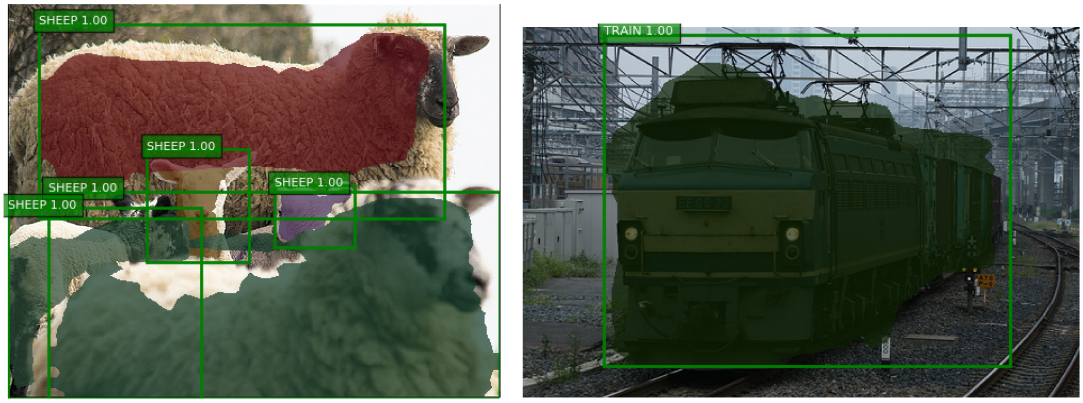
Usporedbu rezultata s drugim radovima u ovom slučaju je malo teže provesti, no možemo ih otprilike usporediti s nekim dostupnim rezultatima. Po pitanju iznosa AP_{50} metrike za okvire, najbližu usporedbu možemo pronaći u Faster R-CNN radu [14] gdje je dobiven rezultat od 0.67 AP_{50} korištenjem kralježnice VGG-16 te modula za predlaganje regija interesa RPN. Isto tako bitno je napomenuti kako je u tom slučaju model treniran na svih preko sedamnaest tisuća slika iz svih podskupova VOC-2012 skupa podataka, a testiranje je provedeno na testnom skupu.

Provođenje usporedbe za mjeru prosječne preciznosti AP_{50} za segmentacijske makse pojedinih objekata još je teže, a potencijalno bi rezultate mogli usporediti koristeći dobivene metrike iz Mask R-CNN rada [7]. Iako je u ovom slučaju model naučen na puno većem skupu podataka COCO nad 135000 slika iz skupa *trainval135k* te testiran na podskupu *minival*, određenu sličnost možemo pronaći u korištenju kralježnice ResNet-50 s rezolucijskom piramidom značajki FPN pri čemu su postignuti rezultati od 0.552 AP_{50}.

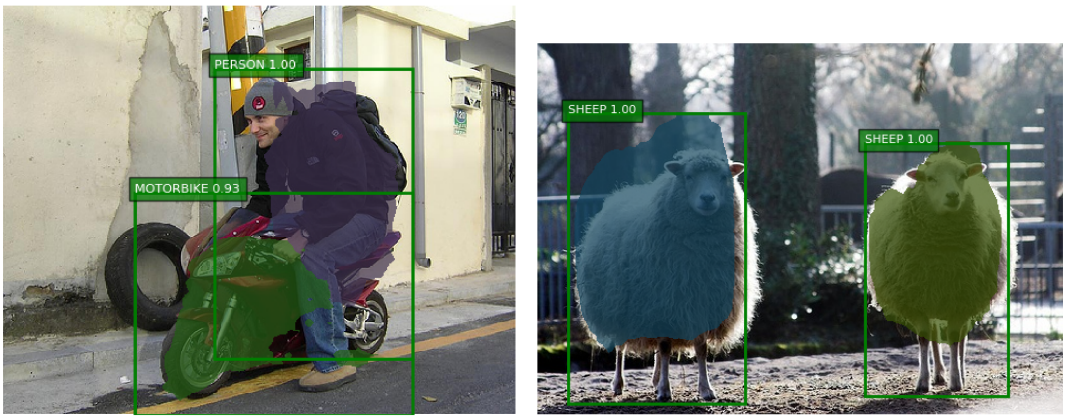
Prikažimo sada i nekoliko konkretnih rezultata lokalizacije i semantičke segmentacije dobivenih na slikama iz VOC-2012 skupa podataka. Na slici 6.1 prikazana su dva rezultata predikcija naučenog modela Mask R-CNN-a nad dvjema slikama iz skupa za učenje. Ovdje je prvo prikazan jedan kompleksniji slučaj fotografije na kojoj se nalazi više objekata te drugi s jednim objektom, a na obje slike možemo primijetiti kako je model vrlo uspješno odradio svoj posao predviđanja klasa, pozicija te segmentacijskih maski svih objekata.

Za usporedbu kvalitete rada Mask R-CNN-a potrebno je testirati njegov rad i na nekoliko slika koje model prethodno nije vidio, tj. na slikama iz skupa za validaciju. Rezultati nad fotografijama iz tog skupa podataka prikazani su na slici 6.2. Unatoč dosta slabijim vrijednostima AP_{50} metrike na validacijskom skupu, možemo primijetiti da su predikcije modela također vrlo dobre. Na prvoj fotografiji uočavamo dva preklapajuća objetka te je usprkos takvoj situaciji klasifikacija, lokalizacija i segmentacija oba objekta provedena vrlo uspješno. Druga slika sastoji se dva odvojena objekta gdje su oba poprilično kvalitetno segmentirana te konačni rezultat izgleda jako dobro.

Iako se najčešće rezultati naučenih modela i njihovog rada prikazuju na skupo-



Slika 6.1: Rezultati na VOC-2012 skupu za učenje

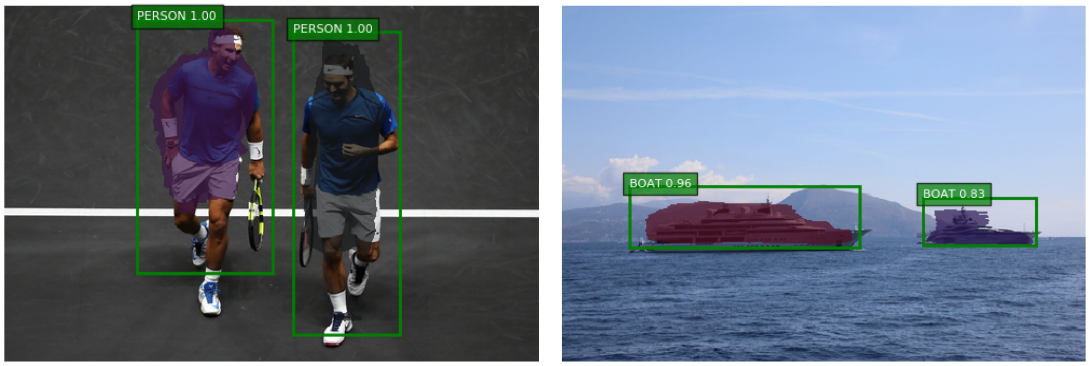


Slika 6.2: Rezultati na VOC-2012 skupu za validaciju

vima za učenje, validaciju i testiranje, u ovom slučaju bilo bi vrlo zanimljivo prikazati dobivene rezultate na nekoliko slika izvan skupa podataka. Upravo time bismo htjeli pokazati svojstvo generalizacija modela Mask R-CNN-a te općenito mogućnost primjene na razne druge fotografije i situacije s kojima se možda nismo susreli u skupu za učenje, a potencijalno niti u ostatku odabranog skupa podataka. Slika 6.3 prikazuje nekoliko rezultata dobivenih na dvije odabrane slike. Na prvoj fotografiji Rogera Federera i Rafaela Nadala [1] naš model vrlo je dobro lokalizirao i segmentirao oba tenisača, dok su na drugoj slici [2] jahte također kvalitetno detektirane.

[1]Fotografija je preuzeta s lokacije https://api.time.com/wp-content/uploads/2017/09/federer-nadal-laver-cup.jpg

[2]Fotografija je preuzeta s lokacije https://yachtharbour.com/static/uploads/1994_ab927.png



Slika 6.3: Rezultati na ostalim slikama

6.3. AOLP

Sljedeći skup podataka koji smo koristili nazivamo AOLP, tj. *Application Oriented License Plate*. Ovaj skup podataka preuzet je sa službenih stranica uz dobivenu dozvolu autora [8]. Fotografije automobila te njihovih registarskih tablica obuhvaćaju veliki raspon različitih vremenskih uvjeta, doba dana, unutarnjeg ili vanjskog prostora, kao i različitih kuteva pod kojim su fotografije snimljene. Ovaj skup podataka sastoji se od 2409 takvih slika podijeljenih u tri podskupa, a sve fotografije prikupljene su na Tajvanu na raznim lokacijama i iz različitih područja primjene.

Podskupovi podataka označeni su kraticama AC (engl. *Access Control*), LE (engl. *Law Enforcement*) te RP (engl. *Road Patrol*) koje označavaju različita područja primjene u kojima su navedene fotografije skupljene. Ovi manji skupovi slika redom se sastoje od 681, 757 i 611 slika koje su za potrebe učenja modela skupljene u jedan veći skup podataka te tako zajedno korištene. AP i RP podskupovi sastoje se od slika dimenzija 320x240, dok su fotografije u LE podskupu nešto veće i uglavnom dimenzija 640x480. Istiniti okviri zadani u u obliku tekstualnih datoteka koje sadrže gornji lijevi i donji desni vrh registarskih tablica, pri čemu postoji određeni broj krivo označeni podataka ili podataka gdje su koordinate donjeg desnog vrha manje od onih gornjeg lijevog na što je potrebno posebno pripaziti pri obradi podataka. Segmentacijske maske u ovom skupu podataka nisu definirane pa su stoga definirane za cijelo područje okvira. Nakon skupljanja svih slika u jednu skupinu, obavljenja je i podjela na skupove za učenje i validaciju te je pokrenut i sam postupak učenja modela.

Pored samog učenja modela Mask R-CNN-a na ovom skupu podataka, cilj je bio provjeriti i konfigurabilnost same implementacije kako bismo se čim brže prilagodili svakom novom skupu podataka. Za potrebe ovog skupa podataka napravljeno je novo učitavanje i obrada istinitih podataka, promijenjene su veličine i omjeri visine i širine

prilagođeni izduženosti tablica kako bismo okvire RPN-a bolje prilagodili prosječnim dimenzijama registarskih tablica. Nakon provedenog postupka učenja sa stopom od 0.0007 te poslije 70 epoha, dobiveni su sljedeći rezultati prikazani u sljedećoj tablici 6.2. Od ostalih hiperparametara valja spomenuti veličinu uzorka sidara od 50, pri čemu su definirane granice udjela presjeka IoU redom 0.3 i 0.7 za podjelu na intervale negativnih, neutralnih i pozitivnih sidara. Površine sidara definare su veličinom stranice kvadratnog oblika sidra te 50 za svaku razinu FPN-a, dok su korišteni omjeri visine i širine u ovom slučaju 2:1, 2.5:1 i 3:1 kako bismo se bolje prilagodili obliku registarskih tablica. Veličina uzorka za učenje Mask R-CNN-a iznosi 100 s omjerom pozitivnih i negativnih prijedloga RPN-a 1:3.

	AP_{50} za okvire (%)	AP_{50} za segmentacijske maske (%)
Skup za učenje	98.32	97.69
Skup za validaciju	98.01	97.26

Tablica 6.2: Metrike na AOLP skupu podataka

Prema rezultatima metrike prosječne preciznost AP_{50} iz tablice možemo primijetiti kako je ovaj problem lokalizacije tablica ipak bio puno jednostavniji te su stoga i iznosi metrika puno bolji. Vrijednosti AP_{50} metrika za segmentacijske maske također su vrlo zadovoljavajuće te su sve u svemu rezultati vrlo dobri. Možemo primijetiti i kako su iznosi metrika gotovo jednaki na skupovima za učenje i validaciju što također mnogo govori o kvaliteti rezultata, ali i većoj jednostavnosti samog problema lokalizacije i segmentacije.

Pogledajmo i nekoliko rezultata prikazanih na fotografijama iz AOLP skupa podataka. Konačni cilj ovog zadatka bio je detektirati i zamagliti registarske tablice radi zaštite privatnosti. Na slici 6.4 prikazana su dva rezultata lokalizacije i zamagljivanja tablica nad slikama iz skupa za učenje. Registarske tablice su vrlo dobro i precizno uočene te su sakriveni konkretni podaci o njihovima vrijednostima.

Usporedimo sada i ove rezultate s onima na slikama koje model Mask R-CNN-a nije vidio u postupku učenja. Dvije takve fotografije na kojima je proveden postupak lokalizacije i segmentacije registarskih tablica prikazane su na slici 6.5. U slučaju AOLP skupa podataka možemo uočiti kako su rezultati identične kvalitete kao i kod skupa za učenje te naučeni model vrlo dobro radi.

Sada kada smo pokazali sve rezultate na oba podskupa podataka iz samog AOLP dataseta, zanimljivo bi bilo provjeriti i mogućnost generalizacije samo modela. Upravo zbog toga nas zanima kakvu kvalitetu lokalizacije i zamagljivanja registarskih tablica

Slika 6.4: Rezultati na AOLP skupu za učenje

Slika 6.5: Rezultati na AOLP skupu za validaciju

možemo postići nad fotografijama koje se poprilično razlikuju od onih iz skupa podataka. Za potrebe takvog testa uslikane su dvije fotografije automobila s hrvatskim registarskim tablicama te je proveden postupak segmentacije i zamagljivanja uočenih tablica. Iako se hrvatske tablice dosta razlikuju od onih s Tajvana, možemo primijetiti da naučeni model Mask R-CNN-a vrlo dobro generalizira te samim time i uočava i ovakve registarske tablice.

Slika 6.6: Rezultati na ostalim slikama

7. Zaključak

U ovom radu detaljno je opisan model Mask R-CNN-a koji predstavlja stanje tehnike dubokih modela za klasifikaciju, lokalizaciju i segmentaciju primjeraka objekata. Pojašnjavajući sve njegove komponentne dotaknuli smo se i rezidualnih neuronskih mreža koje koristimo kao kralježnicu cjelokupnog modela. Sljedeće je opisana i rezolucijska piramida značajki FPN kao jedna ideja organizacije konvolucijskih neuronskih mreža često korištena kod raznih problema lokalizacije i semantičke segmentacije zbog svojih specifičnih svojstava kojima zadržavamo i prostorno i semantički značajne informacije o samim ulaznim podacima. Kao značajan dio modela Mask R-CNN-a prikazan je i modul za predlaganje regija interesa RPN. Ovaj modul zamijenio je prethodno korištene algoritme selektivnog pretraživanja te tako omogućio učenje početnih informacija o pozicijama objekata i klasifikaciji istih od pozadine čime sam modul Mask R-CNN-a u početku dobiva kvalitetnije prijedloge područja na kojima se potencijalno nalaze objekti koje mora klasificirati, lokalizirati i segmentirati. U konačnici je opisan i modul specifičan za sam Mask R-CNN koji se sastoji od glava za klasifikaciju, predviđanje okvira te segmentaciju primjeraka objekata.

Pri samom kraju rada prikazani su i rezultati dobiveni korištenjem naše implementacije Mask R-CNN-a na dva dosta različita skupa podataka za dva različita područja primjene. Prvi skup podataka koji smo obradili bio je PASCAL VOC iz 2012. godine koji se sastoji od dvadeset klasa objekata koji su na slikama zastupljeni u velikom rasponu veličina i oblika. Nad ovim skupom podataka postignuti su solidni rezultati AP_{50} metrike kako za okvire tako i za segmentacijske maske primjeraka objekata. Izdvojeno je i nekoliko slika iz skupa podataka, ali i izvan njega, gdje možemo primijetiti kako naučeni model vrlo dobro prepoznaje objekte i označava njihove piksele. Poboljšanje rezultata i metrika prosječne preciznosti mogli bismo postići rastresanjem dostupnih podataka i poboljšanjem resursa na kojima bismo tada brže mogli naučiti model Mask R-CNN-a nad većim brojem podataka.

Sljedeći skup podataka kojeg smo obradili nazivamo AOLP. Ovaj skup podataka sastoji se od velikog broja fotografija automobila na kojima su istintim okvirima oz-

načene registarske tablice. Upravo zbog toga AOLP smo iskoristili za problem zaštite privatnosti pa smo tako učili pozicije tablica kako bismo ih u konačnici mogli kvalitetno zamagliti i sakriti konkretne podatke. Nakon učenja modela evaluacijom rada modela postignute su vrlo kvalitetne vrijednosti metrike prosječne preciznosti AP_{50}, a i sami vizualni prikaz lokalizacije tablica na fotografijama jako je dobar.

Sve u svemu, kroz ovaj rad detaljno smo se upoznali s modelom i primjenama Mask R-CNN-a koji se pokazao kao vrlo kvalitetan, ali i zahtjevan, model za postizanje odličnih rezultata u području lokalizacije i segmentacije primjeraka objekata. Unatoč solidnim postignutim rezultatima, prostora za napredovanje još uvijek ima te je moguće iste poboljšati korištenjem većeg broja slika, jačih resursa, dužeg vremena učenja i sl.

LITERATURA

[1] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, i A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, Lipanj 2010.

[2] Ross B. Girshick. Fast R-CNN. *CoRR*, abs/1504.08083, 2015. URL `http://arxiv.org/abs/1504.08083`.

[3] Ross B. Girshick, Jeff Donahue, Trevor Darrell, i Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *CoRR*, abs/1311.2524, 2013. URL `http://arxiv.org/abs/1311.2524`.

[4] Ian Goodfellow, Yoshua Bengio, i Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, i Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. URL `http://arxiv.org/abs/1512.03385`.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, i Jian Sun. Identity mappings in deep residual networks. *CoRR*, abs/1603.05027, 2016. URL `http://arxiv.org/abs/1603.05027`.

[7] Kaiming He, Georgia Gkioxari, Piotr Dollár, i Ross B. Girshick. Mask R-CNN. *CoRR*, abs/1703.06870, 2017. URL `http://arxiv.org/abs/1703.06870`.

[8] G. Hsu, J. Chen, i Y. Chung. Application-oriented license plate recognition. *IEEE Transactions on Vehicular Technology*, 62(2):552–561, 2013.

[9] Sergey Ioffe i Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015. URL `http://arxiv.org/abs/1502.03167`.

[10] Ivan Kreso, Sinisa Segvic, i Josip Krapac. Ladder-style DenseNets for semantic segmentation of large natural images. U *Proceedings of the IEEE International Conference on Computer Vision Workshops*, stranice 238–245, 2017.

[11] Ivan Krešo, Josip Krapac, i Siniša Šegvić. Efficient ladder-style DenseNets for semantic segmentation of large images. *IEEE Transactions on Intelligent Transportatoin Systems*, 2020.

[12] Hao Li, Zheng Xu, Gavin Taylor, i Tom Goldstein. Visualizing the loss landscape of neural nets. *CoRR*, abs/1712.09913, 2017. URL `http://arxiv.org/abs/1712.09913`.

[13] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, i Serge J. Belongie. Feature pyramid networks for object detection. *CoRR*, abs/1612.03144, 2016. URL `http://arxiv.org/abs/1612.03144`.

[14] Shaoqing Ren, Kaiming He, Ross B. Girshick, i Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *CoRR*, abs/1506.01497, 2015. URL `http://arxiv.org/abs/1506.01497`.

[15] Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, i Aleksander Madry. How does batch normalization help optimization?, 2018.

POPIS SLIKA

POPIS TABLICA

Konvolucijski modeli za semantičku segmentaciju primjeraka

Sažetak

Lokalizacija i segmentacija primjeraka objekata jedni su od bitnih problema dubokog učenja. Mask R-CNN predstavlja stanje tehnike modela koji rješavaju te zadatke. U sklopu rada pojašnjene su sve komponente Mask R-CNN-a poput kralježnice u vidu rezidualnih neuronskih mreža ResNet, rezolucijske piramide značajki FPN, modula za predlaganje regija interesa RPN te samih glava za klasifikaciju, okvire i semantičku segmentaciju specifičnih za cijeli model. Model Mask R-CNN-a implementiran je i naučen na dva različita skupa podataka PASCAL VOC-2012 i AOLP. Prikazane su dobiveni vrijednosti AP metrika te sami rezultati na slikama gdje su objekti klasificirani, lokalizirani i segmentirani. Za oba skupa podataka i područja primjene također su prikazani i rezultati na slikama izvan navedenih skupova kako bismo prikazali mogućnost generalizacije našeg modela.

Ključne riječi: Konvolucijske neuronske mreže, ResNet, rezolucijska piramida značajki, FPN, modul za predlaganje regija interesa, RPN, Mask R-CNN, klasifikacija, lokalizacija, detekcija, segmentacija primjeraka, VOC-2012, AOLP

Convolutional models for semantic segmentation of object instances

Abstract

Object detection and instance segmentation are one of the most important problems in deep learning. Mask R-CNN represents state-of-the-art model which solves these tasks. As part of the paper, all components of Mask R-CNN are explained, such as backbone in form of residual neural networks ResNet, feature pyramid network FPN, region proposal network RPN and classification, bounding box and semantic segmentation heads specific for the whole model. Mask R-CNN model was implemented and trained on two different datasets PASCAL VOC-2012 and AOLP. Values of AP metrics are shown and also results on images where objects are classified, localized and segmented. For both datasets and areas of application results are also shown on images not included in mentioned datasets to show generalization of our model.

Keywords: Convolutional neural networks, ResNet, Feature Pyramid Network, FPN, Region Proposal Network, RPN, Mask R-CNN, classification, localization, detection, instance segmentation, VOC-2012, AOLP