

Primjena metoda strojnog učenja za prepoznavanje i očitavanje trenutnog stanja strujnog brojila

Application of machine learning methods to recognize and read the current state of an electricity meter

¹Maja Risek, ²Bruno Trstenjak
¹student Međimurskoga veleučilišta u Čakovcu
²Međimursko veleučilište u Čakovcu, Bana
Jelačića 22a, 40000 Čakovec, Hrvatska
e-mail: ¹maja.risek@student.mev.hr,
²btrstenjak@mev.hr

Sažetak

Ručno očitavanje brojila, koje je i dalje često prisutno u praksi, podložno je pogreškama i usporava obradu podataka. Ovaj rad predstavlja razvoj konvolucijske neuronske mreže (CNN) za automatizirano prepoznavanje znamenki na digitalnim brojlama čime se rješava problem pogrešaka i sporosti ručnoga očitavanja. Model razvijen u Pythonu treniran je na vlastitom skupu od 5 500 originalnih slika znamenki sa stvarnoga brojila snimljenih u različitim uvjetima. Kroz predprocesiranje i augmentaciju, skup podataka je proširen na 55 000 slika u sivoj skali. Evaluacijom je potvrđena visoka preciznost i robusnost razvijenoga CNN modela, čak i na neviđenim i izazovnim slikama. Model učinkovito prepoznaje znamenke, a njegova je prednost sposobnost detekcije vlastite nesigurnosti pri predviđanju. Aplikacija interpretira ovu nesigurnost kao potencijalnu pogrešku, vizualno je signalizira korisniku i preporučuje dodatnu provjeru, čime se osigurava visoka pouzdanost cijeloga sustava. Razvijeni model ima značajnu primjenu u mobilnim aplikacijama za automatizirano očitavanje brojila. Za daljnje unaprjeđenje predlaže se obogaćivanje skupa podataka novim, izazovnim realnim fotografijama, čime bi se poboljšala generalizacijska sposobnost modela i smanjila potreba za ljudskom intervencijom.

Ključne riječi: CNN, strojno učenje, računalni vid, digitalna brojila, Python.

Abstract

Manual meter reading, which is still common in practice, is prone to errors and slows down data processing. This paper presents the development of a Convolutional Neural Network (CNN) for automated digit recognition on digital meters, thereby addressing the problem of errors and slowness

associated with manual reading. The model developed in Python was trained on its own dataset of 5 500 original digit images from a real meter, captured under various conditions. Through preprocessing and augmentation, the dataset was expanded to 55 000 grayscale images. Evaluation confirmed the high precision and robustness of the developed CNN model, even on unseen and challenging images. The model efficiently recognizes digits, and its advantage lies in its ability to detect its own uncertainty during prediction. The application interprets this uncertainty as a potential error, visually signals it to the user, and recommends additional verification, thus ensuring the high reliability of the entire system. The developed model has significant application in mobile applications for automated meter reading. For further improvement, it is proposed to enrich the dataset with new, challenging real-world photographs, which would enhance the model's generalization capability and reduce the need for human intervention.

Keywords: *CNN, machine learning, computer vision, digital meters, Python.*

1. UVOD

Automatizacija je postala neizostavan dio modernoga poslovanja i svakodnevnoga života, s posebnim naglaskom na industrijske sektore gdje su preciznost i učinkovitost od presudne važnosti. S kontinuiranim razvojem umjetne inteligencije i strojnoga učenja, otvaraju se značajne mogućnosti za optimizaciju procesa koji su tradicionalno ovisili o ljudskom radu. Jedan od takvih procesa je očitavanje digitalnih brojeva. Trenutačno rasprostranjeno ručno očitavanje opterećeno je rizikom od ljudskih pogrešaka, sporošću obrade podataka te posljedično povećanim operativnim troškovima i smanjenom učinkovitošću sustava. Ovaj rad prikazuje razvoj i evaluaciju sustava za automatsko prepoznavanje znamenki na digitalnim brojevima primjenom konvolucijskih neuronskih mreža (CNN). Naglasak je stavljen na stvaranje robusnoga modela sposobnoga za rad u realnim uvjetima, uključujući različite uvjete osvjetljenja, kutove snimanja i stupnjeve mutnoće, uz istovremenu implementaciju mehanizma za procjenu pouzdanosti predviđanja. Cilj je demonstrirati da metode dubokoga učenja nude učinkovito i konkretno rješenje za složene zadatke računalnoga vida na stvarnim primjerima, čime se značajno pridonosi automatizaciji i pouzdanosti procesa očitavanja brojeva. Glavni doprinosi ovoga rada leže u razvoju vlastitoga, opsežnoga i augmentiranoga skupa podataka, specifično prilagođenoga varijacijama digitalnih znamenki u realnim uvjetima. Nadalje, implementiran je mehanizam za detekciju nesigurnosti modela koji korisniku pruža kontrolu nad rezultatima predviđanja te omogućuje sustavno prepoznavanje i rješavanje problematičnih uvjeta snimanja kroz ciljano obogaćivanje skupa podataka za daljnje poboljšanje performansi.

2. PRIJAŠNJA ISTRAŽIVANJA

U području automatiziranoga očitavanja digitalnih brojila objavljen je niz istraživanja koja predlažu različite metode za detekciju i prepoznavanje znamenki korištenjem tehnika računalnoga vida i strojnoga učenja. Velik broj pristupa koristi duboko učenje posebno konvolucijske neuronske mreže zbog njihove sposobnosti automatskoga učenja značajki iz slike. Predložene su metode koje omogućuju potpuno automatizirano očitavanje brojila u stvarnim uvjetima. Jedan od radova koristi YOLO arhitekturu za detekciju znamenki i njihovo prepoznavanje u realnom vremenu. Model je treniran na slikama digitalnih brojila snimljenima iz različitih kutova i pod različitim osvjetljenjem. Ovakav pristup omogućuje visoku točnost prepoznavanja te vrlo brzo izvođenje što ga čini pogodnim za implementaciju u stvarnim sustavima [12]. U nekim istraživanjima koriste se klasične metode digitalne obrade slike za segmentaciju sedam segmentnih znamenki uz dodatnu primjenu algoritama strojnoga učenja za klasifikaciju. U takvim pristupima važnost se stavlja na točnu segmentaciju znamenki kao ključan korak prije prepoznavanja [7]. Za zadatke prepoznavanja znamenki često se koriste CNN arhitekture. U više radova prikazano je kako se modeli trenirani na standardnim skupovima poput MNIST-a mogu koristiti za precizno prepoznavanje rukom pisanih znamenki. U takvim slučajevima naglasak je na jednostavnosti mrežne arhitekture i malom broju slojeva uz visoku razinu točnosti [6] [10]. Posebna pažnja posvećena je optimizaciji CNN arhitektura. Istraživanja uključuju usporedbu različitih brojeva slojeva, veličina filtera te primjene različitih optimizacijskih algoritama. Cilj ovakvih pristupa je povećanje točnosti uz što manju kompleksnost modela [2]. Neki radovi uvode napredne metode poput višeslojne ekstrakcije značajki pri čemu se koristi fuzija slojeva CNN mreže kako bi se poboljšala klasifikacijska učinkovitost. U takvim modelima informacije se prenose s jednoga sloja na drugi te se kombiniraju za poboljšanje rezultata [8]. Istraživanja su također usmjerena na otpornost CNN modela u uvjetima smanjene kvalitete slike. Prikazani su modeli koji zadržavaju visoku točnost čak i kada su ulazne slike zamućene, s dodanim šumom ili djelomično zaklonjene. Ovakvi pristupi pokazuju važnost proširivanja skupa podataka tehnikama augmentacije kako bi model bio robusan u stvarnim uvjetima [1] [9]. Postoje i primjeri primjene CNN mreža u klasifikaciji znakova različitih pisama. U tim istraživanjima koristi se konvolucijska mreža za izdvajanje značajki, uz dodatne klasifikatore poput SVM-a, čime se pokazuje fleksibilnost CNN pristupa u različitim područjima primjene [11] [4].

Brojni znanstveni radovi potvrđuju visoku učinkovitost konvolucijskih neuronskih mreža (CNN) u zadacima prepoznavanja znamenki. Istraživanjima je pokazano da struktura modela, optimizacija parametara mreže i prilagodba na slike lošije kvalitete značajno utječu na točnost

i robusnost CNN modela. Navedena saznanja činila su temelj za razvoj vlastitoga rješenja u okviru ovoga rada.

3. ISTRAŽIVAČKA TEHNOLOGIJA

3.1. PODATKOVNI SKUP I PREDPROCESIRANJE

Za razliku od rukom pisanih znamenaka pri čemu su linije znamenaka uglavnom spojene, znamenke na digitalnim sedam-segmetnim zaslonima karakterizira jasno odvojeni segmenti s vidljivim prazninama. S obzirom na specifičan izgled ovih znamenki te potrebu za robusnim prepoznavanjem u realnim uvjetima, kreiran je vlastiti skup podataka. Skup podataka sastoji se od slika pojedinačnih znamenki (0-9) te decimalne točke, izreznanih s fotografija stvarnih digitalnih brojila. Slike su prikupljene pod različitim kutovima snimanja i u različitim uvjetima osvjetljenja kako bi se osigurala raznolikost i simulirali realni scenariji primjene.

Kvaliteta i priprema skupa podataka ključni su za učinkovitost predikcija modela. Originalne slike bile su u PNG i JPG formatu, različitih dimenzija (npr., od 50x50 do 120x100 piksela). Za potrebe obrade, sve su slike inicijalno pretvorene u sive tonove (engl. *grayscale*) kako bi se smanjio broj ulaznih kanala (s tri na jedan) i fokusiralo na konture i oblik znamenki umjesto na boju. Nadalje, sve su slike skalirane na dimenzije 32x32 piksela, a vrijednosti piksela normalizirane su dijeljenjem s 255.0 (skaliranje u raspon 0–1). Decimalna točka je uključena kao zasebna klasa radi poboljšanja sposobnosti modela da je razlikuje od numeričkih znamenki.

Kako bi se spriječio problem prekomjernoga učenja (engl. *overfitting*) i povećala generalizacijska sposobnost modela na neviđenim podacima, primijenjena je tehnika augmentacije podataka (engl. *data augmentation*) [14]. Korištenjem klase `ImageDataGenerator` iz biblioteke Keras, za svaku originalnu sliku generirano je 10 novih, augmentiranih varijanti. Na taj način, početni skup od 5 500 originalnih slika (500 slika po svakoj od 11 klasa) proširen je na 55 000 augmentiranih slika, simulirajući realne varijacije poput blagoga nagiba brojila, uvećanja ili manjega pomaka čime je značajno poboljšana robusnost i performanse modela.

3.2. ARHITEKTURA KONVOLUCIJSKE NEURONSKE MREŽE

Model za klasifikaciju znamenki implementiran je kao sekvencijalna konvolucijska neuronska mreža prilagođena za učenje hijerarhijskih značajki iz ulaznih slika znamenki i njihovu klasifikaciju u 11 definiranih kategorija (znamenke od 0 do 9, te decimalna točka). Sekvencijalni model karakterizira linearna struktura slojeva pri čemu izlaz jednoga sloja služi kao ulaz za sljedeći [15]. Detaljna arhitektura modela prikazana je na slici 1.

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 30, 30, 32)	320
max_pooling2d (MaxPooling2D)	(None, 15, 15, 32)	0
conv2d_1 (Conv2D)	(None, 13, 13, 64)	18,496
max_pooling2d_1 (MaxPooling2D)	(None, 6, 6, 64)	0
flatten (Flatten)	(None, 2304)	0
dense (Dense)	(None, 128)	295,040
dense_1 (Dense)	(None, 11)	1,419

Total params: 315,275 (1.20 MB)

Trainable params: 315,275 (1.20 MB)

Non-trainable params: 0 (0.00 B)

Slika 1. Sažetak strukture CNN modela.

Izvor: autorica.

Model se sastoji od sljedećih slojeva:

- ulazni sloj je implicitno definiran prvim konvolucijskim slojem, prima slike dimenzija 32x32 piksela s jednim kanalom (grayscale).
- prvi konvolucijski sloj (Conv2D) primjenjuje 32 filtra veličine 3x3 piksela. Koristi ReLU aktivacijsku funkciju za uvođenje nelinearnosti. Izlazni oblik je (None, 30, 30, 32). Broj treniranih parametara je 320.
- prvi sloj maksimalnoga sažimanja (MaxPooling2D) smanjuje prostornu dimenzionalnost karata značajki koristeći 2x2 prozor, pri čemu se odabire maksimalna vrijednost. Izlazni oblik je (None, 15, 15, 32), bez treniranih parametara.
- drugi konvolucijski sloj (Conv2D_1) primjenjuje 64 filtra veličine 3x3 piksela s ReLU aktivacijskom funkcijom. Izlazni oblik je (None, 13, 13, 64). Ovaj sloj s 18 496 parametara služi za ekstrakciju složenijih i apstraktnijih značajki.
- drugi sloj maksimalnoga sažimanja (MaxPooling2D_1) ponovno primjenjuje 2x2 prozor, smanjujući dimenzije karata značajki na 6x6 piksela, bez treniranih parametara.
- sloj za izravnavanje (Flatten) transformira 3D izlaz iz prethodnoga sloja (6x6x64) u jednodimenzionalni vektor (2 304 elementa). Nema treniranih parametara i služi kao priprema za potpuno povezane slojeve.

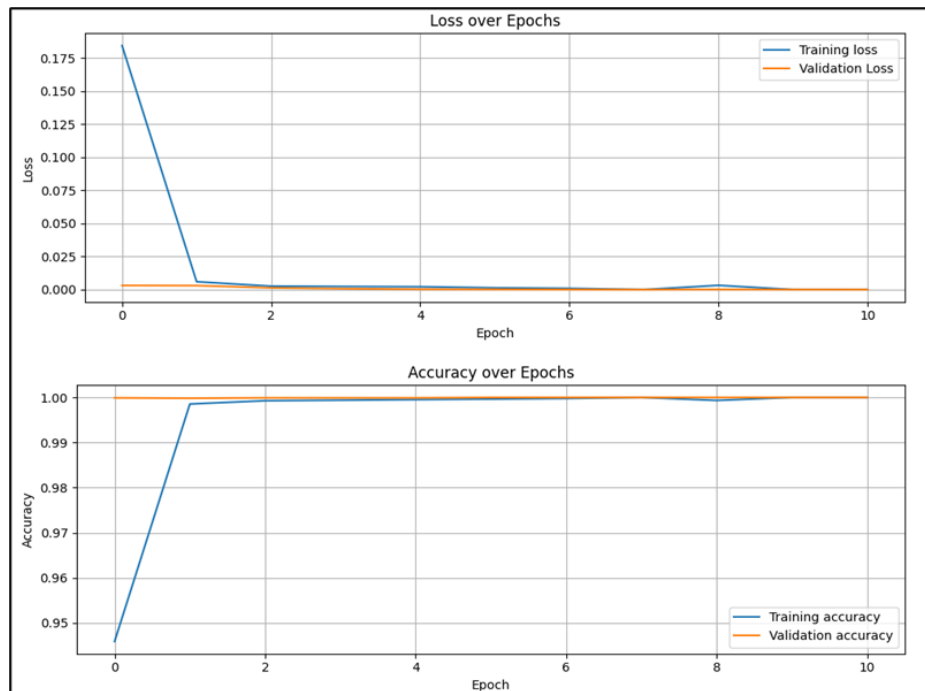
- gusto povezani sloj (Dense) je prvi potpuno povezani sloj s 128 neurona i ReLU aktivacijskom funkcijom. Sadrži 295 040 treniranih parametara, a njegova uloga je integracija i interpretacija naučenih značajki.
- izlazni sloj (Dense_1) posljednji je sloj modela, također gusto povezan, s 11 neurona koji odgovaraju broju klasa. Koristi Softmax aktivacijsku funkciju za pretvaranje numeričkih izlaza u vjerojatnosti, osiguravajući da se zbroj svih vjerojatnosti za pojedine klase zbroji na 1 [13]. Ovaj sloj ima 1 419 treniranih parametara. Neuron s najvećom vjerojatnosti predstavlja konačnu klasifikaciju modela.

4. REZULTATI TESTIRANJA

4.1. PROCES TRENIRANJA I PODJELA PODATKOVNOGA SKUPA

Prije treniranja modela, pripremljeni skup podataka (ukupno 55 000 augmentiranih slika) podijeljen je na skup za treniranje i skup za validaciju. Za treniranje je izdvojeno 80 % podataka, dok je preostalih 20 % korišteno za validaciju. Ova podjela osigurava da se performanse modela evaluiraju na podacima koje nije vidio tijekom faze učenja, čime se objektivno procjenjuje njegova generalizacijska sposobnost. Treniranje CNN modela provedeno je s ciljem minimiziranja pogreške i maksimiziranja točnosti klasifikacije. Korišten je Adam optimizator, poznat po dinamičkoj prilagodbi stope učenja. Kao funkcija gubitka (engl. *loss function*) odabrana je *categorical_crossentropy*, prikladna za probleme višeklasne klasifikacije, koja mjeri odstupanje između predviđenih i stvarnih distribucija klasa [3]. Tijekom treniranja, pratile su se metrike točnosti (engl. *accuracy*) i gubitka (engl. *loss*) na oba skupa podataka. Implementiran je mehanizam ranoga zaustavljanja (engl. *early stopping*) kako bi se spriječilo prekomjerno prilagođavanje modela (engl. *overfitting*) podacima za treniranje. Mehanizam je konfiguriran tako da prati validacijski gubitak i prekida treniranje ako se on ne poboljša kroz tri uzastopne epohe, čuvajući pritom utege modela iz epohe s najboljim validacijskim performansama. Iako je model bio postavljen na 20 epoha, rano zaustavljanje omogućilo je završetak treniranja u prosjeku nakon 10 epoha, značajno optimizirajući vrijeme treniranja s 19 na 9 minuta bez ugrožavanja performansi.

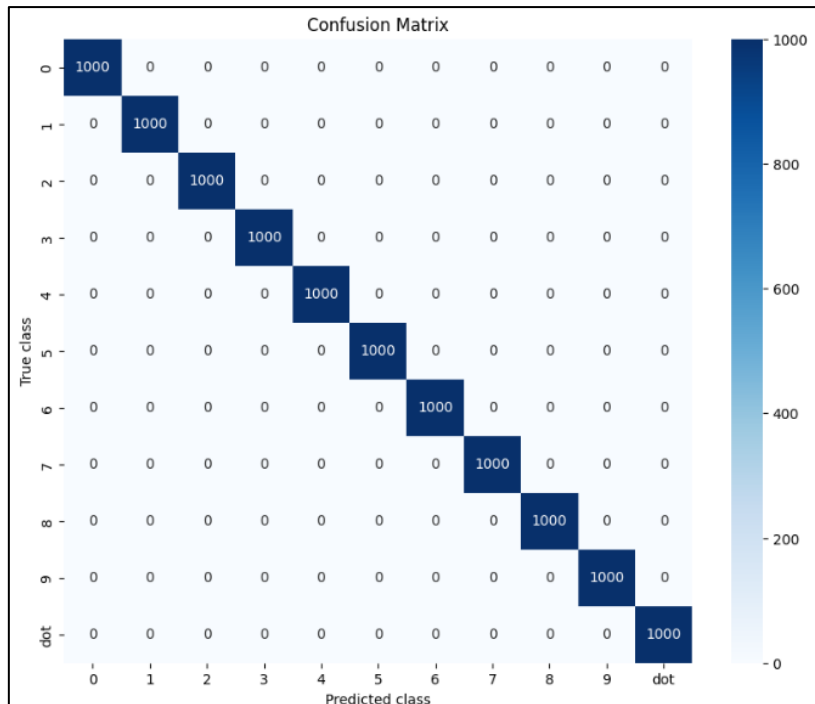
4.2. ANALIZA PERFORMANSI MODELA



Slika 2.. Krivulje gubitka i točnosti tijekom treniranja modela.

Izvor: autorica.

Performanse modela detaljno su evaluirane analizom grafikona gubitka i točnosti te matrice konfuzije. Slika 2 prikazuje krivulje gubitka i točnosti tijekom procesa treniranja. Na gornjem grafikonu i trening gubitak (plava linija) i validacijski gubitak (narančasta linija) znatno opadaju u prvim epohama i stabiliziraju se na gotovo nultim vrijednostima. Njihov paralelni tijek signalizira stabilan proces učenja i odsustvo *overfittinga*. Donji grafikon prikazuje sličan trend za točnost. Trening točnost i validacijska točnost brzo dostižu gotovo 100 % (iznad 0.99) i ostaju iznimno visoke tijekom cijeloga procesa. Činjenica da se validacijska točnost izjednačava s trening točnošću i ostaje visoka potvrđuje iznimnu sposobnost modela da generalizira na nove, neviđene podatke. Odsutnost pada validacijske točnosti pri kraju treniranja također eliminira rizik od *overfittinga* [5]. Glavni razlog visokih performansi modela pripisuje se veličini i raznolikosti korištenoga skupa podataka od 55 000 augmentiranih slika. Matrica konfuzije (slika 3) korištena je za vizualnu procjenu odnosa između stvarnih i predviđenih klasa na validacijskom skupu. Za svaku od 11 klasa (znamenke 0-9 i decimalna točka), svi elementi na dijagonali matrice pokazuju vrijednost 1 000, dok su svi ostali elementi izvan dijagonale jednaki nuli. Ovaj rezultat ukazuje na to da je model ispravno klasificirao sve uzorke u validacijskom skupu za svaku od 11 klasa što potvrđuje iznimnu preciznost i robusnost modela u prepoznavanju znamenki.



Slika 3. Matrica zbunjenosti modela.

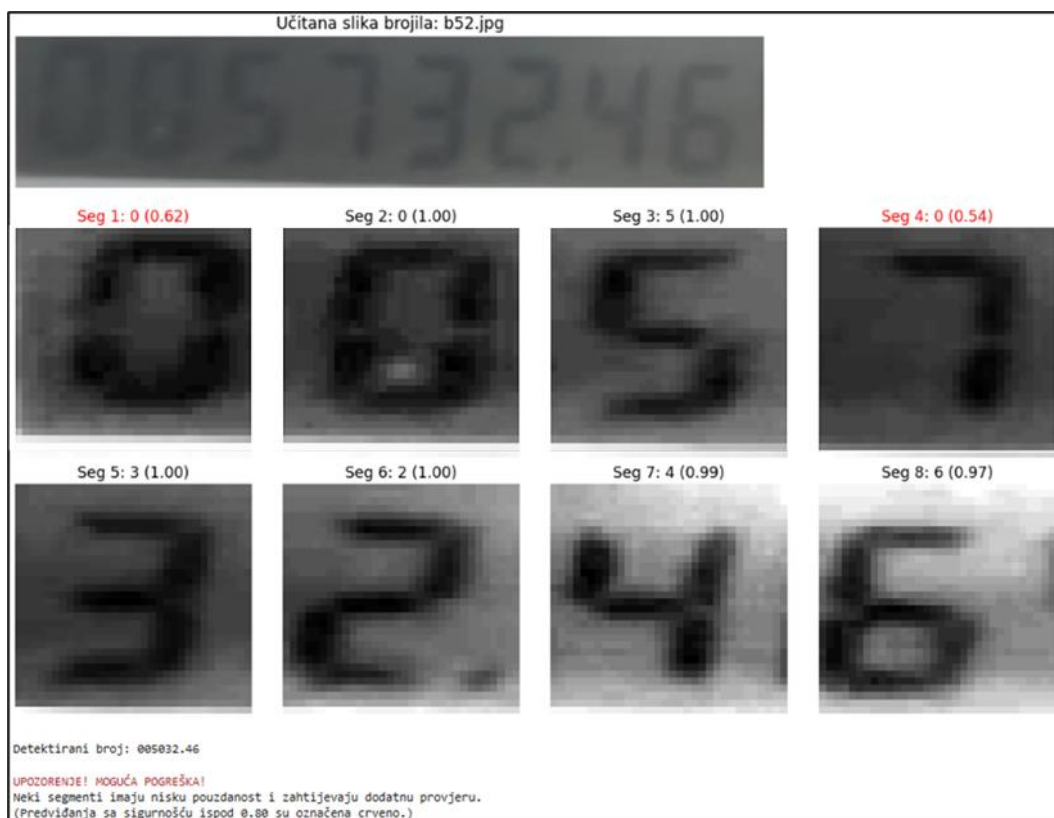
Izvor: autorica.

4.3. PRIMJENA MODELA I DETEKCIJA NESIGURNOSTI

Implementacija modela obuhvaća predprocesiranje ulazne fotografije digitalnoga brojila i segmentaciju očitavanja. Ulazna fotografija prvo se prilagođava standardiziranim dimenzijama (250x52 piksela), a zatim se dijeli na osam jednakih segmenata, od kojih svaki odgovara jednoj očekivanoj poziciji znamenke. Svaki segment se neovisno obrađuje (konverzija u sive tonove, skaliranje na 32x32 piksela, normalizacija) prije prosljeđivanja istreniranom CNN modelu za predikciju. Decimalna točka, zbog varijacija u razmaku na stvarnim brojilima, ručno se umeće na svoju fiksnu poziciju (nakon šeste znamenke) unutar prepoznatoga niza. Robusnost modela osigurana je treniranjem na datasetu koji uključuje slike znamenki u blizini decimalne točke, kao i zasebnu klasu za točku, čime se sprječavaju pogrešne klasifikacije.

Model, koristeći Softmax aktivacijsku funkciju, za svaki segment generira distribuciju vjerojatnosti pripadnosti pojedinoj klasi. Najveća vjerojatnost određuje konačnu predikciju [13]. Kako bi se kontrolirala pouzdanost tih predikcija, implementiran je sustav za detekciju nesigurnosti modela. Definiran je prag sigurnosti (engl. *confidence threshold*) od 0.80 (80 %). Ako model predvidi znamenku s vjerojatnošću nižom od ovoga praga, aplikacija to interpretira kao potencijalnu pogrešku ili situaciju koja zahtijeva dodatnu provjeru. Primjer takvoga ponašanja vidljiv je na slici 4, koja prikazuje očitavanje brojila s ulazne fotografije snimljene u otežanim uvjetima (mutnoća, smanjena jasnoća). Stvarno stanje brojila bilo je 005732.46. Model je predvidio 005032.46, pri čemu je pogriješio u segmentu 4 (predvidio 0 umjesto 7). Vjerojatnost za tu pogrešnu predikciju iznosila je 0.54 što je značajno niže od praga od 0.80. Aplikacija je u ovom slučaju vizualno signalizirala nisku pouzdanost bojenjem segmenta u crvenu boju i prikazala upozoravajuću tekstualnu poruku čime proaktivno zahtijeva ljudsku intervenciju i

potvrdu. Ovaj mehanizam osigurava visoku pouzdanost sustava u realnim, izazovnim uvjetima smanjujući rizik od neprimijećenih pogrešaka.



Slika 1. Primjer detekcije nesigurnosti modela pri otežanim uvjetima snimanja.

Izvor: autorica.

5. ZAKLJUČAK

Ovaj rad uspješno je demonstrirao visoku učinkovitost konvolucijskih neuronskih mreža (CNN) u zadatku automatiziranoga prepoznavanja znamenki na digitalnim brojilima, čime je potvrđen potencijal dubokoga učenja za rješavanje složenih problema računalnoga vida u realnim aplikacijama. Razvijeni model, treniran na vlastitom, opsežnom i augmentiranom skupu podataka te primjenom odabrane CNN arhitekture, postigao je iznimno dobre rezultate. Krivulje treniranja i validacije jasno su pokazale brz pad funkcije gubitka i postizanje gotovo 100 % točnosti, što ukazuje na robustan proces učenja i sposobnost modela da generalizira na neviđene podatke.

Model pokazuje iznimnu preciznost u idealnim uvjetima, a njegova je robusnost posebno naglašena pri obradi izazovnih vizualnih uvjeta. Čak i na fotografijama brojila snimljenima u otežanim uvjetima, poput mutnoće i smanjene jasnoće, model je sposoban točno prepoznati veliku većinu znamenki. Implementacijom mehanizma za detekciju nesigurnosti modela, osigurava se visoka pouzdanost sustava u praksi, prebacujući potrebu za ljudskom provjerom samo na slučajeve kada model nije siguran u svoju predikciju.

Ovakav model treniran za prepoznavanje znamenki s digitalnih brojila ima širok spektar primjene u praksi, značajno unaprjeđujući automatizaciju procesa očitavanja. Potencijalne primjene uključuju

mobilne aplikacije koje korisnicima omogućuju precizno i samostalno slanje očitavanja brojila, smanjujući potrebu za fizičkim obilascima i ručnim unosima.

Za budući rad, predlaže se daljnje unaprjeđenje modela kroz analizu i dodavanje novih, izazovnih primjera slika brojila koje korisnici prijavljuju osobito onih s izraženom mutnoćom, neobičnim osvjetljenjem ili drugim kompleksnim vizualnim izazovima. Dodavanjem takvih podataka u postojeći skup za treniranje i ponovnim treniranjem modela njegova bi se sposobnost generalizacije i otpornost na izrazito nepovoljne uvjete dodatno poboljšala, smanjujući potrebu za ljudskom intervencijom i povećavajući ukupnu pouzdanost automatiziranoga sustava.

6. LITERATURA

[1] Aghdam H.H.; Heravi E.J.; Puig D. (2016). „Analyzing the Stability of Convolutional Neural Networks against Image“. Conference: International Conference on Computer Vision Theory and Applications. Vol. 4, str. 370-382

[2] Ahlawat S.; Choudhary A.; Nayyar A.; Singh S.; Yoon B. (2020). „Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN)“. Sensors Vol 20 (12).

[3] Anand S.M. (2025). „Kickstart Artificial Intelligence Fundamentals: Master Machine Learning, Neural Networks, and Deep Learning from Basics to Build“. Orange Education Pvt. Ltd.

[4] Chen L.; Wang S.; Fan W.; Sun J.; Naoi S. (2015). „Beyond Human Recognition: A CNN-Based Framework for Handwritten Character Recognition“. 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR) , str. 1-5. Izdavač: IEEE.

[5] Chesterfield G. (2024). „Advanced Image Processing with Python and OpenCV“. Independently published.

[6] Gupta K. (2019). „Digit Recognition Using Convolution Neural Network“. Priyadarshini Institute of Engineering & Technology, Nagpur, India.

[7] Haseeb H.; Hassan M.T.; Iftikhar A.; Asmat A. (2024). „Automated Detection and Recognition of Seven-Segment Digits from Electric Meters Utilizing Digital Image Processing and Machine Learning“. VFAST Transactions on Software Engineering. Vol. 12(4), str.87-98.

- [8] Hossain M.A.; Sajib M.A.S. (2019). „Classification of Image using Convolutional Neural Network (CNN)“. Global Journal Of Computer Science And Technology: D Neural & Artificial Intelligence. Vol. 19(2).
- [9] Karahan S.; Yildirim M. K.; Kirtac K.; Rende F. S.; Butun G.; Ekenel H. K. (2016). „How Image Degradations Affect Deep CNN-based Face Recognition?“. Computer Vision and Pattern Recognition.
- [10] Khorsheed E.A.; Al-Sulaifanie A.K. (2024). „Handwritten Digit Classification Using Deep Learning Convolutional Neural Network“. Journal of soft computing and data mining, vol. 5 (1), str. 79-90.
- [11] Maitra D.S.; Bhattacharya U.; Parui S.K. (2015). „CNN Based Common Approach to Handwritten Character Recognition of Multiple Scripts“. 2015 13th International Conference on Document Analysis and Recognition (ICDAR), str. 1021-1025.
- [12] Peng J.; Zhou W.; Han Y.; Liu W. (2023). „Deep Learning-based Autonomous Real-time Digital Meter Reading Recognition Method for Natural Scenes“. Measurement, vol 222.
- [13] Rajput V. (2024). „Ultimate Neural Network Programming with Python“. India: BPB Publications.
- [14] Srivastava P.K.; Yadav A.K. (2024). „Methodologies, Frameworks, and Applications of Machine Learning“. Hershey, PA: IGI Global.
- [15] Wong M. (2023). „Neural Networks with Python: Design CNNs, Transformers, GANs and Capsule networks using Tensorflow and Keras“. Published by Packt Publishing Ltd., UK