SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

SEMINAR

Duboka arhitektura za jednoprolaznu lokalizaciju objekata

Ivan Grubišić
Voditelj: Siniša Šegvić

Zagreb, prosinac 2017.

Sadržaj

1. Uvod

Važna skupina problema kojima se bavi računalni vid je razumijevanje slike. Ono najčešeće uključuje klasifikaciju (prepoznavanje) - pridjeljivanje slike odgovarajućem semantičkom razredu (kategoriji ili klasi). Česta poopćenja problema klasifikacije slike su semantička segmentacija i pronalaženje [1] objekata. Semantička segmentacija je klasificiranje svakog dijela (piksela) slike. Pronalaženje objekata se sastoji od pronalaženja okvira od kojih svaki uokviruje jedan objekt i klasifikaciju svakog od njih. Postoji i pristup koji ujedinjuje navedena dva - semantička segmentacija uz razlikovanje objekata.

U posljednjih nekoliko godina postiže se značajan napredak u računalnom vidu. On se uglavnom temelji na razvoju dubokih konvolucijskih mreža u kojima glavnu ulogu imaju konvolucijski slojevi. Istovrsni objekti mogu se na slici pojavljivati na različitim položajima na slici. Konvolucijske arhitekture neuronskih mreža iskorištavaju to svojstvo tako da se isti neuron unutar nekog sloja može učiti i koristiti na svim položajima ulazne slike (ili sloja).

U poglavlju 2 opisani su osnovni pojmovi vezani uz konvolucijske mreže s naglaskom na konvolucijski sloj. To poglavlje je zasnovano na razradi te teme u završnom radu [5]. U poglavlju 3 opisan je problem pronalaženja objekata na slici i dan je pregled glavnih pristupa rješavanju tog problema. U poglavlju 4 opisan je postupak SSD [8]. U poglavlju 5 pokazani su primjeri rezultata sustava SSD i usporedba rezultata s drugim modelima. U poglavlju 6 je zaključak.

[1] U literaturi na engleskom jeziku koriste se izrazi detection i localization. I jedan i drugi često označavaju različite probleme pronalaženja objekata - ponekad objekti pripadaju jednom razredu, ponekad se traži samo jedan objekt, a ponekad se traži i klasificira veći broj objekata, od čega je ovo zadnje tema ovog seminara.

2. Konvolucijske mreže

Konvolucijske mreže su unaprijedne neuronske mreže koje se sastoje od većeg broja slojeva prilagođenih obradi signala koji u različitim položajima mogu sadržavati slične uzorke. One omogućuje efikasno učenje i obradu i posebno se uspješno primjenjuju u računalnom vidu.

Kod općenite višeslojne unaprijedne neuronske mreže svaki skriveni sloj sastoji se od neurona koji su potpuno povezani s neuronima prethodnog sloja i od kojih svaki ima težine neovisne o težinama drugih neurona. Takva bi mreža za ulaze većih dimenzija poput slika imala prevelik broj parametara, teško bi učila i bila bi sklona prenaučenosti.

Kod konvolucijskih mreža slojevi su rijetko i lokalno povezani i koriste dijeljenje parametara. Svaki neuron povezan je s malim brojem bliskih neurona prethodnog sloja. Osnovna građevna jedinica konvolucijske mreže je konvolucijski sloj. Njega čini skup naučenih filtara od kojih svaki stvara jednu matricu izlaznog tenzora značajki. Te matrice se nazivaju mapama značajki. Filtri se još nazivaju konvolucijskim jezgrama i općenito su trodimenzionalni. Imaju dvije prostorne dimenzija i jednu koja se općenito proteže kroz sve mape značajki ulaznog tenzora. Pomicanjem filtra po prostornim dimenzijama generira se jedna od izlaznih mapa značajki.

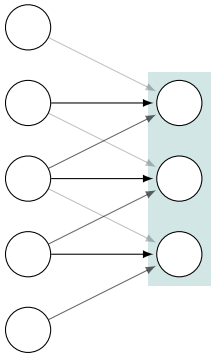
Za klasifikaciju se na kraju mreže obično koristi jedan ili više potpuno povezanih slojeva ili konvolucijski sloj i globalno sažimanje aritmetičkom sredinom. Izlaz takvog konvolucijskog sloja sadrži jednu mapu značajki za svaki razred. Globalno sažimanje aritmetičkom sredinom svaku mapu značajki preslikava u njenu aritmetičku sredinu, čime se dobiva jedna vrijednost za svaki razred. Unutar mreže se nalaze posebni konvolucijski slojevi ili slojevi sažimanja koji služe smanjenju prostornih dimenzija tenzora značajki (među ostalim mogućim ulogama).

Parametri konvolucijskih mreža uče se, slično kao kod klasičnih neuronskih mreža, algoritmom propagacije pogreške unatrag (engl. *backpropagation*) koji se temelji na

pravilu derivacije složene funkcije i dinamičkom programiranju.

2.1. Konvolucijski slojevi

Svojstva konvolucijskih slojeva temelje se na pretpostavci da se istovrsni uzorci mogu pojavljivati na svim položajima u ulazu. To se ostvaruje dijeljenjem parametara između neurona unutar nekog sloja. Djelovanje takvog sloja može se predstaviti konvolucijama dijeljenih težina s ulaznim tenzorom, dodavanjem pragova i primjenom prijenosne funkcije. Na slici 2.1 prikazan je jedan jednostavan konvolucijski sloj.



Slika 2.1: Primjer jednostavnog jednodimenzionalnog konvolucijskog sloja s jednom ulaznom i jednom izlaznom mapom značajki. Svaka izlazna vrijednost se računa na temelju vrijednosti oko istog položaja pomnoženih težinama i dodanim pragom. Težine i prag su zajednički svim neuronima. Iste težine su naznačene jednako tamnim strelicama.

Kod dvodimenzionalnih konvolucijskih slojeva kakvi se koriste u računalnom vidu težine se mogu predstaviti matricama koje se filtrima ili konvolucijskim jezgrama (engl. *kernel*). Jedna izlazna mapa značajki se dobiva konvolucijom pripadajuće konvolucijke jezgre s mapama značajki ulaznog tenzora po prostornim dimenzijama, zbrajanjem i primjenom prijenosne funkcije. Dalje će se n-torka dvodimenzionalnih jezgri, gdje je n općenito broj ulaznih mapa značajki, samo zvati jezgrom. Koriste se jezgre malih prostornih dimenzija, često 1×1, 3×3 ili 5×5.

Konvolucija se provodi pomicanjem jezgre po prostornim dimenzijama (visina i širina) prethodnog sloja i zbrajanjem praga i umnožaka elemenata jezgre s odgovarajućim vrijednostima mapa značajki prethodnog sloja. Na rezultate konvolucije s više različitih jezgri primjenjuje se prijenosna funkcija i dobivaju se konačne izlazne mape značajki. Svaka mapa značajki predstavlja ulaz filtriran na

drugačiji način, s prepoznatim drugačijim značajkama. To se može izraziti ovako:

$$\boldsymbol{H}_{lp} = \varphi\left(-\theta_{lp} + \sum_{q} \boldsymbol{w}_{lpq} * \boldsymbol{H}_{(l-1)q}\right). \tag{2.1}$$

Ovdje $\boldsymbol{H}_{lp}$ označava p-tu izlaznu mapu značajki sloja l, $\boldsymbol{H}_{(l-1)q}$ ulazne mape značajki s rednim brojevima q, θ_{lp} odgovarajući prag, a $\boldsymbol{w}_{lpq}$ q-tu komponentu odgovarajuće jezgre pridruženu q-toj mapi značajki sloja $l-1$. φ predstavlja prijenosnu funkciju, a $*$ označava operator konvolucije koji može biti definiran ovako:

$$(\boldsymbol{h} * \boldsymbol{x})_{mn} = \sum_{i} \sum_{j} h_{ij} x_{(m-i)(n-j)}, \tag{2.2}$$

gdje su $\boldsymbol{h}$ i $\boldsymbol{x}$ matrice (s pomaknutim koordinatama) ili funkcije $\mathbb{Z} \times \mathbb{Z} \to \mathbb{R}$.

Na slici 2.2 prikazan je primjer konvolucije jedne mape značajki i dvodimenzionalne jezgre koja prepoznaje vertikalne rubove. U slučaju da se ulaz sastoji od više komponenata, izlazna matrica jednaka je zbroju rezultata konvolucija svih komponenata s odgovarajućim komponentama konvolucijske jezgre.

$$\begin{bmatrix} 0 & 1 & 1 & 2 \\ 0 & 2 & 2 & 1 \\ 1 & 2 & 2 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} * \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix} \longrightarrow \begin{bmatrix} 3 & 0 & 0 \\ 3 & 0 & -3 \\ 1 & 1 & -3 \end{bmatrix}$$

Slika 2.2: Primjer jednostavne dvodimenzionalne diskretne konvolucije (jedna jezgra dimenzija $2 \times 2 \times 1$) kojom se stvara jedna mapa značajki na temelju jedne ulazne mape značajki

Kao prijenosna funkcija kod konvolucijskih slojeva najčešće se koristi zglobnica (engl. *Rectified linear unit*, *ReLU*) definirana ovako:

$$\varphi(x) = \max(0, x). \tag{2.3}$$

Kako bi izlazne mape značajki bile jednake širine i visine kao ulaz, potrebno je nekako nadopuniti ulaz na mjestima na kojima konvolucijska jezgra prelazi njegove rubove. Rubovi ulaza se najčešće nadopunjuju nulama. Ponekad se jezgra umjesto jedan po jedan piksel pomiče po više piksela odjednom, čime se smanjuju dimenzije izlaza. Taj pomak nazivamo izlazni korak (engl. *stride*).

2.2. Smanjivanje dimenzija slojeva

U konvolucijskim mrežama je važno smanjiti broj parametara i veličine slojeva kako bi se smanjili memorijski zahtjevi modela i ubrzalo izvođenje. To se ostvaruje smanjivanjem dimenzija slojeva i korištenjem manjih konvolucijskih filtara. Za smanjivanje dimenzija slojeva (mapa značajki) donedavno su se uglavnom koristili slojevi sažimanja. Najčešće vrste sažimanja su sažimanje maksimalnom vrijednošću (engl. *max-pooling*) i sažimanje srednjom vrijednošću (engl. *average-pooling*).

Slojevi sažimanja najčešće ostvaruju sažimanje maksimalnom vrijednošću ili aritmetičkom sredinom. Jednostavan primjer sažimanja maksimalnom vrijednošću kojim se dimenzije ulazne mape značajki smanjuju na pola prikazan je na slici 2.3. Dimenzije tih četverokuta su najčešće 2×2, čime se širina i visina ulaza dvostruko smanjuju. Sažimanje se provodi na svakoj mapi značajki nezavisno pa je broj mapa značajki izlaznog tenzora jednak broju mapa značajki ulaznog tenzora.

$$\begin{bmatrix} 1 & 6 & 1 & 2 \\ 0 & 3 & 1 & 0 \\ 2 & 3 & 2 & 0 \\ 2 & 2 & 4 & 1 \end{bmatrix} \longrightarrow \left[\begin{array}{cc|cc} 1 & 6 & 1 & 2 \\ 0 & 3 & 1 & 0 \\ \hline 2 & 3 & 2 & 0 \\ 2 & 2 & 4 & 1 \end{array}\right] \longrightarrow \begin{bmatrix} 6 & 2 \\ 3 & 4 \end{bmatrix}$$

Slika 2.3: Sažimanje maksimalnom vrijednošću

U novijim uspješnim modelima [6], a posebno u nenadziranim generativnim modelima [11], često se umjesto navedenih slojeva sažimanja koriste konvolucijski slojevi s izlaznim korakom većim od 1. Ako je izlazni korak konvolucije n, konvolucijski filtri se, umjesto na svakom položaju ulaznih mapa značajki, postavljaju na svaki n-ti položaj u pojedinoj prostornoj dimenziji. Time se postiže da se dimenzije izlaznih mape značajki uz izlazni korak n smanjuju na $1/n$ odgovarajućih dimenzija ulaza.

3. Lokalizacija objekata

Jedan od osnovnih problema računalnog vida je klasifikacija slika, tj. prepoznavanje vrste (razreda) objekta koji se nalazi na slici. Skup podataka za učenje klasifikacije slika sastoji se od slika i oznaka razreda od kojih je svaka pridružena jednoj slici. Izlaz sustava za klasifikaciju je obično vektor koji predstavlja razdiobu vjerojatnosti pripadanja primjera svakom od razmatranih razreda, tj pouzdanosti (engl. *confidence*) klasifikacije.

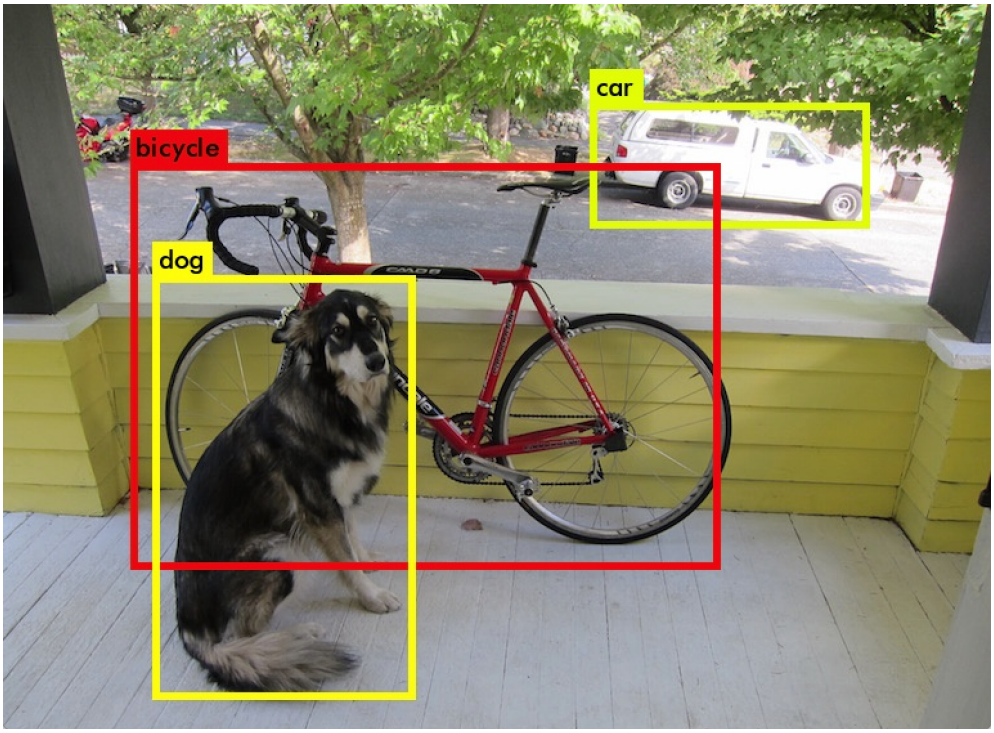
Kod pronalaženja objekata zadatak je prepoznati svaki objekt na slici koji pripada poznatom razredu i pridružiti mu oznaku njegovog razreda i najmanji opisani pravokutnik ili okvir (engl. *minimum bounding box*, *oriented bounding box*) unutar kojeg se nalazi. Skup podataka za učenje postupaka za pronalaženje objekata se sastoji od slika i svakoj od njih pridruženog skupa okvira s odgovarajućim oznakama razreda[1]. Na slici 3.1 prikazan je primjer rezultata sustava za pronalaženje objekata na jednoj slici.

3.1. Osnovni pristupi

Uz riješen problem klasifikacije, najjednostavniji pristup pronalaženju objekata je korištenje pomičnih okana različitih dimenzija i klasificiranje svake na taj način dobivene *podslike*. Na taj način se za neki objekt na slici dobije velik broj okvira unutar kojih se taj objekt prepoznaje s određenom razinom pouzdanosti. Nakon toga potrebno je za svaki objekt pronaći jedan okvir koji ga najbolje opisuje.

Do otprilike 2012. godine najuspješniji su bili modeli koji objekte modeliraju dijelovima koji mogu biti različito raspoređeni (engl. *deformable part model*, *DPM*) [3] i pri tome kao značajke koriste histograme gradijenata (engl. *histogram*

[1] Postoji i jednostavniji problem pronalaženja samo jedne vrste objekata pa u tom slučaju oznake razreda ne bi bile potrebne.



Slika 3.1: Primjer slike iz [12] s objektima kojima su pridruženi okviri.

of gradients, *HoG*).

Malo kasnije su se pokazali uspješnima postupci koji na neki način smanjuju broj okvira koje sustav treba potpuno klasificirati. Na temelju jednostavnijih značajki koje se mogu brže izračunati moguće je odabrati kandidate okvira koji s većom vjerojatnošću opisuju objekte. To se naziva predlaganje okvira (engl. *bounding box proposals*) i time se smanjuje ukupan broj okvira koje treba potpuno klasificirati. Neki primjeri postupaka predlaganja okvira se temelje na segmentaciji slike superpikselima (*Selective search* [16]), korištenju rubova (*Edge boxes* [18]) ili konvolucijskoj mreži (*region proposal network*, *RPN* [14]) koja se može brže izvršavati na grafičkim procesorima. Smanjenje broja okvira koje treba klasificirati omogućuje korištenje jačih modela za učenje značajki i klasifikaciju uz isto ili kraće vrijeme izvođenja.

3.2. Evaluacijske mjere

3.2.1. Jaccardov koeficijent sličnosti

Kod pronalaženja objekata izlaz sustava je skup (klasificiranih) okvira. Mjera koja se koristi kao mjera kvalitete lokalizacije je omjer površine presjeka i unije (IoU, engl. *intersection over union*). Omjer presjeka i unije se još naziva Jaccardov koeficijent sličnosti i označava se s J. Vrijednost koeficijenta sličnosti je 1 ako i samo ako su okviri koje uspoređuje jednaki. Neka su A i B dva okvira koje smatramo skupovima piksela. Onda je koeficijent sličnosti definiran ovako:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}. \tag{3.1}$$

3.2.2. Preciznost, odziv i srednja prosječna preciznost

Preciznost (P, engl. *precision*) se koristi kao mjera relevantnosti rezultata i jednaka je udjelu relevantnih rezultata u svim rezultatima, a odziv (R, engl. *recall*) je udio relevantnih rezultata u svim relevantnim objektima.

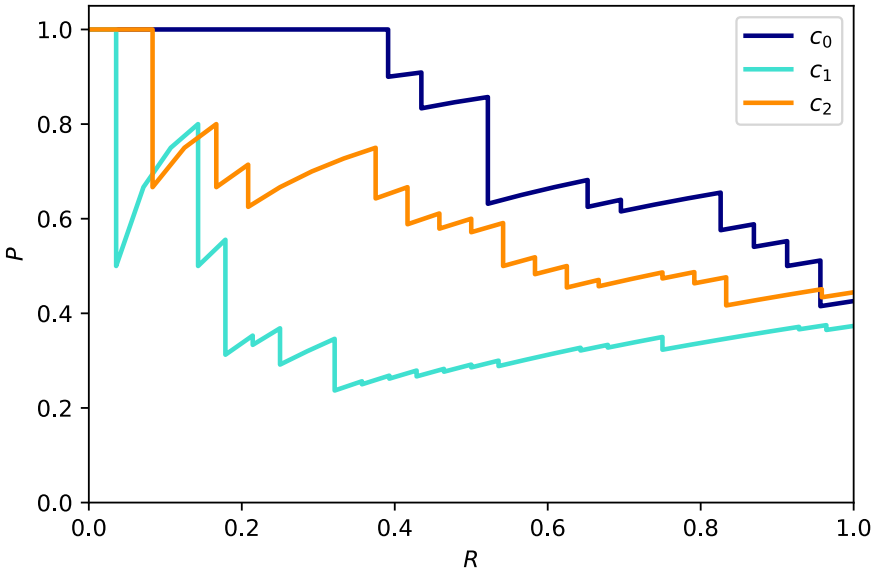
Ako tp (engl. *true positives*) predstavlja broj prepoznatih relevantnih objekata, fp (engl. *false positives*) broj nerelevantnih objekata klasificiranih pod relevantne, a fn (engl. *false negatives*) broj neprepoznatih relevantnih objekata, onda se preciznost i odziv definiraju ovako:

$$P = \frac{tp}{tp + fp}, \tag{3.2}$$

$$R = \frac{tp}{tp + fn}. \tag{3.3}$$

Neka je H skup okvira rezultata, a T skup ciljnih okvira. Bez smanjenja općenitosti, neka svi okviri imaju istu oznaku razreda. Neka je $h \in H$ promatrani okvir rezultat. Prema [2], on se smatra ispravnim (*true positive*) ako postoji $t \in T$ tako da vrijedi $J(t, h) > 0.5$ i $\forall h' \in H\, J(t, h) \geq J(t, h')$. Inače se smatra neispravnim (*false negative*).

Uz položaje i dimenzije okvira, sustav za pronalaženje objekata obično daje i mjeru pouzdanosti klasifikacije okvira. Krenuvši redom od okvira za koji je sustav najsigurniji, uzimanjem u obzir sve više okvira spuštanjem praga pouzdanosti klasifikacije smanjuje se preciznost, a povećava odziv. To se može grafički prikazati krivuljom preciznosti i odziva (slika 3.2).



Slika 3.2: Primjer krivulja preciznosti i odziva za 3 razreda. Slika je preuzeta iz [1] i prilagođena.

Kao evaluacijska mjera za pronalaženja objekata s jednim razredom se najčešće koristi prosječna preciznost (engl. *average precision*), a u slučaju više razreda srednja prosječna preciznost (engl. *mean average precision*). Preciznije, češće se koristi preciznost interpolirana na monotono padajuću krivulju s obzorom na odziv, što je malo dalje opisano.

Prosječna preciznost je često definirana kao površina ispod krivulje preciznosti i odziva:

$$AP = \int_0^1 P(r)\,\mathrm{d}r, \tag{3.4}$$

gdje r predstavlja odzive. Postoje i drugačije definicije [17].

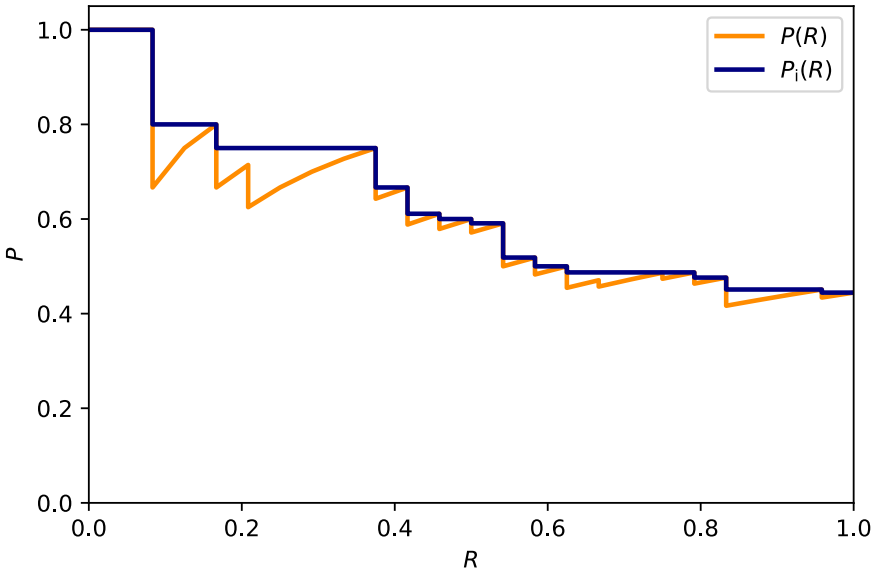
Srednja prosječna preciznost je, uz skup razreda C, definirana ovako:

$$mAP = \frac{1}{|C|} \sum_{c \in C} AP_c. \tag{3.5}$$

Češće se ista oznaka mAP koristi za srednju prosječnu interpoliranu preciznost, tj. umjesto preciznosti se koristi preciznost interpolirana na monotono padajuću krivulju (s obzirom na odziv). Neka P_i označava interpoliranu preciznost, a r odzive. Interpolirana preciznost je definirana ovako:

$$P_\mathrm{i}(r) := \max\{P(r') \mid r' \geq r\}. \tag{3.6}$$

Na slici 3.3 je prikazana krivulja interpolirane preciznosti u ovisnosti o odzivu.



Slika 3.3: Primjer krivulje preciznosti u ovisnosti o odzivu (narančasto) i odgovarajuće krivulje interpolirane preciznosti u ovisnosti o odzivu (plavo).

Dalje u ovom seminaru će se koristiti oznaka *mAP* za srednju prosječnu interpoliranu preciznost.

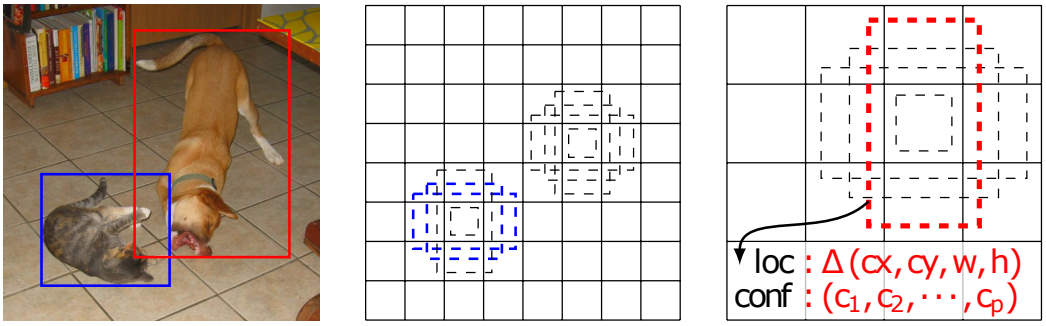
4. SSD: Single Shot MultiBox Detector

SSD (*Single Shot Multibox Detector*) [8] je model za pronalaženje objekata (koji pripadaju većem broju razreda) koji se temelji na dubokoj konvolucijskoj mreži koja u jednoj propagaciji unaprijed generira sve predikcije okvira od kojih se oni najbolji ostavljaju (engl. *single-shot detection*).

Glavna značajka SSD-a je da umjesto korištenja posebne komponente za generiranje prijedloga okvira koristi ravnomjerno raspoređene okvire (slika 4.1) koji će se dalje nazivati razmatranim okvirima i na svima od njih u jednom prolazu provodi klasifikaciju i regresiju položaja i omjera stranica. Za svaki okvir se procjenjuje pouzdanost klasifikacije u moguće razrede i prilagodbe dimenzija i položaja. Još jedna bitna značajka mreže je da dijelovi za prepoznavanje objekata izravno koriste izlazne tenzore značajki konvolucijskih slojeva različitih veličina na različitim dubinama konvolucijske mreže. Time se omogućuje bolje prepoznavanje objekata različitih veličina. Za dodatno poboljšavanje regresije omjera stranica, na istim položajima se koriste razmatrani okviri različitih omjera stranica.

Za ulazne slike dimenzija 300×300 na skupu podataka VOC2007-test SSD ostvaruje $mAP = 0.743$ i brzinu (broj obrađenih slika u sekundi) $59s^{-1}$ uz paralelizirano obrađivanje 8 slika i $46s^{-1}$ uz pojedinačno obrađivanje slika na grafičkoj kartici *Nvidia Titan X*, što je poboljšanje za malo više od 0.01 u *mAP* i više nego šesterostruko ubrzanje u odnosu na tada[1] najbolji model Faster R-CNN [14]. U međuvremenu su se pojavili još neki modeli koji su dostigli ili prestigli performanse SSD-a [7, 13].

[1] Rad je objavljen 2016. godine.



(a) Slika sa željenim okvirima

(b) Mapa značajki dimenzija 8×8

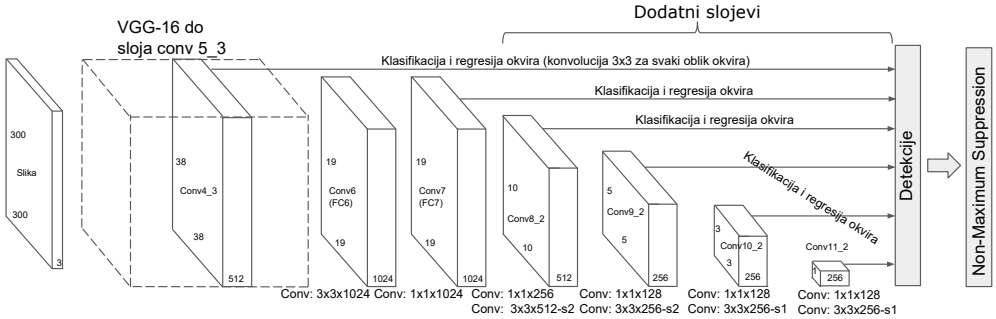
(c) Mapa značajki dimenzija 4×4

Slika 4.1: Slike ilustriraju glavne značajke modela SSD. Svakom pikselu nekih slojeva značajki pridruženo je nekoliko razmatranih okvira različitih omjera stranica. Crtkani okviri na prikazanim mapama značajki po položaju, veličini i omjeru stranica najbolje odgovaraju ciljnim okvirima na slici i zato su najpogodniji za učenje parametara na tom primjeru. Slike su preuzete iz [8].

4.1. Model

Konvolucijska arhitektura za izlučivanje značajki. Model se temelji na prednjem dijelu neke dobre unaprijedne konvolucijske mreže za klasifikaciju. Koriste se svi slojevi te mreže osim onih zadnjih koji služe konačnoj klasifikaciji. U radu je za to korištena mreža VGG-16 [15] do sloja `conv_5_3` (bez posljednja dva potpuno povezana sloja za klasifikaciju). Takvoj mreži je dodano još 10 slojeva od kojih neki koriste izlazni korak 2 za konvoluciju i prema kraju se (u slučaju modela SSD-300) smanjuju od dimenzija 19×19 do dimenzija 1×1. Ti dodatni slojevi služe prepoznavanju značajki više razine i omogućuju bolje prepoznavanje objekata različitih dimenzija.

Za lokalizaciju objekata se izravno koriste tenzori značajki 6 slojeva. Najveći od njih, koji je dimenzija 38×38, je sloj `conv4_3` iz mreže VGG-16. Jedan piksel neke njegove mape značajki po relativnoj veličini odgovara otprilike 8×8 piksela ulazne slike koja je dimenzija 300×300. Zadnji sloj značajki `conv11_2` je dimenzija 1×1 i svaka njegova mapa značajki (od njih 256) sastoji se od samo jedne realne vrijednosti. On se koristi za prepoznavanje objekata koji prekrivaju cijelu sliku. Arhitektura mreže je detaljnije opisana na slici 4.2. Kao prijenosna funkcija se za sve konvolucijske slojeve koristi zglobnica (ReLU).



Slika 4.2: Ilustracija sustava SSD s prikazanim mapama značajki koje se koriste za pronalaženje objekata. Slika je preuzeta iz [8] i malo izmijenjena.

Slojevi za pronalaženje objekata. Tenzor značajki sloja conv4_3 (dimenzija $38 \times 38 \times 512$) i tenzore značajki 5 od 10 dodatnih slojeva (dimenzija od $19 \times 19 \times 1024$ do $1 \times 1 \times 256$) koriste dijelovi mreže za lokalizaciju, tj klasifikaciju i regresiju razmatranih okvira. Svakom položaju u mapama značajki svakog od tih slojeva pridružen je jedan skup razmatranih okvira različitih dimenzija. Pomicanjem konvolucijskih jezgri, od kojih svaka predstavlja za jedan oblik okvira, preko tenzora značajki provodi se klasifikacija i regresija razmatranih okvira. Te jezgre su dimenzija $3 \times 3 \times p$, gdje je p broj mapa značajki ulaznog tenzora. Računaju se pouzdanosti klasifikacije u $|C|$ razreda i prilagodbe okvira (pomak središta i prilagodba dimenzija[2]) kako bi se okviri prilagodili obliku objekta. Neka je R skup brojeva koji predstavljaju omjere stranica. Iz nekog tenzora značajki na svakom se položaju primjenom $|R|$ konvolucijskih jezgri za klasifikaciju i regresiju okvira dobiva $(|C| + 4)\,|R|$ vrijednosti koje definiraju pouzdanosti klasifikacije, položaj i dimenzije okvira rezultata.

Veličine i omjeri stranica razmatranih okvira. Na slici 4.1 prikazana su dva primjera slojeva značajki različitih dimenzija i okviri pridruženi nekim njihovim pikselima. Neka se koristi m tenzora značajki za traženje objekata. Faktor skaliranja za okvire k-tog sloja značajki, uz $k \in \{1..m\}$, određuje se ovako:

$$\sigma_k = \sigma_{\min} + \frac{k-1}{m-1}(\sigma_{\max} - \sigma_{\min}). \tag{4.1}$$

[2]Vrijednosti prilagodbi okvira nisu apsolutne (u pikselima) nego su relativne s obzirom na razmatrani okvir. Pomak u vertikalnom/horizontalnom smjeru je skaliran inverzom visine/širine razmatranog okvira, a prilagodba visine/širine je prirodni logaritam visine/širine podijelene visonom/širinom razmatranog okvira. To je formalnije opisano u ulomku Funkcija gubitka u potpoglavlju 4.2.

Pri tome $\sigma_{\text{min}} = 0.2$ i $\sigma_{\text{max}} = 0.9$. Uz omjere stranica $r \in R = \left\{1, 2, 3, \frac{1}{2}, \frac{1}{3}\right\}$, širine w_{kr} i visine h_{kr} razmatranih okvira k-tog sloja značajki nad kojim se primjenjuje klasifikacija i regresija okvira se računaju ovako:

$$w_{kr} = \sigma_k r^{\frac{1}{2}}, \tag{4.2}$$

$$h_{kr} = \sigma_k r^{-\frac{1}{2}}. \tag{4.3}$$

Također se još koriste okviri omjera 1 s faktorom skaliranja $\sigma'_k = \sqrt{\sigma_k \sigma_{k+1}}$. Tako se dobiva $|R| + 1 = 6$ razmatranih okvira za svaki položaj u nekom od odabranih tenzora značajki.

U konačnom modelu koji je testiran kod nekih slojeva postoje odstupanja od navedenih pravila omjera stranica i faktora skaliranja.

Odbacivanje okvira. Za ulaznu sliku dimenzija 300×300 dobije se 8732 okvira. Većinu njih treba odbaciti i ostaviti samo jedan okvir za svaki objekt pronađen na slici. Za to se koristi algoritam *non-maximim suppression* (*NMS*). Njime se za svaki razred nezavisno pronalaze okviri za koje je Jaccardov koeficijent sličnosti veći od određenog praga i, uzimajući u obzir pozdanosti klasifikacije, pohlepnim algoritmom se uklanja višak okvira. Točna implementacija je dostupna ovdje: https://github.com/intel/caffe/blob/master/src/caffe/util/bbox_util.cpp.

4.2. Učenje

SSD se uči tako da se za svaku sliku s ciljnim okvirima prvo iz skupa za učenje među razmatranim okvirima odaberu oni koji su najprikladniji za učenje pozitivnih i negativnih primjera rezultata, a onda se parametri korišteni za dobivanje odabranih okvira prilagođavaju algoritmom koji se temelji na gradijentnom spustu. S će dalje označavati skup razmatranih okvira, a T skup ciljnih okvira trenutne slike. C će označavati skup razreda koji uključuje i razred *ostalo* (ili *pozadina*).

Odabir razmatranih okvira za učenje. Tijekom učenja na nekoj slici potrebno je odrediti koji od razmatranih okvira su po položajima i dimenzijama najprikladniji za učenje parametara. Prvo se za svaki objekt, tj. ciljni okvir, pronalazi razmatrani okvir koji se s njim najviše preklapa, tj. ima najveći Jaccardov koeficijent sličnosti s njim. Tako se osigurava da je svakom objektu pridružen barem 1 razmatran okvir.

Također se još odabiru i drugi razmatrani okviri koji se s nekim ciljnim okvirom preklapaju s Jacardovim koeficijentom većim od 0.5. Na taj način se pri učenju svakom ciljnom okviru (pozitivnom primjeru) pridružuje neprazan podskup razmatranih okvira koji se s njim dovoljno preklapaju. U nastavku će funkcija $a\colon T \to 2^S$ predstavljati to pridruživanje. Ona se može izraziti ovako:

$$a(t) = \{\operatorname{argmax}_s\{J(t,s) \mid s \in S\}\} \cup \{s \in S \mid J(t,s) > 0.5\}. \tag{4.4}$$

Može se primijetiti da se u slučaju postojanja razmatranog okvira koji se s ciljnim okvorom preklapa s Jaccardovim koeficijentom većim od 0.5 funkcija svodi na svoj drugi član.

Funkcija gubitka. Neka je $S_+ = \bigcup_{t \in T} a(t)$ skup svih razmatranih okvira koji su odabrani za učenje na pozitivnim primjerima. Gubitak je definiran kao težinski zbroj gubitka lokalizacije L_l i gubitka pouzdanosti klasifikacije L_c:

$$L = \frac{1}{|S_+|}(L_\mathsf{l} + \alpha L_\mathsf{c}). \tag{4.5}$$

Neki razmatrani ili ciljni okvir $u = (\boldsymbol{x}_u, \boldsymbol{d}_u, \boldsymbol{c}_u)$ određen je vektorom položaja $\boldsymbol{x}_u$ dimenzije 2, vektorom visine i širine $\boldsymbol{d}_u$ dimenzije 2 i pouzdanostima klasifikacije $\boldsymbol{c}_u$ dimenzije $|C|$ koje su u slučaju ciljnog okvira vektor jednojediničnog koda (engl. *one-hot vector*). Dimenzije razmatranog okvira određuju se kao što je opisano u ulomku Veličine i omjeri stranica razmatranih okvira u potpoglavlju 4.1.

Neka je $t \in T$ ciljni okvir, a $s \in a(t)$ razmatrani okvir čiji se parametri uče. Cilj je postići da okvir rezultat h_s bude što sličniji ciljnom okviru t. Relativne koordinate nekog okvira u u odnosu na razmatrani okvir s su definirane preko transformacije

$$r_s(u) = ((\boldsymbol{x}_u - \boldsymbol{x}_s) \oslash \boldsymbol{d}_s, \ln(\boldsymbol{d}_u \oslash \boldsymbol{d}_s), \boldsymbol{c}_u) \tag{4.6}$$

sa svojstvom $r_s(s) = (\mathbf{0}, \mathbf{0}, \boldsymbol{c}_s)$. Ona ima inverz

$$r_s^{-1}(\hat{u}) = (\boldsymbol{x}_s + \boldsymbol{x}_{\hat{u}} \odot \boldsymbol{d}_s, \exp(\boldsymbol{d}_{\hat{u}}) \odot \boldsymbol{d}_s, \boldsymbol{c}_{\hat{u}}). \tag{4.7}$$

Pri tome su exp, ln, $\odot$ i $\oslash$ redom eksponencijalna funkcija, prirodni logaritam, množenje[3] i dijeljenje po elementima vektora. Slojevi za klasifikaciju i regresiju okvira predikciju $\hat{h}_s$ računaju u relativnim koordinatama s obzirom na razmatrani okvir s, tj. konačni okviri rezultati se računaju ovako: $h_s = r_s^{-1}(\hat{h}_s)$. Gubitak

[3] Množenje po elementima vektora još se naziva Hadamardov produkt.

lokalizacije je definiran kao udaljenost po zaglađenoj L^1-normi $\|\cdot\|_{\tilde{1}}$) predviđenih i ciljnih parametara položaja i dimenzija okvira u relativnim koordinatama:

$$L_{\mathsf{l}} = \sum_{t \in T} \sum_{s \in a(t)} \left(\|\boldsymbol{x}_{\hat{h}_s} - \boldsymbol{x}_{r_s(t)}\|_{\tilde{1}} + \|\boldsymbol{d}_{\hat{h}_s} - \boldsymbol{d}_{r_s(t)}\|_{\tilde{1}} \right). \tag{4.8}$$

Zaglađena L^1-norma $\|\cdot\|_{\tilde{1}}$ iz [4] je poseban slučaj Huberove[4] norme i definirana je ovako:

$$\|\boldsymbol{x}\|_{\tilde{1}} = \sum_i |x_i|_{\tilde{1}}\,, \quad \text{uz} \quad |x|_{\tilde{1}} = \begin{cases} 0.5x^2, & |x| < 1, \\ |x| - 0.5, & |x| \geq 1. \end{cases} \tag{4.9}$$

Druga komponenta funkcije gubitka je gubitak klasifikacije koji je proporcionalan unakrsnoj entropiji između dobivenih i ciljnih jednojediničnih razdioba razreda:

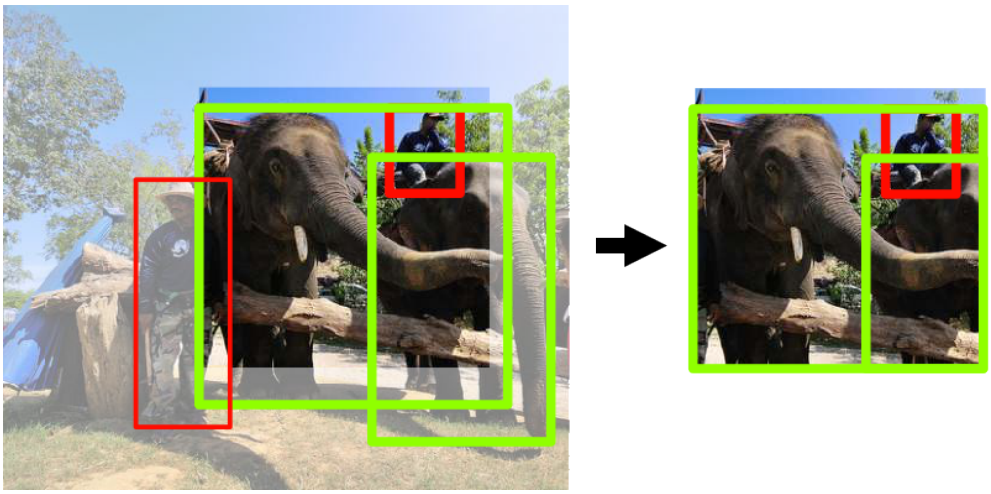
$$L_{\mathsf{c}} = -\sum_{t \in T} \sum_{s \in a(t)} \ln c_{s(\operatorname{argmax} \boldsymbol{c}_t)} - \sum_{s \in S_-} \ln c_{s0}, \tag{4.10}$$

gdje je $(\operatorname{argmax} \boldsymbol{c}_t) \in \{1..|C|-1\}$ oznaka razreda ciljnog okvira t, 0 oznaka razreda *ostalo*, a S_- skup razmatranih okvira koji se ne preklapaju dovoljno ni s jednim ciljnim okvirom, tj. $S_- = S \setminus S_+$.

Odabir negativnih primjera. Nakon odabira razmatranih okvira za učenje na ciljnim okvirima, tj. pozitivnim primjerima, samo mali broj okvira ostane u skupu S_+. S_- je puno veći skup od S_+, zbog čega negativni primjeri imaju prevelik utjecaj na gubitak pouzdanosti klasifikacije definiran kao u jednadžbi 4.10. Umjesto cijelog S_- koristi se samo njegov podskup koji daje najveći gubitak pouzdanosti klasifikacije i nema više elemenata od $3\,|S_+|$. U radu je utvrđeno da se na taj način ostvaruje brža i stabilnija konvergencija.

Proširivanje skupa podataka. Broj i raznolikost primjera za učenje povećava se proširivanjem skupa podataka. Pri učenju se svaka slika uzima u izvornom obliku ili se nasumično uzorkuje neki njen dio uz neko ograničenje minimalnog preklapanja s objektima ili se nasumično uzorkuje neki njen dio. Odabrani dio slike treba imate veličinu barem 0.1 veličine cijele slike. Tako dobivena slika se uvećava na fiksnu veličinu, zrcali s vjerojatnošću 0.5 i još se primjenjuju nasumične transformacije. Zadržavaju se samo ciljni okviri čije središte je unutar uzorkovanog dijela slike i prilagođavaju rubovima slike kao što je ilustrirano slikom 4.3.

[4] https://en.wikipedia.org/wiki/Huber_loss



Slika 4.3: Odabir i odbacivanje ciljnih okvira pri uzorkovanju dijela slike. Na lijevoj slici podebljani su svi okviri čije se središta nalaze unutar uzorkovanog područja slike. Oni se zadržavaju i prilagođavaju uzorkovanom dijelu slike.

5. Rezultati

U eksperimentima autori su koristili mrežu VGG-16 s unaprijed naučenim parametrima na skupu podataka *ILSVRC CLS-LOC*. Slojevi `fc6` i `fc7` su zamijenjeni konvolucijskim slojevima, `fc9` uklonjen, a sloj `pool5` izmijenjen tako da umjesto receptivnog polja dimenzija 2×2 i izlaznog koraka 2 koristi receptivno polje dimenzija 3×3 i izlaznog koraka 1.

Iskorištavaju se naučeni parametri potpuno povezanih slojeva `fc6` i `fc7`. `fc6` je pretvoren u ekvivalentni konvolucijski sloj s jezgrom[1] dimenzija 7×7 koja pokriva cijeli tenzor značajki sloja `pool5`. Promjenom izlaznog koraka sloja `pool5` s 2 na 1, kako bi imali smisla parametri sloja `fc6`, potrebno je povećati njegov ulazni korak (dilataciju) na 2, čime se njegovo receptivno polje udvostručuje na dimenzije 14×14 (proporcionalno povećanju dimenzija sloja `pool5`). Konačno, jezgra sloja `fc6` je poduzorkovana tako da se pretvori u jezgru dimenzija 3×3 s dilacijom 6, čime se dimenzije receptivnog polje ne mijenjaju. Time se postiže otprilike 20-postotno ubrzanje prolaza unaprijed u odnosu na mrežu s nepoduzorkovanom jezgrom sloja `fc6`. Sloj `fc6` je pretvoren u konvolucijeki sloj s jezgrom domenzija 1×1.

Dodatni slojevi se inicijaliziraju Xavierovim inicijalizacijom. Korišten je stohastički gradijentni spust s koeficijentom inercije (engl. *momentum*) 0.9, L^2 regularizacijom, veličinom grupe 32, početnom stopom učenja 10^{-3} i smanjivanjem stope učenja. Detalji su opisani u radu. Slijedi pregled glavnih rezultata i zaključaka.

Za učenje i testiranje korišteni su skupovi *Pascal VOC2007-test*, *Pascal VOC2012-test* i *COCO*. Usporedba rezultata testiranja različitih modela na prikazana je u tablici 5.1. Autori zaključuju da SSD ima manju grešku lokalizacije u odnosu na R-CNN jer SSD uči zajednički regresiju i klasifikaciju okvira na istim mapama značajki. R-CNN prvo određuje okvir, uzorkuje dio slike određen okvirom i onda klasificira taj dio slike.

[1] Ovdje se koristi riječ *jezgra* u jednini, ali ona se zapravo odnosi na 1024 jezgre jer izvorni sloj `fc6` ima izlaz dimenzije 1024.

Autori su također mjerili utjecaj razreda kojemu objekt pripada i veličine objekta. Zaključuju da SSD daje lošije rezultate za male objekte zbog velikog broj slojeva prije prvog sloja čije se mape značajki koriste za lokalizaciju. SSD ima problema kod razlikovanja objekata koji pripadaju nekim razredima (često životinje).
Prepoznavanje malih objekata može se poboljšati korištenjem veće ulazne slike, ali autori navode da još ima prostora za poboljšanje. Također zaključuju da SSD zbog razmatranih okvira različitih omjera stranica jako dobro određuje oblik okvira.

Model	*FPS*	Podaci za učenje	*mAP*/%	
			VOC2007	VOC2012
Fast-RCNN [4]	0.5	VOC07	66.9	-
		VOC07, VOC12	70.0	68.4
Faster-RCNN [14]	7	VOC07	69.9	-
		VOC07, VOC12	73.2	70.4
		VOC07, VOC12, COCO	78.8	75.9
YOLO [13]	45	VOC07, VOC12	63.4	57.9
*YOLOv2-544 [13]	40	VOC07, VOC12	78.6	73.4
*PVANET-c [7]	31	VOC07, VOC12, COCO	84.4	83.7
*PVANET [7]	22	VOC07, VOC12, COCO	84.9	84.2
SSD300	46	VOC07	68.0	-
		VOC07, VOC12	74.3	72.1
		VOC07, VOC12, COCO	79.6	77.5
SSD512	19	VOC07	71.6	-
		VOC07, VOC12	76.8	74.9
		VOC07, VOC12, COCO	81.6	80.0

Tablica 5.1: Usporedba rezultata evaluacije na skupovima *VOC2007-test* i *VOC2012-test* označenih u tablici s "VOC2007" i "VOC2012". U stupcu "*FPS*" su brzine izvođenja (broj slika u sekundi) pri pojedinačnoj obradi (grupe veličine 1) na grafičkoj kartici *Nvidia Titan X*. U stupcu U stupcu "Podaci za učenje" su skupovi podataka (bez dijela koji je korišten za testiranje) koji su korišteni za učenje. "VOC07, VOC12, COCO" označava da je model (osim za PVANET) prvo učen na skupu *COCO-trainval35k*, a nakon toga još prilagođen na uniji odgovarajućih dijelova skupova *VOC2007* i *VOC2012*. Od različitih varijanti modela, odabrane su one koje su po brzini izvođenja najusporedivije s modelom SSD. * označava modele novije od SSD-a.[8, 7, 13]

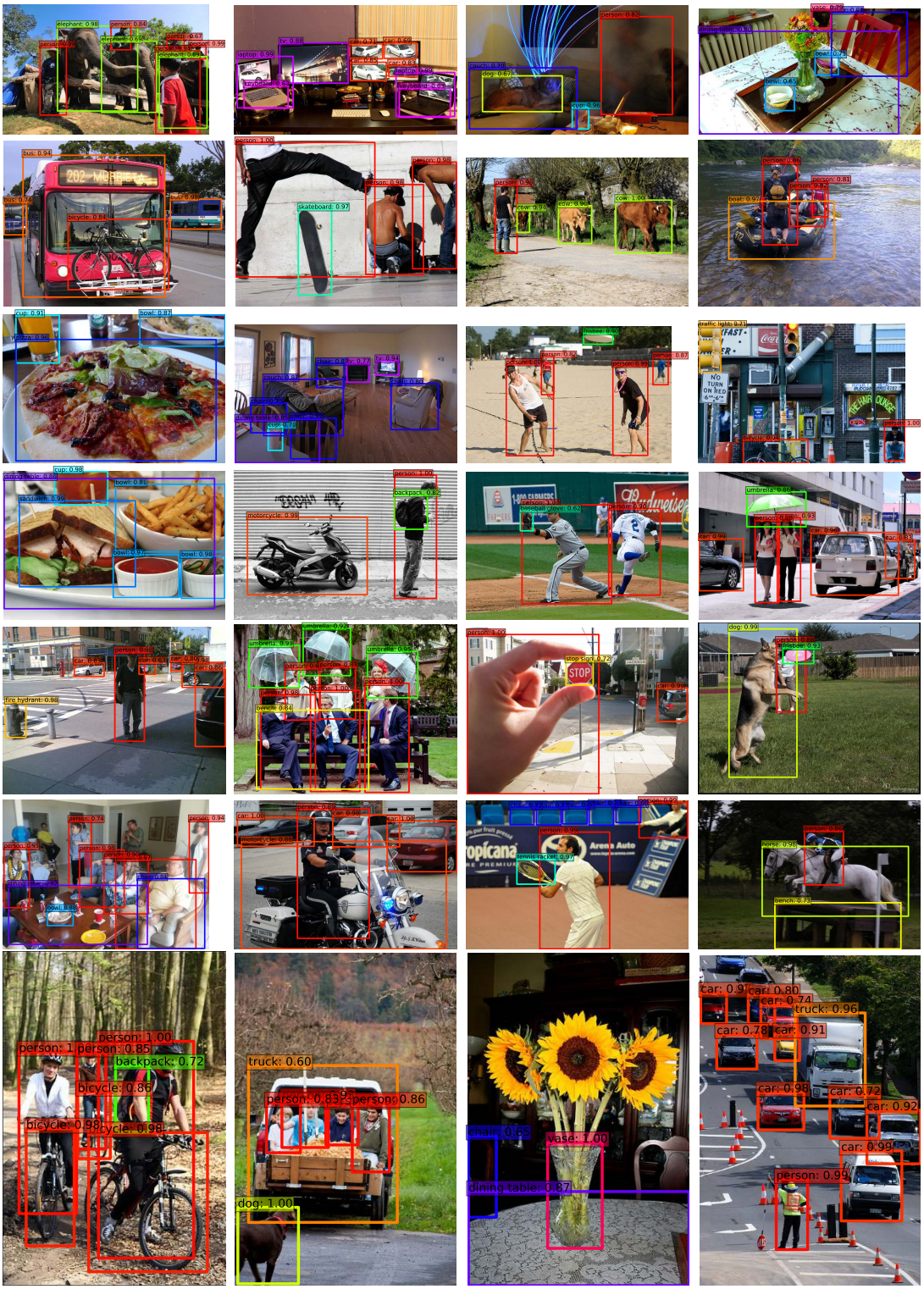
Autori su analizirali utjecaj izmjene različitih komponenata na učinak sustava. U

više slojeva značajki za traženje objekata		•	•	•	•	•
više proširivanja podataka	•		•	•	•	•
korištenje omjera $\left\{\frac{1}{2}, 2\right\}$	•	•		•	•	•
korištenje omjera $\left\{\frac{1}{3}, 3\right\}$	•	•			•	•
jezgra 3×3 s dilatacijom 6 u sloju `fc6`	•	•	•	•		•
mAP/%	62.4	65.5	71.6	73.7	74.2	**74.3**

Tablica 5.2: Utjecaj različitih odabira komponenata na srednju prosječnu pogrešku kod modela SSD300 na skupu *VOC2007-test*. Redak "više slojeva značajki za traženje objekata" označava korištenje slojeva značajki `conv4_3`, `conv7`, `conv8_2`, `conv9_2`, `conv10_2` i `conv11_2` umjesto samo `conv7` za pronalaženje objekata.

tablici 5.2 prikazan je utjecaj različitih odabira komponenata na učinak modela SSD300 na skupu *VOC2007-test*. Vidi se da od navedenog, osim korištenja 6 slojeva značajki u odnosu na samo `conv7` kao ulaze slojeva za pronalaženje objekata, najveći utjecaj ima dodatno proširivanje skupa podataka u odnosu na korištenje samo originalnih slika i slika dobivenih njihovim horizontalnim zrcaljenjem. Vidi se da značajan utjecaj ima i korištenje okvira s omjerima stranica $\left\{2, 3, \frac{1}{2}, \frac{1}{3}\right\}$. Korištenje poduzorkovanih jezgri u sloju `fc6` nema značajan utjecaj na rezultate, ali donosi ubrzanje izvođenja.

Na slici 5.1 su prikazani primjeri rezultata pronalaženja objekata modelom SSD500.



Slika 5.1: Primjeri rezultata pronalaženja objekata na slikama iz skupa *COCO-test-dev* modelom SSD500. Prikazani su okviri s pouzdanošću klasifikacije većom od 0.6. Slike su preuzete iz [8].

6. Zaključak

SSD donosi neke zanimljive ideje koje zajedno omogućuju brzu obradu slika bez smanjenja kvalitete rezultata u odnosu na neke druge modele. Glavne od njih su korištenje slojeva značajki različitih dimenzija na različitima dubinama konvolucijske mreže za pronalaženje objekata i korištenje razmatranih okvira različitih omjera stranica kao osnova za dobivanje konačnih okvira združenom regresijom i klasifikacijom. Time je postignuta relativno jednostavna konvolucijska arhitektura kojom se izbjegavaju višestruki izračuni sličnih značajki i omogućuje dobro prepoznavanje objekata različitih veličina i oblika.

U prepoznavanju objekata, kao i mnogim drugim područjima računalnog vida, još uvijek nisu postignuti rezultati koji dostižu one koje postižu ljudi. Ima puno prostora za poboljšanje i može se očekivati još poboljšanja nad postojećim modelima.

Duboka arhitektura za jednoprolaznu lokalizaciju objekata

Sažetak

Ovaj seminar razmatra postupak pronalaženja orijentiranih okvira (engl. oriented bounding box, OBB) koji obuhvaćaju objekte koji pripadaju različitim kategorijama na slici. Postupak *Single Shot MultiBox Detector* (SSD) objavljen je 2016. i donosi višestruko ubrzanje učenja i izvođenja bez gubitka preciznosti (mAP) s obzirom na dotadašnje najbolje modele Fast R-CNN i Faster R-CNN.

Ključne riječi: lokalizacija objekata, računalni vid, konvolucijske mreže, duboke neuronske mreže, duboko učenje

7. Literatura

[1] Precision-recall — scikit-learn 0.18.1 documentation. URL http://scikit-learn.org/stable/auto_examples/model_selection/plot_precision_recall.html. [27.3.2017.].

[2] M. Everingham i J. Winn. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Development Kit, 2012. URL http://host.robots.ox.ac.uk/pascal/VOC/voc2012/devkit_doc.pdf.

[3] Ross Girshick. Object Detection, Deep Learning, and R-CNNs. 2014. URL https://homes.cs.washington.edu/~shapiro/EE596/notes/DeepLearning.pptx.

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

[5] Ivan Grubišić. Semantička segmentacija dubokim konvolucijskim mrežama, 2016.

[6] 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.

[7] Sanghoon Hong, Byung-Seok Roh, Kye-Hyeon Kim, Yeongjae Cheon, i Minje Park. PVANet: Lightweight Deep Neural Networks for Real-time Object Detection. *CoRR*, abs/1611.08588, 2016. URL http://arxiv.org/abs/1611.08588.

[8] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, i Alexander C. Berg. SSD: Single Shot MultiBox Detector. *CoRR*, abs/1512.02325, 2015. URL http://arxiv.org/abs/1512.02325.

[9] Michael A. Nielsen. Neural Networks and Deep Learning, 2015. URL http://neuralnetworksanddeeplearning.com/.

[10] Christopher Olah. Colah's blog, 2016. URL http://colah.github.io/. [10.5.2016.].

[11] Alec Radford, Luke Metz, i Soumith Chintala. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *CoRR*, abs/1511.06434, 2015. URL http://arxiv.org/abs/1511.06434.

[12] Joseph Redmon. YOLO: Real-Time Object Detection, 2017. URL https://pjreddie.com/darknet/yolo/. [26.3.2017.].

[13] Joseph Redmon i Ali Farhadi. YOLO9000: Better, Faster, Stronger. *CoRR*, abs/1612.08242, 2016. URL http://arxiv.org/abs/1612.08242.

[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] Karen Simonyan i Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR*, abs/1409.1556, 2014. URL http://arxiv.org/abs/1409.1556.

[16] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, i A. W. M. Smeulders. Selective Search for Object Detection. 2012. URL https://ivi.fnwi.uva.nl/isis/publications/2013/UijlingsIJCV2013/UijlingsIJCV2013.pdf.

[17] Wikipedia. Information retrieval — Wikipedia, 2017. URL https://en.wikipedia.org/wiki/Information_retrieval. [28.4.2017].

[18] C. Lawrence Zitnick i Piotr Dollár. Edge Boxes: Locating Object Proposals from Edges. 2014. URL https://www.microsoft.com/en-us/research/wp-content/uploads/2014/09/ZitnickDollarECCV14edgeBoxes.pdf.