

# Raspoznavanje bolesti rajčice uporabom prijenosnog dubokog strojnog učenja na temelju slike razvijenim modulom IncMB

---

Radočaj, Petra

Master's thesis / Diplomski rad

2022

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

*Permanent link / Trajna poveznica:* <https://urn.nsk.hr/urn:nbn:hr:200:484294>

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

*Download date / Datum preuzimanja:* **2024-12-27**

*Repository / Repozitorij:*

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU  
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I  
INFORMACIJSKIH TEHNOLOGIJA OSIJEK**

**Sveučilišni studij**

**RASPOZNAVANJE BOLESTI RAJČICE UPORABOM  
PRIJENOSNOG DUBOKOG STROJNOG UČENJA NA  
TEMELJU SLIKE RAZVIJENIM MODULOM IncMB**

**Diplomski rad**

**Petra Radočaj**

**Osijek, 2022.**

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK****Obrazac D1: Obrazac za imenovanje Povjerenstva za diplomski ispit**

Osijek, 10.07.2022.

Odboru za završne i diplomske ispite

**Imenovanje Povjerenstva za diplomski ispit**

<b>Ime i prezime Pristupnika:</b>	Petra Radočaj
<b>Studij, smjer:</b>	Diplomski sveučilišni studij Računarstvo
<b>Mat. br. Pristupnika, godina upisa:</b>	D-1156R, 12.10.2020.
<b>OIB studenta:</b>	96083571167
<b>Mentor:</b>	Prof.dr.sc. Goran Martinović
<b>Sumentor:</b>	,
<b>Sumentor iz tvrtke:</b>	
<b>Predsjednik Povjerenstva:</b>	Prof. dr. sc. Krešimir Nenadić
<b>Član Povjerenstva 1:</b>	Prof.dr.sc. Goran Martinović
<b>Član Povjerenstva 2:</b>	Izv. prof. dr. sc. Ivan Aleksi
<b>Naslov diplomskog rada:</b>	Raspoznavanje bolesti rajčice uporabom prijenosnog dubokog strojnog učenja na temelju slike razvijenim modulom IncMB
<b>Znanstvena grana diplomskog rada:</b>	<b>Umjetna inteligencija (zn. polje računarstvo)</b>
<b>Zadatak diplomskog rada:</b>	U teorijskom dijelu rada treba istražiti i opisati izazove pri raspoznavanju bolesti rajčice s naglaskom na bolesti lista, te aktualne mogućnosti primjene postupaka dubokog strojnog učenja u rješavanju navedenog problema. Također, treba definirati načine prikupljanja i predobradbe podataka, obaviti izbor značajki, izabrati prikladne skupove podataka i predložiti odgovarajući klasifikacijski postupak zasnovan na konvolucijskim neuronskim mrežama, prijenosnom dubokom strojnom učenju koje uključuje i klasične postupke strojnog učenja, te na korištenju vlastitog razvijenog modula IncMB. U okviru eksperimentalne analize potrebno je opisati korištene platforme i tehnologije i definirati postavke eksperimenta. Na temelju dobivenih
<b>Prijedlog ocjene pismenog dijela ispita (diplomskog rada):</b>	Izvrstan (5)
<b>Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:</b>	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 3 bod/boda Jasnoća pismenog izražavanja: 3 bod/boda Razina samostalnosti: 3 razina
<b>Datum prijedloga ocjene od strane mentora:</b>	10.07.2022.
Potvrda mentora o predaji konačne verzije rada:	<i>Mentor elektronički potpisao predaju konačne verzije.</i>
	Datum:

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK**IZJAVA O ORIGINALNOSTI RADA**

Osijek, 17.07.2022.

Ime i prezime studenta:

Petra Radočaj

Studij:

Diplomski sveučilišni studij Računarstvo

Mat. br. studenta, godina upisa:

D-1156R, 12.10.2020.

Turnitin podudaranje [%]:

15

Ovom izjavom izjavljujem da je rad pod nazivom: **Raspoznavanje bolesti rajčice uporabom prijenosnog dubokog strojnog učenja na temelju slike razvijenim modulom IncMB**

izrađen pod vodstvom mentora Prof.dr.sc. Goran Martinović

i sumentora ,

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija. Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

# Sadržaj

1. UVOD .....	1
2. RASPOZNAVANJE BOLESTI LISTA RAJČICE .....	2
2.1. Važnost pravovremene detekcije bolesti.....	2
2.2. Bolesti rajčice.....	3
2.2.1. Razvoj epidemije .....	4
2.2.2. Patogeni.....	4
2.2.3. Dijagnosticiranje bolesti .....	4
2.3. Primjena raspoznavanja bolesti rajčice .....	5
2.4. Korišteni podatkovni skup .....	6
2.4.1. Segmentacija bolesnih dijelova lista rajčice algoritmom k-najbližih susjeda .....	7
2.4.2. Histogrami za primjere slika u boji.....	8
2.5. Postojeći sustavi i postupci za raspoznavanje bolesti rajčice.....	9
3. KLASIFIKACIJA BOLESTI RAJČICE.....	12
3.1. Razvoj klasifikacijskih metoda .....	12
3.2. Prikupljanje podataka i predobrada.....	14
3.3. Klasifikatori dubokog strojnog učenja .....	15
3.3.1. Neuronska mreža .....	15
3.3.2. Konvolucijska neuronska mreža.....	18
3.3.2.1. Matematička funkcija konvolucije.....	19
3.3.2.2. Konvolucijski sloj .....	20
3.3.2.3. Aktivacijske funkcije .....	21
3.3.2.4. Sloj sažimanja .....	23
3.3.2.5. Potpuno povezani sloj.....	24
3.4. Prijenosno strojno učenje .....	25
3.5. Upotreba metoda klasičnog u dubokom strojnom učenju.....	26
3.6. Razvijeni IncMB modul.....	27
3.7. Postupci analize i usporedba performansi .....	28
3.7.1. Matrica zabune.....	28
3.7.2. Mjere za vrednovanje klasifikatora.....	30
4. EKSPERIMENTALNA ANALIZA I USPOREDBA PERFORMANSI KLASIFIKATORA	31

4.1. Zahtjevi na sustav.....	31
4.2. Opis platformi, alata i tehnologija.....	33
4.2.1. Python i Anaconda.....	33
4.2.2. TensorFlow i Keras.....	33
4.2.3. NumPy, SciPy i Matplotlib.....	34
4.2.4. OpenCV.....	34
4.2.5. MATLAB.....	35
4.3. Provedba eksperimenta.....	35
4.4. Rezultati i diskusija.....	36
4.4.1. Analiza ostvarenih rezultata pojedinih mreža.....	40
4.4.2. Analiza ostvarenih rezultata mreža s IncMB modulom.....	40
4.4.3. Analiza ostvarenih rezultata mreža s metodom potpornih vektora.....	40
4.5. Mogućnosti poboljšanja.....	40
5. ZAKLJUČAK.....	42
LITERATURA.....	43
SAŽETAK.....	49
ABSTRACT.....	50
ŽIVOTOPIS.....	51
PRILOZI.....	52

## 1. UVOD

Rastući trend u populaciji današnjice predstavlja novi izazov u proizvodnji hrane kako bi ona bila proizvedena u većim količinama i kvalitetna. Sa sve većom primjenom u klasifikaciji podataka, pred duboko strojno učenje i preciznu poljoprivredu stavljeni su izazovi u svrhu bržeg uspostavljanja dijagnoze bolesti kako bi se spriječilo njezino napredovanje i epidemija te kako bi se na vrijeme otklonile mogućnosti za nastajanje komplikacija te usmjerilo na odgovarajući tretman. Pri tome je potrebno izraditi modele koji oponašanjem biološke organizacije slijede hijerarhiju i strukturu neuronskih mreža. Takvi modeli predstavljaju objektivnu pomoć stručnjacima pri donošenju odluka te ubrzavaju proces raspoznavanja bolesti. Kako je nemoguće konstantno nadgledanje usjeva zbog manjka resursa, modeli dubokog strojnog učenja kao i sustavi za udaljeno nadgledanje pružaju mogućnost ranijeg i preciznijeg uočavanja biljnih bolesti.

Cilj rada je pokazati kako metode dubokog strojnog učenja ostvaruju bolje performanse od metoda klasičnog strojnog učenja za problem klasifikacije bolesti rajčice kako bi njihova primjenjivost bila veća, pri čemu će se koristiti prijenosno strojno učenje. Također, cilj je razviti IncMB modul koji će predstavljati univerzalni modul ugradiv u mreže, a koji doprinosi ostvarenju visokih performansi prilikom klasificiranja podataka nekog područja primjene. Navedeno će se dokazati usporedbom performansi klasificiranja bolesti rajčice putem konvolucijske neuronske mreže s IncMB modulom, trenirane mreže bez modula te mreže dubokog strojnog učenja i spoja s metodom potpornih vektora koja pripada metodama klasičnog strojnog učenja. U svim slučajevima primjene, koristit će se prijenosno strojno učenje zbog veće brzine i točnosti, kao i ekonomičnije upotrebe računarskih resursa.

U drugom poglavlju se opisuju bolesti rajčice, njihovi simptomi uz način postavljanja dijagnoze. Prikazuju se primjeri iz korištenog skupa podataka te opisuju već postojeći sustavi za raspoznavanje bolesti biljaka. U trećem poglavlju se opisuje način prikupljanja i predobrade podataka, klasifikatori te struktura konvolucijske neuronske mreže, razvijenog IncMB modula i metode potpornih vektora. Opisuju se prednosti korištenja prijenosnog strojnog učenja. U četvrtom poglavlju se zasebno i u kombinaciji analiziraju navedeni klasifikatori te se uspoređuju njihove performanse. Uz to, opisane su korištene tehnologije i pripadajući alati.

## **2. RASPOZNAVANJE BOLESTI LISTA RAJČICE**

U navedenom poglavlju, definira se pojam bolesti, opisuje razvoj simptoma te dinamika epidemije. Opisan je korišteni skup podataka uz prikaz histograma za slike u boji te postupka segmentacije na bolesnim jedinkama rajčice. Nadalje, opisana je primjena raspoznavanja bolesti rajčice i biljaka te je dan uvid u već postojeća rješenja čiji klasifikatori ostvaruju visoke performanse.

### **2.1. Važnost pravovremene detekcije bolesti**

Kao jedna od osnovnih namirnica u svijetu, godišnja vrijednost rajčice prelazi 90 milijardi američkih dolara [1]. Uz globalni utjecaj klimatskih promjena na ukupnu poljoprivrednu proizvodnju, prirodne neprijatelje, štetnike i otpornost na insekticide, uzgoj rajčice je u opasnosti. Model utjecaja klimatskih promjena na usjeve s najtežim predviđanjem po količini uzgoja rajčice i njezine vrijednosti na tržištu, otkrio je značajna ograničenja pogodnih površina za proizvodnju rajčice. Kao rezultat međunarodne trgovine, glavna zabrinutost je natjecanje invazivnih vrsta s autohtonim organizmima kao i širenje genotipova otpornosti [2]. Prema [3], moljac rajčice i koprivina girnja, učestali štetnici rajčice, razvili su otpornost na oko 100 aktivnih kemijskih sastojaka. Nadalje, virus žućenja i kovrčavosti lista uzrokuje potencijalni rizik od gospodarskih gubitaka i smanjenja poljoprivredne proizvodnje nakon što je otkriveno da se na poljoprivredne i druge površine prenosi izravnom ili neizravnom invazijom na embrij posredovanom zaraženim spolnim stanicama [2]. Neadekvatna uporaba pesticida može prouzročiti dugotrajnu otpornost patogena i naštetiti mehanizmu obrane biljke. Za bolesti bez vidljivih simptoma zaraze ili simptoma koji se uoče prekasno, potrebno je kreirati inovativan način njihova otkrivanja i liječenja.

Kako bi se spriječile katastrofalne posljedice vezane uz održivost, dugoročnost i ekološku prihvatljivost obiteljskih gospodarstava i drugih djelatnosti [4], cilj rada je usmjeren na stvaranje modela dubokog strojnog učenja koji će na brz i točan način dijagnosticirati bolest rajčice. Detekcija bolesti temeljena je na boji i strukturi lista jer se najčešće prvi znakovi obolijevanja pojavljuju na listovima [5], ugrožavajući daljnji razvoj jedinke. Pravovremeno i objektivno dijagnosticiranje putem automatiziranih sustava smanjilo bi nepovratni gubitak proizvoda koji iznosi do 30% na godišnjoj razini [6].

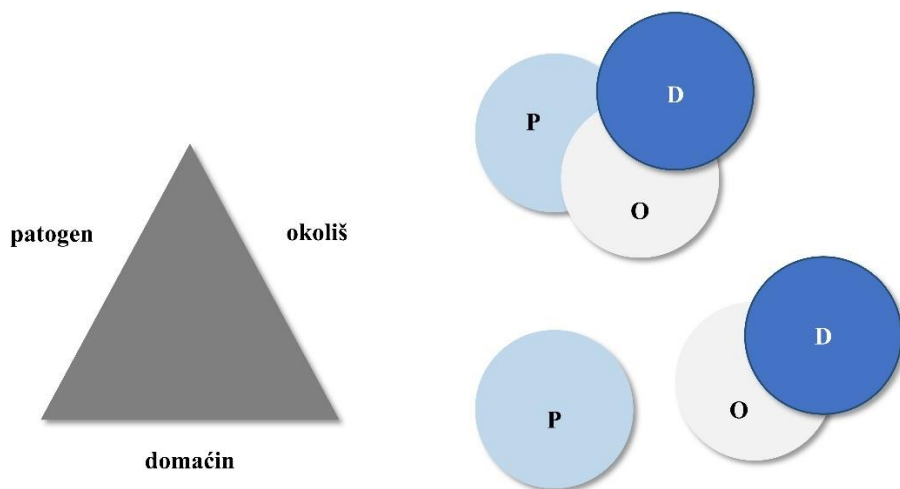
Kako bi se brzo i točno predvidio razvoj bolesti te kontroliralo njezino širenje, stvaraju se modeli dubokog strojnog učenja čiji se postupak raspoznavanja temelji na slikama bolesnih jedinki biljki.



Opisani proces se temelji na izdvajanju karakterističnih značajki bolesnih listova pomoću predobrade podataka te korištenjem odgovarajućeg klasifikatora. Najčešće se izdvajaju značajke poput boje, oblika i teksture lista. Primjerice, virus mozaika iz roda *Tobamovirus* te virus žućenja i kovrčavosti lista iz roda *Begomovirus* imaju zajednički simptom, a to je pojava žute boje lista i promjena u strukturi lista na njegovim krajevima [2]. U ovome radu, cilj je kreirati klasifikator koji uz navedene bolesti ispravno i s visokim performansama određuje kojom bolešću je zaražena promatrana jedinka.

## 2.2. Bolesti rajčice

Svaka bolest štetna je za biljku ili njezin dio, a može se manifestirati kao fiziološki poremećaj i patološke promjene u strukturi. Rezultira nemogućnošću korištenja njezinih plodova i sjemenja što smanjuje ekonomsku vrijednost biljke. Bolest se najčešće dijagnosticira nakon pojave prvih simptoma koji se mogu razlikovati u ovisnosti o razvojnom stadiju biljke [7]. Simptomi su gotovo uvijek vidljivi na lišću koje mijenja boju iz zelene u žutu ili smeđu te po strukturi lista koja može biti šupljikava ili uvenula. U manjem broju slučajeva, moguća je pojava bolesti bez simptoma ili latentna zaraza. Tada znakovi zaraze nisu vidljivi što dovodi do otežane dijagnoze uzročnika koji je i dalje prisutan u biljnom organizmu. Za razvijanje bolesti, potrebna su tri svojstva: prisutnost patogena koji dovodi do razvitka bolesti, prisutnost domaćina na kojem su vidljivi simptomi bolesti te okolinu koja zadovoljava pogodne uvjete za razvoj bolesti poput temperature, vlage i ostalih ekoloških čimbenika [8,9]. Navedeni uvjeti za pojavu biljnih bolesti opisani su na slici 2.1.



Slika 2.1 Uvjeti za pojavu i razvoj biljnih bolesti.

### **2.2.1. Razvoj epidemije**

Zbog velike reproduktivne sposobnosti parazita, prisustva agresivnog uzročnika bolesti, povoljne vlažnosti i temperature te laganog prenošenja najčešće putem vjetra, razvoj epidemije je olakšan i ubrzan, a sve veći broj oboljelih jedinki na širem području postaje otežano tretirati. Pri analizi svake epidemije se uz prostornu dinamiku širenja bolesti, promatra i ona vremenska čija se eskalacija može spriječiti putem biljne karantene poput prestanka trgovanja zaraženim jedinkama, njihovim sjemenom i/ili presadnicom [8]. Nadalje, moguće ju je smanjiti ili zaustaviti korištenjem pesticida te primjenom agrotehničkih, mehaničkih, fizičkih i ostalih mjera. Osim implementacije algoritma za raspoznavanje bolesti, moguće je kreirati algoritam u svrhu predviđanja rizika od obolijevanja biljaka kako bi stručnjaci u fazi planiranja zaštite usjeva, njihovoj količini i potrebama dobili pravovaljane informacije [9].

### **2.2.2. Patogeni**

Velik broj patogena koji napadaju biljke posjeduje ogroman reproduktivni potencijal kojim povećavaju svoju populaciju tijekom pogodnih vremenskih uvjeta. Osim što mogu brzo postati otporni na pesticide i kemijska sredstva kojima se tretiraju, imaju mogućnost brze prilagodbe na učestale klimatske i okolišne promjene [10]. Time predstavljaju opasnost za biljne organizme kojima prijeti zaraza, to jest daljnje širenje bolesti na zdrave usjeve. Birch i sur. navode da biljke kako bi se zaštitile od napada patogena mogu reagirati hipersenzitivno. Odgovaraju na napad patogena urođenim imunološkim sustavom izazivajući programiranu staničnu smrt [11].

### **2.2.3. Dijagnosticiranje bolesti**

Najvažniji korak u svrhu učinkovitog uspostavljanja dijagnoze bolesne rajčice kao i drugih biljki je identifikacija biljne vrste. Poznavajući vrstu i karakteristike biljnog organizma, poznajemo i patogene koji mogu napasti biljku, uzrokovati bolest i njezine simptome. Pri tome je nužno razlikovati izgled i svojstva bolesnih jedinki od onih zdravih kako bi se što ranije otkrilo napada li biljku samo jedan ili više patogena te koje simptome uzrokuje svaki od njih [12]. Varijabilnost simptoma u raznim fazama rasta biljke mogu dovesti do uspostave netočnih dijagnoza te se radi toga često pristupa laboratorijskom ispitivanju uzorka koje ostvaruju visoku točnost, no često se ne provode na velikom uzorku zbog teške primjenjivosti i usporenosti postupka. Kako bi se navedeni nedostaci uspješno nadvladali, moguće je koristiti modele temeljene na dubokim

neuronskim mrežama čije se treniranje i validacija provodi na bazi podataka koja uključuje bolesne jedinke u vanjskim nekontroliranim uvjetima kao i u laboratorijskim uvjetima, na primjerima biljaka tretiranih pesticidima te zdravima [13].

### **2.3. Primjena raspoznavanja bolesti rajčice**

Kako bi bolest rajčice i biljaka uopće bila rano detektirana, time smanjila trošak i spriječila daljnja epidemija te odabrala odgovarajući tretman, cilj rada je stvoriti model za raspoznavanje bolesnih listova rajčice koji će biti efikasan, objektivn i većim dijelom automatiziran. Kao dodatna pomoć stručnjacima pri analizi oboljenja i donošenju odluka, algoritmi dubokog strojnog učenja pružaju jednu od najperspektivnijih metoda računalne dijagnoze [14]. Navedeno potvrđuje osvrt znanstvenika na model za raspoznavanje bolesti listova biljki [15]: „*Algoritmi po svojoj prirodi zahtijevaju daleko veću razinu specifičnosti nego što je obično uključeno u ljudsko donošenje odluka, što je u nekom smislu 'crna kutija'. S odgovarajućim pravnim i regulatornim sustavima, algoritmi mogu poslužiti nečemu poput Geigerovog brojača koji olakšava otkrivanje – a time i – sposobnost razlikovanja.*“.

Prednosti modela dubokog strojnog učenja su uz brzu dostavu rezultata, veća učinkovitost pri detekciji biljnih bolesti, a posebno biljnih lezija i manji troškovi. Skupa s postavljanjem pravovremene dijagnoze, analiziranjem podataka algoritmi predviđaju vjerojatnost pojave biljnih bolesti i štetnog utjecaja nametnika. Rastući trend u populaciji današnjice predstavlja novi izazov u svrhu kvalitetnije i veće proizvodnje hrane. Zbog nemogućnosti konstantnog nadgledanja usjeva, modeli dubokog strojnog učenja kao i sustavi za udaljeno nadgledanje pružaju mogućnost ranijeg i preciznijeg uočavanja biljnih bolesti [14]. Modeli su dodatno prilagođeni uporabi stručnjacima svih vrsta korištenjem interaktivne aplikacije koja uz podatke o vrsti biljke i trenutnoj lokaciji pruža trenutnu pomoć u dijagnostici [16]. Za dobivanje što točnijih rezultata, potrebna je mogućnost unosa i analize trenutnih vremenskih prilika poput temperature i vlažnosti koje pogoduju nastanku bolesti. Također, nužna je mogućnost unosa novih bolesti i prepoznatih simptoma te proširenje trenutno dostupne baze slika bolesnih biljaka. Primjeri bolesnih jedinki koje su korištene u ovome radu prikazani su na slici 2.2.



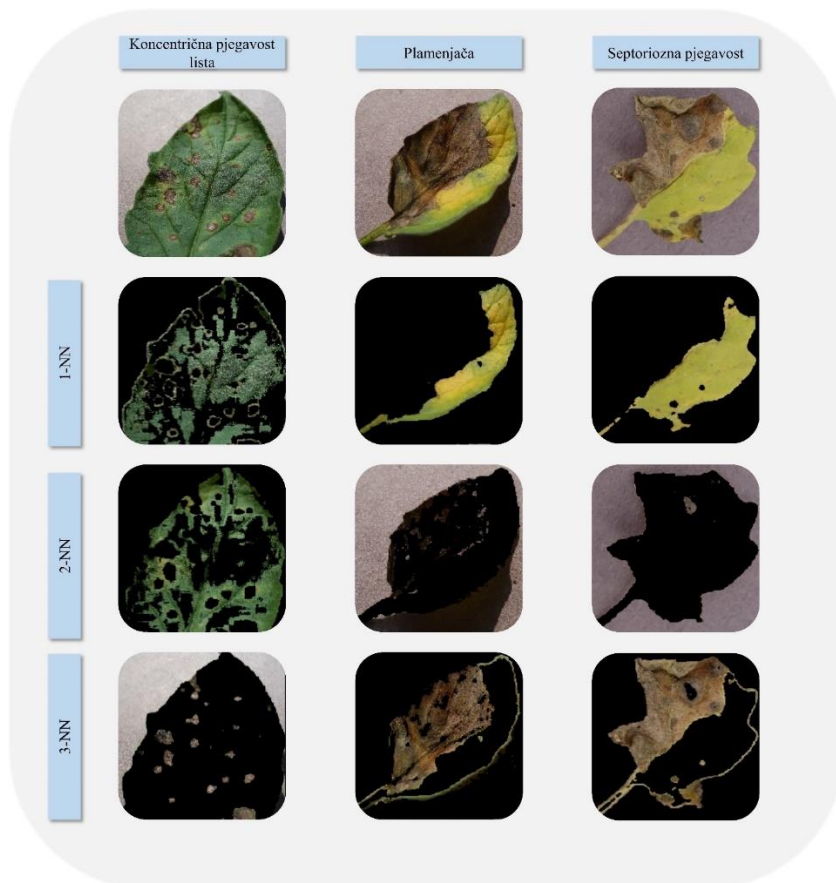
Slika 2.2 Primjeri bolesnih jedinki rajčice.

## 2.4. Korišteni podatkovni skup

Na ukupno 3300 podataka bolesnih jedinki rajčice koji su podijeljeni u šest klasa, provedeno je treniranje i validacija algoritama konvolucijskih neuronskih mreža za raspoznavanje koncentrične pjegavosti lista, plamenjače, septoriozne pjegavosti, koprivine grinje, virusa mozaika te virusa žučanja i kovrčavosti lista [17]. Svaku klasu čini 550 primjera bolesnih jedinki, a omjer podataka za treniranje i validaciju iznosi 70 : 30 pri čemu je podjela podataka odrađena nasumično. Svi podaci dio su javno dostupnog skupa podataka naziva *New Plant Diseases Dataset* [17] koji je proizašao iz skupa *PlantVillage* [18].

### 2.4.1. Segmentacija bolesnih dijelova lista rajčice algoritmom k-najbližih susjeda

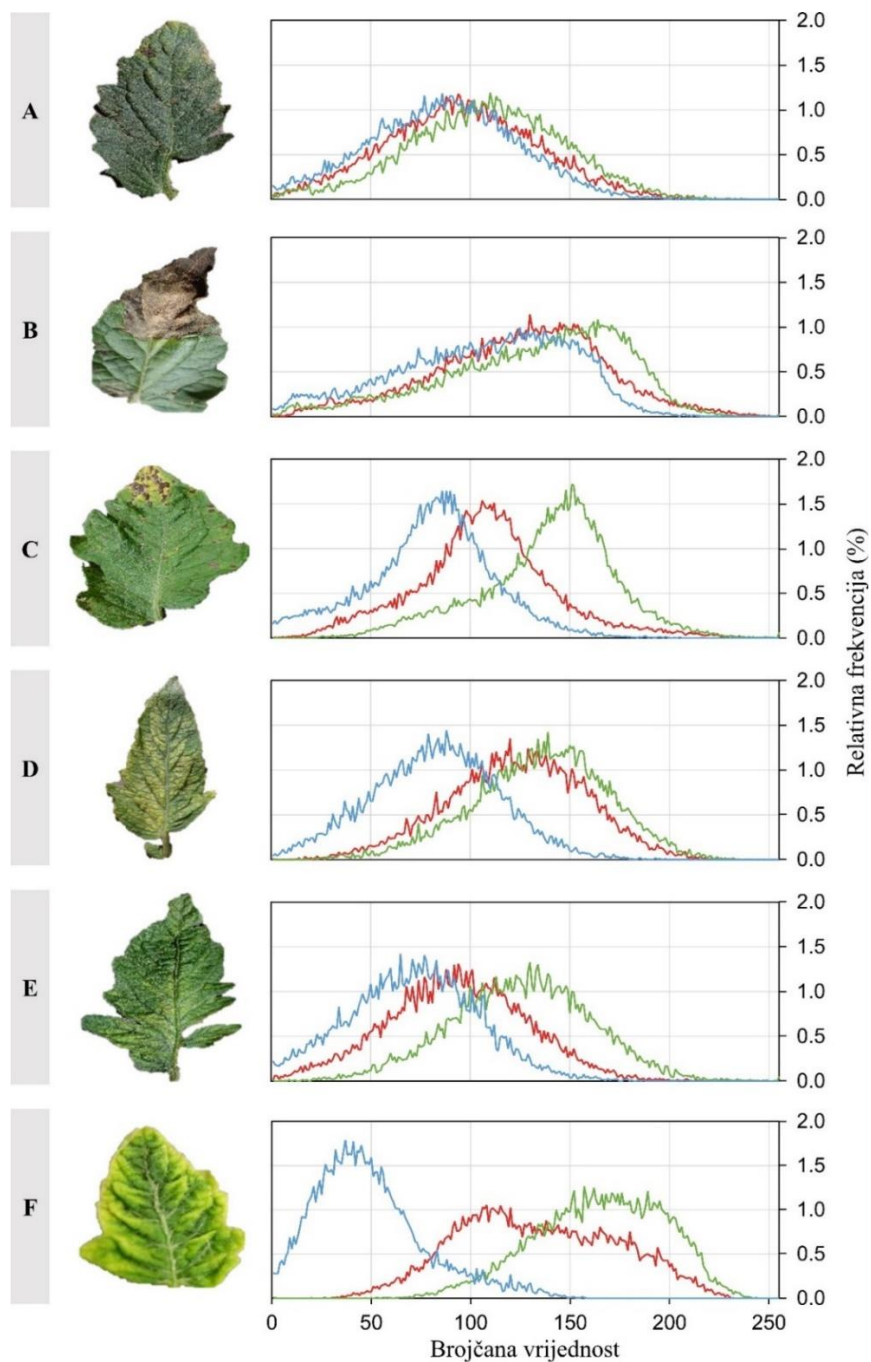
Pomoću brzog, robusnog i lako prilagodljivog algoritma k-najbližih susjeda ili k-NN, obavljena je segmentacija bolesnih jedinki rajčice na primjeru koncentrične pjegavosti lista, plamenjače i septoriozne pjegavosti. Rezultati prerađenog algoritma prikazani su na slici 2.3 [19] gdje je k broj susjeda odabran proizvoljno. Navedeni algoritam nenadziranog strojnog učenja moguće je koristiti i za probleme višeklasne klasifikacije. Metoda se temelji na pronalasku broja k najbližih susjeda određenih središnjom točkom ili centroidom te njihovim klasama. Postupak se odvija bez prethodnog treniranja, a svakom podatku se izračunom udaljenosti, najčešće Euklidske, pronalazi najbliži susjed ili k-najbližih susjeda gdje je k neki prirodni broj. Metoda je često korištena jer efikasno odvaja objekt od pozadine slike kao i zdrav od bolesnog dijela lista biljke te time odrađuje segmentaciju slike. Prilikom postupka segmentacije, dolazi do izražaja nedostatak algoritma kNN koji se očituje u nemogućnosti egzaktnog određivanja središnje točke i optimalnog broja susjeda zbog kompleksnosti teksture lista, njegove boje i oblika [20].



Slika 2.3 Segmentacija bolesnih jedinki rajčice kNN algoritmom.

## 2.4.2. Histogrami za primjere slika u boji

Kako bi se distribucija intenziteta piksela sastavljena od tri kanala R (crvenog), G (zelenog) i B (plavog) grafički opisala, izrađen je histogram za primjere bolesnih jedinki rajčice predstavljen na slici 2.4.



Slika 2.4 Usporedba histograma za bolesne jedinke rajčice. (A) koncentrična pjegavost lista; (B) plamenjača; (C) septoriozna pjegavost; (D) koprivina grinja; (E) virus mozaika; (F) virus žućenja i kovrčavosti lista.

Vrijednosti histograma su relativno nepromjenjive translacijom ili rotacijom slika. Tek manje varijacije moguće je postići pri različitom kutu gledanja. Os apscisa sadrži vrijednosti pojedinih piksela u intervalu [0, 255], dok os ordinata daje relativnu frekvenciju za određeni intenzitet. Za učinkovito raspoznavanje bolesti lista biljke, moguće je ukloniti piksele koji sadrže vrijednosti zelene boje. Time bi se omogućila detekcija bolesnog dijela lista, promjene u strukturi i površina oboljenja [21]. Kod bolesnih jedinki, relativna frekvencija je približno jednake vrijednosti te ona ukazuju na prisutnost bolesti zbog drugačijeg intenziteta boja na listu koji osim zelene, sadrži nijanse žute i/ili smeđe.

## **2.5. Postojeći sustavi i postupci za raspoznavanje bolesti rajčice**

U [22] su Waaleed Albattah i sur. 2022. godine predstavili novi pristup koristeći duboko strojno učenje prilikom detekcije i klasifikacije biljnih bolesti. Uveli su robustan sustav za klasifikaciju sastavljenog od CenterNet mreže koja predstavlja programski okvir i DenseNet-77 mreže kao baznog modela. Dok DenseNet-77 služi za izdvajanje ključnih značajki i pri tome ih sažima, CenterNet služi za otkrivanje i kategorizaciju biljnih bolesti. Za provođenje eksperimenta, koristili su javno dostupni skup podataka *PlantVillage* [18] koja predstavlja standardni skup podataka s primjerima različitog oblika i veličine lišća, promjenama u strukturi i boji te varijabilnim intenzitetom, a sastoji se od 38 kategorija. Provedena kvalitativna i kvantitativna analiza potvrđuju da je navedenim pristupom ostvarena pouzdanija i veća točnost od trenutno korištenih, a koja iznosi 99.98%. Iako su postignuti zapaženi rezultati, predloženi model se zbog svoje veličine i ogromnog broja parametara ne može upotrebljavati u mobilnim uređajima pa daljni razvoj istraživanja namjeravaju usmjeriti u stvaranje manje zahtjevnog modela prema računarskim resursima, zadržavajući visoke točnosti. Također, vremensku složenost namjeravaju smanjiti odabirom učinkovitije mreže za izdvajanje značajki. Planiraju proširiti korišteni skup podataka kako bi model bio primjenjiv u širem području te navedeno validirati.

U [23] su Sue Han Lee i sur. predstavili metode otkrivanja i klasifikacije biljnih bolesti pomoću automatiziranih sustava temeljenih na dubokom strojnom učenju. U istraživanju su ispitivali performanse prethodno treniranih modela i modela treniranih samo na promatranom skupu podataka, nakon čega su otkrili da najbolje performanse ostvaruje prethodno trenirani model temeljen na prijenosnom učenju. Također, potvrdili su da izdvojene značajke ne moraju nužno ukazivati na klasificiranu bolest, zbog razlike u simptomima tijekom razvoja biljke. Pri tome su

koristili *PlantVillage* [18] skup podataka u omjeru 70 : 30 za trening i validacijski skup podataka. Dodatnu evaluaciju proveli su nad IPM i Bing skupom podataka koji sadrže slike biljaka u nekontroliranim vanjskim uvjetima. Zaključili su da je pogrešna klasifikacija česta prilikom validacije bolesnih jedinki kukuruza i krumpira zbog sličnosti u izgledu usjeva, izgledu listova i simptomima bolesti.

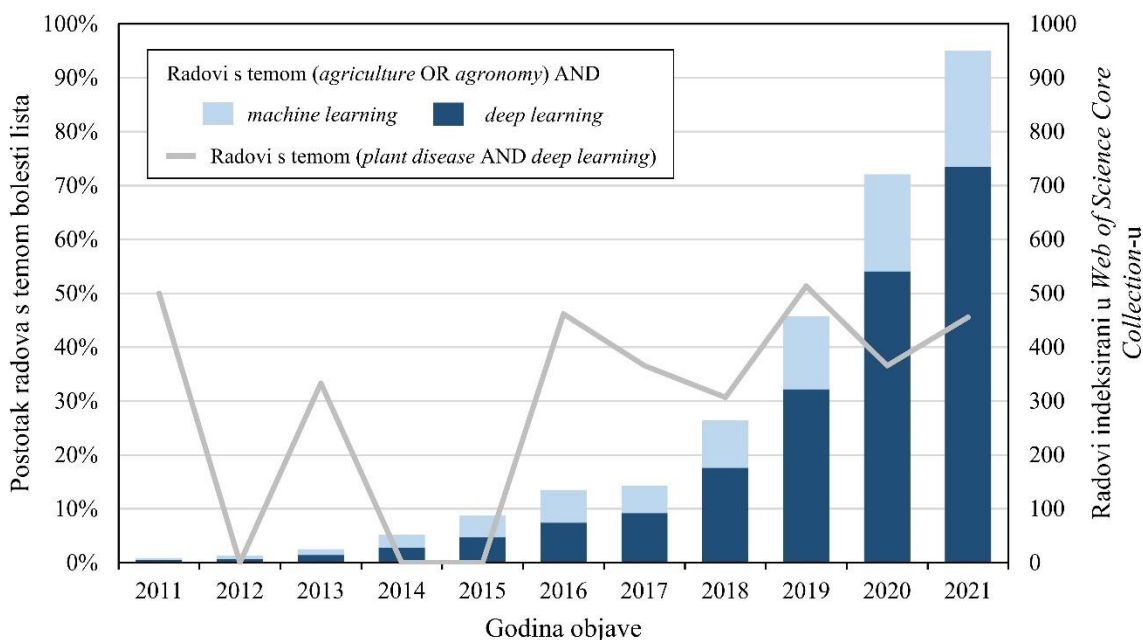
U [24] su Amanda Ramcharan i sur. koristeći prijenosno strojno učenje trenirali duboku konvolucijsku neuronsku mrežu radi uspješnog prepoznavanja tri bolesti te dvije štetočine koje napadaju biljku cassava koja je jedna od tri najveća izvora ugljikohidrata u supsaharskom dijelu Afrike. Mreža je trenirana na primjerima slika bolesnih listova cassave koje su snimljene u Tanzaniji. Korištena baza sadrži slike bolesnih jedinki cassave u različitim fazama razvoja biljke uz promatranje različitih genotipova. Analiziranjem performansi inačica InceptionV3 mreže s Inception softmax slojem, metodom potpodnih vektora i metode k-najbližih susjeda ( $k=3$ ), najbolji model je ostvario točnost od 93.00% te je implementiran u Android uređaje. Od navedenih, inačica s metodom k-najbližih susjeda pokazala je najlošije performanse, dok su ostale dvije inačice ostvarile slične performanse. Trenutno se model integriran u mobilne uređaje koristi u Tanzaniji kao dodatna pomoć stručnjacima radi kontrole rasprostranjenosti bolesti i sprječavanja epidemije.

U [25] je kreiran model kako bi otkrili lezije na listovima biljaka u ranom stadiju kao i bolesti uopće. Rješenje je temeljeno na prijenosnom učenju i modelu konvolucijskih neuronskih mreža. Koristili su javno dostupan skup podataka *PlantVillage* [18] u kojem se uz primjere bolesnih jedinki nalaze i one zdrave. Navedeni skup podataka upotpunili su vlastito izrađenim koji se sastojao od bolesnih jedinki riže i kukuruza. Kako bi osigurali dovoljno raznovrsan skup podataka, slike su u postupku predobrade rotirali, mijenjali im intenzitet osvjetljenja kao i razmjera. Koristeći VGGNet model koji je prethodno treniran, za raspoznavanje bolesti dostupnih u *PlantVillage* bazi [18] ostvarili su točnost od 91.83%, dok je točnost raspoznavanja jedinki kod vlastito izrađene baze iznosio 92.00%. Korištenom modelu su izmijenili posljednji sloj tako što su ga zamijenili dodatnim konvolucijskim slojem koji je za aktivaciju funkciju koristio Swish funkciju umjesto dotad najčešće korištene *ReLU* nelinearne funkcije. Zatim su dodali dva Inception modula dok su radi smanjenja dimenzija područja značajki postavili sloj sažimanja umjesto potpuno povezanog softmax sloja. Daljni rad usmjerili su na razvoj modela kao i na implementaciju istog u mobilne



uređaje kako bi ubrzali proces prepoznavanja bolesti, praćenja njihova razvoja te u slučaju epidemije, smanjenja njezinog intenziteta.

Prikaz broja znanstvenih radova indeksiranih u *Web of Science Core Collection*-u [26] s primjenom klasičnog i dubokog strojnog učenja u raspoznavanju bolesti biljaka tijekom godina, nalazi se na slici 2.5. U porastu je broj objavljenih radova u kojima se pri klasifikaciji koriste metode dubokog učenja, a za problem raspoznavanja bolesti lista biljke povećan je broj studija u drugoj polovici desetljeća.



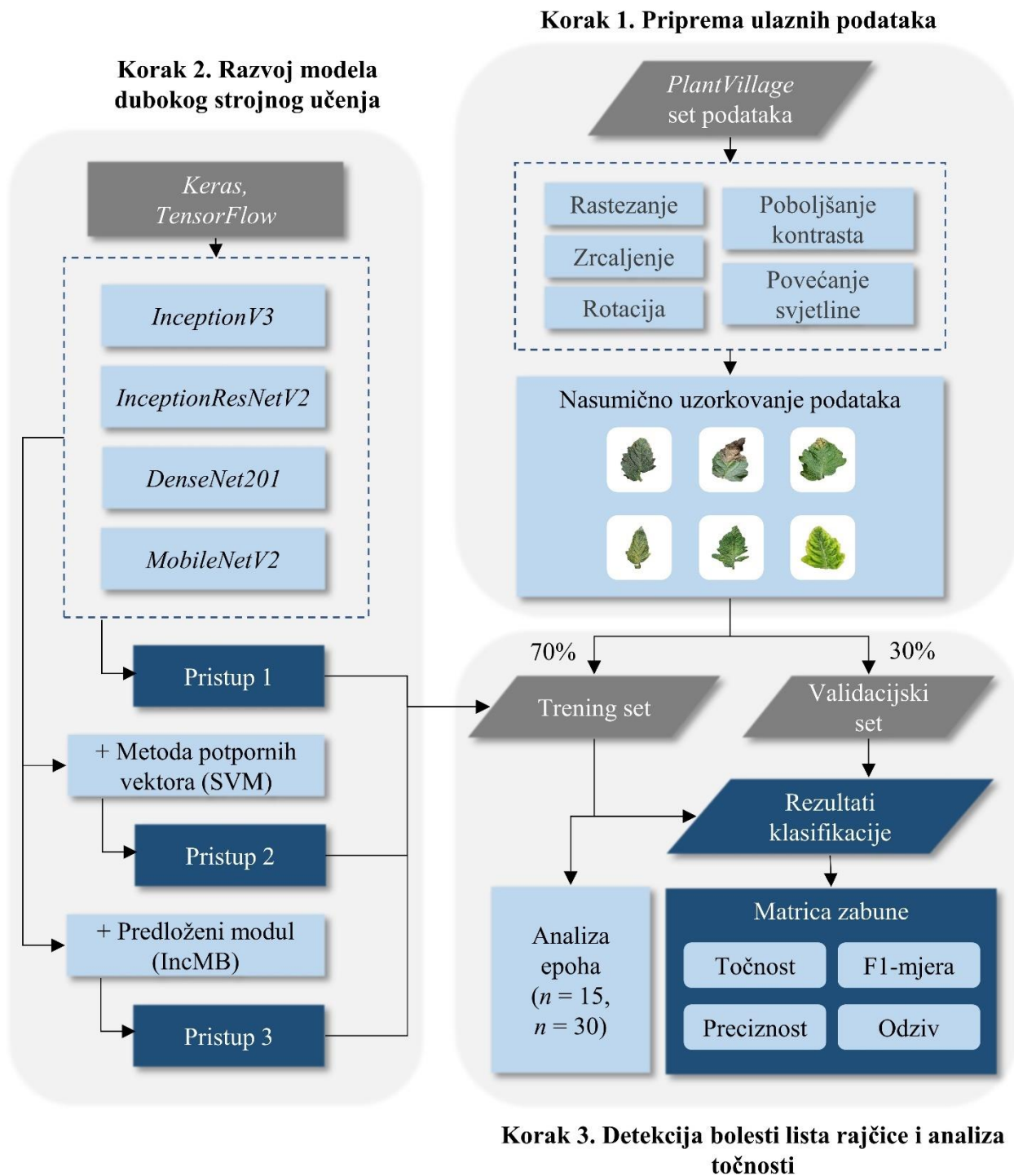
Slika 2.5 Broj radova s primjenom strojnog učenja u raspoznavanju bolesti biljke tijekom godina.

### **3. KLASIFIKACIJA BOLESTI RAJČICE**

U ovome poglavlju se nakon načina prikupljanja podataka i njihove predobrade prikazuju materijali i metode s naglaskom na metodu konvolucijske neuronske mreže. Prikazane su prednosti korištenja prijenosnog strojnog učenja, očekivane performanse korištenja metoda klasičnog strojnog učenja skupa s dubokim učenjem i razvijenog IncMB modula. Naposljetku, prikazani su postupci analize i usporedbi performansi prilikom postupka validacije implementiranih modela.

#### **3.1. Razvoj klasifikacijskih metoda**

Razvoj automatiziranih sustava koji otkrivaju biljne bolesti detekcijom simptoma na lišću, potaknut je unaprjeđenjem konvolucijske neuronske mreže [27], metode dubokog strojnog učenja koja predstavlja najperspektivniji pristup navedenog problema zbog preciznog načina izdvajanja vizualnih značajki i to bez upotrebe metoda segmentacije [28]. Korištenje prijenosnog učenja omogućilo je implementaciju modela koji ostvaruju visoku točnost pri klasifikaciji primjeraka iz specijaliziranog skupa podataka radi preciznog određivanja parametara modela koji je prethodno treniran na općenitom skupu podataka. Na primjerima bolesnih i zdravih jedinki biljaka čiji se životni ciklus odvija pod kontroliranim uvjetima, pokazano je da prethodno trenirani modeli konvolucijskih neuronskih mreža ostvaruju rezultate s većom točnosti u usporedbi s modelima treniranima po prvi put. Nadalje, dokazana je ovisnost pozadinskih značajki pri klasifikaciji bolesti biljaka te se stoga uklanjanje pozadine pomoću metoda segmentacije ne provodi jer smanjuje učinkovitost modela [28]. Uspoređujući algoritme klasičnog strojnog učenja i one dubokog, modeli dubokog strojnog učenja ostvaruju značajne razine točnosti i visoke performanse iako je njihovo provođenje dugotrajnije i sklopovski zahtjevnije. Proces izdvajanja značajki je kod metoda dubokog strojnog učenja potpuno automatiziran jer je već ugrađen u sam algoritam. Iako takvi algoritmi postižu zapaženo bolje performanse, nije ih jednostavno interpretirati. Znanje o procesu izdvajanja značajki i njegova vizualizacija značajno bi pridonijeli razumijevanju dijagnostičkih postupaka, jer unatoč broju objavljenih studija, taj proces ostaje nejasan (engl. *black box*). Poznavajući proces, povećala bi se učinkovitost i primjenjivost uporabe algoritama dubokog strojnog učenja pri raspoznavanju bolesti biljaka [29]. Opisani tijekom rada i korištene metode prikazani su na slici 3.1.



Slika 3.1 Postupci istraživanja prilikom raspoznavanja bolesti lista rajčice.

### 3.2. Prikupljanje podataka i predobrada

Za treniranje modela vezanih uz duboko strojno učenje s ciljem raspoznavanja bolesti biljke i automatsku klasifikaciju, tehnička ograničenja sustava najčešće gotovo neprimjetno utječu na prikaz performansi dok korišteni skup podataka te brojnost primjera podataka uglavnom doprinose netočnim rezultatima predviđanja te neželjenim posljedicama. Zbog izbjegavanja ovog problema, potrebno je osigurati konzistentnost korištenog skupa podataka koji u ovome rasu sadržava primjereke bolesnih listova rajčice pri tome ne promatrajući zarazne cvijetove, stabljike i plodove. Također, nužno je osigurati primjere bolesnih slika u različitim životnim uvjetima poput kontroliranih ili realnih te uzgojenih u laboratorijskim uvjetima [30].

Korišteni podatkovni skup sadržava slike u RGB formatu čije vrijednosti piksela iznose 256x256. Svaki piksel određen je sumom 8-bitnih kanala crvene, zelene i plave boje s vrijednošću od 24 bita po pojedinom pikselu koji označava najveću vrijednost za kolekciju boja vidljivu ljudskim okom. Bit po pikselu kao vrijednost jedinice određuje ukupnu razinu boje koja se nalazi u pojedinom pikselu.

U teorijskom razmatranju, Barbedo navodi kako uklanjanje pozadine slike prilikom predobrade ne bi trebalo utjecati na ishode klasifikacije u negativnom smislu. Međutim, prilikom provođenja postupka, primijećeno je kako konvolucijske neuronske mreže pri klasifikaciji bolesnih jedinki osim simptomatskih, koriste i pozadinske karakteristike slike. Napominje kako je moguće ostvariti iste rezultate klasifikacije ovisno o različitim fazama napredovanja bolesti i pojave drugačijih simptoma u usporedbi s primjerima u korištenom skupu podataka [30]. Sharma i sur. su smanjenjem veličine piksela uklonili prisutnu buku mijenjajući RGB format slika u HSV format u kojem spektar boja sadrži manje šuma [31].

Prilikom kreiranja podatkovnog skupa, poznato je da skup s neuravnoteženim brojem primjeraka bolesnih biljaka u različitim klasama, najčešće rezultira pretjernim prilagođavanjem podataka klasi koja sadržava najveći broj podataka. Za rješavanje ovog problema, moguće je primijeniti postupke transformacije poput rastezanja, zrcaljenja, rotacije slika, poboljšanja kontrasta te promjene svjetline. S ciljem povećanja podatkovnog skupa čije je primjere moguće naći u realnim i drugim životnim uvjetima, prethodno navedeni postupci korišteni su u predobradi podataka koji su dio *New Plant Diseases Dataset*-a [17] korištenog u ovom radu.

### 3.3. Klasifikatori dubokog strojnog učenja

Ograničenje metoda klasičnog strojnog učenja prilikom izdvajanja značajki uspješno prevladavaju metode dubokog strojnog učenja čija je slojevita struktura umjetnog neuronskog sustava nalik na biološku neuronsku mrežu u ljudskom mozgu [32].

Klasifikaciju slika moguće je provesti pomoću implementacije modela strojnog učenja u kojima je ulazni podatak potrebno klasificirati, dok izlazni podatak označava skup kategorija kojima opisuje ulazni podatak ili određuje njegovu pripadnost pojedinoj klasi. Izražava se funkcijom 3-1:

$$f: \mathbb{R}^n \rightarrow \{1, \dots, k\} \quad (3 - 1)$$

gdje je  $k$  prirodni broj. Ako je relacija 3-2 zadovoljena, model svakom ulazu  $x$  određuje izlaznu kategoriju  $y$ .

$$y = f(x) \quad (3 - 2)$$

Navedeni primjer dio je nadziranog strojnog učenja kojemu su za trening skup unaprijed određene diskretne vrijednosti na izlazu s ciljem pronalaska funkcije koja ima najveću učinkovitost i točnost. Za provjeru točnosti funkcije, klasifikator provodi isti postupak na validacijskom skupu podataka koji sadrži do tada nepoznati skup podataka [33].

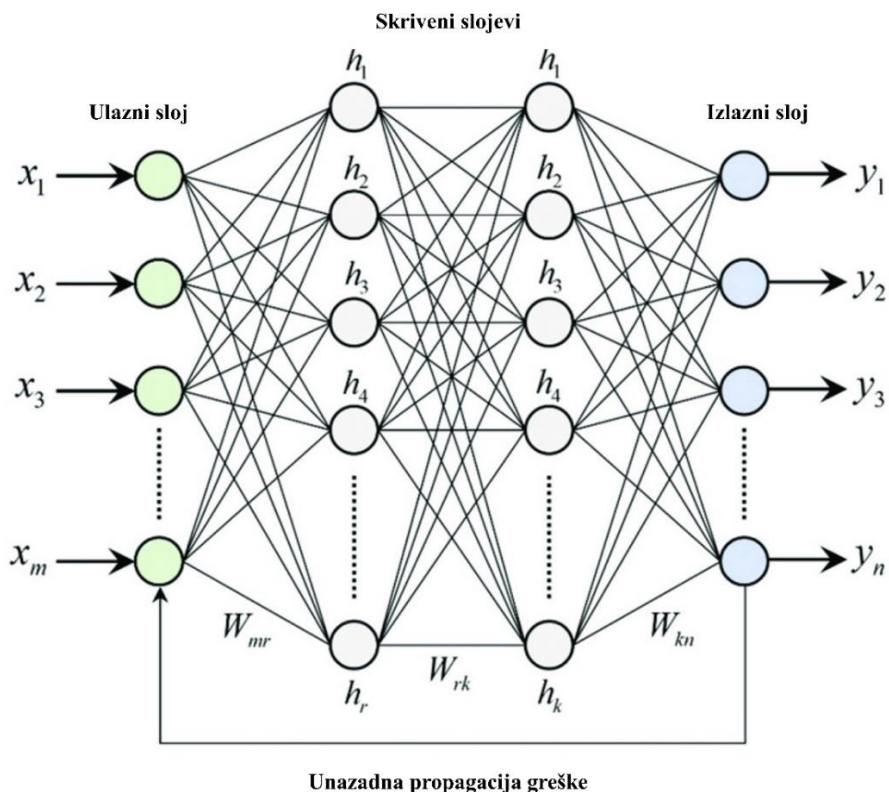
#### 3.3.1. Neuronska mreža

Neuronske mreže ili skup algoritama čija je svrha prepoznavanje obrazaca, percipiraju podatke putem strojne predodžbe tako što označavaju ili grupiraju podatkovni unos. Svi obrasci su numeričkog oblika, a opisuju ih vektori u kojima svi signali iz stvarnog svijeta poput slike, zvuka, teksta ili videa moraju biti prevedeni. Osim grupiranja ulaznih podataka pomoću sličnosti, klasificiraju podatke ukoliko su podaci u trening skupu označeni. Također, neuronske mreže mogu biti posrednici prilikom kojeg se temeljem izdvojenih značajki kreira novi skup podataka kojeg je moguće predati drugom algoritmu na daljnju obradu [34].

Navedeni postupci primjenjuju se tijekom određivanja medicinskih dijagnoza i tretmana, prepoznavanja govora, slika i teksta, detekciji anomalija, stvaranja novih informacija na temelju ispitanih baza podataka, a svakako su neizostavna i objektivna pomoć stručnjacima različitih područja znanosti prilikom donošenja odluka [34].

Hinton i sur. su 2018. godine oponašanjem hijerarhije neuronskih mreža te biološke strukture poboljšali model mreža kapsula, vrste umjetne neuronske mreže. Kako bi smanjili početne smetnje, u neuronske mreže su dodavali strukture nazvane kapsule s izlaznim vektorom pomoću čijih vrijednosti su stvarali nove, sadržajnije kapsule. Kapsula predstavlja skup neurona koji istražuju različita svojstva objekata kao što su položaj ili veličina. Uz uspješno prepoznavanje slika čiji dijelovi nisu u pravilnim prostornim odnosima i usmjeravajući podatke iz kapsula koje se nalaze na nižoj razini u one više razine mreže, doprinose efikasnijem rješavanju Picassovog problema [35].

Neuronska mreža s najmanje četiri razine slojeva skupa s čvorovima, smatra se modelom dubokog strojnog učenja, različito od jednostavnog perceptrona koji osim ulaznog i izlaznog sloja, sadrži najviše jedan skriveni sloj između njih. U dubokoj neuronskoj mreži, svaki sloj čvorova izvršava operacije na novom skupu značajki temeljem dobivenih rezultata iz prethodnog sloja, a primjer takve mreže predstavljen je na slici 3.2 [36]. Apstrakcija mreže povećava se s većom brojnošću slojeva kao i hijerarhije značajki. Ovo svojstvo omogućava obradu složenog višedimenzionalnog skupa podataka kao i brojnih parametara koji postaju dio nelinearnih funkcija.



Slika 3.2 Struktura duboke neuronske mreže [36].

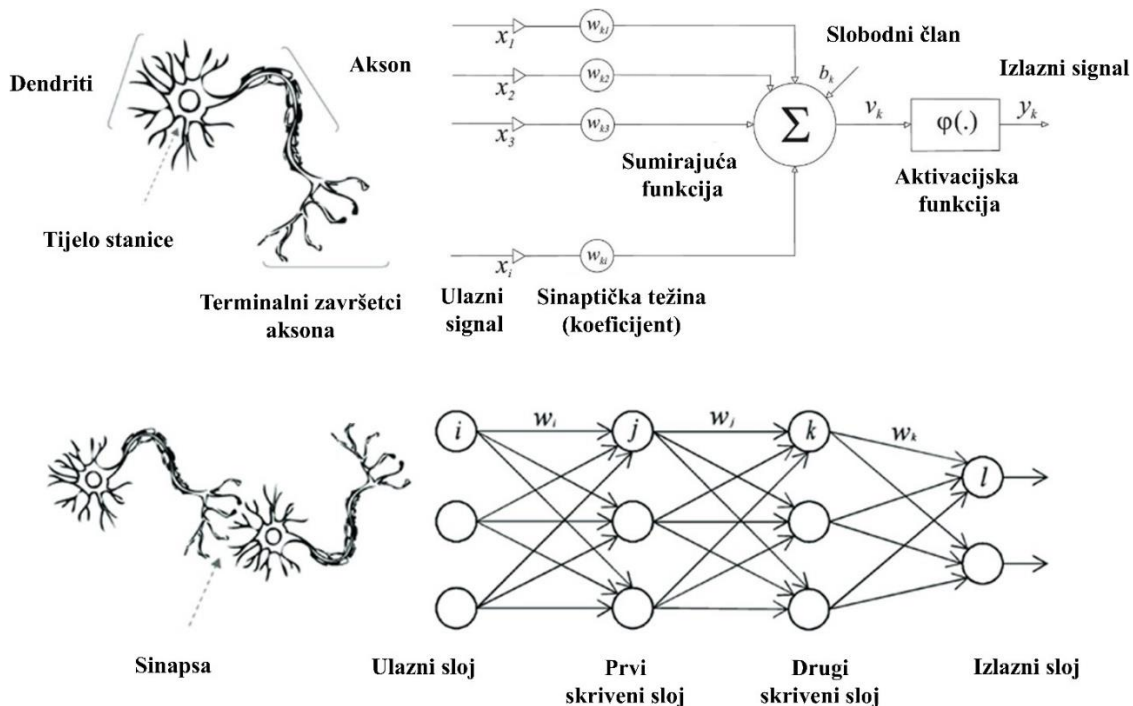
Svi umjetni neuroni oponašaju i provode iste zadaće poput bioloških neurona koji su sastavni dio živčanog sustava. U procesu obrade i prijenosa informacija, ulazni signali označeni simbolima  $x_1, x_2, \dots, x_i$  međusobno su povezani pomoću sinaptičke težine  $w_{ki}$ . Također, vezani su s funkcijom vremena te slobodnim koeficijentom  $b_k$  (engl. *bias*), svojevrsnom graničnom vrijednošću (engl. *threshold*) prilikom kreiranja karte značajki. Sumiranjem vrijednosti ulaznih signala te dobivenih sinaptičkih težina, dobiva se ulaz mreže  $u_k$  koji je opisan izrazom 3-3:

$$u_k = \sum_{j=1}^i w_{kj} x_j \quad (3 - 3)$$

Ulazu mreže se potom pribraja slobodni koeficijent kako bi se stvorio aktivacijski potencijal umjetnog neurona označenog simbolom  $v_k$ . Svaki izlazni signal prethodnog sloja, ujedno je ulazni sljedećem sloju. Za dobivanje izlaza mreže, potrebno je međudjelovanje aktivacijske funkcije  $\varphi(\cdot)$  i aktivacijskog potencijala neurona opisanog jednadžbom 3-4 [37]:

$$y_k = \varphi(u_k + b_k) \quad (3 - 4)$$

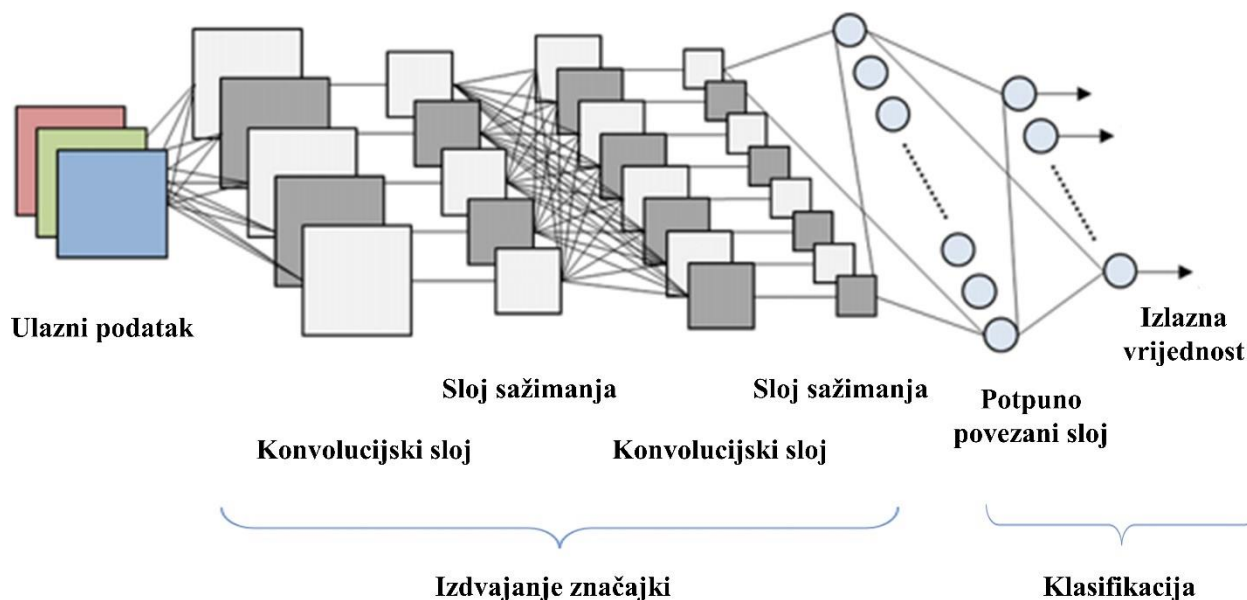
Kako bi se modelirali složeni nelinearni skupovi podataka, potrebno je osigurati nelinearno svojstvo neurona pomoću aktivacijske funkcije. Nadalje, izlazni signal postaje dio neurona sljedećeg sloja [37], a prikazan je na slici 3.3 uspoređujući umjetni s biološkim neuronom.



Slika 3.3 Usporedba strukture bioloških i umjetnih neurona [37].

### 3.3.2. Konvolucijska neuronska mreža

Konvolucijska neuronska mreža (engl. *convolutional neural network*) imitira funkcije poput onih u vidnom sustavu ljudi izdvajanjem značajki putem operacije konvolucije koja se obavlja na ulaznom podatku mreže. Potom se izdvaja najveća ili prosječna vrijednost dobivenih značajki pomoću kojih se na izlazu mreže stvara rezultat klasifikacije nakon potpuno povezanog sloja. Između konvolucijskih slojeva na izlazu iz trenutnog, a na ulazu u sljedeći neuron, numeričke operacije obavlja aktivacijska funkcija predstavljajući filter. Primjer strukture i slojeva konvolucijske neuronske mreže opisan je na slici 3.4. Kako bi se ubrzala konvergencija i kontrola treniranja mreže, moguće je koristiti *dropout* slojeveve kao i *batch normalization* [38].



Slika 3.4 Struktura i prikaz slojeva konvolucijske neuronske mreže [39].

Pri ocjenjivanju performanse mreže kao i izračunavanja vrijednosti, gubitak prilikom treniranja i validacije prikazuje se pomoću funkcije izračuna gubitka (engl. *loss function*). Rezultat formira pomoću parametara izlaza modela i ciljane vrijednosti s kojom ga uspoređuje. Izračune obavlja u dva smjera: u prolazu naprijed izračunava vrijednost gubitka pomoću ulazne i ciljane vrijednosti, dok prolaskom unatrag izračunava gradijent funkcije gubitka pomoću algoritma propagacije pogreške unatrag (engl. *backpropagation*) [39]. Prvo se izračunava gradijent posljednjeg sloja, potom prethodnog sloja s djelomičnim izračunima gradijenta trenutnog sloja, a posljednje se određuje gradijent prvog sloja u konvolucijskoj neuronskoj mreži.



### 3.3.2.1. Matematička funkcija konvolucije

Matematička funkcija konvolucija nastala je integracijom umnoška dvije funkcije po intervalu njihovih definicija. Funkcije su međusobno ravnopravne te pojedinačna infinitezimalna promjena svake funkcije utječe na onu drugu tijekom cijelog intervala definicije. Primjenom funkcije konvolucije moguće je smanjiti smetnje koje su sastavni dio ulaznog signala koji time postaje otporan na njih kao i na pozadinsku buku [39]. Metoda konvolucijske neuronske mreže prilikom izračuna koristi diskretnu konvoluciju definiranu pomoću izraza 3-5:

$$(u * v)(n) = \int_{m=-\infty}^{\infty} u(n - m)v(m)dt \quad (3 - 5)$$

gdje se konačni broj koraka označava s  $n$  i  $m$ , dok su signali prikazani simbolima  $u$  i  $v$ . Matematička funkcija konvolucija prikazana je simbolom  $*$  [39].

Rezultat konvolucije pohranjuje se u registar akumulatora koji predstavlja sklopovsku jedinicu. Slijedi opis algoritma matematičke funkcije konvolucije u pseudojeziku [39], dok je postupak opisan na slici 3.5 gdje je ulazna slika veličine  $7 \times 7$ , dok je filter (engl. *kernel*) veličine  $3 \times 3$ .

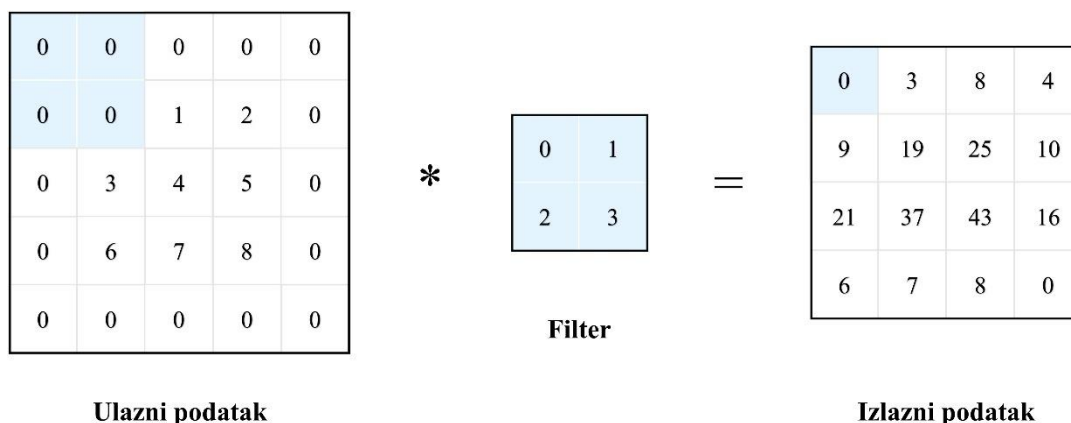
---

#### Algoritam 1 Matematička operacija konvolucija

---

```
for each red matrice slike in ulazna slika:  
  for each piksel in red matrice slike:  
    set akumulator na nula  
    for each red konvolucijske matrice in konvolucijska matrica:  
      for each element in red konvolucijske matrice:  
        if pozicija elementa odgovara poziciji piksela then  
          multiply vrijednost elementa za odgovarajuću vrijednost piksela  
          add rezultat u akumulator  
        end if  
      end for  
    end for  
  end for  
end for  
set piksele izlazne slike na akumulator
```

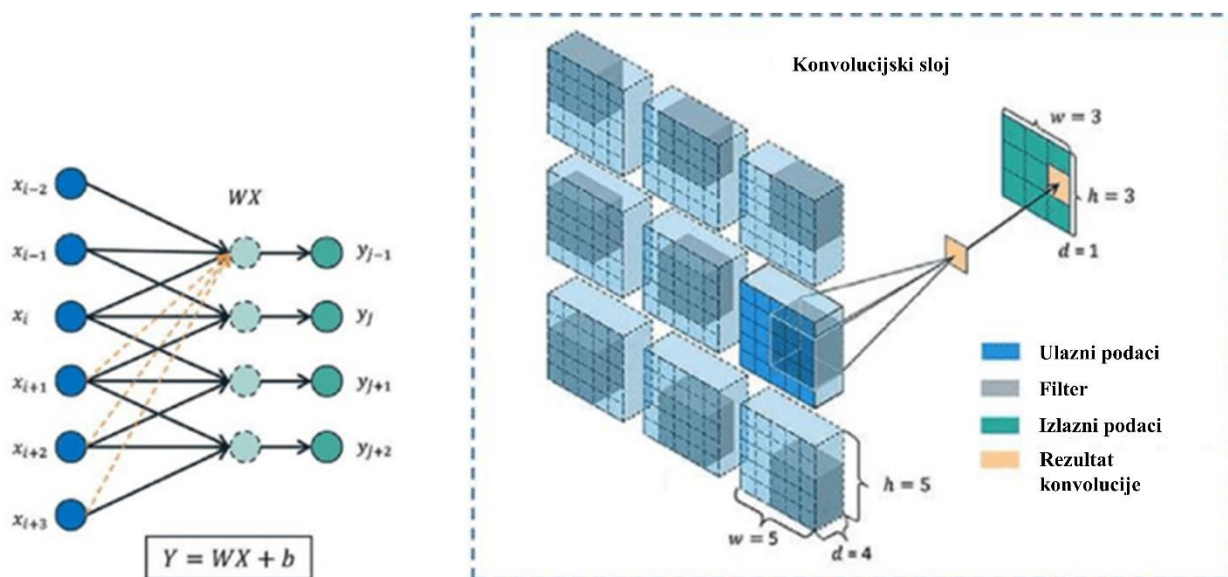
---



Slika 3.5 Izvođenje matematičke operacije konvolucije.

### 3.3.2.2. Konvolucijski sloj

Iako konvolucijske neuronske mreže ostvaruju visoke točnosti i zavidne performanse prilikom raspoznavanja objekata, najefikasnija i najučinkovitija implementacija arhitekture mreže se istražuje. Uobičajena struktura konvolucijskog sloja prikazana je na slici 3.6. Prilikom stvaranja konvolucijskih neuronskih mreža, u prvom sloju se obavljaju operacije konvolucije pomoću filtera čija brojnost definira broj detektora značajki koji stvaraju pripadajuće karte značajki. Povećanjem broja filtera također se povećava rizik od pretjeranog prilagođavanja podataka [40].



Slika 3.6 Struktura konvolucijskog sloja u modelu CNN [41].

Navedeni sloj određuju parametri kao što su dubina izlaznog volumena pomoću kojeg je moguće kontrolirati broj neurona povezanih na istom području ulaznog volumena. Svi neuroni bit će aktivirani prilikom izračuna različitih značajki ulazne slike. Putem koraka pomaka filtera opisuje se prostorna dubina stupaca koji su zahvaćeni filterom. Ako je za korak pomaka filtera definirana vrijednost 1, tada će filter prelaziti svaki pojedini piksel slike. Tada će rezultirajući volumen obuhvaćati prostor manjih dimenzija zbog manje preklapljenosti receptivnih polja. S posljednjim parametrom, opisuje se širenje ulaznog volumena pomoću polja koja sadrže vrijednost nula (engl. *zero-padding*), kako bi se očuvale prostorne dimenzije i dizajn dublje mreže. Središnji piksel filtera je tada pozicioniran kao rubni piksel slike što dovodi do sprječavanja smanjenja karti značajki [40].

Pomoću jednadžbe 3-6 izražava se izlaz konvolucijskog sloja:

$$M_j^p = f \left( \sum_{i \in M_j} M_i^{p-1} * k_{ij}^p + N_j^p \right) \quad (3 - 6)$$

pri čemu  $M_j$  označava ulazne karte značajki,  $p$  predstavlja redni broj sloja,  $k_{ij}$  predstavlja konvolucijsku matricu dok je  $N_j$  slobodni koeficijent. Težina konvolucijske matrice i slobodni koeficijent predstavljaju parametre čija je vrijednost rezultat treniranja mreža pomoću metoda nenadziranog strojnog učenja.

Korištenjem konvolucijskog sloja, osigurava se invarijantnost prostornih pozicija objekta te lokalna povezanost koja nastaje korelacijom susjednih piksela. Smanjenje ukupnog broja parametara moguće je postići mehanizmom dijeljenja težina na pojedinoj karti značajki [39].

Poput ostalih neuronskih mreža, u konvolucijskoj neuronskoj mreži zastupljena je uporaba nelinearne aktivacijske funkcije nakon dobivenih vrijednosti u konvolucijskom sloju. Njezina uporaba se definira u pojedinom konvolucijskom sloju, no moguće je kreirati zaseban sloj zbog ostvarenja veće fleksibilnosti implementiranog modela [40].

### 3.3.2.3. Aktivacijske funkcije

Svrha aktivacijske funkcije je uvođenje nelinearnosti u mrežu koja tada može učiti iz skupova podataka, provoditi složene izračune te predviđati. Prethodi joj linearna transformacija svake matematičke operacije koja se izvodi prilikom treniranja modela [42]. Osim što prilikom treniranja

i određivanja performansi pojedine mreže aktivacijske funkcije imaju integralnu ulogu, predstavljaju matematički filter između neurona u trenutnom i idućem sloju [43]. *ReLU* aktivacijska funkcija definirana izrazom 3-7 određuje standard prilikom svake primjene:

$$f(x) = \max(0, x) \quad (3 - 7)$$

pri čemu povratna vrijednost iznosi nula u slučaju negativne vrijednosti ulaznog podatka, dok je pozitivna vrijednost podatka vraćena u istom iznosu. Ograničenost domene definirana je u intervalu  $[\approx -0.31, \infty)$  [42]. *ReLU* funkcija ostvaruje brzo konvergiranje mreže te provedbu unazadne propagacije greške osim kada je vrijednost gradijenta funkcije nula [43], a moguće ju je zamijeniti skalarnom funkcijom koja sadrži broječanu vrijednost podataka pri tome ne mijenjajući arhitekturu mreža dubokog učenja [42].

Iako definirana na istom intervalu kao *ReLU* funkcija, zaglađena i nemonotona Swish aktivacijska funkcija ostvaruje bolje performanse ili ih u iznimnim slučajevima dosljedno prati. Ramachandran i sur. su na *ImageNet* bazi podataka proveli istraživanje performanse Swish funkcije te zaključili kako njena uporaba rezultira povećanjem preciznosti klasifikacije za 0.9%. Definirana je izrazom 3-8:

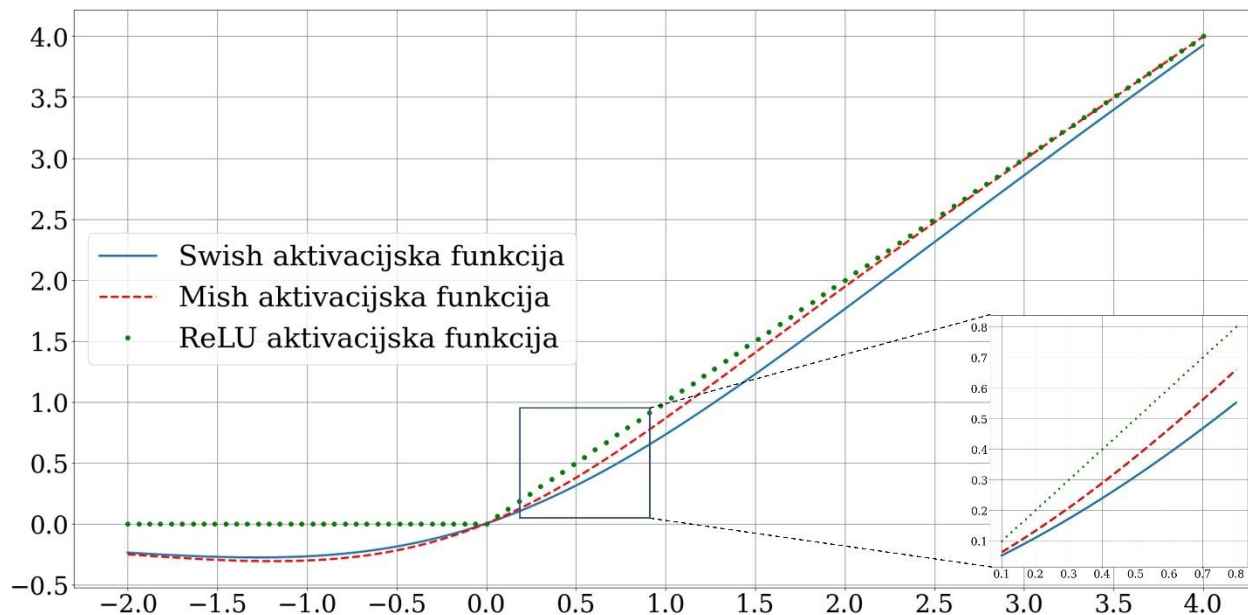
$$f(x) = x * \text{sigmoid}(\beta x) \quad (3 - 8)$$

u kojem simbol  $\beta$  označava konstantu ili parametar definiran za sigmoidnu funkciju pri treniranju mreže. Swish aktivacijska funkcija može poprimiti oblik *ReLU* kada parametar  $\beta$  teži beskonačnosti. Točnije, interpolira među linearnom i *ReLU* funkcijom. Najvažnija razlika između promatranih funkcija je nemonotoničan „skok“ vrijednosti u Swish aktivacijskoj funkciji u trenutku preaktivacijskog procesa kada su vrijednosti na osi apscisa negativne [42].

Ispitivanjem performansi dubokih mreža na složenim podatkovnim skupovima, uočeno je da Mish aktivacijska funkcija donosi bolje rezultate od mreža koje implementiraju standardne funkcije poput *ReLU* i Swish. Po definiciji, Mish funkcija slična je Swish aktivacijskoj funkciji, no zbog ostvarivanja boljih performansi, točnijeg učinka i jednostavnije izvedbe, češći je odabir prilikom stvaranja modela dubokog strojnog učenja. Mish funkcija definirana je izrazom 3-9.

$$f(x) = x * \text{tanh}(\text{softplus}(x)) = x * \text{tanh}(\ln(1 + e^x)) \quad (3 - 9)$$

Osim što povećava preciznost klasifikacije, zaglađena i nemonotonična Mish funkcija omogućuje postavljanje niže vrijednosti brzine učenja (engl. *learning rate*) prilikom treniranja modela [43]. Grafička usporedba *ReLU*, Swish i Mish aktivacijske funkcije opisana je na slici 3.7.

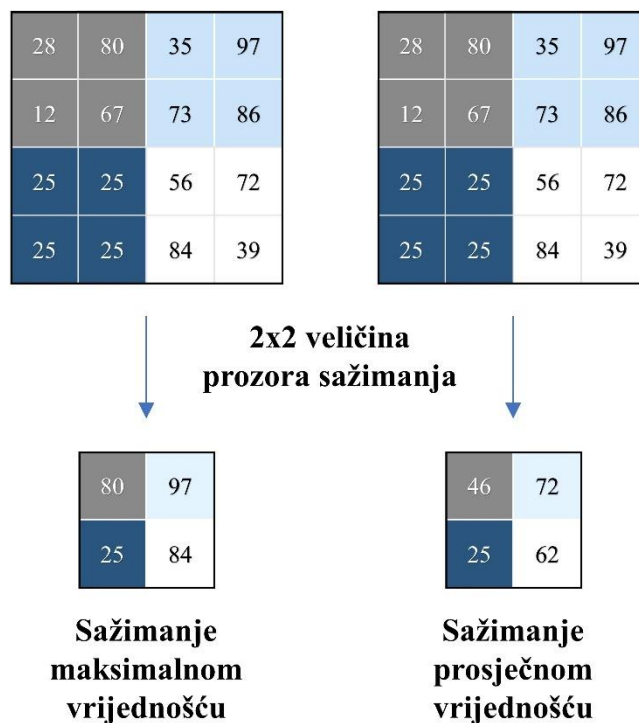


Slika 3.7 Grafički prikaz aktivacijskih funkcija.

#### 3.3.2.4. Sloj sažimanja

Radi pojednostavljivanja ili smanjivanja prostornih dimenzija informacija dobivenih iz karti značajki, u modelima konvolucijskih neuronskih mreža upotrebljava se sloj sažimanja. Osim sažimanja prosječnom vrijednošću i najčešće korištenog sažimanja maksimalnom vrijednošću za veću brzinu i poboljšanu konvergenciju, moguće je upotrebljavati *L2-norm* vrstu sloja na temelju različitih primjena. Na slici 3.8 prikazan je pomak filtera veličine 2x2 preko ulaznog volumena, gdje je rezultat postupka prosječna ili maksimalna vrijednost podregija [39]. Drugačijom bojom su prikazani mogući položaji filtera tijekom sažimanja.

Različito od konvolucijskog sloja u kojem se transformacija odvija pomoću konvolucijskog kernela, u sloju sažimanja se operacije odvijaju putem tenzora sažimanja [44].



Slika 3.8 Primjeri sažimanja vrijednosti u sloju sažimanja CNN modela.

### 3.3.2.5. Potpuno povezani sloj

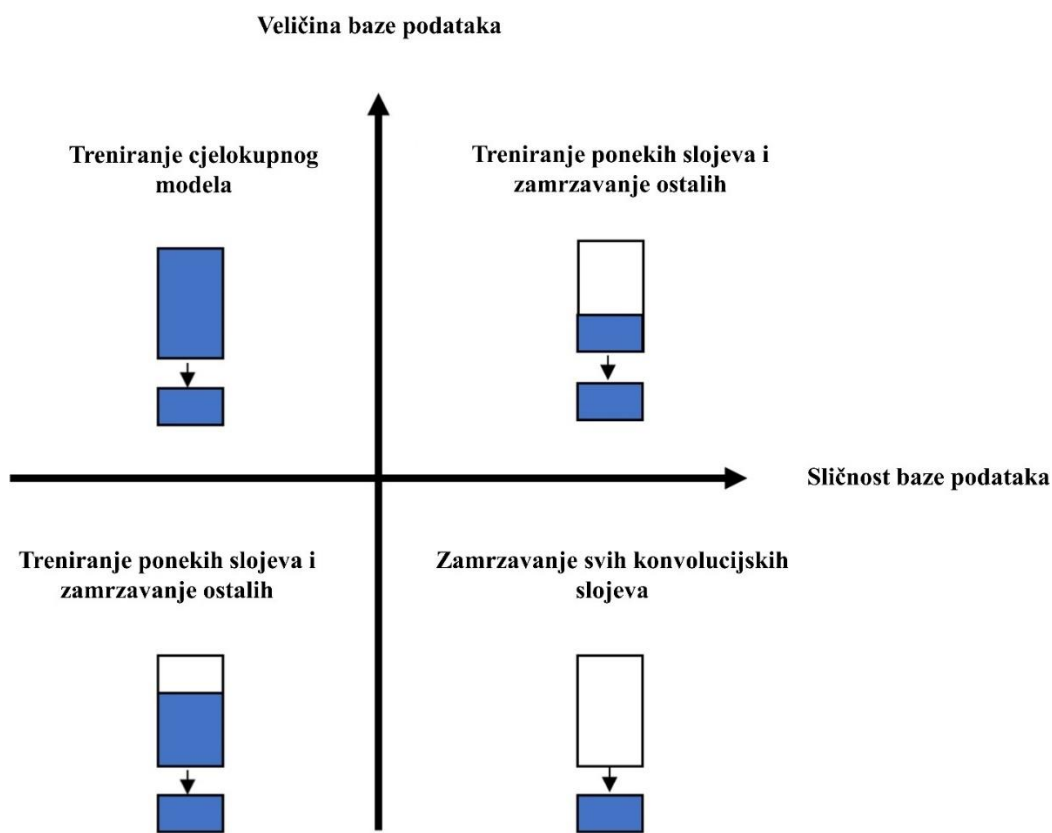
Prilikom implementacije konvolucijske neuronske mreže, njezin posljednji sloj je potpuno povezani sloj (engl. *fully connected layer*). Njegova glavna uloga je klasifikacija značajki proizašlih iz niza konvolucijskih slojeva čija je uloga izdvajanje značajki i slojeva sažimanja koji provode sažimanje vrijednosti tih značajki maksimalnom ili prosječnom vrijednošću [40]. Svoj naziv temelji na povezivanju svakog ulaza i svakog izlaza mreže koji zasebno stvaraju vektor. U ukupnom broju parametara modela, ovaj sloj ih sadržava približno 90%.

Posljednji sloj sažimanja u mreži čini ulaz u potpuno povezani sloj, sastojeći se od vektora unaprijed poznate veličine koji određuje broj klasa. Prosljeđivanje navedenih vektora u specifične kategorije moguće je ostvariti implementacijom softmax funkcije koja prevođenjem numeričkih vrijednosti izračunava distribuciju vjerojatnosti, a primjenu je moguće analizirati pomoću VGGNet modela. Izlazom iz pojedine mreže, prikazuje se visina točnosti klasifikacije pojedinog primjerka u bazi podataka namijenjenog za validaciju. Navedeni izlaz mreže može se primijeniti u ostalim modelima strojnog učenja, a da se pri tome ne mora provesti postupak normalizacije.

### 3.4. Prijenosno strojno učenje

Koristeći prijenosno strojno učenje moguće je postići visoke točnosti i bolje performanse u usporedbi s modelima koji su trenirani po prvi put. Takvi modeli, prethodno su trenirani na *ImageNet* skupu podataka koji je javno dostupan, a sadržava tisuće slika. Prethodnim treniranjem, modeli sadržavaju već naučene težine pomoću kojih je moguće kreirati karte značajki prije treniranja modela na specijaliziranom skupu podataka za koje se drastično smanjuje vrijeme treniranja. Vrijednosti hiperparametara kod prijenosnog strojnog učenja sadržavaju standardizirane vrijednosti [34,38].

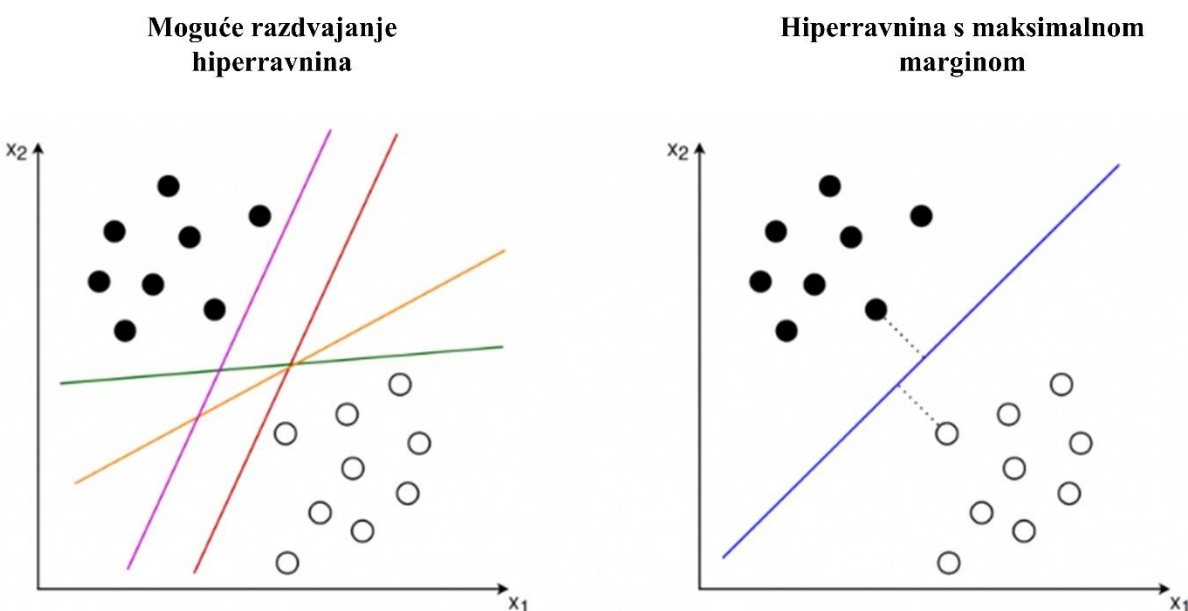
Na slici 3.9, prikazane su moguće opcije prilikom korištenja prijenosnog strojnog učenja. Uzimajući u obzir veličinu baze podataka i sličnosti među podacima, moguće je zamrzavanje pojedinih slojeva mreže čime se zadržavaju već naučene težine. U ovome radu, zamrznuli su se slojevi osnovnog modela, dok je ostatak mreže ponovno treniran, a dodani su novi klasifikatori poput metode potpornih vektora ili razvijeni IncMB modul u cilju ostvarivanja visokih performansi.



Slika 3.9 Mogući načini upotrebe prijenosnog strojnog učenja.

### 3.5. Upotreba metoda klasičnog u dubokom strojnom učenju

Najčešće korištena metoda klasičnog strojnog učenja prije pojave dubokog učenja, metoda potpornih vektora ostvaruje visoke točnosti prilikom klasifikacije raznih skupova podataka. Metoda je dio nenadziranog strojnog učenja s ciljem pronalaska hiperravnine koja će najtočnije odvojiti promatrane klase putem određivanja najveće moguće margine, a navedeno je prikazano na slici 3.10. Računanje maksimalne udaljenosti, provodi se računanjem udaljenosti između dva vektora. Prilikom provođenja klasifikacije bolesti biljke cassave, ova metoda je ostvarila najbolje performanse od onih analiziranih, a taj model je potom ugrađen u Android uređaje u svrhu pravovremenog i točnog raspoznavanja bolesti [24].



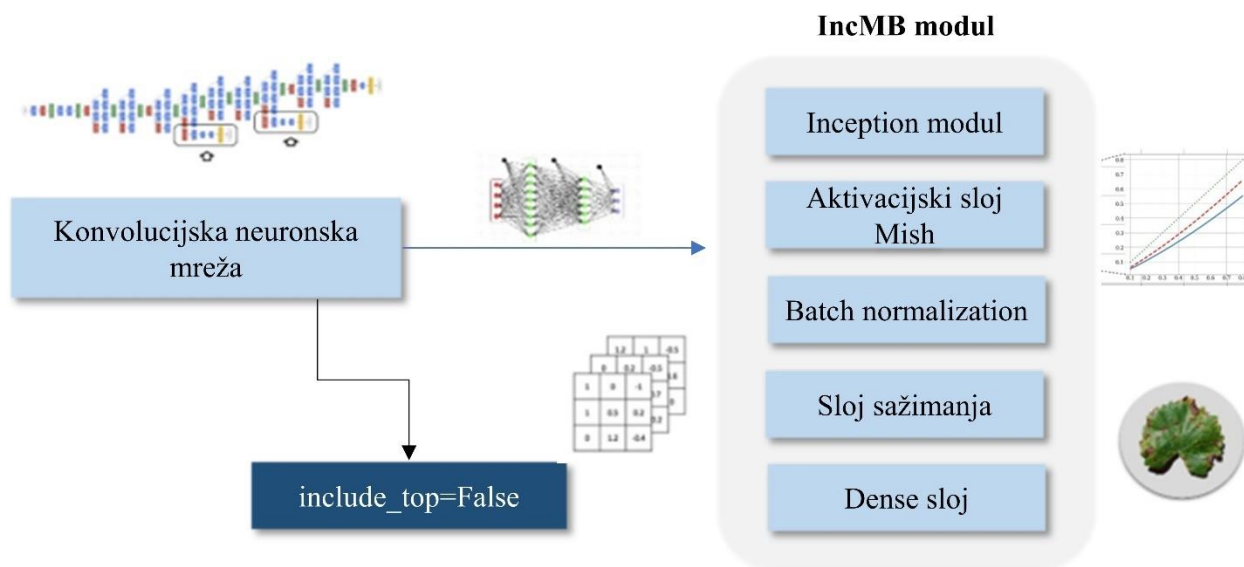
Slika 3.10 Metoda potpornih vektora [45].

Kao ponajbolji klasifikator klasičnog strojnog učenja, analizirane su performanse metode potpornih vektora skupa s osnovnom mrežom dubokog strojnog učenja kako bi se istražio način rada spoja metoda klasičnog i dubokog strojnog učenja. Pri tome je korišteno prijenosno strojno učenje sa zamrznutim konvolucijskim slojevima.



### 3.6. Razvijeni IncMB modul

Ideja stvaranja IncMB modula jest omogućiti jedinstveni sustav slojeva ugradivog u sve konvolucijske neuronske mreže u svrhu ostvarivanja boljih performansi i većih točnosti prilikom postupka klasifikacije raznih skupova podataka. Navedeni pristup je računalno skup zahtijevajući ogromne računalne resurse, a uz navedeno je dugotrajan. U usporedbi konvolucijskih neuronskih mreža s IncMB modulom i treniranih mreža bez modula ili mreža dubokog strojnog učenja i spoja s metodom potpunih vektora koja pripada metodama klasičnog strojnog učenja, ideja je pokazati da je inačica duboke mreže s IncMB modulom točnija i preciznija za problem raspoznavanja bolesti lista rajčice. Naziv IncMB modula obuhvaća najvažnije dijelove blokovskog sustava, a to su, uz Inception modul, korištenje aktivacijske funkcije Mish u zasebnom sloju [43] i *batch normalization* sloj. Struktura razvijenog modula, prikazana je na slici 3.11., a modul je implementiran koristeći Tensorflow i Keras biblioteke. Eksperimentalno vrednovanje provedeno je na isti način kao na ostalim inačicama mreža opisanih u ovome radu.



Slika 3.11 Struktura razvijenog IncMB modula.

Kako bi bio ugradiv u sve modele, prethodni slojevi konvolucijske neuronske mreže su zamrznuti, a na njezin kraj postavljen je modul. Inspiriran arhitekturom mreže Inception, na osnovni gradivni dio Inception modula, postavljen je sloj s nemonotoničnom Mish funkcijom koja ostvaruje bolje performanse u usporedbi s najčešće korištenom *ReLU* i Swish aktivacijskom funkcijom [43]. Potom je na prethodni aktivacijski sloj primijenjen *batch normalization* sloj koji regulira učenje

mreže te smanjuje pogrešku prilikom propagacije. Radi smanjivanja prostornih dimenzija informacija dobivenih iz karti značajki, upotrijebio se globalni sloj sažimanja [44]. Potom je implementiran *dense* sloj u kojem svaki neuron dobiva informacije od svih neurona prethodnog sloja.

### **3.7. Postupci analize i usporedba performansi**

Produbljivanje CNN mreže, a time i povećanje složenosti, otežava optimizaciju mreže te dovodi do povećanja vjerojatnosti za pretjeranim prilagođavanjem podataka (engl. *overfitting*). Ovisno o domeni i količini ulaznih podataka, navedeno se može izbjeći, no zapažene bolje rezultate, u algoritmima dubokog učenja u usporedbi s metodama klasičnog učenja, teže je interpretirati. Postupak izdvajanja značajki dubokim učenjem doprinuo bi boljem razumijevanju dijagnostičkih postupaka u klasifikaciji bolesti rajčice i ostalih, ali za sada ostaje nepoznat (engl. *black box*). Pri postupcima analize, moguće je koristiti metode vizualizacije gdje bi iz pojedinih iteracija treniranja bilo moguće izdvojiti slikovni podatak kakvog model dubokog učenja trenutno „vidi“. Također, specijaliziranim algoritmima vizualizacije, moguće je lokalizirati i prepoznati simptome bolesti na području zaraženih dijelova lista. Zbog potpunog nerazumijevanja ovog procesa, modela dubokog strojnog učenja najčešće se evaluira pomoću preciznosti, odziva, F1-mjere i točnosti koje je moguće izvesti iz vrijednosti matrice zabune koja prikazuje broj identificiranih i neidentificiranih podataka ovisno o dodijeljenoj klasi koju je odredio klasifikator i stvarnoj, kojoj zaista pripadaju. Analizu je moguće upotpuniti grafičkim prikazima točnosti treniranja i validacije, kao i gubicima pri treniranju i validaciji.

#### **3.7.1. Matrica zabune**

Matrica zabune je kvadratna matrica koja pruža vizualni i sažeti prikaz performansi klasifikatora. Može se koristiti pri vrednovanju binarne i višeklase klasifikacije. Sastoji se od redaka koji predstavljaju identificiranu ili neidentificiranu stvarnu klasu podatka i stupaca koji predstavljaju klase koje je klasifikator dodijelio podatku. Moguće vrijednost klase i identifikacije algoritma prikazani su na slici 3.12. Matricu je moguće zapisati i u obratnom obliku s promijenjenim iznosima izračunatih mjera. Broj ispravnih klasifikacija određuje se prema vrijednostima na glavnoj dijagonali čije visoke vrijednosti osiguravaju visoku točnost i preciznost modela, dok se izvan nje nalaze niske vrijednosti [46].

Navedena analiza koristi se pri prepoznavanju slučajeva u kojima model ostvaruje visoke točnosti, no pogrešno klasificira iste podatke. U tom slučaju, parametre je potrebno ponovno podesiti kako bi model ostavio bolje performanse [46].

	Dodijeljena klasa IDENTIFICIRANO	Dodijeljena klasa NEIDENTIFICIRANO
Stvarna klasa IDENTIFICIRANO	TP	FN
Stvarna klasa NEIDENTIFICIRANO	FP	TN

Legenda	Vrijednost klase	Identifikacija algoritma
TP (engl. True Positives)	1	1
FN (engl. False Negatives)	1	0
FP (engl. False Positives)	0	1
TN (engl. True Negatives)	0	0

Slika 3.12 Matrica zabune s popisom mogućih slučajeva.

TP predstavlja broj listova koji su ispravno klasificirani kao bolesni gdje zaražene listove karakterizira promjena boje u žutu ili smeđu, promjena oblika lista poput šupljikavosti te uvenuće biljke.

FN ili greška Tipa II predstavlja broj listova koji su pogrešno klasificirani kao zdravi, dok u stvarnosti pripadaju jednoj od klasa koje određuju bolest.

FP ili greška Tipa I predstavlja broj pogrešno klasificiranih listova kao bolesnih, dok su oni u stvarnosti zdravi.

TN predstavlja broj ispravno klasificiranih listova koji su zdravi, a koji ne pripadaju niti jednoj klasi koja određuje bolest.

### 3.7.2. Mjere za vrednovanje klasifikatora

Preciznost (engl. *precision*) predstavlja udio točno klasificiranih instanci u ukupnom broju dohvaćenih instanci iste kategorije. Vrijednost pokazuje točnost predviđanja klasifikatora koji podacima dodjeljuje klase, a da su one ujedno i stvarne. Naziva se i pozitivna prediktivna vrijednost [47], a prikazana je izrazom 3-10.

$$P = \frac{TP}{TP + FP} \quad (3 - 10)$$

Odziv (engl. *recall*) predstavlja udio točno klasificiranih instanci otkrivenih u ukupnom broju instanci iste kategorije. Prikazuje točnost kojom model prepoznaje pojedinu kategoriju [47], a prikazuje ju izraz 3-11.

$$R = \frac{TP}{TP + FN} \quad (3 - 11)$$

F1-mjera (engl. *F1-score*) je harmonijska sredina preciznosti i odziva, a predstavlja rezultat točnosti testa. Opisuje ju izraz 3-12. Najbolja preciznost i odziv ostvaruje najvišu vrijednost F1-mjera koja iznosi 1, dok suprotno, najlošija preciznost i odziv ostvaruje najnižu vrijednost F1-mjera koja iznosi 0. Također, F1-mjera usklađujući vrijednosti preciznosti i odziva predstavlja bolju mjeru za vrednovanje klasifikatora od točnosti kada su podaci neravnomjerno raspoređeni u promatranim kategorijama [48].

$$F1\text{-mjera} = 2 * \frac{P * R}{P + R} \quad (3 - 12)$$

Točnost (engl. *accuracy*) je omjer broja točno klasificiranih podataka u ukupnom broju ulaznih podataka. Opisuje broj točnih predikcija prilikom treniranja i validacije algoritama [49], a prikazana je izrazom 3-13.

$$A = \frac{TP + TN}{TP + FP + TN + FN} \quad (3 - 13)$$

## **4. EKSPERIMENTALNA ANALIZA I USPOREDBA PERFORMANSI KLASIFIKATORA**

Za svaki se sustav, prilikom njegova nastajanja, definira točan i opširan skup zahtjeva koje sustav mora zadovoljiti kako bi bio učinkovit, ostvario svoju svrhu i doprinuo razvoju područja za koje je namijenjen. Prilikom neispunjenja nekog od zahtjeva sustava za raspoznavanje bolesnih jedinki rajčice, moguće je kako predviđene dijagnoze neće biti točne, a time će i predložen tretman biti neučinkovit i pogrešan. Loše performanse sustava, vremenom će dovesti do širenja bolesti i pogoršanja epidemije [50]. Zbog visoke točnosti i performansi, konvolucijska neuronska mreža koja je metoda dubokog strojnog učenja, zauzela je vodeću poziciju za upotrebu u problemima klasifikacije na raznim skupovima podataka. Iako primarno, ostvarena točnost ovisi o kvaliteti podataka, također ovisi o broju konvolucijskih slojeva i načinu izdvajanja značajki [50].

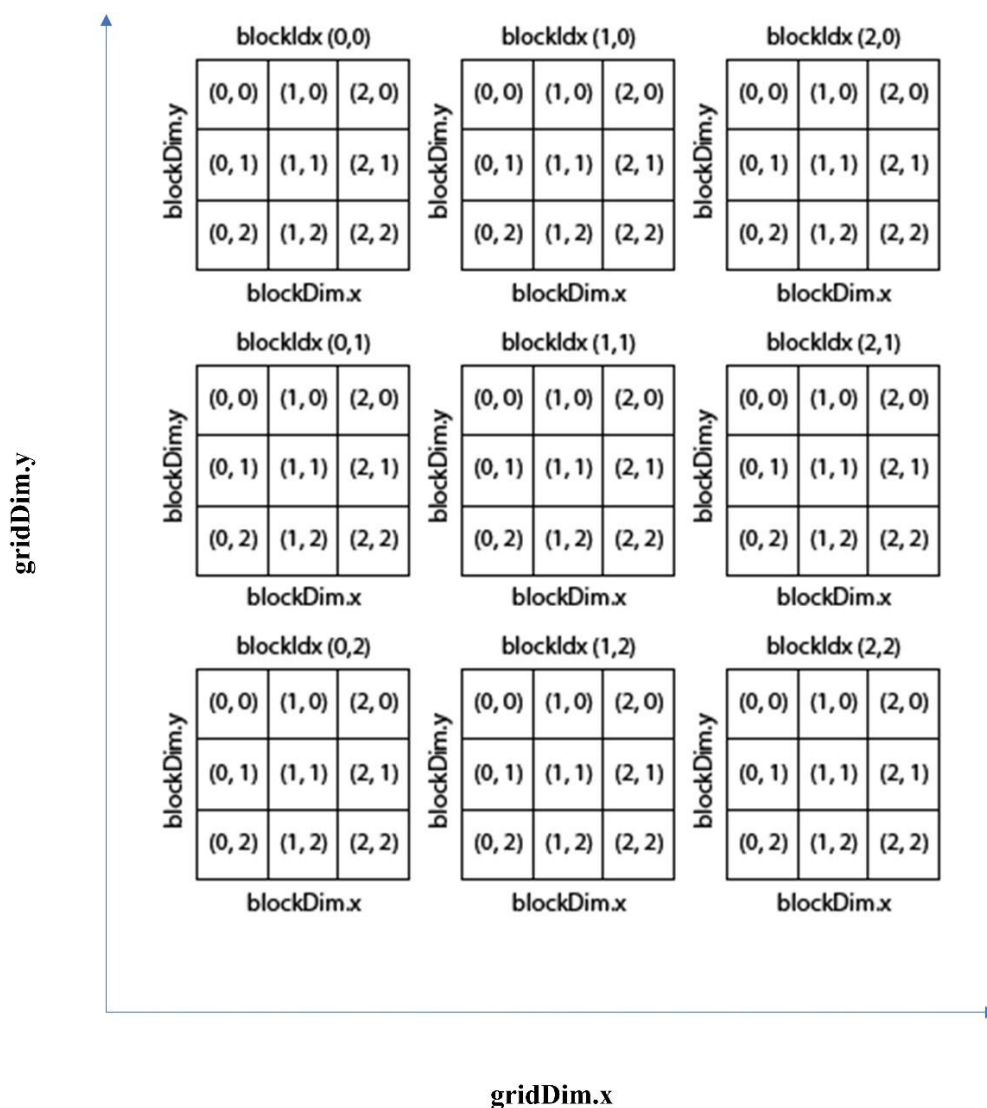
U ovome poglavlju, nakon opisa zahtjeva na sustav i korištenih tehnologija, prikazat će se uvjeti i oprema pod kojom je proveden eksperiment. Nadalje, prikazat će se i diskutirati rezultati koje su ostvarile implementacije pojedine mreže, njezine inačice s IncMB modulom i inačice s metodom potpornih vektora. Naposljetku, predložit će se mogućnosti poboljšanja eksperimenta s naglaskom na IncMB modul.

### **4.1. Zahtjevi na sustav**

Veliki broj podataka za kreiranje modela dubokog strojnog učenja i veliki broj parametara pojedine mreže predstavljaju izazov za središnju procesorsku jedinicu u svrhu efikasne obrade podataka. Pri tome, zbog pogodne arhitekture moguće je koristiti paralelizam kod grafičke procesorske jedinice kako bi se istovremeno obradio veći broj podataka pomoću procesa i niti na dvorazinskoj hijerarhiji. Takve procese moguće je modelirati pomoću CUDA (engl. *Compute Unified Device Architecture*) paralelne računarske platforme na arhitekturi NVIDIA grafičkoj kartici. CUDA je ujedno model aplikacijskog programskog sučelja koji kreira heterogeni sustav središnje i grafičke procesorske jedinice za obavljanje složenih postupaka. Za vrijeme obavljanja logičkih i matematičkih operacija kod središnje jedinice, grafička u isto vrijeme omogućuje paralelizam pomoću niti. Za izvođenje i treniranje dubokih neuronskih mreža, računanje konvolucije naprijed i natrag, normalizacije, aktivacijskih funkcija kao i ostalih matematičkih operacija koje se odvijaju u konvolucijskom sloju i sloju sažimanja, namijenjena je biblioteka cuDNN koja se temelji na

modelu CUDA. Nadalje, programske biblioteke korištene u radu, TensorFlow i Keras oslanjaju se na visoke performanse CUDA-inih biblioteka i grafičke procesorske jedinice [51].

Na slici 4.1 prikazan je paralelizam grafičke procesorske jedinice pomoću modela CUDA. Više niti se izvodi paralelno, a njihov skup se naziva blok dok više blokova čini koordinatnu mrežu. Svaki blok u koordinatnoj mreži sadrži isti broj niti. Pri tome, `gridDim.x` označava broj blokova u mreži, a `blockIdx.x` označava indeks trenutnog bloka u mreži. Unutar bloka, `blockDim.x` opisuje broj niti u bloku. Svaka od njih ima vrijednost `threadIdx.x` koja sadrži indeks niti [52].



Slika 4.1 Korištenje paralelne računarske platforme CUDA [53]

Prema izrazu 4-1, moguće je mapirati niti i vrijednost pojedinog elementa [52].

$$int\ idx = threadIdx.x + blockIdx.x * blockDim.x \quad (4 - 1)$$

## 4.2. Opis platformi, alata i tehnologija

Stvaranjem složenijih neuronskih mreža pomoću velikog skupa neurona, povećala se točnost modela kao i ostalih performansi. Optimizacijske tehnike koje su specijalizirane za sklopovlje i algoritme omogućile su efikasnu obradu modela dubokog učenja i velikih skupova podataka [54]. U nastavku je dan opis korištenih tehnologija uz prikaz prednosti njihove upotrebe za implementaciju mreža dubokog strojnog učenja.

### 4.2.1. Python i Anaconda

Zbog širokog raspona odabira biblioteka s prethodno implementiranim modulima, time olakšavajući ostvarivanje pojedine funkcionalnosti koda, objektno-orijentirani programski jezik Python jedan je od najkorištenijih prilikom izrađivanja mreža dubokog strojnog učenja. Uz navedeni, podržava proceduralni i funkcionalni stil programiranja. Opsežno je dokumentiran i jednostavne je sintakse, a programski je jezik više razine. Moguće ga je upotrebljavati skupa s drugim programskim jezicima kako bi se problem učinkovito riješio. Omogućava korištenje alata za vizualizaciju pomoću kojih se prikazuju podaci i uspoređuju učinkovitosti pojedinih modela. Za pregledniji i praktičniji način implementacije modela dubokog strojnog učenja, korištena je Anaconda - distribucija Python i R programskih jezika. Upravljanje paketima ostvareno je pomoću Anacondinog vlastitog sustava conda, zasebnog projekta otvorenog koda [55].

### 4.2.2. TensorFlow i Keras

TensorFlow je programski okvir otvorenog koda koji osim za implementaciju algoritama dubokog i klasičnog strojnog učenja, služi za implementaciju višeslojnih neuronskih mreža, stvaranje predikcije i rješavanje problema klasifikacije [56]. Izrađivanje i treniranje modela ostvaruje putem grafa protoka podataka čiji se skup kreiranih čvorova može izvoditi u isto vrijeme ili paralelno, dok mu je tok oblikovan usmjerenim granama. Nadalje, sami čvorovi (engl. actors) označavaju operacije koje se provode nad podacima. Na razini poslužitelja, najbolje rezultate ostvaruje izvođenjem višenitnih procesa. Zbog učestalih potreba za praktičnu primjenu rezultata izvođenja modela strojnog učenja, dizajnirana je inačica TensorFlow Lite za Android uređaje koja pruža

aplikacijsko programsko sučelje za izvršavanje operacija potrebnih za kreiranje modela. Svoje ime dobila je prema matematičkoj veličini tenzoru koja je određena s više od tri broja, a često ima i dodatna svojstva. Primjenjujući zakon transformacije tenzora, njegove komponente se mijenjaju prilikom prelaska između koordinantnih sustava [54]. Na najvišoj razini programskog okvira TensorFlow, izvodi se razvojni okvir (engl. *framework*) Keras koji je pisan u Pythonu, a olakšava implemetaciju slojeva neuronskih mreža, korištenje aktivacijskih funkcija i odgovarajuće optimizacije. Iako se treniranje modela dubokog strojnog učenja može izvoditi na središnjoj procesorskoj jedinici, zbog uštede i pravilnog rukovanja resursima, pomoću navedenih biblioteka može se izvoditi na grafičkoj procesorskoj jedinici s kombinaciji s CUDA paralelnoj računskoj platformi ili na tenzorskim procesnim jedinicama [57].

#### **4.2.3. NumPy, SciPy i Matplotlib**

Biblioteka pisana u Pythonu, NumPy, najčešće se koristi pri izvođenju računskih operacija u znanstvene svrhe. Omogućava algebarske operacije nad matricama i višedimenzionalnim nizovima kao i statističke te trigonometrijske operacije. Omogućava generiranje slučajnih brojeva i obavljanje trigonometrijskih funkcija poput Fourierove transformacije. Također, moguće je integriranje koda u C/C++ programske jezike pomoću odgovarajućih alata [58]. Biblioteka SciPy sadržava module pomoću kojih je moguće obrađivati signale, odrađivati računske operacije iz linearne algebre, interpolacije i optimizacije. Osnovna struktura podataka ove biblioteke su NumPy nizovi pa se navedene biblioteke često upotrebljavaju zajedno [59]. Biblioteka Matplotlib također je pisana u Pythonu, a služi za vizualizaciju podataka. Pomoću nje, moguće je izrađivati grafičke prikaze za opću namjenu pomoću objektno-orijentiranog aplikacijskog programskog sučelja.

#### **4.2.4. OpenCV**

OpenCv je programska biblioteka za kreiranje modela strojnog učenja i modela računalnog vida s primjenom u stvarnom vremenu. Biblioteka je otvorenog koda, a sadrži više od 2500 algoritama s visokom optimiziranošću koji primjenu nalaze u interakciji čovjeka i računala, prepoznavanju lica, emocija i gestikulacija, detekciji i praćenju objekata, proširenoj stvarnosti. Iako pisana u programskim jezicima C/C++, podržava i programske jezike Python, Javu i MATLAB [60].



#### 4.2.5. MATLAB

Programski jezik MATLAB upotrebljen je za postupak segmentacije slika bolesnih jedinki rajčice kako bi se odredila ukupna površina lista zahvaćena bolešću te odvojio i prepoznao bolesni dio lista od zdravog. Pri tome je korišten prerađeni algoritam koji je implementirao metodu k-najbližih susjeda. Sve računске operacije u MATLAB-u svedene su na računске operacije s matricama. Osim što omogućuje korištenje paradigmi objektno-orijentiranog programiranja, podržava i upotrebu programskog jezika C, Javu i SQL. Primjenjuje se prilikom analiziranja podataka, u implementaciji algoritama za strojno učenje i razvoju robotskog vida. Također, korišten je alat pri obradi signala, kreiranju matematičkih modela, robotici i segmentaciji slika [61].

#### 4.3. Provedba eksperimenta

Prilikom postavljanja eksperimenta, korišteni skup podataka podijeljen je u omjeru 70 : 30 za trening i validacijski skup podataka pri čemu je njihova podjela odrađena nasumično. Ukupan broj primjera za treniranje iznosi 2640, dok za validaciju iznosi 660. Sve slike su dimenzionirane prilikom predobrade u veličinu 224x224 piksela te su u boji. Tijekom provedbe eksperimenta, korišteno je prijenosno duboko strojno učenje, četiri modela te tri inačice svakog od njih. Osim inačice pojedine mreže, implementirana je inačica s razvijenim IncMB modulom kao i inačica s metodom potpornih vektora. Promatrane klase sadrže primjere bolesnih jedinki rajčice i to koncentrične pjegavosti lista, plamenjače, septoriozne pjegavosti, koprivine grinje te dvije virusne bolesti, virus mozaika te virus žućenja i kovrčavosti lista. Vrijednosti postavljenih hiperparametara su predstavljene u tablici 4.1.

Tablica 4.1 Vrijednosti hiperparametara.

Naziv hiperparametra	Vrijednost
Stopa učenja	<i>ReduceLROnPlateau</i> (automatizirano)
Broj epoha	30
Batch size	32
Optimizacijski algoritam	Adam

Za provedbu eksperimenta korištena je platforma računarstva u oblaku, Google Colab koja omogućuje besplatan pristup grafičkim procesorskim jedinicama NVIDIA Tesla K80 s 12 GB

memorije. Pomoću Keras-GPU i TensorFlow-GPU paketa, odrađena je predobrada podataka i implementacija modela. Treniranje svakog modela je u prosjeku trajalo 5 sati.

#### 4.4. Rezultati i diskusija

Eksperimentalna analiza provedena je na četiri različite mrežne arhitekture koristeći prijenosno duboko strojno učenje kojim su modeli prethodno trenirani na javno dostupnom skupu podataka *ImageNet* čiji je svaki čvor u hijerarhiji prikazan tisućama slika. Time modeli sadržavaju naučene težine iz kojih se kreiraju karte značajki prije treniranja na odabranom skupu podataka iz *New Plant Diseases Dataset*-a [17]. Vrijednosti hiperparametara kod prethodno treniranih konvolucijskih neuronskih mreža sadrže standardizirane vrijednosti. Treniranje i validacija pojedine mreže, njezine inačice s IncMB modulom i inačice s metodom potpornih vektora s ciljem ostvarenja visokih performansi pri raspoznavanju šest bolesti rajčice, ostvarila je rezultate prikazane u tablici 4.2, a mjere korištene za vrednovanje pri tome su točnost, odziv, preciznost i F1-mjera.

Tablica 4.2 Vrednovanje performansi modela.

Model	F1-mjera	Preciznost	Odziv	Točnost
InceptionV3	0.9329	0.9338	0.9335	0.9333
InceptionV3 + SVM	0.8866	0.8976	0.8888	0.8889
InceptionV3 + IncMB	0.9778	0.9779	0.9778	0.9778
InceptionResNetV2	0.9662	0.9673	0.9667	0.9667
InceptionResNetV2 + SVM	0.9309	0.9337	0.9312	0.9311
InceptionResNetV2 + IncMB	0.9733	0.9734	0.9733	0.9733
MobileNetV2	0.9125	0.9325	0.9155	0.9156
MobileNetV2 + SVM	0.8443	0.8829	0.8445	0.8444
MobileNetV2 + IncMB	0.9496	0.9502	0.9500	0.9500
DenseNet201	0.8569	0.8610	0.8588	0.8589
DenseNet201 + SVM	0.9247	0.9288	0.9246	0.9244
DenseNet201 + IncMB	0.9195	0.9198	0.9200	0.9200

Prema analizi rezultata, s točnošću od 97.78%, najbolje performanse ostvarila je InceptionV3 mreža koja sadrži IncMB modul. Slijede ju InceptionResNetV2 i MobileNetV2 mrežna arhitektura, obje u kombinaciji s IncMB modulom. Uspoređujući pojedinu mrežu, njezinu inačicu s IncMB modulom i inačicu s metodom potpornih vektora, u 4/5 slučajeva, najbolju točnost ostvarila je mreža s IncMB modulom. U preostalom slučaju, točnost inačice DenseNet201 mreže s IncMB modulom iznosila je 92.00%, dok je u kombinaciji s metodom potpornih vektora iznosila

92.44%. U usporedbi ostalih metoda vrednovanja rezultata za taj slučaj, primijećeno je kako mreža s IncMB modulom ostvaruje zapažene rezultate te povećava performanse klasifikatora pri raspoznavanju bolesti rajčice. Daljnim razvojem, performanse je moguće poboljšati proširivanjem korištenog skupa podataka i/ili korištenjem drugačijeg optimizatora.

Iako je korišteni skup podataka u eksperimentu uravnotežen po broju upotrebljenih primjera bolesnih jedinki iz pojedine kategorije, izrađena je analiza vrijednosti F1-mjere za pojedinu bolest rajčice jer je zbog harmonijske sredine preciznosti i odziva objektivn pokazatelj točnosti testa. U tablici 4.3, prikazane su osvarene vrijednosti F1-mjere za pojedine bolesti rajčice za kombinaciju mrežne arhitekture i razvijenog IncMB modula.

Tablica 4.3 Vrijednosti F1-mjere za bolesti rajčice. (A) koncentrična pjegavost lista; (B) plamenjača; (C) septoriozna pjegavost; (D) koprivina grinja; (E) virus mozaika; (F) virus žućenja i kovrčavosti lista.

Bolest rajčice	InceptionV3 + IncMB	InceptionResNet V2 + IncMB	MobileNetV2 + IncMB	DenseNet201 + IncMB
A	0.9498	0.9533	0.8993	0.8407
B	0.9637	0.9530	0.9408	0.8829
C	0.9797	0.9701	0.9199	0.9247
D	0.9834	0.9835	0.9610	0.9373
E	0.9967	0.9865	0.9933	0.9699
F	0.9934	0.9934	0.9835	0.9615

Prema ostvarenim rezultatima, najveća vrijednost F1-mjere primijećena je kod kategorije s primjerima jedinki oboljelih od virusa mozaika, a potom i od virusa žućenja i kovrčavosti lista. Algoritmi dubokog strojnog učenja bolje prepoznaju primjere navedenih klasa vjerojatno zbog specifičnosti simptoma virusnih bolesti. Naime, kod jedinki oboljelih od virusa mozaika glavni simptom je žućenje listova dok kod virusa žućenja i kovrčavosti lista dodatno dolazi do zakrivljenosti lisne površine. Kod ostalih promatranih bolesti, glavni simptom su crne točkice i/ili crni krugovi nepravilnog oblika pa zbog preklapanja simptoma, ali i prisutnosti istih u raznim fazama razvoja, algoritmi teže raspoznaju od čega je jedinka rajčice oboljela.

Kako bi se pri treniranju i validaciji modela osiguralo da nije došlo do prevelikog (engl. *overfitting*) ili premalog (engl. *underfitting*) prilagođavanja podataka, u tablici 4.4 prikazane su vrijednosti točnosti treniranja i validacije te gubitci validacije tijekom 15 i 30 epoha.

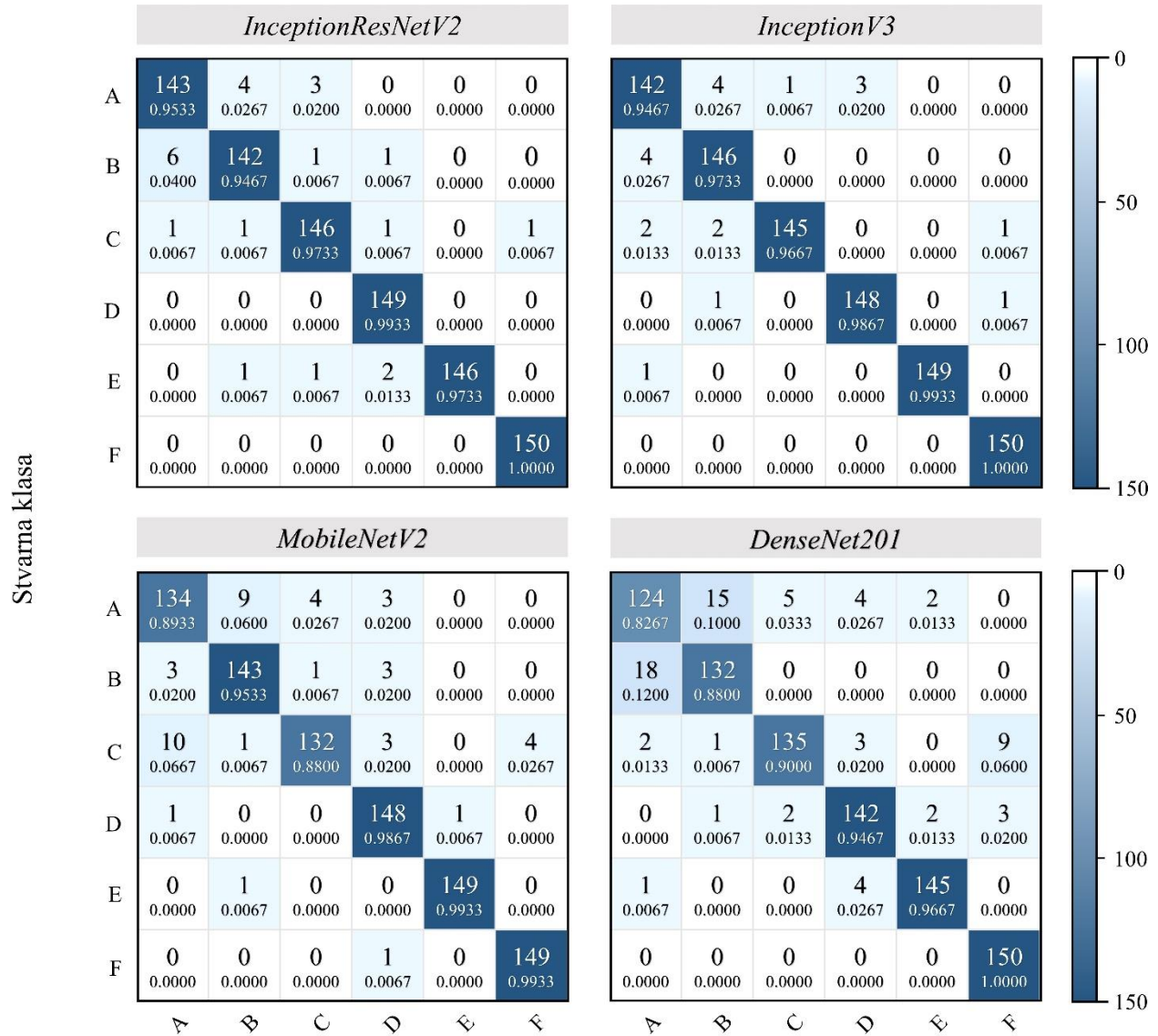
Tablica 4.4 Točnost treniranja/validacije i gubitak validacije tijekom 15 i 30 epoha. (TT) točnost treniranja; (TV) točnost validacije; (GV) gubitak validacije.

Model	15 epoha			30 epoha		
	TT	TV	GV	TT	TV	GV
InceptionV3	0.9529	0.8650	1.2822	0.9859	0.9330	0.3108
InceptionV3 + SVM	0.9890	0.8549	0.5262	0.9956	0.8884	0.3553
InceptionV3 + IncMB	0.9743	0.9143	0.7250	0.9996	0.9762	0.2369
InceptionResNetV2	0.8813	0.8125	0.6906	0.9862	0.9665	0.2297
InceptionResNetV2 + SVM	0.9496	0.8411	0.8032	0.9950	0.9344	0.0567
InceptionResNetV2 + IncMB	0.9325	0.8578	1.3071	0.9983	0.9733	0.1889
MobileNetV2	0.9892	0.9167	1.0077	0.9967	0.9156	0.5614
MobileNetV2 + SVM	0.8908	0.8074	0.4561	0.8935	0.8380	0.3068
MobileNetV2 + IncMB	0.9235	0.8750	0.4299	0.9693	0.9489	0.2386
DenseNet201	0.6874	0.6060	0.5483	0.9122	0.8583	0.4427
DenseNet201 + SVM	0.8154	0.8811	0.4195	0.9063	0.9244	0.2126
DenseNet201 + IncMB	0.8104	0.8678	0.4792	0.8942	0.9200	0.2657

Tijekom procesa treniranja, model u svim epohama uči na primjerima iz korištenog skupa podataka. Validacija se provodi na primjerima iz skupa podataka koji nisu dio trening skupa. Povećanjem broj epoha, povećavaju se vrijednosti točnosti treniranja kao i točnosti validacije koje su u gotovo svim slučajevima nezamjetno manjih vrijednosti od onih treniranja. Pri tome se i gubitak validacije smanjuje povećanjem epoha što potvrđuje trend učenja algoritma. Prilikom treniranja, nema značajih odstupanja u spomenutim vrijednostima. Proces treniranja i validacije je optimiziran i uredno proveden.

Izračunom vrijednosti mjera za vrednovanje klasifikatora s IncMB modulom, ostvaren je vizualni prikaz rezultata u obliku matrice zabune prikazane na slici 4.2.

Prema apsolutnim i relativnim vrijednostima na glavnoj dijagonali dobiven je broj ispravnih klasifikacija čije visoke vrijednosti osiguravaju visoku točnost i preciznost modela. U matrici zabune, uočavaju se vrijednosti pogrešno klasificiranih primjera bolesnih jedinki i to kod koncentrične pjegavosti lista, plamenjače, septoriozne pjegavosti i koprivine grinje, bolesti koje karakteriziraju slični simptomi poput pojave crnih točkica na površini lista. Kod raspoznavanja virusa mozaika te virusa žućenja i kovrčavosti lista, najmanji je broj pogrešno klasificiranih primjera jedinki. Najbolje performanse i najveći broj točno klasificiranih primjera ostvaruje InceptionV3 mreža s IncMB modulom, a slijedi ju InceptionResNetV2 mreža s IncMB modulom. Najlošije performanse ostvaruje inačica DenseNet201 mreže s IncMB modulom.



Slika 4.2 Matrica zabune za klasifikaciju bolesti rajčice. (A) koncentrična pjegavost lista; (B) plamenjača; (C) septoriozna pjegavost; (D) koprivina grinja; (E) virus mozaika; (F) virus žućenja i kovrčavosti lista.

#### **4.4.1. Analiza ostvarenih rezultata pojedinih mreža**

Najbolju točnost uspoređujući performanse pojedinih mreža, ostvaruje InceptionResNetV2 mrežna arhitektura s točnošću od 96.67% koju slijedi InceptionV3 mreža. Najlošije performanse, kao i u ukupnoj usporedbi modela ostvaruje DenseNet201 mreža s točnošću od 85.89%. Ostvarenim visokim točnostima nakon 30 epoha doprinosi upotreba prijenosnog strojnog učenja kojim su mreže prethodno trenirane na velikom skupu podataka nepovezanim s korištenim skupom podataka bolesnih jedinki rajčice.

#### **4.4.2. Analiza ostvarenih rezultata mreža s IncMB modulom**

Najbolje performanse uspoređujući ukupne performanse modela korištenih u eksperimentu, ali i uspoređujući ostvarene performanse mreža s IncMB modulom, ostvaruje InceptionV3 mreža s točnošću od 97.78%. Slijedi ju InceptionResNetV2 mreža, dok DenseNet201 i u ovom slučaju ostvaruje najlošije performanse. Zapažene performanse MobileNetV2 mreže otvaraju mogućnost implementacije modela u mobilne uređaje kako bi se olakšala uporaba iste poljoprivrednim i drugim stručnjacima te osigurala rana detekcija bolesti usjeva.

#### **4.4.3. Analiza ostvarenih rezultata mreža s metodom potpornih vektora**

InceptionResNetV2 mreža u kombinaciji s metodom potpornih vektora ostvaruje najvišu točnost od 93.11%, a slijedi ju DenseNet201 s točnošću od 92.44%. Iako DenseNet201 mreža ostvaruje lošije performanse u usporedbi s ostalim mrežama analiziranih u eksperimentu, u kombinaciji s metodom potpornih vektora ostvaruje zapažene performanse. Navedeno je ostvareno vjerojatno zbog specifične arhitekture u kojoj su karte značajki svakog sloja spojene s onim prethodnim te time mreža sadrži replicirane podatke više puta. Iako metoda potpornih vektora pripada metodama klasičnog učenja, dokazano je kako u kombinaciji s metodama dubokog učenja doprinosi boljoj i točnijoj klasifikaciji podataka te se kao takva nastavlja upotrebljavati.

### **4.5. Mogućnosti poboljšanja**

Dostupnost većeg skupa podataka za treniranje modela dubokog strojnog učenja uz prikladnu programsku podršku svakako će poboljšati točnost modela. Osim bolesnih jedinki rajčice, skup se može proširiti ostalim primjerima bolesnih listova rajčice kao i zdravih, a moguće je kreirati skup podataka koji sadrži primjere bolesnih listova ostalih poljoprivrednih kultura kako bi se proširio opseg uporabe modela. Obogaćivanje korištenog skupa podataka novim primjerima, također bi

dao uvid u trenutnu širinu područja primjene i pomoći stručnjacima. Za ostvarivanje boljih performansi, nužno je nastaviti razvijati IncMB modul, koristiti dodatne konvolucijske slojeve te proširiti primjenu suvremenih aktivacijskih funkcija koje dokazano pridonose točnosti modela.

## 5. ZAKLJUČAK

U ovome radu stvoren je jedinstveni blok slojeva koji čine modul IncMB ugradiv u kreirane konvolucijske neuronske mreže s idejom korištenja pri klasifikaciji skupova podataka iz svih područja primjene, a koje ostvaruju bolje performanse i veće točnosti od mreža koje ne sadržavaju ovaj modul. Kako navedeni pristup zahtijeva ogromne ljudske i računalne resurse, u ovome radu prikazane su performanse konvolucijske neuronske mreže s IncMB modulom u usporedbi s treniranim mrežama bez modula ili s mrežama dubokog strojnog učenja i spoja s metodom potpornih vektora koja pripada metodama klasičnog strojnog učenja, na primjeru prepoznavanja bolesti lista rajčice. U svim slučajevima, korišteno je prijenosno učenje koje označava da je mreža prethodno trenirana na velikom skupu podataka koji najčešće nije povezan sa skupom podataka čiji je sadržaj potrebno klasificirati.

Eksperimentalni rezultati pokazuju da je najbolje performanse ostvarila InceptionV3 mreža koja sadrži IncMB modul. U usporedbi pojedine mreže, njezine inačice s IncMb modulom i inačice s metodom potpornih vektora, u 4/5 slučajeva, najbolju točnost ostvarila je mreža s IncMB modulom. U preostalom slučaju, točnost od 92.44% je ostvarila inačica mreže s metodom potpornih vektora, dok je inačica DenseNet201 mreže s IncMB modulom iznosila 92.00%. Uspoređujući ostale metode vrednovanja rezultata za taj slučaj, primijećeno je kako mreža s IncMB modulom i dalje ostvaruje zapažene rezultate, a moguće ju je povećati proširivanjem korištenog skupa podataka i/ili korištenjem drugačijeg optimizatora. Iako je način na koji metode strojnog učenja izdvajaju značajke prilično nepoznat, a tek manji broj studija je proučavao navedeni problem, duboko učenje pruža najtočnije algoritme za klasificiranje. IncMB modul pruža dodatni način poboljšanja rada trenutnih mreža, a preostaje istražiti kako utječe na klasifikaciju pri korištenju skupova podataka iz drugih područja znanosti kako bi bio upotrebljiv u svima. Poboljšanje IncMB bloka tek predstoji.



## LITERATURA

- [1] FAOSTAT 2019., Food and Agriculture Organization of the United Nations, fao.org, dostupno na <https://www.fao.org/faostat/en/> [2.7.2022.]
- [2] R. S. Ramos, L. Kumar, F. Shabani & M. C. Picanço, Risk of spread of tomato yellow leaf curl virus (TYLCV) in tomato crops under various climate change scenarios, *Agricultural Systems*, br. 173., str. 524.-535., 2019.
- [3] C. Ferracini, V. H. Bueno, M. L. Dindo, B. L. Ingegno, M. G. Luna, N. G. Salas Gervasio, ... & L. Tavella, Natural enemies of *Tuta absoluta* in the Mediterranean basin, Europe and South America, *Biocontrol Science and Technology*, br. 29., sv. 6., str. 578.-609., 2019.
- [4] S. W. Zhang, Y. J. Shang, L. Wang, Plant disease recognition based on plant leaf image, *J. Anim. Plant Sci*, br. 3., sv. 25, str. 42.-45., siječanj 2015.
- [5] S. P. Mohanty, D. P. Hughes, M. Salathé, Using deep learning for image-based plant disease detection, *Frontiers in plant science*, sv. 7., str. 1419., rujan 2016.
- [6] D. C. He, J. S. Zhan, L. H. Xie, Problems, challenges and future of plant disease management: from an ecological point of view, *Journal of Integrative Agriculture*, br. 4., sv. 15, str. 705.-715., travanj 2016.
- [7] P. D. Sharma, *Plant pathology*, Alpha Science International Limited, Oxford, 2006.
- [8] A. Van Maanen, X. M. Xu, Modelling plant disease epidemics, *European Journal of Plant Pathology*, br. 7., sv. 109, str. 669.-682., rujan 2003.
- [9] P. S. Ojiambo, J. Yuen, F. Van den Bosch, L. V. Madden, Epidemiology: past, present, and future impacts on understanding disease dynamics and improving plant disease management—a summary of focus issue articles, *Phytopathology*, br. 10., sv. 107, str. 1092.-1094., rujan 2017.
- [10] K. A. Garrett, M. Nita, E. D. De Wolf, P. D. Esker, L. Gomez-Montano, A. H. Sparks, *Plant pathogens as indicators of climate change*, Climate change, Elsevier, Amsterdam, Oxford, Waltham, 2016.

- [11] P. R. Birch, A. O. Avrova, A. Dellagi, C. Lacomme, S. S. Cruz, G. D. Lyon, Programmed cell death in plants in response to pathogen attack, Annual Plant Reviews online, str. 184.-208., travanj 2018.
- [12] M. B. Riley, M. R. Williamson, O. Maloy, Plant Disease Diagnosis, The Plant Health Instructor, sv. 10., siječanj 2002.
- [13] S. S. Abu-Naser, K. A. Kashkash, M. Fayyad, Developing an expert system for plant disease diagnosis, Journal of Artificial Intelligence, br. 2., sv. 1., str. 78.-85., 2008.
- [14] N. Petrellis, A smart phone image processing application for plant disease diagnosis, 2017 6th International Conference on Modern Circuits and Systems Technologies (MOCAST), str. 1 - 4., Thessaloniki, 2017.
- [15] R. Baraniuk, D. Donoho & M. Gavish, The science of deep learning, Proceedings of the National Academy of Sciences, br. 117., sv. 48., str. 30029.-30032., 2020.
- [16] S. Paul, Application of Biomedical Engineering in Neuroscience, Springer, Singapur, 2019.
- [17] S. Bhattarai, New Plant Disease Dataset, kaggle.com, dostupno na:  
<https://www.kaggle.com/vipooooool/new-plant-diseases-dataset> [20.5.2022.]
- [18] D. Hughes & M. Salathé, An open access repository of images on plant health to enable the development of mobile disease diagnostics, arXiv preprint arXiv:1511.08060, 2015.
- [19] B. N. Manu, Plant Leaf Disease Detection and Classification using Multiclass SVM Classifier, MATLAB Central File Exchange, dostupno na:  
<https://www.mathworks.com/matlabcentral/fileexchange/55098-plant-leaf-disease-detection-andclassification-using-multiclass-svm-classifier> [11.6.2022.]
- [20] S. Zhang, H. Wang, W. Huang, Z. You, Plant diseased leaf segmentation and recognition by fusion of superpixel, K-means and PHOG, Optik, sv. 157., str. 866.-872., ožujak 2018.
- [21] M. R. Tejoindhi, B. R. Nanjesh, JagadeeshGujanuru Math, AshwinGeetD'sa, Plant Disease Analysis Using Histogram Matching Based On Bhattacharya's Distance Calculation, International Conference on Electrical, Electronics and Optimization Techniques (ICEEOT), str. 1546.-1549., Chennai, 2016.

- [22] W. Albattah, M. Nawaz, A. Javed, M. Masood & S. Albahli, A novel deep learning method for detection and classification of plant diseases, *Complex & Intelligent Systems*, br. 8., sv. 1., str. 507.-524., 2022.
- [23] S. H. Lee, H. Goëau, P. Bonnet, & A. Joly, New perspectives on plant disease characterization based on deep learning, *Computers and Electronics in Agriculture*, br. 170., str. 105220., 2020.
- [24] A. Ramcharan, K. Baranowski, P. McCloskey, B. Ahmed, J. Legg, D. P. Hughes, Deep learning for image-based cassava disease detection, *Frontiers in plant science*, sv. 8., str. 1852., listopad 2017.
- [25] J. Chen, J. Chen, D. Zhang, Y. Sun, Y. A. Nanekharan, Using deep transfer learning for imagebased plant disease identification, *Computers and Electronics in Agriculture*, sv. 173., str. 105393., siječanj 2020.
- [26] C. Analytics, Web of science core collection, Citation database, Web of Science, 2017.
- [27] P. Sharma, Y. P. S. Berwal, W. Ghai, Performance analysis of deep learning CNN models for disease detection in plants using image segmentation, *Information Processing in Agriculture*, studeni 2019.
- [28] S. H. Lee, H. Goëau, P. Bonnet, A. Joly, New perspectives on plant disease characterization based on deep learning, *Computers and Electronics in Agriculture*, sv. 170., str. 105220., ožujak 2020.
- [29] Y. Toda, F. Okura, How convolutional neural networks diagnose plant disease, *Plant Phenomics*, sv. 2019., str. 9237136., svibanj 2019.
- [30] J. G. A. Barbedo, Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification, *Computers and electronics in agriculture*, sv. 153., str. 46.-53., listopad 2018.
- [31] P. Sharma, P. Hans, S. C. Gupta, Classification Of Plant Leaf Diseases Using Machine Learning And Image Preprocessing Techniques, 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence), str. 480.-484., Noida, 2020.

- [32] M. Manoj krishna, M. Neelima, M. Harshali, M. Venu Gopala Rao, Image classification using Deep learning, *International Journal of Engineering & Technology*, br. 2.7, sv. 7., str. 614.-617., ožujak 2018.
- [33] T. Hope, Y. S. Resheff, I. Lieder, *Learning tensorflow: A guide to building deep learning systems*, O'Reilly Media, Inc., Sebastopol, 2017.
- [34] What is a Neural Network?, DeepAI, deepai.org, dostupno na <https://deepai.org/machinelearning-glossary-and-terms/neural-network> [24.6.2022.]
- [35] Z. Xinyi, L. Chen, Capsule graph neural network, ICLR 2019 Conference Blind Submission, New Orleans, 2019.
- [36] M. H. Esfe, S. A. Eftekhari, M. Hekmatifar & D. Toghraie, A well-trained artificial neural network for predicting the rheological behavior of MWCNT–Al<sub>2</sub>O<sub>3</sub> (30–70%)/oil SAE40 hybrid nanofluid, *Scientific Reports*, br. 11., sv. 1., str. 1.-11., 2021.
- [37] I. A. Samborska, V. Alexandrov, L. Sieczko, B. Kornatowska, V. Goltsev, M. D. Cetner, H. M. Kalaji, Artificial neural networks and their application in biological and agricultural research, *J. NanoPhotoBioSciences*, sv. 2., str. 14.-30., kolovoz 2014.
- [38] M. Loukadakis, J. Cano, M. O'Boyle, *Accelerating deep neural networks on low power heterogeneous architectures*, 2018.
- [39] B. B. Traore, B. Kamsu-Foguem, F. Tangara, Deep convolution neural network for image recognition, *Ecological Informatics*, sv. 48, str. 257.-268., studeni 2018.
- [40] R. Garcia-Dias, A. Mechelli, W. H. Lopez Pinaya, S. Vieira, *Autoencoders, Machine Learning: Methods and Applications to Brain Disorders*, Academic Press, Cambridge, 2019.
- [41] S. Cai, S. Zhou, C. Xu & Q. Gao, Dense motion estimation of particle images via a convolutional neural network, *Experiments in Fluids*, br. 60., sv. 4., str. 1.-16., 2019.
- [42] P. Ramachandran, B. Zoph, Q. V. Le, Searching for activation functions, arXiv preprint arXiv:1710.05941, listopad 2017.

- [43] 7 Types of Neural Network Activation Functions: How to Choose?, missinglink.ai, dostupno na <https://missinglink.ai/guides/neural-network-concepts/7-types-neural-network-activationfunctionsright> [4.6.2022.]
- [44] A Beginner's Guide to Neural Networks and Deep Learning, pathmind.com, dostupno na <https://pathmind.com/wiki/neural-network#define> [24.6.2022.]
- [45] Extensive Guide to Support Vector Machines, inovex.de, dostupno na <https://www.inovex.de/de/blog/support-vector-machines-guide/> [1.7.2022.]
- [46] Confusion Matrix, DeepAI, deepai.org, dostupno na <https://deepai.org/machine-learningglossary-and-terms/confusion-matrix> [15.6.2022.]
- [47] Precision and Recall, DeepAI, deepai.org, dostupno na <https://deepai.org/machine-learningglossary-and-terms/precision-and-recall> [18.6.2022.]
- [48] F-Score, DeepAI, deepai.org, dostupno na <https://deepai.org/machine-learning-glossary-andterms/f-score> [18.6.2022.]
- [49] What are Evaluation Metrics?, DeepAI, deepai.org, dostupno na <https://deepai.org/machinelearning-glossary-and-terms/evaluation-metrics> [18.6.2022.]
- [50] S. Kaur, S. Pandey, S. Goel, Plants disease identification and classification through leaf images: A survey, Archives of Computational Methods in Engineering, br. 2., sv. 26., str. 507.- 530., travanj 2019.
- [51] A. Parvat, J. Chavan, S. Kadam, S. Dev, V. Pathak, A survey of deep -learning frameworks, 2017 International Conference on Inventive Systems and Control (ICISC), str. 1.-7., Coimbatore, 2017.
- [52] Fundamentals of Accelerated Computing with CUDA C/C++, NVIDIA Deep Learning Institute, dostupno na <https://www.nvidia.com/en-us/training/> [2.7.2022.]
- [53] CUDA Parallel Thread Management, microway.com, dostupno na <https://www.microway.com/hpc-tech-tips/cuda-parallel-thread-management/> [2.7.2022.]
- [54] W. G. Hatcher, W. Yu, A survey of deep learning: platforms, applications and emerging research trends, IEEE Access, sv. 6., str. 24411.-24432., 2018.

- [55] Anaconda, anaconda.com, dostupno na <https://www.anaconda.com/> [18.6.2022.]
- [56] TensorFlow, TensorFlow, tensorflow.org, dostupno na <https://www.tensorflow.org/about> [18.6.2022.]
- [57] Keras, keras.io, dostupno na <https://keras.io/> [22.6.2022.]
- [58] NumPy, Numpy.org, dostupno na <https://numpy.org/> [22.6.2022.]
- [59] Scientific Applications, Python Guide, dostupno na <https://docs.pythonguide.org/scenarios/scientific/> [22.6.2022.]
- [60] OpenCV, opencv.org, dostupno na <https://opencv.org/about/> [22.6.2022.]
- [61] MATLAB Overview, MathWorks, mathworks.com, dostupno na <https://www.mathworks.com/products/matlab.html> [22.6.2022.]

## SAŽETAK

U teorijskom dijelu rada prikazane su bolesti rajčice, prateći simptomi i načini dijagnosticiranja bolesti. Opisani su modeli za raspoznavanje bolesti lista rajčice temeljeni na prijenosnom dubokom strojnom učenju. Problem klasifikacije pomoću konvolucijskih neuronskih mreža unaprjeđen je razvojem IncMB modula. Znanje o načinu na koji algoritmi dubokog strojnog učenja izdvajaju značajke povećalo bi točnost detekcije bolesti, osiguralo bržu dijagnostiku i odabir pravilnog tretmana. U eksperimentalnom dijelu rada se na mrežnim arhitekturama InceptionResNetV2, InceptionV3, MobileNetV2 i DenseNet201 provela klasifikacija bolesti lista rajčice i to implementacijom pojedine mreže, njezine inačice s IncMB modulom i inačice s metodom potpornih vektora. Svi modeli su trenirani i validirani na javno dostupnom skupu podataka u kojem su bolesne jedinke rajčice određene u šest klasa. Najbolje rezultate ostvarila je InceptionV3 mreža koja sarži IncMB modul čija je točnost naveća.

**Ključne riječi:** dijagnosticiranje bolesti rajčice, klasifikacija, konvolucijske neuronske mreže, modul IncMB, prijenosno duboko strojno učenje

## **ABSTRACT**

Recognition of tomato diseases from images by using transfer deep machine learning and developed IncMB module

The theoretical part of the thesis describes tomato diseases, symptoms, and diagnostic methods. Models for recognizing tomato leaf disease based on transfer deep machine learning are implemented. The classification problem using convolutional neural networks has been improved by the development of IncMB module. Knowledge of how deep machine learning algorithms extract features would contribute to the accuracy of disease detection, faster diagnosis, and selection of the proper treatment. In the experimental part of the thesis, tomato leaf disease classification was performed on InceptionResNetV2, InceptionV3, MobileNetV2, and DenseNet201 network architectures by implementing an individual network, its version with IncMB module and version with support vector machine. All models were trained and validated on a public dataset containing diseased tomato individuals divided into six classes. The best results were achieved by the InceptionV3 network, which contains an IncMB module with an accuracy of 97.78%.

**Keywords:** diagnosing tomato diseases, classification, convolutional neural networks, IncMB module, transfer deep machine learning



## ŽIVOTOPIS

Petra Radočaj rođena je 20.6.1998. godine u Našicama. Završila je prirodoslovno-matematičku gimnaziju u Đakovu, nakon čega upisuje Preddiplomski sveučilišni studij Računarstvo na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku 2017. godine. Pritom je ostvarila izravan upis zbog izvrsnog uspjeha. Stipendistica je u više kategorija tijekom studiranja, uključujući kategoriju za darovite učenike, sveučilišne stipendije za izvrsnost te STEM stipendiju. Dobitnica je Rektorove nagrade za rad „Detekcija raka gušterače metodama strojnog učenja pomoću biomarkera urina“ i nagrade Fakulteta za uspjeh u studiranju na diplomskom bloku Programsko inženjerstvo. Demonstratorica je na kolegijima Programiranje 1, Programiranje 2, Objektno-orijentirano programiranje, Signali i sustavi, Dizajn računalnih sustava. Trenutno je zaposlena kao Software Developer u tvrtki „Ericsson Nikola Tesla“ za razvoj i održavanje 5G ugrađenih sustava, a prethodno iskustvo u radu na Messaging sustavu za masovno slanje poruka ostvarila je u tvrtki „NTH Group“ tijekom 2 godine. Sudjelovala je na „OSC Hackathon 2015.“ u kategoriji „Juniora“ te je volontirala u Code Clubu Gimnazije. Položila je certifikate iz područja arhitekture i razvoja ugrađenih sustava, dubokog strojnog učenja i paralelnog računarstva na CUDA platformi.

## **PRILOZI**

Prilog 1. „Raspoznavanje bolesti rajčice uporabom prijenosnog dubokog strojnog učenja na temelju slike razvijenim modulom IncMB“ u .docx formatu

Prilog 2. „Raspoznavanje bolesti rajčice uporabom prijenosnog dubokog strojnog učenja na temelju slike razvijenim modulom IncMB“ u .pdf formatu

Prilog 3. Izvorni kod modela konvolucijske neuronske mreže s razvijenim modulom IncMB