

KLASIFIKACIJA TIPA RADARSKOG OBJEKTA KORISTEĆI LSTM NEURONSKE MREŽE SA KONVOLUCIJAMA NA SKUPOVIMA TAČAKA**RADAR OBJECT TYPE CLASSIFICATION USING KERNEL POINT CONVOLUTION LSTM NEURAL NETWORKS**

Vladimir Lunić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – Primenjene računarske nauke i informatika

Kratak sadržaj – Rad predstavlja model neuronskih mreža za klasifikaciju tipa radarskog objekta na osnovu skupa tačaka koje definišu objekat. Arhitektura koristi konvolucije na neuređenim skupovima tačaka u okviru kratkotrajno-dugotrajne memorijske ćelije da enkodira geometriju i osobine tačaka kroz vreme. Model pokazuje visoke mere performansi na skupu podataka View of Delft sa 94% tačnosti i 0.94 F1 merom na klasama pešaka, automobila i biciklista.

Ključne reči: Veštačka inteligencija, Neuronske mreže, Radarska klasifikacija

Abstract – This paper presents a neural network model for type classification of radar objects based on associated point clouds. The architecture uses kernel point convolutions on unordered point sets within a long-short term memory cell to encode the geometric data and features of the point cloud. The model shows high performance scores on the View of Delft dataset, with 94% accuracy and 0.94 F1 Score for car, pedestrian and bicycle classes.

Keywords: Artificial Intelligence, Neural Networks, Radar type classification

1. UVOD

Radari su sistemi koji koriste elektromagnetne talase u radio spektru kako bi odredili distancu objekta u odnosu na mesto gde se radar nalazi. Takođe određuju smer u kojem se objekat nalazi spram mesta gde je radar postavljen koristeći azimutni ugao i ugao elevacije. Ključna prednost radara u odnosu na druge sisteme koji mogu da odrede gde se objekat nalazi jeste i mogućnost radarskih sistema da odrede i radijalnu brzinu objekta u odnosu na mesto gde je radarski sistem postavljen, koristeći Doplerov efekat.

Radarski sistemi, za razliku od kamera, su velikim delom otporni na vremenske uslove u kojima kamera ne bi mogla da vrši svoj zadatak. Osim mogućnosti da radi u vremenskim uslovima kao što su kiša, magla i sneg, radarskim sistemima nije potreban stalni izvor svetlosti, stoga mogu da rade i u mraku ili tokom noći.

Radarski sistemi imaju i svoje mane. Prostor koji radar vidi je mnogo manje gust u odnosu na LiDAR koji radi na sličan način, osim toga, radarski sistemi zahtevaju da objekat ka kojem je poslat elektromagnetni talas, bude reflektivan kako bi ga radarski sistem uopšte mogao videti. Sem nedostatka refleksivnosti nekih objekata, čest problem jeste i prevelika količina refleksivnosti, koja može da zbuni radarski klasifikator.

Kako bi se radarski senzori koristili u okviru automobilskih sistema, potrebna je velika pouzdanost da senzor tačno klasifikuje objekat, kako bi se izbegla materijalna ili fizička šteta. Kroz istoriju, zbog ograničenja radarskih senzora, ali i ograničenja računarske moći, algoritmi za klasifikaciju su bili prosti statistički modeli koji razaznaju da li se nešto kreće ili ne. Povećanjem računarske moći i prelaskom na mašinsko učenje i duboko učenje, modeli su mogli da klasifikuju različite učesnike u saobraćaju u nameri da se razviju napredni sistemi za pomoć pri vožnji, kao i sistemi za autonomnu vožnju.

Radarski senzori u automobilskoj industriji vide svet kao skup tačaka u prostoru, koje se nazivaju detekcije. Detekcije koje su međusobno blizu i imaju slične osobine se grupišu u grupe detekcija koje se nazivaju klasteri ili liste detekcija. Liste detekcija se dalje koriste kako bi se kreirali objekti u prostoru.

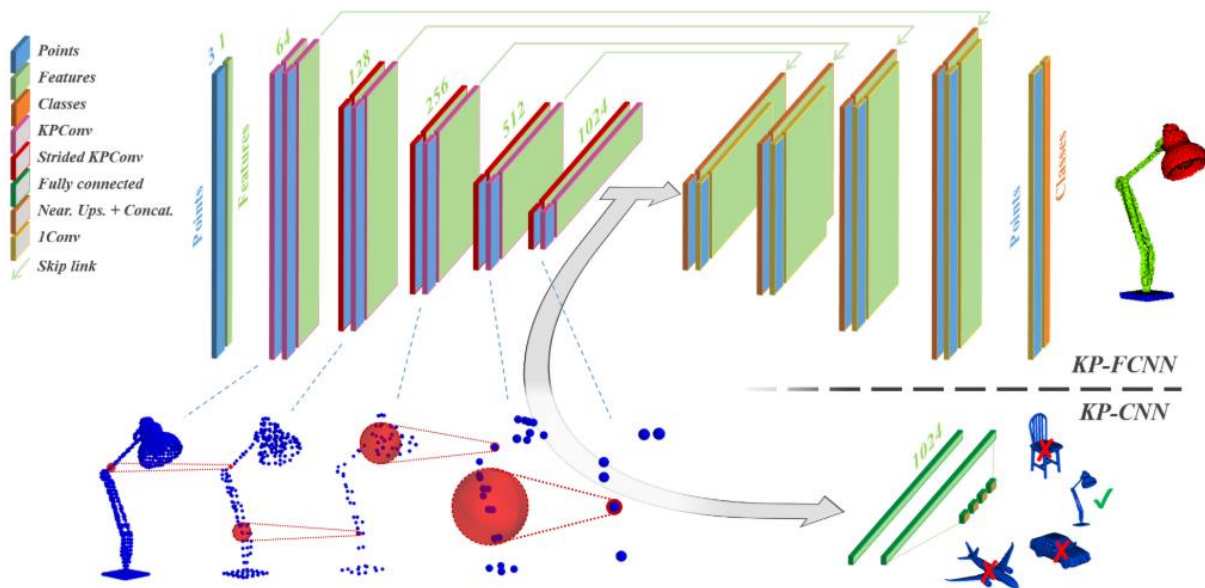
Ovaj rad će predstaviti model dubokog učenja, baziran na rekurentnim mrežama, koji će koristiti liste detekcija kako bi klasifikovao tip objekta, odnosno klasifikovao vrstu učesnika u saobraćaju.

2. PREGLED STANJA U OBLASTI

Prvobitni pokušaji klasifikacije radarskih objekata u domenu dubokog učenja bili su bazirani na dubokim neuronskim mrežama i klasifikaciji na osnovu jednog primera (eng. single shot classification). Koristili su informacije iz objektnog nivoa apstrakcije kao što su brzina objekta, dimenzije objekta i radarski presek objekta. Ovakvi modeli nisu pokazali visoke mere performansi usled činjenice da se izgubio veliki deo informacija prelaskom na viši nivo apstrakcije. Pokušaji da se produbi mreža nisu doprineli velikom prosatu metrika, kao ni promene u arhitekturi sa običnih dubokih mreža na konvolucione mreže.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Milan Rapačić, red. prof.



SLIKA 1: ARHITEKTURA KERNEL POINT KONVOLUTIVNE MREŽE

Nakon pokušaja sa klasičnim dubokim arhitekturama, modeli su počeli da koriste rekurentne mreže kako bi primenili latentne informacije iz vremenskog domena. Rekurentnost je napravila prvi veći skok u performansama modela, pošto je sada bilo lakše razaznati pešake, bicikliste i vozila usled razlika u mikrodopleru. Ali i dalje postoji problem što su modeli radili na nivou apstrakcije objekata. U ranijim radarima ovakvi modeli su predstavljali vrhunac zbog činjenice, da, na kraju razvoja modela, model treba da vrši inferenciju na samom radaru u realnom vremenu, što je veliko usko grlo dubokog učenja na radarskim sistemima usled skupih procesorskih operacija.

Prvi rad koji se smatra savremenim pokušajem klasifikacije i segmentacije neuređenog skupa tačaka je PointNet [1]. PointNet rezonuje nad trodimenzionim geometrijskim podacima kao što su skupovi tačaka, odnosno oblaci tačaka. Arhitektura PointNet neuronske mreže se sastoji iz tri glavne komponente koje predstavljaju rešenje za ograničenja sa radom na neorganizovanim skupovima tačaka. Prvom komponentom realizuje se embedding ulaznog skupa tačaka u globalni vektor. Druga komponenta se odnosi na dalje korišćenje embedovanog skupa tačaka za klasifikaciju ili segmentaciju. Poslednja komponenta rešava problem transformacija ulaznog skupa, tako što pomoću manjih neuronskih mreža, transformiše ulazne skupove tačaka u referentni sistem, koji se uči, uporedo sa klasifikatorom.

PointNet++ [2] arhitektura je predstavljena kao unapređenje koje specifično cilja da reši problem zavisnosti tačke od lokalne strukture susedstva. Arhitekturu predstavljenu u radu [2] čini komponenta za hijerarhijsko učenje osobina skupa tačaka kojom se ostvaruje i prikaz lokalnih susedstava skupova tačaka, umesto samo globalni vektor.

Rad [3] uzima inspiraciju iz klasičnih konvolutivnih mreža i predstavlja *kernel point* konvolucije (nadalje KPC) i konvolutivne modele za segmentaciju i

klasifikaciju neuređenih skupova tačaka. Primer segmentacionog i klasifikacionog KPC modela predstavljen je na slici 1. Rad uvodi pojam kernela tačaka, kao modifikaciju kernel matrica koje koriste konvolutivne mreže. Centriranjem kernela na pojedinačne tačke, operatorom konvolucije, računaju se izlazne osobine te tačke, kreirajući 1 – 1 mapiranje između ulaznog i izlaznog skupa tačaka.

3. METODOLOGIJA

Za rad je korišćen programski jezik *Python* verzije 3.11.2. Za implementaciju modela korišćene su isključivo funkcije prisutne u *Tensorflow* i *Keras* bibliotekama uz hardverska ubrzanja korišćenjem anotacija `@tensorflow.function()`. Minimalna verzija biblioteke *Tensorflow*, kako bi model radio kako je namenjeno, je verzija 2.16. Za *Keras* je potrebna minimalno verzija 3.0, dok je u radu korišćena verzija 3.3

3.1. Postavljanje kernel tačaka

Kreiranje kernela se svodi na proces optimizacije pozicije tačaka u okviru lopte koja predstavlja kernel. Ako pretpostavimo da imamo K kernel tačaka $\tilde{x}_k \in R^d$ u procesu optimizacije želeli bismo da postavimo tačke tako da im je međusobna distanca maksimalna i da su i dalje u okviru lopte poluprečnika r , za koji se obično uzima vrednost $r = 1$. Stoga svakoj tački dodeljujemo odbojnu energiju spram tačke \tilde{x}_k , $E_k^{rep}(x)$, i privlačnu energiju, $E_k^{att}(x)$, ka centru lopte, koja je centrirana na $\vec{0}$. Optimizacioni proces teži da minimizuje ukupnu energiju sistema koji se sastoji od K tačaka. Dodatno, u optimizacionom procesu se mogu postaviti fiksne tačke koje ne učestvuju u optimizacionom procesu ali pridodaju ukupnoj energiji. Optimizacioni proces se izvršava iterativno pomoću gradijentnog spusta.

$$E_k^{rep}(x) = \frac{1}{\|x - x_k\|} \quad (1)$$

$$E^{\text{att}}(x) = ||x|| \quad (2)$$

$$\text{minimize } E^{\text{tot}} = \sum_{k < K} \left(E^{\text{att}}(\tilde{x}_k) + \sum_{l \neq k} E_k^{\text{rep}}(\tilde{x}_l) \right) \quad (3)$$

3.2. Operatori kernel point konvolucije

Nakon definicije kernela, operator konvolucije može da se izvrši. KPC nad skupom osobina \mathcal{F} kernelom g u tački x je definisana sledećim izrazom (4).

$$(\mathcal{F} * g)(x) = \sum_{x_i \in \mathcal{N}_x} g(x_i - x) f_i \quad (4)$$

Tačke x_i predstavljaju tačke ulaznog skupa tačaka $\mathcal{P} \in R^{N \times d}$, gde N predstavlja broj tačaka skupa i d predstavlja dimenzionalnost tačaka. Tačke f_i predstavljaju vektore osobina tačaka skupa \mathcal{P} i pripadaju svom odvojenom skupu $\mathcal{F} \in R^{N \times D}$, gde D predstavlja dimenzionalnost prostora osobina. Cilj KPC je onda da transformiše skup osobina \mathcal{F} u novi skup osobina, konvolucijama nad tačkama skupa \mathcal{P} .

Izraz (4) dodatno definiše susedstvo tačke x , \mathcal{N}_x , skup tačaka koji će učestvovati u konvoluciji tačke x . Susedstvo \mathcal{N}_x je definisan kao $\{x_i \in \mathcal{P} \mid ||x_i - x|| \leq r\}$, odnosno svaka tačka skupa \mathcal{P} čija distanca od tačke x spada u loptu poluprečnika r .

Kernel funkcija g predstavlja ključnu funkciju pri KPC. Funkcija g prima kao ulaz tačke susedstva tačke x , centrirane oko same tačke x , nadalje definisane sa $y_i = x_i - x$. Definisano susedstvo onda predstavlja loptu poluprečnika r , $\mathcal{B}_r^d = \{y \in R^d \mid ||y|| \leq r\}$, te je domen kernel funkcije lopta \mathcal{B}_r^d . Ideja kernel funkcije je da primeni različite težine tačkama \mathcal{B}_r^d , nalik konvolucijama uređenih skupova tačaka. Stoga potreban je utvrditi način na koji će se izračunati koje težine treba primeniti na koje tačke lopte.

Definišimo sada tačke kernela $\{\tilde{x}_k \mid k \leq K\} \subset \mathcal{B}_r^d$ i težine asocirane za svaku od tačaka kernela sa $\{W_k \mid k \leq K\} \subset R^{D_{in} \times D_{out}}$, gde D_{in} predstavlja dimenzionalnost prostora osobina ulaznog skupa osobina, a D_{out} dimenzionalnost prostora osobina izlaznog skupa osobina. Sa tako definisanim tačkama i težinama definišemo kernel funkciju g za svaku tačku $y_i \in \mathcal{B}_r^d$ izrazom (5).

$$g(y_i) = \sum_{k \leq K} h(y_i, \tilde{x}_k) W_k \quad (5)$$

Funkcija $h(y_i, \tilde{x}_k)$ definiše korelaciju između tačke susedstva y_i i tačke kernela \tilde{x}_k i njena vrednost ukazuje na vednost korelacije ove dve tačke. Mera korelacije se može izračunati na veliki broj načina, imajući na umu da se radi o tačkama u realnom svetu, dobra mera korelacije bi bila rastojanje između te dve tačke. Dodatno mera korelacije bi trebalo da raste sa smanjenjem rastojanja između dve tačke i da konvergira ka 0 što je rastojanje veće. Stoga funkciju korelacije $h(y_i, \tilde{x}_k)$ definišemo izrazom (6).

$$h(y_i, \tilde{x}_k) = \max\left(0, 1 - \frac{||y_i - \tilde{x}_k||}{\sigma}\right) \quad (6)$$

Parametar σ izraza (6) definiše snagu kernela. Parametar se može protumačiti kao zona unutar koje je korelacija sa kernelom velika. Odabir parametra snage kernela zavisi od definicije drugih parametara KPC sloja.

3.1. Kernel point LSTM ćelija

LSTM ćelija nastoji da premesti KPC u vremenski domen. To vrši zamenjujući operaciju linearne kombinacije u kapijama sa operacijom KPC. Izrazi (7-13) definišu nove preračune kapija KPC LSTM ćelije.

$$\tilde{X}_t = \text{KPC}(X_t, K^X) \quad (7)$$

$$F_t = \sigma\left(\text{KPC}([\tilde{X}_t, H_{t-1}], K^F)\right) \quad (8)$$

$$I_t = \sigma\left(\text{KPC}([\tilde{X}_t, H_{t-1}], K^I)\right) \quad (9)$$

$$O_t = \sigma\left(\text{KPC}([\tilde{X}_t, H_{t-1}], K^O)\right) \quad (10)$$

$$\hat{C}_t = \sigma\left(\text{KPC}([X_t, H_{t-1}], K^C)\right) \quad (11)$$

$$C_t = F_t \odot C_{t-1} + I_t \odot \hat{C}_t \quad (12)$$

$$H_t = O_t \odot \tanh(C_t) \quad (13)$$

$\text{KPC}(X, K)$ predstavlja KPC nad X kernelima K , gde X predstavlja uređeni par skupa tačaka i tenzora osobina. X , predstavljen uređenim parom $(\mathcal{P}_X, \mathcal{F}_X)$, je jedan element niza uređenih parova skupova tačaka i tenzora osobina koji predstavljaju vremensku seriju. \mathcal{P}_X je definisan matricom $N \times d$, gde N predstavlja broj tačaka, a d dimenzionalnost tačaka, dok je \mathcal{F}_X definisan tenzorom $N \times D^{in} \times c$, gde D^{in} predstavlja dimenzionalnost ulaznog prostora osobina, a c broj kanala tenzora osobina. Sa druge strane, H predstavlja skriveno stanje LSTM ćelije. Skriveno stanje je takođe definisano uređenim parom tačaka i osobina, $(\mathcal{P}_H, \mathcal{F}_H)$. \mathcal{P}_H je matrica veličine $N' \times d$, dok \mathcal{F}_H predstavlja tenzor veličine $N' \times D^{out} \times c'$, gde N' predstavlja broj tačaka skrivenog stanja, D^{out} predstavlja dimenzionalnost skrivenog prostora osobina, koji je ujedno i dimenzionalnost izlaznog prostora osobina, a c' broj kernela korišćen u kapijama KPC. Ćelijsko stanje C , nalik H , je definisano uređenim parom $(\mathcal{P}_C, \mathcal{F}_C)$, istih dimenzionalnosti kao $(\mathcal{P}_H, \mathcal{F}_H)$. $[x, y]$ definiše operaciju vertikalne konkatenacije, dok operator \odot_m predstavlja modifikovano Adamarovo množenje.

Adamarovo množenje zahteva da oba operanda budu matrice iste veličine, ali u ovom slučaju, izlazi kapija predstavljaju uređene parove matrica i tenzora veličina $\hat{N} \times d$ i $\hat{N} \times D^{out} \times c'$, dok C_{t-1} i \hat{C}_t predstavljaju uređene parove matrica i tenzora veličina $N' \times d$ i $N' \times D^{out} \times c'$. Usled toga, Adamarovo množenje se ne može izvršiti zbog neslaganja dimenzija \hat{N} i N' . Stoga, operator Adamarovog množenja je izmenjen da može da radi sa operandima različitih veličina. Operator za

operande prima dva uređena para tačaka i njihovih vektora osobina. Broj tačaka ne mora biti isti, ali dimenzionalnost tačaka i osobina tačaka mora biti ista. Rezultat operacije vraća uređeni par tačaka i izlaznih osobina. Prvi element uređenog para prima vrednost matrice tačaka levog operanda, dok drugi element uređenog para nakon računa ostaje iste dimenzionalnosti kao tenzor osobina levog operanda. Izrazi (14-16) definišu modifikovani operator Adamarovog množenja.

$$h(\mathcal{P}_x, \mathcal{P}_y) = \max\left(0, 1 - \left\| \mathcal{P}_x - \mathcal{P}_y \right\| \right) \quad (14)$$

$$\mathcal{F} = \mathcal{F}_x \odot (h(\mathcal{P}_x, \mathcal{P}_y) \circ \mathcal{F}_y) \quad (15)$$

$$(\mathcal{P}_x, \mathcal{F}_x) \odot_m (\mathcal{P}_y, \mathcal{F}_y) = (\mathcal{P}_x, \mathcal{F}) \quad (16)$$

4. SKUP PODATAKA

Skup podataka korišćen za sve eksperimente potiče iz rada [4], univerziteta u Delftu i naziva se *View of Delft* (nadalje VoD). Skup podataka je kreiran vožnjom kroz kampus, predgrađa i centar Delfta. Radar korišćen za kreiranje skupa podataka je *ZF FRGen21 3+1* dimenzioni radar, frekvencije 13Hz, postavljen iza prednjeg branika automobila. Dodatno za prikupljanje podataka korišćena je stereo kamera, frekvencije 30Hz postavljena na vetrobran i *Velodyne HDL-64 S3 LiDAR*, frekvencije 10Hz. Pozicije senzora predstavljene su na slici 2.

Frejmovi svakog od senzora su sinhronizovani, tako da svaki frejm sadži tačke koje je LiDAR izmerio, tačke koje je radar izmerio, sliku stereo kamere i kalibraciju senzora za trenutni frejm. LiDAR je uzet kao glavni frejm spram kojeg su traženi najbliži frejm radara i kamere, maksimalne tolerancije razlike vremenskih odrednica frejmova od 0.05 sekundi. Stoga frejmovi predstavljaju sekvence snimljene na svakih 10Hz.

Skup podataka je organizovan što je bliže moguće KITTI formatu, koji definiše kako se skladišti jedan frejm. Podaci su podeljeni na trening i test frejmove. VoD nudi labele samo za trening frejmove, te test frejmovi nisu mogli biti korišćeni za svrhe rada, bez da se cela implementacija rada pošalje univerzitetu u Delftu.



SLIKA 2: POSTAVKA SENZORA KORIŠĆENIH U "VJU OF DELFT" SKUPU PODATAKA

Ukupan broj trening i test frejmova je 8693, od čega, 26% predstavljaju test frejmove, ostavljajući samo 6433 frejmova koji su labelirani. Trening je dalje podeljen u direktorijume za LiDAR frejmove, i tri vrste radar frejmova koji predstavljaju 1, 3 ili 5 frejmova spojenih u jedan frejm. Za svrhe rada, korišćeni su radar podaci sačinjeni od jednog frejma detekcija, jer takva vrsta podataka najbliže predstavlja realne podatke.

5. REZULTATI I DISKUSIJA

Eksperimenti na arhitekturi mreže će biti podeljeni u tri kategorije. Prva kategorija je uticaj veličine i broja kernela na rezultate. Druga vrsta eksperimenata će se baviti uticajem koraka na rezultate. Treća kategorija eksperimentiše sa količinom ćelija u rekurentnom bloku. Sve arhitekture, osim ako nije dodatno navedeno, će koristiti isti skup podataka, sačinjen od najvećeg podskupa podataka koji sadrži iste količine labela automobila, biciklista i pešaka. Biće trenirane u 100 epoha, koristeći *Adaptive Moment Estimation* (ADAM) optimizacionu metodu, sa invarijantnom stopom učenja od 1.0×10^{-3} , funkcijom greške unakrsne entorpije (CEL), metrikama tačnosti i makro F1 mere i bez mehanizma za ranije zaustavljanje. Evaluacija će biti odrađena na celokupnom test skupu podataka, odvojenom u fazi preprocesiranja, i biće učitana kontrolna tačka sa najmanjom validacionom greškom.

Prvi model za evaluaciju se sastojao od minimalne arhitekture koja prati preporuke za KPC predstavljene u radu [3]. Arhitektura se sastoji od jedne KPC LSTM ćelije. Ćelija na ulazu prima RCS detekcije i absolutnu vrednost kompenzovane Dopler brzine, te je ulazni prostor osobina dvodimenzijoni. Dimenzionalnost skrivenog prostora osobina je dva puta veći od ulaznog po preporuci [3] koja nalaže dupliranje dimenzionalnosti između slojeva. Broj kernela je 1 kao i kod običnih KPC mreža predstavljenih u [3], dok je broj tačaka u kernelu 15. Korak je u ovoj vrsti eksperimenta fiksiran i ima vrednost 1.0. Drugi model prati istu arhitekturu kao i prvi model, sa razlikom da sadrži 3 kernela. Dok treći model sadrži 1 kernel i 7 tačaka. Rezultati su predstavljeni u tabeli 1.

TABELA 1: REZULTATI EKSPERIMENATA KERNEL TAČAKA

Oblik Kernela	Tačnost	Makro F1 Mera
[1, 15]	90%	90%
[3, 15]	91.5%	91.5%
[1, 7]	91.6%	92%

Povećanje broja kernela je donelo usporenje od 10%, dok je smanjenje ubrzalo treniranje za 30%. Povećanje broja kernela je učinilo proces treniranja manje stabilnim, sa mnogo više skokova u grešci za razliku od modela sa manjim brojem jezgara, usled činjenice da mreža ima veliki broj parametara. Fenomen eksplozivnog gradijenta je stoga bio primećen u određenim epohama, što se može rešiti praćenjem performansi modela za

potrebe ranijeg zaustavljanja i biranjem najboljeg sačuvanog modela. Model sa jednim jezgrom i sedam tačaka je imao najstabilniji proces učenja, sa minimalnim oscilacijama u grešci, koje najverovatnije potiču usled upotrebe algoritma stohastičkog gradijenta.

Sledeća grupa eksperimenata se fokusira na parametar koraka. Mreže korišćene u ovim eksperimentima imaju fiksiranu arhitekturu na jednu ćeliju sa 4 dimenzije u skrivenom prostoru osobina, sa jednim kernelom od 14 tačaka i uticajem kernel tačaka od 4.0. Mreže su testirane sa korakom 1.0, 0.25, 0.5 i 0.75. Rezultati eksperimenata su predstavljeni u tabeli 2.

TABELA 2: REZULTATI EKSPERIMENTA KORAKA

Korak	Tačnost	Makro F1 Mera
1.0	90%	90%
0.75	94.3%	94.3%
0.5	90.7%	90.8%
0.25	89.6%	89.7%

Smanjenjem koraka se mreža očekivano kraće trenirala, pitanje koje ovaj eksperiment hoće da odgovori jeste da li uopšte korak utiče na performanse. Dodatno, ako se korak pokaže da pozitivno utiče na proces treniranja, koja vrednost je najbolja, odnosno za koliko smanjivati prostor svakom sledećom KPC. Pozivajući se na puling funkcije običnih konvolutivnih mreža, očekivana vrednost bi bila 0.5, ali usled skupa koji nije gust kao što su slike, vrednost se očekuje da bude veća. Korak između 1.0 i 0.75 se pokazao kao najbolji u očuvavanju relevantnih tačaka i prostora u kojem se kernel kreće. Uz smanjenje prostora koji kerneli pokušavaju da embeduju, proces učenja ispod koraka 0.5 postaje haotičniji i vrednost greške češće skače. Ako korak skрати broj centara konvolucije previše, susedstva će često biti retka i uglavnom prazna, što naravno, ne pomaže procesu učenja.

Poslednja grupa eksperimenata trenira mrežu sa dve rekurentne ćelije. Prva arhitektura enkodira 4 dimenzije skrivenog prostora stanja, ima 1 kernel od 14 tačaka, korak od 1.0 i uticaj od 4.0. Dok druga ćelija, kao što [3] nalaže, duplira dimenzionalnost skrivenog prostora osobina, u ovom slučaju sa 4 na 8. Kernel ostaje isti, ali se uticaj kernel tačaka menja sa porastom enkodiranih dimenzija, sa 4.0 na 8.0, opet po preporuci [3]. Poređenje ovakve dve mreže je predstavljeno tabelom 3.

Mreža koja je sadržala veći broj ćelija nije bila u stanju da generalizuje pravilno na problemu. Proces treniranja je trajao duplo duže i bio je podložan pojavi fenomena eksplozivnih gradijenata, gde u poslednjem slučaju kada je gradijent eksplozivirao, mreža nije mogla da se vrati na logaritamski pad greške kako su epohe tekle i zatekla se u lokalnom minimumu iz kojeg nije mogla da izađe.

TABELA 3: REZULTATI EKSPERIMENTA BROJA ĆELIJA

Broj Ćelija	Tačnost	Makro F1 Mera
1	90%	90%
2	74%	74%

Pretpostavka je da je nova dimenzionalnost uvedena drugim slojem negativno uticala na mnogostrukost koji deli klasifikacioni prostor, te mreža nije mogla da nađe adekvatne parametre koji bi izmenili mnogostrukost na bolje usled većeg izbora dimenzija koji je donela nova ćelija.

6. LITERATURA

- [1] Qi, C.R., Su, H., Mo, K. and Guibas, L.J., 2017. Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 652-660).
- [2] Qi, C.R., Yi, L., Su, H. and Guibas, L.J., 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems, 30.
- [3] Thomas, H., Qi, C.R., Deschaud, J.E., Marcotegui, B., Goulette, F. and Guibas, L.J., 2019. Kpconv: Flexible and deformable convolution for point clouds. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 6411-6420).
- [4] Palffy, A., Pool, E., Baratam, S., Kooij, J.F. and Gavrila, D.M., 2022. Multi-class road user detection with 3+ 1D radar in the View-of-Delft dataset. IEEE Robotics and Automation Letters, 7(2), pp.4961-4968.
- [5] Nobis, F., Fent, F., Betz, J. and Lienkamp, M., 2021. Kernel point convolution LSTM networks for radar point cloud segmentation. Applied Sciences, 11(6), p.2599.
- [6] Fent, F., 2020. Machine Learning-Based Radar Point Cloud Segmentation.
- [7] Fan, H. and Yang, Y., 2019. PointRNN: Point recurrent neural network for moving point cloud processing. arXiv preprint arXiv:1910.08287.
- [8] Schumann, O., Hahn, M., Dickmann, J. and Wöhler, C., 2018, July. Semantic segmentation on radar point clouds. In 2018 21st International Conference on Information Fusion (FUSION) (pp. 2179-2186). IEEE.

Kratka biografija:

Vladimir Lunić rođen je u Novom Sadu 2000. god. Master rad na Fakultetu tehničkih nauka iz oblasti Primenjene računarske nauke i informatika – Inteligentni sistemi odbranio je 2024.god.
kontakt: vladimir.lunic2000@gmail.com