

Detekcija objekata metodama strojnog učenja

Bastijanić, Luka

Master's thesis / Diplomski rad

2020

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Pula / Sveučilište Jurja Dobrile u Puli**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:137:616314>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2025-03-31**



Repository / Repozitorij:

[Digital Repository Juraj Dobrila University of Pula](#)



Sveučilište Jurja Dobrile u Puli
Fakultet informatike u Puli

LUKA BASTIJANIĆ

**DETEKCIJA OBJEKATA METODAMA STROJNOG
UČENJA**

Diplomski rad

Pula, 2020.

Sveučilište Jurja Dobrile u Puli
Fakultet informatike u Puli

LUKA BASTIJANIĆ

**DETEKCIJA OBJEKATA METODAMA STROJNOG
UČENJA**

Diplomski rad

JMBAG: 0303047679, redoviti student

Studijski smjer: Informatika

Predmet: Umjetna Inteligencija

Znanstveno područje: Tehničke znanosti

Znanstveno polje: Računarstvo

Znanstvena grana: Umjetna inteligencija

Mentor: doc. dr. sc. Darko Etinger

Pula, rujan 2020.



IZJAVA O AKADEMSKOJ ČESTITOSTI

Ja, dolje potpisani Luka Bastijanić, kandidat za magistra informatike ovime izjavljujem da je ovaj Diplomski rad rezultat isključivo mogega vlastitog rada, da se temelji na mojim istraživanjima te da se oslanja na objavljenu literaturu kao što to pokazuju korištene bilješke i bibliografija. Izjavljujem da niti jedan dio Diplomskog rada nije napisan na nedozvoljen način, odnosno da je prepisan iz kojega necitiranog rada, te da ikoji dio rada krši bilo čija autorska prava. Izjavljujem, također, da nijedan dio rada nije iskorišten za koji drugi rad pri bilo kojoj drugoj visokoškolskoj, znanstvenoj ili radnoj ustanovi.

Student

U Puli, _____, _____ godine



IZJAVA

o korištenju autorskog djela

Ja, Luka Bastijanić dajem odobrenje Sveučilištu Jurja Dobrile

u Puli, kao nositelju prava iskorištavanja, da moj diplomski rad pod nazivom „Detekcija objekata metodama strojnog učenja“ koristi na način da gore navedeno autorsko djelo, kao cjeloviti tekst trajno objavi u javnoj internetskoj bazi Sveučilišne knjižnice Sveučilišta Jurja Dobrile u Puli te kopira u javnu internetsku bazu završnih radova Nacionalne i sveučilišne knjižnice (stavljanje na raspolaganje javnosti), sve u skladu s Zakonom o autorskom pravu i drugim srodnim pravima i dobrom akademskom praksom, a radi promicanja otvorenoga, slobodnoga pristupa znanstvenim informacijama.

Za korištenje autorskog djela na gore navedeni način ne potražujem naknadu.

U Puli, _____ (datum)

Potpis

Sadržaj

1. UVOD.....	1
2. Umjetna inteligencija	2
2.1. pristupi umjetnoj inteligenciji	3
2.1.1. Ljudsko ponašanje: Pristup Turingovog testa	3
2.1.2. Ljudsko razmišljanje: Pristup kognitivnog modeliranja	4
2.1.3. Racionalno razmišljanje: Pristup „Zakona misli“	5
2.1.4. Racionalno ponašanje: Pristup racionalnog agenta	5
2.2. Temelji umjetne inteligencije	6
2.2.1. Filozofija.....	6
2.2.2. Matematika	7
2.2.3. Ekonomija	8
2.2.4. Neuroznanost.....	9
2.2.5. Psihologija.....	9
2.2.6. Računalno inženjerstvo.....	9
2.2.7. Kibernetika i teorija kontrole.....	10
2.2.8. Lingvistika	10
2.3. Povijest umjetne inteligencije	11
3. Strojno učenje	14
3.1. Područja primjene.....	14
3.2. Zadaci strojnog učenja	16
3.2.1. Klasifikacija (eng. classification).....	16
3.2.2. Regresija (eng. regression).....	17
3.2.3. Rangiranje (eng. ranking)	17
3.2.4. Detekcija anomalija (eng. anomaly detection).....	17
3.2.5. Klasteriranje (eng. clustering)	18

3.2.6.	Transkripcija (eng. transcription)	18
3.2.7.	Smanjenje dimenzionalnosti	19
3.3.	Učenje u strojnom učenju	19
3.3.1.	Nadgledano učenje	19
3.3.2.	Polu-nadgledano	20
3.3.3.	Nenadgledano učenje	20
3.3.4.	Podržano/ojačano učenje	20
4.	Računalni vid	22
4.1.	Slika	25
4.1.1.	Boja	26
4.1.2.	Rubovi i njihova detekcija	27
4.1.3.	Tekstura	28
4.1.4.	SIFT algoritam	29
5.	Primjena strojnog učenja u računalnom vidu	31
5.1.	Klasifikacija	31
5.1.1.	NN algoritam	31
5.1.2.	Bayes'-ova teorija odlučivanja	32
5.2.	Klasifikacija slika strojnim učenjem	33
5.2.1.	k-NN	33
5.2.2.	Logistička regresija	33
5.2.3.	Stablo odlučivanja	34
5.2.4.	SVM – Support Vector Machine	34
6.	Duboko učenje	35
6.1.	ANN	35
6.2.	CNN (eng. Convolutional neural network)	36
6.1.1.	Kako CNN funkcionira?	36
6.3.	Koraci za izgradnju klasifikatora slike dubokim učenjem	37

7.	Detekcija objekata dubokim učenjem	39
7.1.	Jednostupanjske i dvostupanjske metode	39
7.1.1.	Jednostupanjske metode.....	40
7.1.2.	Dvostupanjske metode	41
7.2.	Izazovi detekcije objekata.....	42
7.2.1.	Problem u varijaciji razmjera.....	42
7.2.2.	Problem zaklanjanja	42
7.2.3.	Problem deformacije	43
8.	Scenariji korištenja detekcije objekata.....	44
8.1.	Korištenje kod detekcije prekršaja u prometu.....	44
8.2.	Korištenje u zdravstvenim ustanovama	45
8.3.	Korištenje u poljoprivredi	45
8.4.	Korištenje u komercijalne svrhe.....	45
8.5.	Korištenje u prirodi.....	46
9.	Programska rješenja detekcije objekata	47
9.1.	Python	47
9.2.	OpenCV.....	47
9.3.	TensorFlow i Keras.....	48
10.	ZAKLJUČAK.....	50
	LITERATURA	51
	Popis slika	52
	Mrežni izvori	53
	Sažetak.....	55
	Summary	56

1. UVOD

Umjetna inteligencija je trenutno jedna od najvrućih tema. Nekad se samo u filmovima moglo vidjeti pametne sustave koji su neovisni o djelovanju čovjeka, a danas već slušamo o prvim autonomnim automobilima ili robotima koji izvršavaju radnje opažajući okolinu. Sastavni dio umjetne inteligencije je i strojno učenje koje je velikim dijelom zaslužno za sve jači razvoj umjetne inteligencije.

Strojno učenje nam pomaže u raznim zadacima bili oni lagani ili nešto kompleksniji poput raspoznavanja neželjene pošte. Problem prepoznavanja neželjene pošte naizgled se čini kao banalni primjer, no, u našem sve brže rastućem digitalnom svijetu, takve sitnice nam štede velike količine vremena i stvaraju sigurniju digitalnu okolinu.

Jedno od područja strojnog učenja je i računalni vid. To je vitalno područje za omogućiti računalu da postane „bolje“ u izvršavanju nekih novih zadataka koje mu zadajemo. Omogućuje autonomnim vozilima da vide kuda voze i procjenjuju kako se trebaju ponašati u prometu na osnovu detekcije objekata. Osim toga, omogućuje robotima da uz pomoć vida mogu sami učiti izvršavati neke radnje. Time dolazimo do detekcije objekata koja je i tema ovog rada. Ovaj rad sadrži teorijsku osnovu dostupnih metoda strojnog učenja koje služe za detekciju objekata. Od onih „zastarjelih“ gdje smo sami označavali značajke slika do onih, poput dubokih mreža, gdje računalno samo izvršava takve zadatke.

Drugo poglavlje bavi se poviješću i osnovama umjetne inteligencije, treće osnovama strojnog učenja i tipovima učenja, četvrto se bavi računalnim vidom, peto objašnjava na koji se način strojno učenje primjenjuje u računalnom vidu, dok se šesto i sedmo bave dubokim učenjem i detekcijom objekata uz pomoć njega, te se i zadnje osmo poglavlje bavi korištenjem detekcije objekata u stvarnom svijetu.

2. Umjetna inteligencija

Umjetna inteligencija ili na engleskom jeziku Artificial Intelligence (skraćeno AI) jedno je od najmlađih i po svemu sudeći najzanimljivijih polja znanosti. Njezin nastanak datira ubrzo nakon Drugog svjetskog rata, dok je sam naziv toj novoj znanosti dodijeljen 1956. godine.

Čovjeka je oduvijek fascinirala inteligencija, s obzirom da je to ono što ga najviše ističe među ostalim živim bićima na planeti Zemlji. Iz tog razloga su nas stručnjaci i nazvali lat. Homo sapiens, što u prijevodu znači čovjek umni ili čovjek razumni. Od kada smo svjesni sebe i svoje inteligencije postavljamo si razna pitanja kao što su: „Što je to inteligencija?“, „Kako mi to razmišljamo?“, „Kako naš mozak funkcionira?“. Sva ta pitanja su relevantna i u razvoju umjetne inteligencije s obzirom da nastojimo stvoriti računalo koje može razmišljati, shvaćati, predviđati, opažati i zaključivati baš poput nas ljudi. Odnosno pokušavamo stvoriti nove inteligentne entitete.

Tijekom povijesti razvoja ove znanosti mnogi znanstvenici pokušali su definirati što bi to umjetna inteligencija bila, pa je tako prvu definiciju zadao jedan od pionira umjetne inteligencije John McCarthy 1955. godine, a koja glasi: „*Cilj umjetne inteligencije je razviti stroj koji se ponaša kao da je inteligentan.*“. Druga definicija koja se može pronaći u Encyclopedii Britannici glasi: „*Umjetna inteligencija je sposobnost digitalnih računala ili računalno kontroliranih robota da rješavaju probleme koji su inače asocirani s višim intelektualnim sposobnostima čovjeka.*“, međutim prema toj definiciji svako računalo je umjetna inteligencija s obzirom da izvršava već sada veliki broj operacija koje može i čovjek, poput pamćenja nekakvog teksta ili matematičkog računa. Najbolju definiciju ponudila je Elaine Rich koja glasi: „*Umjetna inteligencija je studij o tome kako učiniti računala sposobnima da čine stvari u kojima su ljudi trenutno bolji.*“ Ova definicija će biti valjana i u budućnosti s obzirom da je čovjek još uvijek puno inteligentniji u nekim radnjama koje su mu urođene, poput snalaženja u prostoru ili reakcije na nekakve podražaje iz okoline. Umjetna inteligencija podijeljena je na više polja, a to su: strojno učenje (ovim poljem bavit ćemo se detaljnije s obzirom da je predmet ovog rada), obrada prirodnog jezika (eng. NLP – natural language processing), govor, ekspertni sustavi, robotika i vid.

2.1. Pristupi umjetnoj inteligenciji

Prema Russell-u i Norvig-u (2010.) postoje četiri pristupa umjetnoj inteligenciji, a to su:

- Ljudsko ponašanje;
- Ljudsko razmišljanje;
- Racionalno razmišljanje;
- Racionalno ponašanje.

2.1.1. Ljudsko ponašanje: Pristup Turingovog testa

Alan Turing je 1950. godine predstavio takozvani Turingov test. To je test koji je osmišljen tako da nam pruža dovoljno dobru operativnu definiciju inteligencije. Turingov test dolazi u obliku više pitanja, ta su pitanja istovremeno postavljena računalu i jednom čovjeku. Nakon što oni daju odgovore na postavljena pitanja, odgovore pregledava ispitivač, međutim on ne zna tko je dao koje odgovore. Ukoliko ispitivač ne uspije razaznati jesu li napisani odgovori došli od računala ili osobe, računalo je prošlo test. Pripremiti računalo za ovakav test nije mala stvar i iziskuje da računalo posjeduje neke sposobnosti, a to su:

- Obrada prirodnog jezika (eng. NLP – natural language processing) – kako bi bilo sposobno uspješno komunicirati na ljudskom jeziku;
- Prikaz znanja – sposobnost da spremi što zna ili čuje;
- Automatizirano rasuđivanje - sposobnost da iskoristi spremljene informacije za odgovaranje na pitanja i donošenje novih zaključaka;
- Strojno učenje – sposobnost da se prilagodi novim okolnostima i da otkrije nove obrasce.

Razvijen je također potpuni Turingov test, a takav test uključuje video signal kako bi ispitivač mogao testirati sposobnosti uočavanja subjekta. Isto tako pruža priliku ispitivaču da računalu predstavi fizičke objekte.

Kako bi računalo prošlo takav test potrebne su mu dvije stvari:

1. Računalni vid kako bi mogao opaziti objekte;
2. Robotika za pomicanje objekata i kretanje.

Turingov test ostao je relevantan 60 godina, dok znanstvenici na području umjetne inteligencije nisu odlučili napustiti takav pristup s obzirom da su počeli vjerovati da je važno proučavati osnovne principe inteligencije, a ne stvarati duplikate primjerka.

2.1.2. Ljudsko razmišljanje: Pristup kognitivnog modeliranja

Temelj pristupa kognitivnog modeliranja je taj da program razmišlja kao čovjek, a kako bi se to postiglo potrebno je za početak utvrditi kako čovjek razmišlja, odnosno utvrditi kako ljudski um funkcionira. Tri su načina za učiniti navedeno:

1. Samoispitivanjem – pokušavanje hvatanja vlastitih misli;
2. Psihološkim eksperimentima – promatranjem osobe u akciji;
3. Snimanjem mozga – promatranje mozga u akciji.

Kada dokučimo kako ljudski um zapravo radi, to znanje možemo prenijeti u računalni program. Ukoliko bi računalo stvorilo input ili output koji bi odgovarao ljudskom ponašanju tada bi imali dokaz da mehanizam programa uspješno obavlja mehanizam kakav se može pronaći i kod ljudi.

Primjer ovog pristupa možemo pronaći u slučaju Allena Newella i Herberta Simona koji su 1961. godine razvili GPS („General Problem Solver“) odnosno Opći rješavač problema. Njih nije zadovoljavalo kada bi računalni program problem riješio točno, već je predmet njihovog interesa bio taj da utvrde u kojoj su mjeri koraci za rješavanje problema računalnog programa istovjetni onima kod ljudskih ispitanika koji su rješavali isti problem. Tu je vrlo bitna kognitivna znanost koja spaja računalne modele umjetne inteligencije kao i eksperimentalne tehnike iz psihologije kako bi se stvorile precizne i provjerljive teorije ljudskog uma.

2.1.3. Racionalno razmišljanje: Pristup „Zakona misli“

Prvi koji je pokušao dešifrirati načine razmišljanja bio je Aristotel. Njegovi silogizmi ponudili su primjere za strukturu argumenata koja bi uvijek dala točan zaključak ukoliko bi bile ponuđene točne pretpostavke. Primjer toga je: „Sokrat je čovjek; svi ljudi su smrtni“ stoga proizlazi da je Sokrat smrtan. Iz ovoga je nastalo područje logike. U 19. stoljeću logičari su razvili svu potrebnu notaciju za izraziti izjave o bilo kojim objektima na svijetu, kao i veze među njima. Uz pomoć ove notacije već do 1965. godine stvoreni su prvi programi koji su mogli riješiti problem koji je bio opisan takvom logičkom notacijom. Ova takozvana logička tradicija u umjetnoj inteligenciji se služi tim načelima kako bi stvorila inteligentne sustave. Međutim ovaj pristup ima i nekoliko mana, prva od njih je ne baš jednostavno prenošenje neformalnog znanja u formalno iz kojeg bi se definirala logička notacija za određeni problem. Druga velika mana je što postoji velika razlika između rješavanja problema u teoriji i u praksi. Ovaj problem se može javiti već i s nekoliko stotina činjenica ukoliko program nije dobro vođen odnosno nema jasno definirano gdje početi i koji su mu daljnji koraci.

2.1.4. Racionalno ponašanje: Pristup racionalnog agenta

Sastavni elementi ovog pristupa su agenti. Agent je nešto što djeluje (odnosno izvršava zadaće), a od svakog računalnog agenta očekuje se da:

- Djeluje samostalno;
- Opaža svoju okolinu;
- Bude ustrajan u dužem vremenskom periodu;
- Bude prilagodljiv promjenama;
- Stvara i slijedi zadane ciljeve.

Kod ovog pristupa konkretno se koriste racionalni agenti, a oni moraju djelovati tako da postignu najbolji mogući ishod ili kad to nije moguće eventualno najbolji očekivani ishod. Jednim dijelom racionalni agenti se koriste prethodno opisanim pristupom odnosno „Zakonom misli“ s obzirom da je jedan od načina racionalnog djelovanja logikom zaključiti da će dana akcija postići zadane ciljeve te potom djelovati prema

dobivenim zaključcima. S druge strane nije uvijek sve u racionalnosti jer ponekad nije jasno što se treba postići, ali je svejedno potrebno nešto napraviti. Postoje isto tako i načini racionalnog djelovanja za koje se ne mogu izvesti zaključci.

Prva prednost ovog pristupa u odnosu na prethodno opisane pristupe, je ta što je puno opširniji nego što je to recimo pristup „Zakona misli“, iz razloga što je donošenje ispravnih zaključaka samo jedan od načina racionalnog djelovanja. Druga prednost je što je ovaj pristup podložniji znanstvenom napretku nego što su to pristupi temeljeni na ljudskom ponašanju ili ljudskom razmišljanju.

2.2. Temelji umjetne inteligencije

Temelje za umjetnu inteligenciju kroz povijest su postavile mnoge discipline, a najveći trag su ostavile: filozofija, matematika, ekonomija, neuroznanost, psihologija, računalstvo, kibernetika i lingvistika. Ponešto o utjecajima svake od navedenih disciplina navodi se u nastavku.

2.2.1. Filozofija

Filozofija je dala puno velikih imena koja su pridonijela razvoju umjetne inteligencije, među njima su Aristotel, Thomas Hobbes, Leonardo da Vinci, Blaise Pascal, Gottfried Wilhelm Leibniz i René Descartes. Obilježilo ih je sljedeće:

- Racionalizam koji smatra da se jedino rasuđivanjem može spoznati objektivna stvarnost;
- Dualizam koji označava mišljenje da postoje dvije ravnopravne, iskonske i nesvodljive kategorije stvarnosti, a to su duh i tvar (ideja i materija). Međutim na životinje se u toj ideji gledalo kao na nešto što ne posjeduje te kvalitete već su sličnije strojevima;
- Materijalizam u kojem je glavna teorija da se naša stvarnost sastoji od fizičke tvari, a koja je uzrok ljudske misli, osjećaja i akcije. Slobodna volja pritom se smatra načinom percepcije dostupnih izbora koji se pojavljuju subjektu;

- Empirizam zastupa stav da je iskustvo osnovni izvor spoznaje, pa je prema tome iskustvo ono što određuje domet, granice i mogućnosti;
- Indukcija kao logička metoda kod koje je osnovni zaključak da ono što vrijedi za svaki pojedinačni slučaj jedne vrste vrijedi za cijelu vrstu;
- Logički pozitivizam koji objedinjuje racionalizam i empirizam, a osnovna premisa mu je da se svo znanje može okarakterizirati logičkim teorijama povezanim s prikupljenim iskustvom;
- Teorija konfirmacije koja analizira prikupljanje znanja iz iskustva.

Filozofi su svojim razmatranjem ideja o tome kako je ljudski um na neki način sličan stroju svakako pridonijeli umjetnoj inteligenciji. Funkcionira tako što koristi znanje spremljeno u obliku internog jezika, a potom misao koristi kao alat s kojim bira koju će radnju poduzeti.

2.2.2. Matematika

Matematika je znanost koja je formalno poduprla filozofska razmišljanja u područjima logike, računanja i vjerojatnosti. Kako bi istražili mogućnosti logike i računanja, znanstvenici na području matematike stvorili su algoritme, a među njima je, možda i najpoznatiji, Euklidov algoritam za računanje najvećeg zajedničkog djelitelja. Kreirali su i algoritme za logičku dedukciju te su uz pomoć toga dokazali da postoji procedura koja dokazuje svaku točnu izjavu logike prvog reda. Međutim, logika prvog reda nije imala mogućnost dohvatiti principe kojima bi matematička indukcija mogla okarakterizirati prirodne brojeve. Kurt Gödel je stoga 1931. godine svojim teoremom nepotpunosti pokazao da postoje ograničenja u korištenju dedukcije, odnosno da postoje točne izjave za koje ne postoji dokaz u teoriji. Uvidjevši problem da neke funkcije nad cijelim brojevima ne mogu biti predstavljene algoritmom, Alan Turing počeo je karakterizirati funkcije s eng. *computable* odnosno karakterizirati ih kao podložnima računanju. Još jedan termin koji je ostavio snažan otisak u matematici je *tractability*, odnosno svaki problem koji nije moguće riješiti algoritmom s polinomijalnim rastom nazivamo *intractable*. Kako bi se takvi problemi prepoznali, predstavljena je teorija NP-potpunosti od strane Stevena Cooka i Richarda Karpa. Oni su dokazali postojanje velike klase kombinatoričnih pretraga i problema rasuđivanja koji su NP-

potpuni. Osim logike i računanja veliki doprinos umjetnoj inteligenciji s područja matematike je teorija vjerojatnosti. Nastala je iz potrebe da se opišu mogući ishodi kockarskih događaja te se uglavnom kroz povijest i razvijala korištenjem kockarskih primjera. Međutim vrlo brzo vjerojatnost je postala neprocjenjiv dio svake kvantitativne znanosti. Matematičari su dakle kroz povijest osmislili alate kojima su omogućili manipuliranje izjavama logičke sigurnosti kao i onima nesigurne probabilističke izjave. Jednako tako udarili su temelje za razumijevanje računanja i rasuđivanje algoritama.

2.2.3. Ekonomija

Ekonomija kao grana znanosti pojavljuje se 1776. godine i uglavnom se bavi proučavanjem načina na koji ljudi donose odluke koje vode do željenih ishoda. Najčešće se veže uz pojam novca. Ono čime je ekonomija pridonijela jesu razne teorije i različiti procesi. Među njima je teorija odlučivanja koja omogućava formalni i cjeloviti okvir za donošenje odluka u uvjetima neizvjesnosti. Ova teorija koristi se uglavnom u većim ekonomijama gdje ekonomski agent ne mora pratiti ponašanje drugih individualnih agenata. No, za manje ekonomije osmišljena je teorija igara s obzirom da takve situacije izgledaju poput igre u kojoj svaki igrač može značajno utjecati na korisnost drugih, pozitivno ili negativno. Do tada su se u ekonomiji uglavnom bavili pitanjima o trenutnim rezultatima, međutim bilo je bitno baviti se i pitanjem kako učiniti nešto čiji će rezultat biti vidljiv tek u budućnosti. Stoga je u Drugom svjetskom ratu razvijeno područje operativnih istraživanja, koje je potom dovelo do procesa odlučivanja. Ekonomija i operativna istraživanja su uvelike pridonijeli razvoju racionalnih agenata, a sami ekonomisti formalizirali su problem donošenja odluka koje maksimiziraju očekivane ishode donositelju odluka.

2.2.4. Neuroznanost

Neuroznanost je znanost koja proučava živčani sustav, a osobito mozak. Znanstvenicima je predmet fascinacije oduvijek bilo funkcioniranje mozga, te na koji način omogućava razmišljanje. Ljudski mozak sastavljen je od neurona koji su međusobno povezani i komuniciraju pomoću signala, a znanstvenici su dokučili da skup neurona dovodi do misli, akcija i svijesti. Mozak i računala se u neku ruku mogu usporediti s obzirom na „performanse“, ali vrijeme ciklusa kod računala je milijun puta brže nego kod mozga. Međutim mozak ima daleko više memorijskog „prostora“ nego osobno računalo, dok su neka superračunala dosegula količinu memorije mozga. Ono u čemu je računalo možda u najvećoj prednosti je činjenica da čovjek, iako ima veliki potencijal kad govorimo o memoriji i moždanim funkcijama, koristi daleko manji postotak svog mozga.

2.2.5. Psihologija

Psihologija postavlja pitanje kako ljudi i životinje razmišljaju i djeluju. Ono što je ova znanstvena grana pridonijela temi od interesa jest razvoj kognitivne psihologije. To je znanost koja je usmjerena na otkrivanje psihičkih procesa koji se nalaze u osnovi ponašanja. Ona vidi mozak kao uređaj koji procesira informacije, a predmet njenog istraživanja su kreativnost, mišljenje, pamćenje, percepcija, pozornost, predstavljanje znanja i rješavanje problema. Iz kognitivne psihologije rodila se kognitivna znanost u Sjedinjenim Američkim Državama 1956. godine, a njeno stvaranje je potaknuto razvojem računalnog modeliranja.

2.2.6. Računalno inženjerstvo

U jeku Drugog svjetskog rata napravljene su prve verzije digitalnog elektroničkog računala. Računalo je uz inteligenciju osnova uspješne umjetne inteligencije. Od kreacije prvih računala do danas, svakom iteracijom njihovog razvoja postajala su sve

manja, brža i cijena njihove proizvodnje je postajala sve niža. No, nije samo hardver zaslužan za razvoj umjetne inteligencije već je tu i softver, kao što su operativni sustavi, programski jezici i sve ostalo što je veoma bitno kod kreacije modernih računalnih programa.

2.2.7. Kibernetika i teorija kontrole

Kibernetika se definira kao dio znanosti o općim zakonitostima procesa upravljanja, reguliranja, dobivanja, pohranjivanja, pretvorbe i prijenosa informacija u sustavima neovisno o njihovoj prirodi. Začetnikom kibernetike smatra se Norbert Wiener koji je 1948. godine izdao knjigu naziva „Cybernetics“, a koja je postala bestseller i probudila zanimanje javnosti za idejom umjetno inteligentnih strojeva.

Moderna teorija kontrole, a posebno grana poznata kao optimalna kontrola, ima za cilj projektirati sustave koji maksimiziraju ciljnu funkciju kroz vrijeme. Teorija kontrole bavi se dakle projektiranjem uređaja koji će se ponašati optimalno na osnovu povratnih informacija iz okoline.

2.2.8. Lingvistika

Moderna lingvistika i umjetna inteligencija stvorene su otprilike u isto vrijeme te se iz njih izrodilo novo područje nazvano računska lingvistika ili obrada prirodnog jezika, poznatije pod nazivom „NLP“ odnosno eng. Natural Language processing. „NLP“ se bavi interakcijom između računala i ljudskih jezika, a posebno načinima kako procesirati i analizirati velike količine podataka iz prirodnog jezika pomoću računala.

2.3. Povijest umjetne inteligencije

Umjetna inteligencija imala je kroz povijest svojih uspona i padova, ali uvijek se nastojalo predstavljati nove kreativne pristupe i poboljšavati one najbolje. Umjetna inteligencija se posljednjeg desetljeća rapidno razvija zbog sve češćeg korištenja znanstvenih metoda u eksperimentima i uspoređivanja pristupa razvoju. Prema Wolfgang Ertelu (2011.) glavne prekretnice razvoja umjetne inteligencije su sljedeće:

1931. Austrijanac Kurt Gödel pokazao je da su u logici prvog reda sve izjave koje su točne ujedno i dokazive, dok kod logike drugog reda postoje točne izjave koje nisu dokazive.

1937. Alan Turing ukazuje na ograničenja inteligentnih strojeva uz pomoć problema zaustavljanja.

1943. McCulloch i Pitts modeliraju mreže neurona i stvaraju vezu na izjavnu logiku.

1950. Alan Turing definira strojnu inteligenciju s Turingovim testom te piše o strojevima koji uče i genetskim algoritmima.

1951. Marvin Minsky razvija stroj s neuronskom mrežom koja je sastavljena od 3000 elektronskih cijevi i uspijeva simulirati četrdeset neurona.

1955. Arthur Samuel (IBM) razvija program koji uči igrati šah i koji potom pobjeđuje svog stvoritelja.

1956. McCarthy organizira konferenciju na Dartmouth College-u. Na toj konferenciji predstavljen je termin umjetne inteligencije. Newell i Simon sa Sveučilišta Carnegie Mellon (CMU) predstavljaju prvi računalni program koji je obrađivao simbole, nazvan „Logic Theorist“.

1958. na MIT-u (Massachusetts Institute of Technology) McCarthy izumljuje viši programski jezik nazvan LISP. Piše programe koji su sposobni sami sebe modificirati.

1959. Gelernter (IBM) izrađuje dokazivač geometrijskih teorema.

1961. Newellov i Simonov opći rješavač problema (eng. The General Problem Solver - GPS) imitira ljudske misli.

1963. McCarthy osniva „AI Lab“ (Laboratorij umjetne inteligencije) na sveučilištu Stanford.

1965. Robinson izumljuje rješavajući račun za predikatnu logiku.

1966. Weizenbaumov program Eliza ostvaruje dijalog s ljudima na prirodnom jeziku.

1969. Minsky i Papert u svojoj knjizi „Perceptrons“ pokazuju da perceptron, jednostavna neuronska mreža, može predstavljati samo linearne funkcije.

1972. francuski znanstvenik Alain Colmerauer izumljuje logički programski jezik PROLOG. Britanski fizičar de Dombel razvija ekspertni sustav za dijagnozu akutne abdominalne boli.

1976. Shortliffe i Buchanan razvijaju MYCIN, ekspertni sustav za dijagnozu zaraznih bolesti, koji je sposoban nositi se s nesigurnostima.

1981. Japan započinje “Fifth Generation Project” (projekt pete generacije) s ciljem da sagradi moćan PROLOG stroj.

1982. ekspertni sustav za konfiguraciju računala nazvan R1 šteti korporaciji „Digital Equipment Corporation“ 40 milijuna dolara na godinu.

1986. je došlo do renesanse neuronskih mreža, a najzvučnija imena razdoblja su bila Rumelhart, Hinton i Sejnowski. Sustav imena Nettek naučio je čitati tekst na glas.

1990. Pearl, Cheeseman, Whittaker i Spiegelhalter prenose teoriju vjerojatnosti u umjetnu inteligenciju s Bayesov-im mrežama. Više-agentni sustavi postaju popularni.

1992. Tesauroov program TD-gammon demonstrira prednosti učenja ojačanja (eng. reinforcement learning).

1993. inicijativa Worldwide RoboCup izrađuje autonomnog robota koji igra nogomet.

1995. Vapnik iz teorije statističkog učenja razvija metodu potpornih vektora koja je vrlo važna danas.

1997. IBM-ov računalo za igranje šaha Deep Blue pobjeđuje svjetskog prvaka u šahu Gary Kasparova. Prvo RoboCup natjecanje održava se u Japanu.

2003. roboti na RoboCup-u demonstriraju što su u stanju postići umjetna inteligencija i robotika.

2006. servisna robotika postaje od velikog istraživačkog interesa unutar umjetne inteligencije.

2010. autonomni roboti počinju učiti svoja pravila.

2011. IBM-ov program za razumijevanje prirodnog jezika i odgovaranje na pitanja nazvan „Watson“ pobjeđuje dva ljudska prvaka u američkom TV kvizu „Jeopardy!“.

Obzirom da knjiga ne pokriva događaje iza 2011. godine, u nastavku slijede događaji istaknuti na internetskim stranicama besplatne enciklopedije „Wikipedia“ za vremenski period do 2019. godine.

2011.-2014. Appleova „Siri“ (2011), Googleov „Google Now“ (2012) i Microsoftova „Cortana“ (2014) su aplikacije za pametne telefone koje koriste prirodni jezik za odgovaranje na pitanja, davanje preporuka i izvršavanje radnji na uređaju.

2013. SCHAFT Inc-ov Robot HRP-2 pobjeđuje 15 timova i osvaja DARPA-in „Robotics Challenge Trials“. Natjecanje je sadržavalo više različitih izazova kao što su vožnja vozila, hodanje po ostacima, penjanje po ljestvama, rađanje prolaza u zidu, spajanje cijevi i zatvaranje ventila. „NEIL“ (eng. Never Ending Image Learner) kreiran je na sveučilištu Carnegie Mellon kako bi konstantno uspoređivao i analizirao veze između različitih slika.

2015. Hawking, Musk, Wozniak i 3000 drugih znanstvenika s područja umjetne inteligencije i robotike potpisuju otvoreno pismo kojim bi se zabranila proizvodnja autonomnog oružja. Google DeepMind-ov AlphaGo (verzija: Fan) pobjeđuje trostrukog europskog prvaka u igrici GO Fan Hui-a, 5 prema 0. Razvoj AlphaGo-a nastavlja se i kroz kasnije godine te ostvaruje zapažene rezultate.

2017. OpenAI-ov strojno učeni bot igrao je Dotu 2 na „The International 2017 Dota 2 tournament“ i pobijedio je u igri 1 na 1 protiv profesionalnog igrača Dendi-ja.

2018. „Alibaba“ umjetna inteligencija koja obrađuje jezik nadigrava najuspješnije osobe na sveučilištu Stanford u testu čitanja i razumijevanja. Najavljen Google Duplex, servis koji omogućava asistenta (umjetnu inteligenciju) da vrši rezervacije termina sastanaka.

3. Strojno učenje

Strojno učenje je pod-područje računalne znanosti, te je vrlo važan dio umjetne inteligencije. Bavi se izgradnjom algoritama koji se oslanjaju na izgrađenoj zbirci primjera nekog fenomena, dakle oslanja se na informacijama prikupljenima u prošlosti što možemo nazvati iskustvom. Takve informacije mogu se dobiti iz prirode, može ih stvoriti čovjek ili ih može generirati neki drugi algoritam.

Strojno učenje se isto tako definira kao proces rješavanja praktičnih problema skupljajući podatke i stvaranjem statističkog modela uz pomoć prikupljenog seta podataka. Za taj statistički model pretpostavlja se da se na neki način koristi kako bi riješio praktični problem. Zadatak strojnog učenja je stvoriti učinkovit i precizan prediktivni algoritam, a ono što odlikuje ovakve algoritme jesu vremenska i prostorna kompleksnost.

Uspješnost algoritma ovisi o korištenim podacima, stoga je strojno učenje usko vezano uz analizu podataka i statistiku. Tehnike učenja bazirane su na informacijama naučenima iz samih podataka (engl. Data-driven) te kombiniraju osnovne koncepte računalne znanosti s idejama iz statistike, vjerojatnosti i optimizacije.

3.1. Područja primjene

Strojno učenje ima široki spektar primjena u realnom svijetu, a možda najpoznatiji slučaj koji se najčešće koristi kao primjer je otkrivanje neželjene elektroničke pošte (eng. spam e-mail). Nitko ne voli u sandučiću e-pošte vidjeti poštu sumnjivog sadržaja jer u nju najčešće spada e-pošta koja sadrži računalni virus ili nekakav vid prijevare, a i ponekad nas jednostavno sadržaj određene e-pošte jednostavno ne zanima. Zbog toga pružatelji usluga elektroničke pošte nastoje implementirati način da takva pošta odlazi u sandučić neželjene pošte ili da uopće ne pristigne primatelju. U tom slučaju koristi se strojno učenje. Prema istraživanju Gbenga Dada i sur. (2019.) kod rješavanja problema neželjene pošte najčešće se koristi tehnika filtriranja prema sadržaju, tj. analizira se sadržaj svake pristigle pošiljke što obuhvaća pošiljatelja, naslov i sadržaj, a iz prikupljenih podataka stvara se model prema kojem će ubuduće prepoznati je li e-

pošta neželjena. To će učiniti tako što će klasificirati svu poštu prema povratnoj informaciji od strane korisnika je li određena pošiljka bila neželjena. Što je više prikupljenih primjera, s vremenom će algoritam biti bolji u filtriranju takve pošte.

No, otkrivanje neželjene e-pošte je samo jedan od mnogih područja primjene, a prema Mohri, Rostamizadeh i Talwalkar (2018.) neki od češćih primjera su:

1. Klasifikacija teksta ili dokumenata – ovdje spada i prije opisani problem neželjene pošte, a obuhvaća probleme kao što su dodjeljivanje teme tekstu ili dokumentu i određivanje koje su mrežne stranice maliciozne ili neprimjerene. Dobar primjer su članci na Internet portalima za koje se želi da se samo-označe u koju kategoriju vijesti spadaju, ukoliko im se ne odredi eksplicitno gdje pripadaju;
2. Obrada prirodnog jezika (eng. Natural Language processing) – koristi se kod npr. označavanja vrste riječi (eng. part-of-speech tagging) u govoru, prepoznavanje dijelova teksta kao i njihova kategorizacija (eng. named-entity recognition), gramatičko raščlanjivanje bez konteksta (eng. Context-free parsing) koristi se za prepoznavanje ukoliko neka riječ pripada određenom jeziku s obzirom na gramatiku i drugi;
3. Obrada govora – ovdje spadaju prepoznavanje govora, sinteza govora, prepoznavanje govornika i drugi problemi poput modeliranje jezika ili akustike;
4. Računalni vid – ova kategorija bavi se prepoznavanjem i identifikacijom objekata, prepoznavanjem lica, itd. Ovaj primjer je ujedno i tema ovog rada stoga će ova problematika biti detaljnije obrađena u daljnjim poglavljima, a pogotovo prepoznavanje i identifikacija objekata;
5. Računalna biologija – uključuje analizu gena i proteinskih mreža te predikciju proteinskih funkcija;

Uz navedeno, u svakodnevnom životu postoje i mnoge druge primjene poput igranja računalnih igara, društvenih igara poput šaha, utvrđivanje medicinskih dijagnoza, Internet pretraživači, detekcija raznih prijevara.

3.2. Zadaci strojnog učenja

Strojno učenje osmišljeno je kako bi nam pomoglo u savladavanju zadataka koji mi kao ljudi ne možemo samostalno riješiti ili nismo u stanju napisati dovoljno dobar program koji bi mogao izvršiti te zadatke, tj. oponašati čovjeka pri njihovom izvršenju. Zbog toga nam je potrebno sredstvo koje će iz iskustva samo naučiti kako riješiti zadatke, s obzirom da se okolnosti konstantno mijenjaju te je nemoguće ručno predvidjeti svaki poseban slučaj za koji bi bilo potrebno implementirati programsko rješenje.

Raspoznamo dva tipa zadataka. Prvi su oni koje ljudi i životinje već rutinski izvršavaju odnosno oni za čije izvršavanje ni sami ne znamo kako bismo isprogramirali računalo. To su zadaci poput vožnje automobila, razumijevanja slike ili raspoznavanje govora. Drugi su oni koje ljudi uopće nisu u mogućnosti izvršiti, a tiče se zadataka poput vremenskih prognoza ili mrežnih pretraživača kod kojih je potrebno obraditi veliku količinu podataka. Sve je to moguće jer računala, za razliku od ljudi, danas mogu posjedovati beskonačno mnogo memorije za pohranu podataka.

Sukladno navedenom, zaključujemo da su nam potrebna adaptivna rješenja poput strojnog učenja koja će za razliku od konvencionalnih računalnih programa, koji jednom kada se instaliraju nisu u mogućnosti prilagoditi se izmjenjivoj okolini, doskočiti tako da će učeći iz primjera i okoline pronaći rješenje.

Zadaci koji se najčešće spominju i najviše izučavaju su sljedeći: klasifikacija, regresija, rangiranje, klasteriranje, detekcija anomalija, transkripcija i drugi. Neki od njih bit će detaljno opisani u daljnjem tekstu.

3.2.1. Klasifikacija (eng. classification)

Zadatak klasifikacije je svakome objektu, odnosno ulaznom parametru dodijeliti neku kategoriju. Kao dobar primjer klasifikacije je već spomenuto dodjeljivanje kategorije novinskom članku za kojeg to nije eksplicitno učinjeno. Dakle, recimo da autor članka piše o ozbiljnoj prometnoj nesreći koja se dogodila toga dana, ali nije naveo kategoriju

u kojoj se članak na mrežnim stranicama portala mora pojaviti. U ovom slučaju potrebno je članak klasificirati u kategoriju crne kronike. Na isti način klasifikacija se koristi kod prepoznavanja i identifikacije objekata na slikama, pa će se tako recimo s obzirom na sadržaj, tu sliku smjestiti u odgovarajuću kategoriju. Ukoliko je na slici mačka klasificirat će ju kao takvu, isto će se dogoditi i ukoliko je na slici pas, te za mnogo drugih kategorija.

3.2.2. Regresija (eng. regression)

Regresija ima za zadaću predvidjeti numeričku vrijednost za dani ulazni parametar odnosno svakom predmetu predvidjeti stvarnu vrijednost. Koristi se najčešće za predviđanje vrijednosti nekakvih vrijednosnih papira ili predviđanje određenih ekonomskih pokazatelja.

3.2.3. Rangiranje (eng. ranking)

Zadatak rangiranja je naučiti poslagati predmete prema nekom kriteriju. Dobar primjer toga su Internet pretraživači poput Google-a i Yahoo-a. Dakle uz pomoć rangiranja prema ključnoj riječi kojom se pretražuje, pretraživači će rezultate pretraživanja posložiti ovisno o numeričkoj ili ordinalnoj vrijednosti koja je tom predmetu dodijeljena u tom slučaju.

3.2.4. Detekcija anomalija (eng. anomaly detection)

Detekcija anomalija ima zadatak proći kroz kolekciju objekata ili događaja te označiti one koje odskaču ukoliko su neobični ili atipični. Primjer za detekciju anomalije možemo pronaći u svakodnevnom životu kada dođe do krađe identiteta ili podataka što rezultira prijevarama s bankovnim karticama. Banke ili izdavači kartica uz pomoć ovakvih algoritama za svakog korisnika kreiraju model prema njegovim kupovnim

navikama i onog trenutka kada se dogodi nekakva atipična kupovina, npr. u vrlo kratkom vremenskom periodu kupili smo tri mobilna uređaja, banka će staviti na čekanje transakcije te nas nazvati da provjeri jesmo li uistinu mi zadali takve transakcije. Algoritam se isto tako koristi za prepoznavanje osobe kod prijavljivanja na korisnički račun. Ukoliko živimo u Hrvatskoj i svaki dan se prijavljujemo s uređaja marke Samsung, a odjednom se netko prijavi iz Kanade s uređaja marke Sony, najčešće budemo obaviješteni putem e-pošte o takvom događaju.

3.2.5. Klasteriranje (eng. clustering)

Zadatak klasteriranja je grupirati razne objekte u manje podgrupe odnosno klasterne. Dakle cilj je imati objekte vrlo slične prema određenim kriterijima u istim klasterima, a da nisu slični objektima iz drugih grupa. Algoritam se koristi kod analize velikih setova podataka. Primjer klasteriranja je biologija gdje se grupiraju različite vrste biljaka i životinja ili knjižnice gdje zbog velike količine knjiga, iste treba na neki način grupirati radi lakšeg pronalaska prema temi ili sadržaju same knjige.

3.2.6. Transkripcija (eng. transcription)

Transkripcija ima za zadatak proučiti relativno nestrukturiranu reprezentaciju podataka i prepisati ih u neki tip tekstualnog oblika. To može biti npr. prepoznavanje govora, na način da se na računalo učita zvučna datoteka iz koje će on, prepoznajući što čuje na snimci, zapisati riječi u tekstualnom obliku. Također, primjer transkripcije je i prepoznavanje rukopisa gdje će algoritam iz slike na kojoj se vidi rukopis, prepisati set znakova u ASCII obliku u novi tekstualni oblik.

3.2.7. Smanjenje dimenzionalnosti

Kod problema smanjenja dimenzionalnosti cilj je pretvoriti početnu reprezentaciju podataka u niže-dimenzionalnu reprezentaciju vodeći brigu o tome da se očuvaju neka svojstva one početne. Vrlo čest primjer toga je pred-procesiranje fotografija kod računalnog vida.

3.3. Učenje u strojnom učenju

Strojno učenje razlikuje nekoliko vrsta učenja, a to su prema Burkovu (2019.): nadgledano (eng. supervised), polu-nadgledano (eng. semi-supervised), nenadgledano (eng. unsupervised) i podržano/ojačano (eng. reinforcement) koji će u nastavku biti detaljno opisani. Drugi izvori spominju i dodatne tipove učenja, poput Mohri i sur. (2018.) koji navode još i sljedeće tipove: transduktivni zaključak (eng. transductive inference), on-line učenje (eng. on-line learning), aktivno/pasivno učenje. Također se navodi kako je moguće pronaći primjere drugih vrsta učenja koji nisu zastupljeni u toj mjeri kao do sada navedeni.

3.3.1. Nadgledano učenje

Nadgledano učenje sastoji se od skupa podataka koji mogu poprimiti neku oznaku, s obzirom na obilježja kojima su opisani. Dakle, poznamo obilježja podataka i imamo očekivani rezultat. To se može dobro prikazati na primjeru neželjene pošte koji je prethodno spomenut. Svaka pošiljka ima pošiljatelja, naslov, sadržaj i eventualno privitke te ju prema tome možemo obilježiti kao željenu ili neželjenu poštu. Na osnovu obilježja stoga kreiramo model prema kojem će se utvrditi je li pošiljka neželjena. Osim toga, jedan od poznatih primjera je utvrđivanje ima li osoba prema svojim obilježjima određenu bolest ili pak najpoznatiji primjer korištenja je određivanje vrste cvijeta irisa kada se prema obilježjima cvijeta nastoji utvrditi vrsta.

Nadgledano učenje je najuobičajeniji scenarij korištenja već opisanih problema klasifikacije, regresije i rangiranja.

3.3.2. Polu-nadgledano

Polu-nadgledano učenje u svom skupu podataka za treniranje sadrži označene i neoznačene podatke. Gotovo uvijek je veći broj neoznačenih, dok je broj označenih značajno manji. Ova vrsta učenja ima isti cilj kao nadgledano učenje, međutim koristi se u nadi da bi korištenjem mnogo neoznačenih primjera na neki način pomogli takvim algoritmima da pronađu bolje modele. Primjer takvog učenja je prepoznavanje govora kada bi čovjeku bilo vrlo nezgodno označiti obilježja prilikom preslušavanja svake audio datoteke.

3.3.3. Nenadgledano učenje

Kod nenadgledanog učenja poznajemo obilježja podataka, no od algoritma očekujemo da sam utvrdi ciljne vrijednosti. Prilikom toga koriste se klasteriranje i smanjenje dimenzionalnosti. Dobar primjer nenadgledanoga učenja korištenjem klasteriranja je grupiranje kupaca prema njihovim kupovnim navikama, gdje će kupce, ovisno o tome što su kupili, smjestiti u odgovarajuću skupinu. Primjerice one koji kupuju hranu za pse smjestit će u odvojenu grupu s obzirom na one koji će kupiti hranu za mačke. Isto tako će one koji kupuju hranu za životinje smjestiti u grupu vlasnika kućnih ljubimaca i one koji to nisu u posebnu grupu.

3.3.4. Podržano/ojačano učenje

Mohri i sur. (2018.) definirali su podržano učenje na sljedeći način: „učenik“ kako bi skupio informacije ima aktivnu interakciju s okolinom, a u nekim slučajevima i sam djeluje na okolinu te za svaku tu radnju dobije nagradu. Cilj „učenika“ je maksimizirati

nagradu tijekom izvršavanja radnji s okolinom. Ovaj tip učenja rješava specifične probleme gdje je donošenje odluka sekvencijalno, a cilj je dugoročan, poput igranja igara ili robotika.

4. Računalni vid

Većina ljudi podarena je s pet osjetila kojima doživljavamo okolinu. Naravno radi se o dodiru, njuhu, okusu, sluhu i vidu. Osjetilo koje se među njima možda najviše ističe je osjetilo vida s obzirom da se na njega najčešće i oslanjamo u raznim situacijama. Korištenje vida vrši se organom oka uz čiju pomoć razaznajemo boje, oblike, svjetlo i udaljenosti. Vidom ljudski mozak prima enormnu količinu informacija te je zanimljivo kako zapravo ljudski mozak s velikom lakoćom interpretira sve dobivene informacije. Tako nam ne treba dugo da shvatimo gledamo li u psa, kuću ili slično. Razvojem tehnologija stvorila se želja i potreba da i računala „progledaju“, međutim tu dolazi do velikih izazova s obzirom da je ljudski vid rezultat evolucije i to ne samo oka već i mozga koji procesira sve što dolazi iz oka. Mozak sa svojim neuronima djeluje poput kompleksnog super-računala s ogromnim brojem mikroprocesora koji čine vid takvim kakav jest te na svijetu ne postoji elektronsko računalo koje bi ga moglo nadmašiti. Svakim danom nastoji se omogućiti računalima da zamijene čovjeka u raznovrsnim zadacima i to uglavnom s uspjehom, pa će tako računalo izvršiti većinu repetitivnih zadataka, dok za neke kompleksnije zadatke koji zahtijevaju interakciju s okolinom, potrebno mu je omogućiti da vidi. Mnogo je izazova na putu za ostvarivanje navedenog. Slika 1 na lijevo prikazuje oko što je ljudski organ vida, dok na desno je strojni „organ“ vida tj. kamera.



Slika 1: Ljudski i strojni organi vida

Peters (2017.) je iznio definiciju glavne zadaće računalnog vida, a to jest interpretirati i rekonstruirati prirodne scene na osnovu sadržaja fotografija koje su napravljene uz pomoć digitalnih fotoaparata. Prirodna scena se karakterizira kao dio vizualnog polja koje je ulovljeno ljudskim okom ili optičkim sensorima. Kad govorimo o optičkim sensorima, prirodna scena može biti jedna digitalna fotografija uhvaćena fotoaparatom ili videozapis ulovljen video kamerom. Sadržaj odnosno značajke svake slike su pikseli, rubovi, kutovi, geometrija slike, oblici, teksture i boje.

Russell i Norvig (2010.) istaknuli su tri pristupa problemu računalnog vida, a to su: izvlačenje značajki koje se oslanja na jednostavne proračune koji se primjenjuju na zapažano, pristup prepoznavanjem gdje agent na osnovu vizualnih i drugih podataka pravi razliku među objektima koje sreće (primjer je prepoznavanje lica osoba) te pristup rekonstrukcijom gdje se stvara geometrijski model svijeta iz jedne ili seta slika.

Davies (2018.) je na vrlo jednostavnom primjeru predstavio koje su to poteškoće i rješenja kod implementacije računalnog vida, na primjeru prepoznavanja znakova. Ako se uzme da se svaki znak smatra 24 bitnim uzorkom i da je predstavljen nekom klasom, recimo da računalo nauči svaki uzorak i njegovu klasu napamet. Tada bi se svaki novi uzorak koji nema klasu klasificirao uspoređujući ga s već naučenim setom uzoraka te bi mu se dodijelila klasa najsličnijeg uzorka. No, ovdje vrlo lako dolazi do problema zbog velikog broja varijacija, te je moguće da će doći do krive klasifikacije uzorka ukoliko set podataka za treniranje nije dovoljno velik. Nekoliko je stvari koje mogu dovesti do toga. Primjerice ako testni uzorak ima nekakav višak ili iskrivljenje već u jednom bitu, kao i onda kada su bitovi pozicionirani ili je njihova orijentacija drukčija, u usporedbi s uzorcima koji već postoje u setu podataka za treniranje. Dakle, imamo jasan problem s nedovoljnom količinom podataka za treniranje kao i s time da i ti koji su prisutni, neće nužno moći predstaviti svaki uzorak koji se može pojaviti u praksi.

Kako bi doskočili problemu prepoznavanja potrebna je neka vrsta standardizacije slika. Potrebno je izvršiti normalizaciju položaja i orijentacije slike čime bismo umanjili broj mogućih kombinacija. Metode kojima se to može postići uključuju centralizaciju objekta tako da njegove centroide budu u središtu normalizirane slike i postavljanje njegovih glavnih osi okomito ili vertikalno. Sljedeće što možemo napraviti jest utvrditi red koji je prisutan na slici te tako možemo uočiti da se veoma mali broj uzoraka koji nas zanimaju

mogu raspoznati iz uzoraka nastalih postavljanjem nasumičnih točaka. Jednako tako sve vidljive izolirane točke koje su rezultat šuma mogu biti eliminirane ukoliko se radi o uzorku koji nije nasumičan. Ovdje su opisani neki od općenitih koraka kod standardizacije slike međutim svaki slučaj je različit te je potrebno pristupiti mu na jedinstven način.

4.1. Slika

Singh (2019.) je dao definiciju slike kao dvodimenzionalni vizualni prikaz objekta iz naše okoline s kojima se susrećemo svakodnevno, a to mogu biti životinje, osobe, razni predmeti ili bilo kakav drugi objekt. Slika je zapravo skup piksela koji mogu biti različitih boja i bilo koje boje. Pikel je dakle najmanji element svake digitalne slike, a potpuna slika dobije se udruživanjem velikog broja piksela. Svaka slika je predstavljena rezolucijom, a to je ukupni zbroj piksela koji su prisutni na slici. Što je veći broj piksela prisutan na slici, to će slika biti kvalitetnija. Primjer rezolucije je 800x600 što znači da će takva slika imati 480.000 piksela, što se dobije množenjem ta dva broja. Na slici 2 vidljivi su pikseli raznih boja na uvećanoj slici.



Slika 2: Pikseli

Računalni vid oslanja se na raspoznavanje značajki slike odnosno točaka, rubova, objekata, boja ili tekstura. Značajke slike mogu se podijeliti na globalne i lokalne metode. Neke od značajki poput boje, rubova i tekstura detaljnije su opisani u sljedećim potpoglavljima.

Globalne metode uzimaju značajke iz cijele slike bez da ju podijele na puno manje značajnih područja. U ovu grupu metoda svrstavaju se algoritmi bazirani na histogramima kao što je histogram orijentiranih gradijenata (eng. histogram of oriented gradients - HOG) ili koherencija vektora u boji (eng. colour coherence vector - CCV). Dobra strana ovih metoda je što konstantno stvaraju opisne podatke koje je kasnije

lakše pohraniti i uspoređivati, dok je loša strana ovog tipa metoda što korisniku daju samo nejasnu sličnost.

Lokalne metode temeljene na značajkama međutim pokušavaju pronaći značajna karakteristična polja slike uz pomoć Razlika Gausa (eng. Difference of Gaussian - DoG) ili Laplace od Gausa (eng. Laplacian of Gaussian - LoG) algoritmima. Lokalne metode su puno preciznije te mogu generirati veliku količinu opisnih podataka, ali ta količina ovisi puno od slike do slike. Ovaj tip metoda vrlo je efikasan kod raspoznavanja razlika na slikama, ali nije toliko učinkovit kod prepoznavanja sadržaja slike. Metode odnosno algoritmi ovog tipa koje se najčešće koriste jesu SIFT, SURF, ORB, BRIEF i FAST.

Neki od navedenih algoritama poput SIFT-a i algoritam RANSAC zbog svoje česte upotrebe opisani su i sljedećim potpoglavljima.

4.1.1. Boja

Ljudsko oko zaprima svjetlost koje varira u količini energije različitih valnih duljina, a što se predstavlja spektralnom funkcijom gustoće. Oko ima tri vrste stanica receptora za boju, te može raspoznati raspon od 380 do 750 nm duljine vala. Unutar toga raspona tri su ključne boje: crvena, zelena i plava, stoga se ljudi smatraju trikromatima (eng. trichromatic). Trikromatsko načelo (eng. principle of trichromacy) kaže da je za svaku spektralnu gustoću energije moguće stvoriti drugu koja je spoj tri boje (najčešće crvene, zelene i plave) takvu da ljudsko oko može raspoznati razliku među njima. To je i razlog zbog kojeg su i razni ekrani kojima se služimo na računalima, pametnim uređajima, televizorima napravljeni tako da emitiraju u te tri boje. Isto tako to je ono što algoritmima za računalni vid olakšava posao. Svaka površina tijela se može predstaviti s tri različita albeda, odnosno brojem koji naznačuje u kolikoj se mjeri svjetlost reflektira od površine tog tijela. Potom se uz pomoć Lambertovog kosinusnog zakona utvrđuju R/G/B (eng. kratica za red/green/blue ili na hrv. crveno/zeleno/plava) vrijednosti svakog piksela.

4.1.2. Rubovi i njihova detekcija

Rubovi su linije različitih oblika (ravne ili zakrivljene) na slici, gdje nastaju značajne promjene u svjetlini slike. Rubovi u računalnom vidu su ekstremno bitni jer omogućuju raspoznavanje oblika objekta što je bitno kod klasifikacije slike. Rubove je moguće detektirati kod dviju različitih površina na slici, što znači da se pojavljuju kada se na slici promijeni intenzitet svjetlosti. Sam proces detekcije rubova je baziran na raspoznavanju značajnih razlika na slici.

Scherer (2020.) opisuje četiri tipa rubova:

1. Rampa – vrijednost ruba raste postepeno, a ovaj tip moguće je pronaći na fotografijama nakon što se fotografija zamuti (eng. blur);
2. Korak – ovo je najpoželjniji tip ruba, a do njega dolazi kada se vrijednost intenziteta iznenada promijeni iz jedne krajnosti u drugu. Najčešće se pojavljuju na računalnim slikama;
3. Linija – profil ruba sastoji se od dva vrlo blizu postavljena ruba, njih se može detektirati prvom derivacijom i otporni su na sve tehnike zamagljivanja;
4. Krov – ovaj tip pronalazi se na fotografijama.

Slika 3 prikazuje primjenu detekcije ruba na slici daljinskog upravljača. Na lijevoj polovici slike bijelim linijama označen je detektirani rub.



Slika 3: Slika daljinskog upravljača s primjenom detekcije ruba

4.1.3. Tekstura

Teksturu se može opisati kao vanjski izgled ili vizualni osjećaj neke površine. Ona može biti mat i/ili sjajna te isto tako glatka i/ili hrapava, a već samo gledajući u nju imamo osjećaj kao da ju dodirujemo jer smo poučeni prošlim iskustvom kada smo takvu površinu osjetili drugim osjetilom odnosno dodirom. Kod računalnog vida tekstura je zapravo niz ponavljajućih uzoraka koji se pojavljuju na površini, a moguće ih je uočiti. Opća definicija koju je ponudio Davies (2018.) glasi da je tekstura u računalnom vidu zapravo karakteristična varijacija u intenzitetu dijela slike koja nam omogućava da ju prepoznamo, opišemo i definiramo njezine granice. To su npr. domine poslagane u neku formu, valovi na površini mora ili lišće na stablima. Tekstura se najčešće dobije od manjih komponenti ili predmeta koji su po strukturi slični, ali su zajedno posloženi na različite načine. Tekstura se promatra na većem uzorku piksela za razliku od npr. svjetline koja je karakteristika pojedinog piksela. Na slici 4 prikazane su dvije teksture. Na lijevoj slici je prikazana tekstura deke ovaj tip teksture raspoznaje ljudsko oko, a na desnoj bademi koji čine teksturu kakvu raspoznaje računalni vid.



Slika 4: Primjer tekstura

4.1.4. SIFT algoritam

Transformacija obilježja razmjera (eng. Scale-Invariant Feature Transform) je algoritam koji služi za detekciju i opisivanje lokalnih značajki slike u računalnom vidu. Princip rada ovog algoritma je da za svaku ključnu točku koja opisuje lokalne značajke slike, stvaramo vektor značajki koji se kasnije koristi za daljnje procesiranje slike. Ono što odlikuje ovaj algoritam je to što nema problema sa slikama kojima je promijenjen razmjer, rotacija ili promjena u svjetlini. Kao primjer toga možemo uzeti dvije slike. Na jednoj slici imamo recimo sliku šume, dok druga slika ima uvećan prizor vjeverice na jednom od stabala u šumi. Uzmimo da je druga slika s vjevericom dodatno zarotirana za 90 stupnjeva. Ovaj algoritam neće imati nikakav problem u pronalaženju sličnosti na slici. Značajke koje algoritam pokušava razlučiti tijekom procesuiranja slike jesu: razmjer, rotacija, osvjetljenje i perspektiva.

SIFT algoritam se izvršava u četiri glavna koraka:

1. Kreacija prostora – dohvaćaju se potencijalne ključne točke pregledavanjem kompletne slike. Uzima se originalna slika nad kojom se vrši gausovo zamućivanje (eng. gaussian blur) kako bi se uklonile nebitne točke i dodatan šum koji je prisutan na slici. U ovom koraku slici se više puta mijenja razmjer (povećava se ili smanjuje) te se ponavlja prijašnji proces. Tu se vrši i traženje razlika između slika uz pomoć razlika Gausa;
2. Traženje ključnih točaka i odbacivanje onih koje to nisu – u ovom koraku pronalaze se ključne točke koje su otporne na razmjer i rotaciju slike. Uz pomoć razlike Gausa na slici pronalazimo lokalne maksimume i minimume. Uzimamo svaki pojedinačni piksel te analiziramo njegove susjede. Piksel će biti označen kao ključna točka ukoliko ima najveći maksimum ili najmanji minimum među svojim susjedima. Potom isto to radimo za pod-piksele koje pronalazimo uz pomoć Taylorove ekspanzije. Moguće je za ključne točke izabrati samo kutove koji su vidljivi na slici s obzirom da se oni smatraju najboljim ključnim točkama. Kod odbacivanja ključnih točaka koje nisu važne, svaki piksel se uspoređuje sa zadanom graničnom vrijednošću intenziteta, a ako je manji od te vrijednosti takav piksel se odbacuje i smatramo ga nebitnim;

3. Određivanje orijentacije – traži se orijentacija ključnih točaka koja je otporna na transformaciju slike. Svaka ključna točka mora sadržavati informaciju o tome u kojem smjeru je okrenuta (orijentacija), a gledaju se orijentacije od svih susjeda ključne točke kao i nje same te se ključnoj točki dodjeljuje smjer koji dominira. Za dobivanje toga služimo se histogramima;
4. Generiranje deskriptora ključne točke – Generiraju se vektori koji opisuju ključne točke. Svaki deskriptor ključne točke sačinjen je od dva vektora. Prvi od njih sadrži poziciju ključne točke (x , y koordinate), razmjerni odgovor otkrivene značajke tj. snagu, orijentaciju i Laplaciana, dok drugi vektor sadrži deskriptor.

5. Primjena strojnog učenja u računalnom vidu

Strojno učenje implementira nekolicinu algoritama za računalni vid. U ovom poglavlju opisane su osnove klasifikacije kao što je prepoznavanje i klasifikacija uzoraka. To nas dovodi do modernijeg pristupa tim osnovama u obliku strojnog učenja koje pruža probabilističke metode za interpretaciju uzoraka.

5.1. Klasifikacija

Raspoznavanje uzoraka je za čovjeka već rutinska radnja koju može izvršiti uz minimalan trud, analizom i raspoznavanjem oblika. U usporedbi s time, statističko prepoznavanje uzoraka obrađuje skupove ekstrahiranih značajki koji se koriste za klasifikaciju objekata na statističkoj razini, najčešće matematičkom sličnošću skupova značajki objekata s klasama koje su nam već poznate. Sljedeća potpoglavlja bavit će se NN algoritmom koji je jedna od metoda statističkog prepoznavanja uzoraka, Bayes'ovom teorijom i SVM pristupom.

5.1.1. NN algoritam

NN je eng. kratica za „The Nearest Neighbor“ što bi na hrvatskom jeziku značilo najbliži susjed, dakle riječ je o algoritmu najbližeg susjeda. Prema Daviesu (2018.) načelo NN algoritma je uspoređivanje uzoraka ulazne slike s postojećim primjerima te potom klasifikacijom istih u klasu primjera koji se pokazao kao najbližiji. Kod ovog algoritma moguće je naići na dva osnovna problema. Prvi je da su trening uzorci različitih klasa previše blizu tj. razlikuju se u premalo dijelova da bi ih se moglo razlikovati, a drugi je da minimalni šum, translacija ili rotacija uzrokuju varijacije koje otežavaju prepoznavanje. Drugi problem nastaje uglavnom zbog premalog uzorka trening podataka, tj. jer nije moguće pronaći svaku moguću klasu koja bi se mogla pojaviti u testnim podacima, dok prvi problem nalaže da je moguće imati uzorke dviju različitih klasa koji će u nekim slučajevima biti slični u toj mjeri da niti jedan algoritam neće biti

u stanju ispravno klasificirati. NN algoritam u mogućnosti je klasificirati različite tipove podataka, međutim kako bi ispravno klasificirao potreban je ogroman skup trening podataka. No, to iziskuje veliku količinu računalne snage i memorije.

5.1.2. Bayes'-ova teorija odlučivanja

Bayes'ova teorija odlučivanja bazira se na korištenju pojedinih značajki podataka i apriorna vjerojatnost da će se određeni uzorak pojaviti. Bayes'-ovo pravilo glasi da je za pronaći klasu nekog objekta potrebno poznavati dva seta informacija o objektu: osnovnu vjerojatnost da će se pojaviti neka klasa i distribucija vrijednosti značajke za svaku klasu. Setove podataka koji su nam potrebni moguće je pronaći promatrajući slijed objekata, te se takav slijed objekata naziva trening podacima. Sve najčešće tehnike analize slike daju nam značajke koje se kasnije mogu koristiti u klasifikaciji i prepoznavanju objekata. Klasifikacija se kod ovog pristupa može bitno poboljšati ukoliko se simultano koristi više značajki što uvelike smanjuje potencijalno krivo klasificiranje. Međutim uvođenjem prevelikog broja značajki pogoršavamo situaciju. Isto kao kod NN algoritma ovdje je nekad potrebna velika količina računalne snage i memorije kako bi što rjeđe došlo do pogrešne klasifikacije.

Varijacija na ovu Bayes'ovu klasifikaciju je Naivni Bayes'-ov klasifikator. Razlika je u tome da je ovaj tip klasifikatora u mogućnosti kod puno slučajeva smanjiti potrebu za velikim količinama računalne snage. Ovaj klasifikator se može primijeniti u raznim aplikacijama gdje je moguće odrediti značajke koje su u nekoj mjeri samostalne. To npr. mogu biti boja ili veličina objekta.

5.2. Klasifikacija slika strojnim učenjem

Puno je algoritama strojnog učenja koji se koriste za klasifikaciju slika no ovo poglavlje baviti će se onim najpoznatijima, a to su: k-NN, logistička regresija, stablo odlučivanja i SVM. U sljedećim potpoglavljima biti će opisane njihove osnove.

5.2.1. k-NN

Postoji dodatna varijacija NN algoritma zvana k-NN klasifikator koja uzima „k“ najbližih set uzoraka za treniranje. Ovo je najjednostavniji algoritam za strojno učenje i klasifikaciju slika. Ovaj se algoritam oslanja na udaljenost među vektorima značajki što bi u našem slučaju kada govorimo o klasifikaciji slika značilo intenzitet RGB piksela na slici. K-NN algoritam je napravljen tako da nije učen, već klasificira nepoznate objekte tako da pronade najsličniji njemu među k najbližih primjera. Dakle svaka klasa koja je među k klasama je potencijalna klasa novog objekta.

5.2.2. Logistička regresija

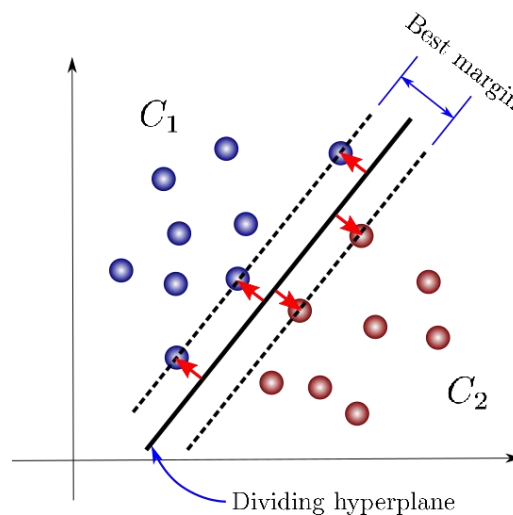
Logistička regresija je izmijenjena forma linearne regresije u kojoj koristimo logite kako bi utvrdili vjerojatnost da se određeni objekt pojavi u određenoj klasi. Rezultat je u rangu između 0 i 1. Ukoliko je element iznad 0,5 tada pripada prvoj klasi, u suprotnom drugoj. Linearna regresija je pristup modeliranju relacija jedne ili više varijabli označene sa Y, te jedne ili više varijabli označene sa X, na način da takav model linearno ovisi o nepoznatim parametrima estimiranih iz podataka (definicija preuzeta s Wikipedije), dok je Logit funkcija logaritam izgleda $\frac{p}{p-1} - p$ je vjerojatnost).

5.2.3. Stablo odlučivanja

Stablo odlučivanja bazira se na principu koji je vrlo sličan onome kada kod donošenja odluka tražimo prednosti i nedostatke. Stablo odlučivanja oslanja se na statističku graničnu vrijednost koja nam za određeni objekt govori pripada li objekt u prvu ili drugu klasu. Ovo možemo iskoristiti ukoliko recimo želimo ustanoviti za nekog psa je li rasni ili mješanac. Prvo uzmemo u obzir boju psa, potom možemo promatrati veličinu, dužinu njuške kao i oblik repa. Zatim iz tih vrijednosti stvaramo stablo odlučivanja koje će nam pomoći kod donošenja odluke u koju kategoriju pas spada.

5.2.4. SVM – Support Vector Machine

Support vector machine ili na hrvatskom metoda potpornih vektora (u daljnjem tekstu SVM). Glavna ideja ove metode je pronaći par paralelnih hiper-ravnina koje maksimalno razdvajaju dvije klase značajki kako bi se umanjila mogućnost greške. SVM stvara ravninu između dviju klasa kojom pokušava maksimizirati udaljenost od suprotstavljenih klasa, tj. hiper-ravninu koja ima najveću marginu razdvajanja klasa. Slika 5 prikazuje primjer metode potpornih vektora.



Slika 5: Metoda potpornih vektora

6. Duboko učenje

Duboko učenje ili eng. deep learning je potpodručje strojnog učenja, što znači da ujedno spada u područje umjetne inteligencije. Cilj umjetne inteligencije je, kako je već opisano u drugom poglavlju, pružiti skup tehnika i algoritama koji se koriste kod rješavanja problema koje čovjek već izvršava automatizmom, a računalu je isto vrlo teško. Primjer toga je i tema ovoga rada, a to je prepoznavanje i razumijevanje onoga što se nalazi na slikama. Dakle, za nas ljude, zadatak prepoznavanja objekata, okruženja i drugog na slikama je zadatak koji iziskuje minimalan trud i minimalnu količinu energije, ali za računalo je taj zadatak iznimno zahtjevan. Polje dubokog učenja specijalizirano je za raspoznavanje uzoraka i učenje iz podataka.

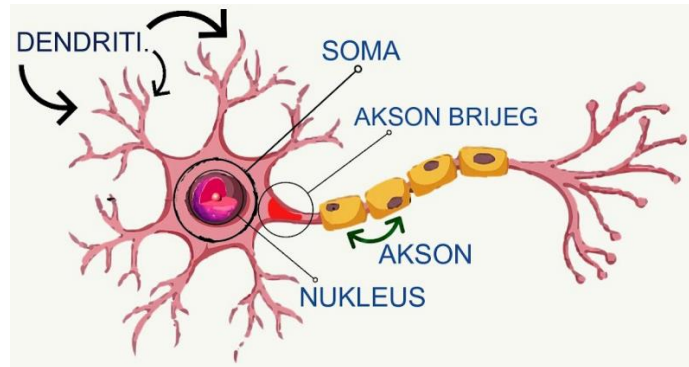
Duboko učenje spada u obitelj algoritama naziva „Umjetna neuronska mreža“, za što je engleska kratica ANN (eng. Artificial neural network) koja će biti korištena u daljnjem tekstu. ANN algoritmi napravljeni su tako da oponašaju strukturu i funkcioniranje mozga. Ovaj tip algoritma spada u klasu algoritama strojnog učenja te kao takav uči iz podataka i bavi se raspoznavanjem uzoraka.

U prethodnim poglavljima bili su opisani algoritmi i metode kojima je bilo potrebno „ručno“ odrediti koje će se značajke slike, poput boje, teksture i drugih, kvantificirati. Odredili bismo piksele na slici na kojima bismo upotrijebili određeni algoritam, uz pomoć kojeg bi dobili vektor značajki koji bi opisao sadržaj slike, te bi na kraju taj vektor koristili kao input za naš model strojnog učenja. Duboko učenje doskočilo je tome.

6.1. ANN

Neuronske mreže su građevni blokovi sustava dubokog učenja. Nastale su po uzoru na ljudski živčani sustav, odnosno po uzoru na neurone u mozgu. Neuronska je pridjev riječi neuron, dok mreža predstavlja strukturu nalik grafu. Zaključujemo stoga da je neuronska mreža računalni sustav koji nastoji oponašati neuronsku povezanost u našem živčanom sustavu. Kako bi sustav bio smatran neuronskom mrežom mora posjedovati označeni usmjereni graf (ovaj tip grafa sastoji se od skupa čvorova i skupa veza koje te čvorove povezuju u parove) u kojem svaki čvor izvršava neku jednostavnu

operaciju. Neuronska mreža sastoji se od više slojeva: ulaznog sloja, skrivenih slojeva i izlaznog sloja. Na slici 6 prikazan je izgled i sastavni dijelovi neurona.



Slika 6: Neuron

6.2. CNN (eng. Convolutional neural network)

CNN ili konvolucijska neuronska mreža je nadogradnja na ANN mrežu. Za razliku od dosadašnjih pristupa kod kojih se ručno izabiru pravila i algoritmi za dohvat značajki iz slike, prema Rosebrocku (2017.) CNN mreža te značajke izučava automatski tijekom procesa treniranja. Korištenjem CNN-a pokušavamo shvatiti problem kao koncept hijerarhije, jer se svaki koncept nadograđuje jedan na drugog i kreiraju se slojevi koncepata. Niži slojevi sadržavat će osnovnu reprezentaciju problema, dok će oni viši slojevi koristiti niže kako bi stvorili apstraktnije koncepte. Zbog toga smo u mogućnosti u potpunosti ukloniti proces u kojem se „ručno“ izvlače značajke i imati, u vidu CNN-a, sredstvo koje uči od početka do kraja. Ono što najviše odlikuje duboko učenje, pa tako i CNN, je to što omogućuju preskakanje koraka u kojemu se dohvaćaju značajke i baš zbog toga su u mogućnosti bolje se fokusirati na proces treniranja naše mreže da nauči sama.

6.1.1. Kako CNN funkcionira?

CNN funkcionira na način da za priloženu sliku, mreži dajemo vrijednost intenziteta piksela. CNN u sebi koristi skrivene slojeve koji se koriste za dohvatajki značajki iz slike, a ti slojevi stvaraju hijerarhijsku strukturu. Prvo će biti otkrivena područja koja nalikuju rubovima ili jesu rubovi te će biti smješteni u niže slojeve mreže, zatim će se ta područja iskoristiti za definiranje kontura (obrub objekata) i kutova (mjesto gdje se sijeku rubovi). Kombiniranjem kontura i kutova dobit ćemo apstraktne dijelove objekta koji će biti smješteni u sljedeći sloj. Za kraj imamo izlazni sloj, na kojeg na neki način utječu svi slojevi naše mreže, klasificirat će našu sliku i dodijeliti joj klasu. Ukratko, naša mreža koristi hijerarhijsko učenje, a svaki sloj naše mreže automatski uči korištenjem izlaznog podatka prethodnog sloja kako bi izgradili puno apstraktnije koncepte.

6.3. Koraci za izgradnju klasifikatora slike dubokim učenjem

Rosebrock (2017.) je definirao četiri koraka koji su potrebni kako bi se izgradio klasifikator slika baziran na dubokom učenju. Ti koraci su:

1. Prikupljanje skupa podataka – u prvom koraku potrebno je prikupiti početni skup podataka. Potrebne su nam slike, te ih je ujedno potrebno i označiti s klasom kojoj pripadaju. Te klase moraju biti dio konačnog skupa kategorija, a one mogu biti npr. automobil, motocikl, romobil. Svaka od kategorija mora imati približno jednak broj slika, jer ako neka kategorija ima značajno više slika, naš klasifikator će postati pristran tim prekomjerno predstavljenim kategorijama. Naziv toga je klasna neuravnoteženost i to je jedan od češćih problema u strojnom učenju.;
2. Razdvajanje skupa podataka – kada smo stvorili skup podataka sljedeća zadaća je razdvojiti ga u dva dijela: u set za treniranje i set za testiranje. Training set nam služi kako bi klasifikator mogao naučiti kako svaka od kategorija izgleda tako što radi predikcije nad ulaznim podacima te se onda ispravlja kada predikcije nisu bile točne. Kada je klasifikator istreniran, možemo započeti s njegovom evaluacijom uz pomoć set za treniranje. U ovom koraku vrlo je važno da se training i test setovi podataka ne preklapaju, jer bi na taj način klasifikator već imao uvid u testne primjerke i naučio bi iz njih, na neki način bi varao. Potrebno je održati podatke za testiranje potpuno neovisnima od procesa treniranja te ih isključivo koristiti za evaluaciju mreže. Najčešće korištene

podjele podataka jesu (u formatu trening, pa test): 66,7%/33,3%, 75%/25% i 90%/10%;

3. Treniranje mreže – kada smo formirali naš trening set slika možemo započeti s treniranjem naše mreže. U ovom koraku naša mreža ima za cilj naučiti kako prepoznati pojedinu zadanu kategoriju među našim označenim podacima i svaki puta kada model bude napravio pogrešku iz nje će nešto naučiti te se tako poboljšati.;
4. Evaluacija – posljednji korak je evaluacija istrenirane mreže. Mreži predstavljamo slike iz seta za testiranje i očekujemo od nje da predvidi kojoj klasi ta slika pripada, te tablično spremamo sve predviđene klase testnog seta podataka. Na samom kraju dobiveni model s predviđenim oznakama klasa uspoređujemo s pravim oznakama iz našeg seta za testiranje, dakle onim oznakama koje govore kojoj kategoriji slike stvarno pripadaju. Iz toga izvlačimo brojke predviđanja koje je klasifikator uspješno predvidio i stvaramo evaluacijske mjere kao što su preciznost, odziv i f-mjera, a koje kasnije koristimo za procjenu performansa naše mreže.

7. Detekcija objekata dubokim učenjem

Detekcija objekata je radnja u kojoj, kako i sam naziv govori, nastojimo ustanoviti objekte koje će ljudsko biće prepoznati u prirodi, dok će kod računala to biti nekakav objekt na slici i videu. Naravno kod ljudi je taj proces evolucijski uznapredovao u toj mjeri da će za njega čovjek potrošiti minimalno energije i odraditi ga bez puno razmišljanja. Računalo već ovdje ima dosta problema s obzirom da se radi o stroju koji zna samo za dvije vrijednosti, 0 i 1. Naš izazov je stoga naučiti računalo da detektira različite objekte na snimci koju mu predstavimo. Sama detekcija objekata vrlo je važno istraživačko područje u procesiranju slika i računalnom vidu. Detekcija objekata može se primijeniti na razna područja u računalnom vidu, npr. robotika ili video nadzor. S pojavom tehnologije dubokih mreža taj proces se uvelike unaprijedio. Metode koje se najčešće koriste su metode konvolucijskih neuronskih mreža (CNN). CNN metode mogu se podijeliti u dvije osnovne klase: jednostupanjske (eng. one-stage) i dvostupanjske (eng. two-stage) metode.

Detekcija objekata je velik i težak izazov s obzirom na brojne faktore poput varijacija u osvjetljenju ili razmjeru, kompleksnosti pozadine, deformacija objekta i mnogi drugi. Zbog toga se već desetljećima ustrajno radi na poboljšanju tog procesa. Danas je moguće metode detekcije objekata podijeliti u dvije klase:

1. Ručne metode zasnovane na značajkama;
2. Metode zasnovane na dubokom učenju.

Ručne metode zasnovane na značajkama bile su vrlo popularne početkom 2000.-ih godina kada su i nastali najpoznatiji deskriptori značajki slika poput SIFT-a, te klasični klasifikatori poput SVM-a, a koji su bili detaljnije razrađeni u četvrtom i petom poglavlju.

7.1. Jednostupanjske i dvostupanjske metode

Kako je ranije navedeno, CNN metode dijele su u dvije klase: jednostupanjske i dvostupanjske metode. Način funkcioniranja dvostupanjskih metoda je taj da prvo generiraju objekte koje zatim klasificiraju u kategorije, dok kod jednostupanjskih, stvari

funkcioniraju tako da se istovremeno generiraju i klasificiraju svi objekti. Razlika u performansama je u brzini detekcije gdje jednostupanjske uvelike prednjače, dok, kada se govori o preciznosti, detekcije ostvaruju otprilike jednake rezultate. U nastavku su detaljnije opisane ove dvije klase metoda „duboke“ detekcije objekata.

7.1.1. Jednostupanjske metode

Jednostupanjske metode detekcije objekata imaju za cilj istovremeno pronaći objekt i predvidjeti u koju kategoriju on pripada. Ovaj tip metoda odlikuje velika brzina, ali i nešto manja preciznost detekcije. Najpoznatije metode ovog tipa su: OverFeat, SSD, YOLO i RetinaNet. Neki od njih detaljnije su opisani u daljnjem tekstu.

SSD (eng. Single-shot detector) ili na hrvatskom „Detektor jednim pucnjem“ je metoda koja služi za detekciju generičkih objekata. SSD omogućava da se samo jednim prolaskom kroz sliku detektira veći broj objekata na slici. SSD-ova mreža je izgrađena na VGG16 arhitekturi zbog njenih jakih performansi kod klasifikacija slika i zbog popularnosti kod rješavanja problema kod kojih učenje pomaže u dobivanju boljih rezultata. Uz osnovnu VGG mrežu generira se nekoliko dodatnih konvolucijskih slojeva različitih rezolucija, a nakon toga koristit će veći broj konvolucijskih slojeva različitih rezolucija kako bi predvidio objekte različitih mjera. U slučaju zadnjeg sloja dane rezolucije to bi značilo da prvo koristi 3 x 3 konvolucijski filter kako bi generirao novu mapu značajki i na osnovu toga predviđa kategoriju objekta kao i lokaciju objekta na novoj mapi značajki.

Yolo (eng. You only look once) ili na hrvatskom „Samo jednom pogledaš“ je metoda koja izvršava više procesa samo jednim prolaskom kroz konvolucijsku mrežu. Prilikom tog jednog prolaska bilježi svaki mogući granični okvir (eng. bounding box), postepeno radi procjenu svake mogućnosti i tako sve dok ne prođe mrežom do kraja kada je već došao do skupa dosljednih odluka. Yolo se pokazao kao vrlo uspješna metoda kod detekcije objekata koji se nalaze u pozadini slike s vrlo malim brojem pogrešnih detekcija, međutim ima i svoje nedostatke poput netočne lokalizacije objekata. Arhitektura Yoloa je takva da sadrži 24 konvolucijska sloja koje slijede dva potpuno spojena sloja i završava sa završnim slojem veličine 7 x 7 x 30. Ova metoda funkcionira

na način da podijeli ulaznu sliku na područja za koja će potom vršiti predviđanja graničnih okvira i vjerojatnosti da se objekt u njima nalazi. YOLO dijeli sliku u $k \times k$ koordinatnu mrežu jednakih ćelija gdje svaka ćelija predviđa B graničnih okvira s ocjenama pouzdanosti i vjerojatnošću c uvjetne klase. Svaki granični okvir ima predviđanja u obliku (x, y, w, h, s) gdje su (x, y, w, h) lokacija graničnog okvira, a „ s “ je pouzdanost da se objekt nalazi u okviru.

Metoda RetinaNet bazirana je na FPN mreži (eng. Feature pyramid network). FPN mreža kombinira semantički jake značajke niske rezolucije s onima semantički slabim značajkama koje su znatno veće rezolucije top-down pristupom i bočnim vezama. FPN posjeduje bogatu semantiku na svim razinama te je sagrađen od samo jedne ulazne slike fiksnog razmjera. RetinaNet radi tako da za izvršenje klasifikacije i regresiju okvira, četiri 3×3 konvolucijska sloja vežu na izlazni sloj FPN-a, nakon toga 3×3 slojevima s $K \times C$ filterima koriste se za klasifikaciju okvira te 3×3 slojevi s $4 \times K$ filterima za regresiju okvira. Jedna pozicija svakog sloja posjeduje K sidra i C klasa. Sidro će biti upareno s klasom ukoliko ima najveće preklapanje IoU sa stvarnom situacijom odnosno kada je IoU preklapanje veće od 0,5. Ukoliko je IoU preklapanje manje od 0,4 sidro će biti dodijeljeno pozadini. IoU (eng. Intersection over union) je mjera preklapanja detektiranog graničnog okvira sa stvarnom lokacijom graničnog okvira.

7.1.2. Dvostupanjske metode

Dvostupanjske metode izvršavaju proces detekcije kroz više faza. Metodi se daje slika za koju se prvo izdvajaju prijedlozi mogućih objekata, a nakon toga se ti prijedlozi klasificiraju u specifične kategorije objekata uz pomoć istreniranog klasifikatora. Prednost ovih metoda su: smanjen broj prijedloga koji se potom predaju klasifikatoru i poboljšana preciznost detekcije s obzirom da kod prijedloga mogućih objekata klasifikator izvršava klasifikaciju s minimalnim utjecajem pozadine u fazi treniranja. Najpoznatiji primjer ovih metoda su: RCNN, SPPnet, Fast RCNN i Faster RCNN. RCNN je novonastala vrsta CNN-a čija je osnovna zadaća detektirati i locirati objekte na slikama. Izlazna vrijednost ove mreže je skup graničnih okvira koji se podudaraju sa svakim detektiranim objektom, a isto tako svakom detektiranom objektu dodjeljuju pripadajuću klasu.

7.2. Izazovi detekcije objekata

Iako je detekcija objekata puno napredovala tijekom zadnjih godina, i dalje postoje razni izazovi odnosno problemi koji na neki način koče njezin razvoj. Jiang et al. (2019.) predstavili su tri glavna tipična izazova koji će detaljnije biti opisani u nastavku.

7.2.1. Problem u varijaciji razmjera

Problem s razmjerima nastaje iz razloga što je kod stvaranja slike udaljenost raznih objekata od kamere različit kao što će i objekti na slici biti različitih dimenzija. Dakle zbog navedene varijacije dolazi do problema kod detekcije objekata. Postoje dva tipa rješenja za varijaciju razmjera:

1. Metode bazirane na slikovnoj piramidi (eng. Image pyramid-based methods) – ova skupina metoda prvo mijenja veličinu originalne slike u više različitih razmjera i onda uz pomoć istog detektora radi pojedinačnu detekciju nad slikama s promijenjenim razmjerima;
2. Metode bazirane na piramidi značajki (eng. Feature pyramid-based methods) – ova skupina metoda radi tako da prvo generira više mapi značajki različitih rezolucija na bazi ulazne slike te potom koristi različite mape značajki za detekciju objekata različitih razmjera.

Metode „duboke“ detekcije slika koriste slikovnu piramidu za detekciju objekata različitih razmjera, a u to spadaju: RCNN, Fast RCNN, Faster RCNN i SPPnet.

7.2.2. Problem zaklanjanja

Ovaj problem česta je pojava na slikama. Do njega može doći zbog nekakve sjene pa će slika na tom području biti zamračena, a isto tako do toga može doći ukoliko se neki objekt prepriječi te na taj način zakloni objekt promatranja. Ovaj problem dovodi do toga da je otežana detekcija objekata na slikama. Kao primjer tome, Jiang et al. (2019.)

navode kako su ulične slike koje se koriste kod detekcije pješaka u otprilike 70% slučajeva zaklonjene barem u jednom okviru.

7.2.3. Problem deformacije

Kod stvaranja slika često dolazi do deformacije objekata na slici. Najčešće se to događa jer je objekt, uzmimo za primjer čovjeka, u nekom pokretu, recimo da trči ili skače. Stoga dobar algoritam za detekciju objekata mora biti „imun“ na deformaciju objekata. Znanstvenici su kroz povijest na razne načine pokušavali doskočiti deformaciji.

8. Scenariji korištenja detekcije objekata

8.1. Korištenje kod detekcije prekršaja u prometu

S obzirom da policijski službenici prometne jedinice nisu u mogućnosti biti uvijek svugdje i na svakome mjestu, moguće je implementirati detekciju objekata strojnim učenjem u promet.

Kao primjer toga mogu biti javna parkirališta, gdje bi se nadzirala parkirna mjesta, a posebno ona predviđena za osobe s invaliditetom. Osobe s invaliditetom na vjetrobranu svojih vozila imaju naljepnicu s oznakom koju bi sustav vrlo lako moglo prepoznati. Ukoliko bi se vozilo bez takve oznake parkiralo na invalidsko mjesto, kamerom bi se slikala tablica koju bi strojno učenje pročitalo te bi na osnovu nje sustav poslao kaznu vlasniku nepropisno parkiranog vozila. Isto tako mogu se očitavati parkirne karte, bile one papirnate, kada bi promatrali kokpit automobila ili plaćene drugim metodama, kada bi prepoznajući tablice čitali u sustavu ima li zapisa za njih. S uvođenjem navedenog ne bi bilo potrebe za fizičkim prisustvom zaposlenika na parkiralištima. Slika 7 prikazuje naljepnicu koja označava da je vozilo osobe s invaliditetom.



Slika 7: Naljepnica za parkiranje vozila osoba s invaliditetom

U prometu se postavljanjem kamera na razna mjesta mogu očitati razne stvari. Moguće je prepoznati ukoliko vozač motornog vozila drži ruku s mobilnim uređajem na uhu, te na osnovu toga i tablice vozila, poslati kaznu za razgovaranje u vožnji na kućnu adresu.

Može se koristiti i za banalnu situaciju gdje vozilu nedostaje tablica ili je vozilo pokvareno svjetlo.

8.2. Korištenje u zdravstvenim ustanovama

Ovakav sustav se može pokazati korisnim i u zdravstvenim ustanovama. Pretpostavimo da je u tijeku epidemija ili pandemija nekog virusa, te je za ući u zdravstvenu ustanovu potrebno nošenje maske za lice kako ne bi zarazili druge. Scenarij je sljedeći: pri dolasku ispred ustanove na skener stavljamo zdravstvenu iskaznicu koju očitava sustav te strojnim učenjem iz teksta prepoznaje o kome se radi. Termalna kamera potom očitava temperaturu te radi sliku glave osobe. Zatim provjerava ima li ta osoba masku na licu i, ukoliko sustav prepozna masku na licu, a tjelesna temperatura je u granicama, sustav otvara vrata. U suprotnom slučaju sustav signalizira osoblju da na vratima imaju potencijalno rizičnu osobu.

8.3. Korištenje u poljoprivredi

U poljoprivredi se oduvijek nastoji olakšati branje plodova s obzirom da ista kultura uglavnom sazrijeva u isto vrijeme te je potreban veliki broj ljudi odjednom. No, kada bi imali robota koji bi samostalno prolazio kroz ogromna polja, s ugrađenom kamerom koja mu služi za snalaženje u prostoru i promatranje plodova, mogao bi se broj ljudi potrebnih za branje smanjiti u velikoj mjeri. Takav robot bi imao sustav sa strojnim učenjem koji bi prepoznavao plod te bi na osnovu boja i oblika mogao prepoznati je li spreman za branje te ga ujedno i ubrati. Isto tako uz pomoć prepoznavanja takav robot mogao bi prepoznati ukoliko je neka biljka u lošem stanju ili napadnuta od strane nekakve bolesti, a to bi mogao postići prepoznavanjem lista te analizom ima li kakvih nepravilnosti.

8.4. Korištenje u komercijalne svrhe

U trgovačkom centru trgovine bi mogle potencijalno imati velike koristi od nekakvog vida prepoznavanja objekata. Mogao bi biti ovakav scenarij: kamere snimaju na ulaznim vratima te prikupljaju podatke. Prvi korak bio bi prepoznavanje jedinstvenih kupaca. Time bi imali uvid u stvarnu brojku osoba koje su posjetile trgovački centar taj dan. Sustav bi pritom mogao dodatno zabilježiti npr. spol jedinstvenih osoba. Potom, kad smo već identificirali jedinstvenu osobu, na njoj možemo prepoznati odjeću koju nosi, odnosno detektirati znak marke odjeće te na osnovu toga prikupljamo podatke što ljudi najčešće nose. Na taj način trgovcima se može omogućiti da npr. stavljaju ciljane akcije na ono što ljudi manje kupuju ili na ono što se često nosi kako bi se dodatno prodalo. Osim toga, trgovci bi tako mogli istaknuti najčešće proizvode kako bi namamili posjetitelje da kupuju, a u najgorem slučaju, ono što se analizom utvrdi da se slabo nosi, prestane držati u ponudi. Isto tako moguće je analizirati boju odjeće i iz toga napraviti dodatne zaključke o trenutnim trendovima. Također, kao dodatna mogućnost, može se izmjeriti visina osobe i po tome procijeniti veličina odjeće koja se najčešće nosi. Sve je to korisno kako bi trgovci mogli naručivati dovoljne količine određenog broja s obzirom da uz današnje analize i dalje u trgovinama nekih veličina ima previše dok drugih uvijek manjka.

8.5. Korištenje u prirodi

Biolozima bi jedan sustav detekcije objekata bio vrlo koristan. Za primjer uzmimo slučaj gdje bi postavili kamere u more. Kamera bi detektirala sve organizme te ih popisivala, na taj način znanstvenici su u mogućnosti bez fizičkog sudjelovanja ili ručnog popisivanja nakon pregledavanja snimki imati u svakom trenutku uvid u podatke o bioraznolikosti lokacije od interesa.

Isto tako, postavljanjem ovog tipa sustava lovačka društva mogu detektirati broj divljači koji trenutno obitava na određenom lovnom području. Ako uzmemo za primjer divlje svinje koje su česta napast i čiji se broj nastoji regulirati, kada bi se detektirao povišen broj jedinki, sustav bi alarmirao lovce koji bi onda pravovremeno mogli poduzeti radnje.

9. Programska rješenja detekcije objekata

Razvojem umjetne inteligencije, a posebno strojnog učenja razvijeno je i puno alata koji pomažu u tom pogledu. Zanimljivost ovog područja je što je većina alata besplatno dostupna široj populaciji. Zato i ne čudi sve veći broj ljudi koji se bave ovim područjem. Detekcija objekata je proces koji uzima maha svakim danom sve više te je inkorporiran u našu svakodnevicu. U nastavku su opisani neki od alata.

9.1. Python

Python je programski jezik stvoren 1990. godine od strane Guida van Rossuma. Trenutno je jedan od najpopularnijih programskih jezika zbog svoje jednostavnosti i fleksibilnosti pogotovo zato jer omogućava programiranje u tri različita stila: aspektno orijentirano, objektno orijentirano i strukturalno. Radi se o jeziku visoke razine koji je opće namjene, te je otvorenog koda što ga čini besplatnim za korištenje i distribuiranje. S obzirom na njegovu dostupnost velika zajednica je okupljena oko njega te stoga postoji ogromna standardna biblioteka kao i veliki broj modula koji su dostupni za korištenje.

9.2. OpenCV

OpenCV engleska je kratica za „Open Source Computer Vision Library“ što bi u prijevodu na hrvatski bilo „Knjižnica za računalni vid otvorenog koda“.

Otvoreni kod ili softver otvorenog koda je naziv za softver čiji je izvorni kod dostupan na korištenje svim korisnicima pod „open source“ licencom bez ikakve naplate za korištenje. Korisnici su dobrodošli i ohrabreni da takav kod mijenjaju, poboljšavaju, a sve s ciljem stvaranja suradnje kako bi kod bio što dostupniji i jednostavniji za korištenje svima.

Na službenim stranicama, OpenCV opisan je kao knjižnica otvorenog koda za računalni vid i strojno učenje. Napravljen je kako bi se omogućila zajednička infrastruktura za aplikacije računalnog vida i kako bi se ubrzala uporaba strojne percepcije u komercijalnim proizvodima. Unutar knjižnice postoji više od 2500 optimiziranih algoritama što uključuje razne algoritme za računalni vid kao i strojno učenje. Njegova upotreba je rasprostranjena pa se među ostalim koristi za identifikaciju objekata, detekciju i prepoznavanje lica, praćenje objekata u pokretu, dobivanje 3D modela iz objekata, pronalaženje sličnih slika u bazi slika, uklanjanje crvenih očiju s fotografija napravljenih s bljeskalicom, koristi se za proširenu stvarnost i mnoge druge namjene. OpenCV okuplja zajednicu od preko 47 tisuća ljudi, a broj preuzimanja se procjenjuje na više od 18 milijuna. Koriste ga velike kompanije (Microsoft, Google, Intel, IBM, Sony i druge), startup-ovi (kompanije s ciljem brzog širenja i rasta, koje imaju neku inovativnu ideju), razne istraživačke skupine i državna tijela. Neki od primjera velikih projekata gdje se koristi su: detekcija utapanja na bazenima u Europi, nadzor opreme za rudarenje u Kini, za interaktivnu umjetnost u Španjolskoj i New York-u, pregledavanje etiketa na proizvodima u tvornicama svugdje u svijetu i brza detekcija lica u Japanu. Napisan je u programskom jeziku C++, a ima sučelja za C++, Python, Java-u i MATLAB. Ima podršku za širok spektar operacijskih sustava, a to su Windows-i, Linux, Android i Mac OS.

9.3. TensorFlow i Keras

Prema opisu sa službenih mrežnih stranica, TensorFlow je platforma za strojno učenje otvorenog koda koja omogućava jednostavnu izgradnju i postavljanje modela strojnog učenja. Predstavljen je kao kompletni eko-sustav koji pomaže u rješavanju izazovnih stvarnih problema u svijetu sa strojnim učenjem.

TensorFlow nudi više stupnjeva apstrakcije iz kojih je moguće izabrati onaj koji nam treba. Omogućava građenje i treniranje modela koristeći Keras API-a koji omogućava lako korištenje samog TensorFlowa i strojnog učenja. Moguće ga je koristiti iz raznih programskih jezika i platformi.

Keras je najkorišteniji framework za duboko učenje, izgrađen je na TensorFlow 2.0 platformi zbog čega ga je moguće koristiti na više platformi, a napisan je u Pythonu. Keras je pristupačan, visoko-produktivan interface za rješavanje problema u strojnom učenju s fokusom na duboko učenje. On omogućava osnovnu apstrakciju i građevne blokove za razvoj i implementaciju raznih programskih rješenja strojnog učenja.

10. ZAKLJUČAK

Čovjeku je priroda podarila vid uz pomoć kojega može promatrati svoju okolinu i prepoznavati ono što nas okružuje bez prevelikog tjelesnog napora. Razvojem umjetne inteligencije, a posebno strojnog učenja, pojavila se potreba da i strojevi „progledaju“.

Računalni vid vrlo je važno područje strojnog učenja, a ujedno i umjetne inteligencije. Omogućavanjem vida strojevima pridonijeli smo razvoju novih mogućnosti koje nam strojevi pružaju. Možemo konstruirati robote koji više neće biti ograničeni na automatizirane radnje, već će se moći prilagoditi s obzirom na okolinu koja ih okružuje. Isto tako omogućen je veliki napredak u svijetu biznisa gdje se uz pomoć kamera mogu analizirati razne situacije u kojima se nalazimo. Primjer toga je kupnja u trgovačkom centru gdje se uz pomoć prepoznavanja osoba i njihovih određenih karakteristika prodajni centri mogu prilagoditi na način da će kupcima ponuditi točno ono što najviše preferiraju. Uz navedeni primjer, u stvarnom životu nalazimo i mnogo drugih situacija u kojima računalni vid olakšava rad ljudima te poboljšava njihovu efikasnost.

Kroz rad su opisane razne metode detekcije objekata, od onih „ručnih“ koje su već u jednu ruku zastarjele, do onih novijih poput dubokog učenja. U duboko učenje spadaju i neuronske mreže koje su zadnji „krik“ tehnologije te se pokazuju kao dobar način detekcije objekata s obzirom na njihovu brzinu i preciznost detekcije.

Za očekivati je da će umjetna inteligencija, a pogotovo strojno učenje, nastaviti rasti i prosperirati u znanstvenim krugovima stoga će biti vrlo zanimljivo vidjeti što će nam u predstojećim godinama ovo područje iznjedriti.

LITERATURA

1. RUSSELL, S. J: i NORVIG, P. (2010.) Artificial Intelligence, A Modern Approach. Third edition. New Jersey: Prentice Hall.
2. BURKOV, A. (2019) The Hundred-Page Machine Learning Book
3. COPPIN, B. (2004.) Artificial Intelligence Illuminated. 1st ed. Sudbury: Jones and Bartlett Publishers.
4. ERTEL, W. (2011.) Introduction to Artificial Intelligence. London: Springer-Verlag.
5. SHAEV-SCHWARTZ, S. i BEN-DAVID, S. (2014.) Understanding Machine Learning. New York: Cambridge University Press.
6. MOHRI, M., ROSTAMIZADEH, A. i Talwalkar, A. (2018.) Foundations of Machine Learning. Second edition. Cambridge: The MIT Press.
7. JIANG, X. et al. (2019.) Deep Learning in Object Detection and Recognition. Singapore: Springer Nature Singapore Pte Ltd.
8. ROSEBROCK, A. (2017.) Deep Learning for Computer Vision with Python. Starter Bundle. 1st Edition. PyImageSearch
9. CYGANEK, B. (2013.) Object Detection and Recognition in Digital Images. West Sussex: John Wiley & Sons, Ltd.
10. SCHERER, R. (2020.) Computer Vision Methods for Fast Image Classification and Retrieval. Cham: Springer Nature Switzerland AG.
11. DAVIES, E. R. (2018.) Computer Vision. Principles, Algorithms, Applications, Learning. Fifth Edition. London: Elsevier Inc.
12. SINGH, H. (2019.) Practical Machine Learning and Image Processing. For Facial Recognition, Object Detection and Pattern Recognition Using Python. Apress.
13. PETERS, J. F. (2017.) Foundations of Computer Vision. Computational Geometry, Visual Image, Structures and Object Shape Detection. Cham: Springer International Publishing AG.
14. DADA, E. G. et al. (2019.) Machine learning for email spam filtering: review, approaches and open research problems. Heliyon, Vol. 5, Issue 6.

Popis slika

Slika 1: Ljudski i strojni organi vida	22
Slika 2: Pikseli	25
Slika 3: Slika daljinskog upravljača s primjenom detekcije ruba	27
Slika 4: Primjer tekstura.....	28
Slika 5: Metoda potpornih vektora	34
Slika 6: Neuron	36
Slika 7: Najlepnica za parkiranje vozila osoba s invaliditetom	44

Mrežni izvori

1. Dualizam(filozofija) (2019.) Dostupno na: [https://hr.wikipedia.org/wiki/Dualizam_\(filozofija\)](https://hr.wikipedia.org/wiki/Dualizam_(filozofija)) [17. rujna 2019.]
2. Racionalizam (2018.) Dostupno na: <https://hr.wikipedia.org/wiki/Racionalizam> [17. rujna 2019.]
3. Materijalizam (2017.) Dostupno na: <https://hr.wikipedia.org/wiki/Materijalizam> [17. rujna 2019.]
4. Empirizam (2018.) Dostupno na: <https://hr.wikipedia.org/wiki/Empirizam> [17. rujna 2019.]
5. Indukcija (2013.) Dostupno na: <https://hr.wikipedia.org/wiki/Indukcija> [17.9.2019]
6. Logički pozitivizam (2018.) Dostupno na: https://hr.wikipedia.org/wiki/Logički_pozitivizam [17. rujna 2019.]
7. Kognitivna psihologija (2016.) Dostupno na: https://hr.wikipedia.org/wiki/Kognitivna_psihologija [19. rujna 2019.]
8. Kibernetika (2017.) Dostupno na: <https://hr.wikipedia.org/wiki/Kibernetika> [20.09.2019.]
9. Obrada prirodnog jezika (2019.) Dostupno na: https://hr.wikipedia.org/wiki/Obrada_prirodnog_jezika [20.09.2019.]
10. Timeline of artificial intelligence (2019.) Dostupno na: https://en.wikipedia.org/wiki/Timeline_of_artificial_intelligence [20.09.2019]
11. How Machine Learning Can Enable Anomaly Detection (2020.) Dostupno na: <https://medium.com/datadriveninvestor/how-machine-learning-can-enable-anomaly-detection-eed9286c5306> [17.06.2020.]
12. Clustering in Machine Learning Dostupno na: <https://www.geeksforgeeks.org/clustering-in-machine-learning/> [17.06.2020.]
13. Rodriguez, J. (2017.) Understanding Semi-supervised Learning Dostupno na: <https://medium.com/@jrodthoughts/understanding-semi-supervised-learning-a6437c070c87> [18.06.2020.]

14. Brownlee, J. (2019.) 14 Different Types of Learning in machine Learning Dostupno na: <https://machinelearningmastery.com/types-of-learning-in-machine-learning/> [18.06.2020.]
15. OpenCV. Dostupno na: <https://opencv.org/about/> [28.8.2020.]
16. Tsang, S. (2019.) Review: FPN – Feature Pyramid Network (Object Detection) Dostupno na: <https://towardsdatascience.com/review-fpn-feature-pyramid-network-object-detection-262fc7482610> [02.09.2020.]
17. TensorFlow. Dostupno na: <https://www.tensorflow.org/about> [02.09.2020.]
18. Keras. Dostupno na: <https://keras.io/> [02.09.2020.]
19. CARRASCO, O. C. (2019.) Support Vector Machines for Classification Dostupno na: <https://towardsdatascience.com/support-vector-machines-for-classification-fc7c1565e3>

Sažetak

Umjetna inteligencija je danas sve prisutnija u našem okruženju. Od autonomnih vozila do humanoidnih robota. Strojno učenje je potpodručje umjetne inteligencije koje se zadnjih godina sve više razvija. Ono nam pomaže u svakodnevnom životu rješavanjem nekih problema poput detekcije neželjene e-pošte, do onih nešto zahtjevnijih korištenjem većeg broja senzora poput pametnih usisivača koji pamte rute i prepreke. Važan dio strojnog učenja je i računalni vid. Ono što ga čini posebnim je to što omogućava strojevima da „progledaju“ te na taj način budu u mogućnosti rješavati neke probleme analizom onoga što takve zadaće prije nisu u stanju bili izvoditi. Vid je evolucijom čovjeku omogućio da prepozna objekte i okolinu bez puno muke stoga se ovaj rad bavi teorijom dostupnih alata za računalni vid. Ovaj rad bavi se detekcijom objekata odnosno dostupnim metodama unutar strojnog učenja. Obradene su najpoznatije tehnike detekcije objekata, one koje se smatraju „ručnima“ i onim nešto modernijima poput dubokog učenja.

Ključne riječi: umjetna inteligencija, strojno učenje, duboko učenje, neuronske mreže, računalni vid

Summary

Today artificial intelligence is everywhere around us. From autonomous vehicles to the humanoid robots. Machine learning is a subfield of the artificial intelligence which is developing more and more with each day. It has shown to be very helpful for us by solving various problems, such as: e-Mail spam detection or robot vacuum cleaners which by using various sensors are remembering routes and detecting obstacles. Computer vision is important part of the machine learning. What makes it special is that we are „giving“ our machine possibility to see the world and by that fact we make it possible for them to perform some new tasks they weren't able to do before. This thesis is all about theory of available tools for computer vision. It is about available methods for object detection using machine learning. In this thesis we pass through some of the known techniques for the object detection which can be „hand-crafted“ methods or those more modern one slike deep learning.

Keywords: artificial intelligence, machine learning, deep learning, neural network, computer vision