

Sistem nadgrajene resničnosti za verifikacijo predmetov v skladiščnih okoljih

Janez Križaj, Janez Perš, Simon Dobrišek in Vitomir Štruc

Univerza v Ljubljani, Fakulteta za elektrotehniko, Tržaška 25, 1000 Ljubljana, Slovenija
E-pošta: janez.krizaj@fe.uni-lj.si

Povzetek. V članku je predstavljen podporni sistem za skladiščne delavce, ki temelji na principu nadgrajene resničnosti. Predlagani sistem zajema slike objektov, s katerimi se srečuje skladiščni delavec pri svojem delu, in sproti preverja istovetnost objektov, kar je ključnega pomena za učinkovito delo skladiščnikov. Rezultat preverjanja se sproti izrisuje na zaslon pametnih očal, ki jih delavec uporablja kot vmesnik za prikazovanje povratne vizualne informacije. V sistemu zajem slik, potrebnih za preverjanje istovetnosti objektov, poteka prek kamere na pametnih očalih, računsko zahtevni del postopka preverjanja istovetnosti pa se izvaja v oblaku in je udejanjen s pomočjo sodobnih postopkov globokega učenja. Predstavljena zasnova sistema nam omogoča obdelavo slik blizu stvarnemu času ter visoko uspešnost preverjanja istovetnosti, kar je razvidno tudi iz rezultatov predstavljenih eksperimentov.

Ključne besede: pametna očala, skladiščni sistemi, verifikacija objektov, nadgrajena resničnost, globoko učenje

Augmented reality system for object verification in warehouse environments

The paper proposes an augmented reality system for visual object verification that helps warehouse workers perform their work. The system sequentially captures images of objects that the warehouse workers encounter during their work and verifies whether the objects are the ones that the workers are supposed to fetch from storage. The system uses Android-powered smart glasses to capture image data and display results to the user, whereas the computationally-intensive verification task is carried out in the cloud and is implemented using recent deep-learning techniques. By doing so, the system is able to process images in near real-time and achieves a high verification accuracy as shown by the experimental results.

Keywords: smart glasses, warehouse systems, object verification, augmented reality, deep learning

1 UVOD

V članku predstavimo sistem nadgrajene resničnosti (ang. augmented reality), ki skladiščnim delavcem omogoča zanesljivejše in hitrejše izvajanje svojih nalog. Pri obstoječem načinu dela skladiščni delavec v informacijskem sistemu najprej poišče lokacijo določenega proizvoda v skladišču, na dotični lokaciji pa nato z ustreznim čitalnikom preveri njegovo istovetnost. Razviti sistem takšen postopek v veliki meri avtomatizira, saj s pomočjo pametnih očal zajema sliko objektov (oz. proizvodov, predmetov), s katerimi se srečuje skladiščni delavec pri svojem delu, in sproti preverja istovetnost objektov na zajetih slikah. Rezultat preverjanja se izrisuje v živo sliko, ki se prikazuje na zaslonu pametnih očal in s tem delavcu daje sprotno povratno informacijo



Slika 1: Ilustrativni prikaz zgradbe podpornega sistema navidezne resničnosti. Sistem sestavljajo tri komponente: pametna očala, skladiščni računalnik in strežnik. Uporabnik na pametnih očalih vidi sliko objekta z detekcijskim oknom, ki se glede na rezultat preverjanja istovetnosti obarva zeleno (pravilen objekt) oz. rdeče (nepravilen objekt).

– glej sliko 1. S predstavljenim tehnologijom je torej v delovni proces skladiščnih delavcev vpeljan avtomatiziran proces preverjanja istovetnosti proizvodov, ki je izveden z računalniško podprtou tehnologijo razpoznavanja objektov iz slikovnih podatkov.

Glavni izziv in razvojni problem, s katerim se srečamo pri udejanjenju sistema nadgrajene resničnosti je problem preverjanja istovetnosti (oz. verifikacije) objektov na podlagi slikovnih podatkov. Procesno verigo verifikacijskega postopka sestavimo iz klasičnih gradnikov, kot so detektor objektov, modul za izračun značilk in modul za razvrščanje, pri čemer za izvedbo večine komponent uporabimo sodobne postopke, ki temeljijo na globokih nevronskeih mrežah [1], [2].

Zaradi relativno visoke računske zahtevnosti teh postopkov večji del procesne verige prenesemo v oblak, kjer so na voljo zadostni računski viri in grafične procesne enote, ki podpirajo pospešeno izvajanje in učenje globokih mrež. Pametna očala, na katerih teče operacijski sistem Android, nam pri tem služijo le kot uporabniški vmesnik ter strojna oprema za zajem in prenos slike v oblak. Ob predpostavki dovolj hitre spletne povezave tako zagotovimo sprejem povratne informacije v obliki rezultata preverjanja istovetnosti in koordinat detekcijskega okna v okvirno 0,2 s po tem, ko sliko posredujemo v obdelavo v oblak.

2 IMPLEMENTACIJA SISTEMA NADGRAJENE RESNIČNOSTI

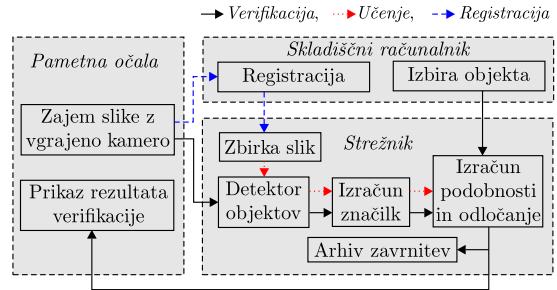
2.1 Način delovanja

Sistem začne s poskusi razpoznavanja proizvodov na zajetih slikah, ko uporabnik z izbiro želenega objekta na skladiščnem računalniku sproži postopek preverjanja istovetnosti. Od začetka razpoznavanja se z določenim intervalom vzorčenja zajemajo slike in pošiljajo v obdelavo na strežnik. Rezultat obdelave se prikaže na zaslonu v pametnih očalih v obliki detekcijskega okna, ki nakujuje položaj proizvoda v trenutni sliki (glej sliko 1). Če je istovetnost proizvoda potrjena, se detekcijsko okno objekta obarva zeleno, v nasprotnem primeru rdeče. Kadar sistemu istovetnosti proizvoda na uspe potrditi, v oblaku shrani sliko nepotrenjenega proizvoda za namene arhiviranja in poznejšega pregledovanja.

2.2 Zgradba sistema

Kot je prikazano na sliki 2, podporni sistem v grobem sestavlja naslednji trije deli:

- *Pametna očala*: Na pametnih očalih teče spletna aplikacija, ki omogoča zajem slike z vgrajeno kamero očal, prikaz slike na zaslonu očal, pošiljanje slike v obdelavo na strežnik v formatu JPEG, sprejem rezultata preverjanja istovetnosti iz strežnika in prikaz rezultata vključno z detekcijskim oknom na zaslonu očal.
- *Strežnik*: Ob sprejemu slike strežnik pošlje dano sliko skozi procesno verigo, zgrajeno iz klasičnih gradnikov razpoznavnega sistema, to so *i*) modul za predobdelavo slik, *ii*) modul za izračun značilk in *iii*) modul za merjenje podobnosti in odločanje. Strežnik nato izračunan rezultat postopka preverjanja istovetnosti in koordinate detekcijskega okna za vsako dobljeno sliko pošlje nazaj do pametnih očal in obenem arhivira slike pri katerih je prišlo do zavrnitev pri preverjanju istovetnosti.
- *Skladiščni računalnik*, na katerem se izvaja registracija novih objektov v sistem in izbira objekta, za katerega želimo preveriti istovetnost. Objekt registriramo s slikanjem vsake od njegovih stranic. Teh šest slik prenesemo na strežnik, kjer postopek za samodejno tvorjenje (umetnih) slikovnih podatkov



Slika 2: Shematski prikaz podpornega sistema s prikazanimi posameznimi gradniki

generira večje število slik tega objekta, nato pa se na strežniškem delu izvede še učenje detektorja in razvrščevalnika na novo generirani učni zbirki.

Podporni sistem deluje v treh načinih. Način registracije omogoča registracijo novih objektov v sistem. V načinu učenja s postopki strojnega učenja prilagodimo parametre detektorja objektov in razvrščevalnika učnim podatkom, ki jih pridobimo v postopku registracije. Način verifikacije oz. preverjanja istovetnosti pa omogoča preverjanje istovetnosti objektov, ki jih zajamemo s kamero pametnih očal. Sledenča podpoglavlja vsebujejo podrobnejši opis naštetih sklopov in delovanja razvitega sistema.

2.3 Registracija novih objektov v sistem

Podporni sistem deluje na predpostavki, da imamo opravka zgolj z objekti (oz. proizvodi) kvadraste oblike – postopek registracije zato zahteva zajem 6 slik vsake od stranic kvadra. Ob prenosu vseh šestih slik stranic objekta na strežnik se na strežniškem delu sistema izvede učenje detektorja in razvrščevalnika na novo dograjeni učni zbirki. Za učinkovitejše učenje dodamo še postopek samodejnega tvorjenja umetnih slik, ki temelji na gradnji 3D modela objekta iz njegovih slik stranic in omogoča preprosto povečanje učnega gradiva.

Postopek gradnje 3D modela iz slik stranic zahteva, da pri postopku registracije v grafičnem vmesniku ročno označimo vse štiri kote stranice na posamezni sliki, preostali koraki tvorjenja slik in učenja pa so samodejni. Na podlagi lokacije označenih koton se s perspektivično transformacijo izvede poravnava stranice v frontalno lego. Model danega objekta nato sestavimo iz “frontaliziranih” slik vseh šestih stranic objeta. Umetne slike, ki jih uporabimo za učenje detektorja in razvrščevalnika, končno tvorimo z rotacijo modela v 3D prostoru in dodajanjem različnih ozadij, kot prikazuje slika 3.

2.4 Detekcija objektov

Za detekcijo objektov uporabimo postopek Faster-RCNN [3], implementiran v knjižnici CNTK [4]. Omenjeni postopek uporablja za detekcijo objektov konvolucijske nevronske mreže (implementacija trenutno podpira mreži VGG16 [5] in AlexNet [6]) in za učenje potrebuje veliko učnega gradiva. Ker z zahtevano količino



(a) Slike stranic objekta z označenimi robnimi točkami



(b) Frontalizirane (oz. poravnane) slike stranic objekta



(c) 3D model



(d) Umetno tvorjene slike

Slika 3: Postopek registracije in tvorjenja umetnih slik, s katerim iz šestih slik stranic registriranega objekta tvorimo poljubno število slik tega objekta

učnih slik ne razpolagamo, v sistemu uporabimo globoko mrežo, ki je bila vnaprej naučena na zbirkki slik ImageNet [7], parametre mreže pa zgolj doučimo na lastni zbirkki slik. Postopek se izkaže za hitrega in učinkovitega, z deležem detektiranih objektov preko 96% in povprečnim IOU (ang. intersection over union) enakim 0,78. Detekcijsko okno služi za kvadratniz izrez predela, ki ga zahteva modul za izračun značilk.

2.5 Izračun značilk

Ker postopki globokega učenja v zadnjih letih dosegajo visoke rezultate na področju računalniškega vida, se odločimo, da izračun značilk podobno kot detekcijo prav tako izvedemo s pomočjo globokih mrež. V ek-

sperimentalni primerjavi (poglavlje 3.2) se izkaže, da največjo učinkovitost verifikacije zagotavlja značilke, ki jih dobimo z mrežo Resnet [8] na predzadnji plasti nevronov (izpustimo zadnjo polno povezano plast). Zato se odločimo, da za izračun značilk najprej uporabljamo mrežo ResNet. Značilke izračunamo iz kvadratnega območja, ki zajema detekcijsko okno objekta, pri čemer v eksperimentalnem delu v razdelku 3.3 preverimo, kakšen vpliv ima na končno uspešnost verifikacije ohranitev oz. odstranitev ozadja zunaj detekcijskega okna.

2.6 Izračun podobnosti in odločanje

Objekt, ki ga posnamemo s kamero pametnih očal, primerjamo zgolj z registriranim objektom, ki ga pred tem izberemo v uporabniškem vmesniku. Na podlagi mere podobnosti med značilkami danega objekta in značilkami izbranega objekta izvedemo sprejem oz. zavrnitev objekta glede na to, ali je izračunana podobnost večja ali manjša od vnaprej nastavljenega praga. Določitev praga in preizkus različnih vrst mer podobnosti eksperimentalno ocenimo v razdelku 3.4.

2.7 Uporabniški vmesnik

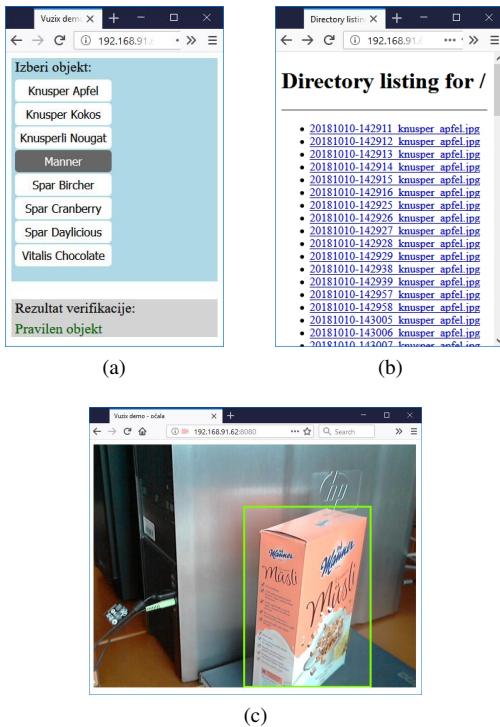
Komunikacija uporabnika s sistemom je omogočena prek več uporabniških vmesnikov, ki so udejanjeni v obliki spletnih aplikacij, zato delovanje podpornega sistema ni odvisno od operacijskega sistema in razpoložljivosti razvojnega kompleta (SDK). Vmesniki so pisani v jezikih HTML5, CSS, JavaScript in Python.

Na začetku preverjanja istovetnosti na skladiščnem računalniku prek ustreznegra vmesnika (slika 4a) najprej izberemo tip objekta, za katerega želimo preveriti istovetnost. Ko izberemo objekt, vmesnik na pametnih očalah (slika 4c) začne prikazovati zajete slike in jih hkrati pošilja v obdelavo na strežnik. Iz smeri strežnika z majhno zakasnitvijo pridobi podatek o detekcijskem oknu, ki ga nato izriše na živo sliko. Če je postopek na strežniku potrdil istovetnost objekta se detekcijsko okno obarva zeleno, drugače rdeče.

Vsi neuspešni poskusi preverjanja istovetnosti se shranijo na strežnik. Do arhiva slik, pri katerih je prišlo do zavrnitve, lahko dostopamo z vpisom ustreznegra spletnega naslova, pri čemer se nam odpre okno, kot ga prikazuje slika 4b. Z izbiro posamezne povezave v seznamu se nam prikaže slika objekta, ki je bil zavrnjen. Vmesnik za beleženje zavrnitev udejanjimo kot HTTP strežnik slik s pomočjo programske knjižnice ImageMe [9].

2.8 Komunikacijski kanali

Podporni sistem deluje na treh enotah, ki vključujejo strežniški računalnik, kjer se izvaja glavnina preverjanja istovetnosti, pametna očala s pripadajočo spletno kamero ter skladiščni računalnik, na katerem dostopamo do vmesnika za registracijo in izbiro vrste objekta. Ko sistem deluje v načinu verifikacije, je potrebno sprotno pošiljanje podatkov med pametnimi očali in skladiščnim



Slika 4: Uporabniški vmesniki podpornega sistema, ki so izvedeni v obliki spletnih aplikacij: (a) vmesnik za izbiro objekta, (b) vmesnik za dostop do arhiva slik, (c) vmesnik na pametnih očalih

računalnikom na eni strani ter strežnikom na drugi. V ta namen z uporabo spletnih vtičnic (ang. websockets) udejanjimo komunikacijska kanala, ki omogočata obojestranski prenos podatkov. Komunikacijski kanal med vmesnikom pametnih očal in strežnikom skrbi za prenos slike v smeri do strežnika in za prenos rezultata verifikacije ter koordinat detekcijskega okna v nasprotni smeri. Na komunikacijskem kanalu med vmesnikom za izbiro objekta in strežnikom pa v smeri proti strežniku poteka prenos oznake izbranega objekta, v nasprotni smeri pa teče prenos rezultata preverjanja istovetnosti.

Implementirani komunikacijski kanali omogočajo, da sistem z intervalom vzorčenja 0,2 s zajema slike iz kamere na pametnih očalih in jih pošilja v obdelavo na strežnik. Povratno informacijo v obliki rezultata preverjanja istovetnosti in koordinat detekcijskega okna ob predpostavki zadosti hitre internetne povezave pridobimo v okvirno 0,2 s po oddaji zajete slike.

2.9 Programska oprema

Jedro podpornega sistema, tj. proces preverjanja istovetnosti, ki teče na strežniku, udejanjimo v programskejem jeziku Python, pri čemer si pomagamo z različnimi programskimi knjižnicami. S pomočjo knjižnice CNTK izvedemo detekcijo objektov, knjižnici Tensorflow [10] in Keras [11] uporabimo pri izračunu značilk, knjižnica Mayavi [12] nam služi za 3D modeliranje objektov v procesu registracije, knjižnico Sklearn pa uporabimo pri

merjenju podobnosti med vektorji značilk. Pri udejanjenju komunikacijskih kanalov s strežnikom pa uporabimo knjižnico Websockets. Spletni uporabniški vmesniki so udejanjeni s pomočjo programskih jezikov HTML5, CSS in JavaScript.

Ker privzeti Androidov spletni brskalnik Google Chrome ne omogoča uporabe spletnih vtičnikov, ampak zahteva uporabo varnih spletnih vtičnikov (ang. web-socket secure), na pametnih očalih namestimo brskalnik Mozilla Firefox, ki ne pogojuje uporabe varnih spletnih vtičnikov, in posledično na njem lahko zaženemo uporabniški vmesnik.

2.10 Strojna oprema

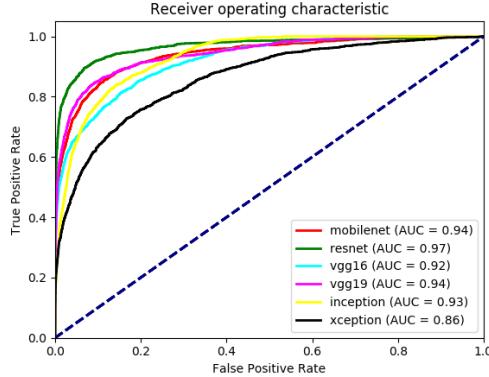
Demonstracijski sistem je izведен s pomočjo strežniškega računalnika, kjer se izvaja računska zahodna del procesa verifikacije, pametnih očal s pripadajočo spletno kamero ter skladiščnega računalnika, na katerem dostopamo do vmesnika za registracijo in izbiro objekta. Predvideva se, da je našteta oprema povezana s hitro spletno povezavo. Strežniški računalnik je opremljen z zmogljivo grafično kartico GeForce GTX TITAN X, ki omogoča hitro učenje in inferenco globokih mrež. Pri razvojnem delu uporabimo pametna očala Vuzix M300, na katerih teče operacijski sistem Android 6.0. Zaradi izvedbe uporabniških vmesnikov v obliki spletnih aplikacij podporni sistem ni omejen le na delovanje v povezavi s pametnimi očali, ampak lahko deluje na katerikoli napravi, ki podpira novejše spletne brskalnike in je opremljena s kamero ter z brezžično povezavo.

3 EKSPERIMENTI

V tem poglavju predstavimo rezultate izvedenih eksperimentov. S pomočjo pridobljenih eksperimentalnih rezultatov nastavimo oz. izberemo določene odprte komponente sistema, pri čemer težimo k čim večji učinkovitosti verifikacije, ki jo merimo v obliki deleža pravilnih potrditev istovetnosti (ang. True Positive Rate, TPR) pri 0,1% napačnih potrditvah istovetnosti (ang. False Positive Rate, FPR). V okviru testiranja preverimo tudi učinkovitost delovanja posameznih postopkov, ki jih integriramo v sistem, kot tudi delovanje prototipnega sistema kot celote. Eksperimente izvedemo na lastni zbirki slik kosmičev. Ker sistem deluje v načinu verifikacije, uspešnost delovanja prikažemo v obliki krivulj ROC (ang. Receiver Operating Characteristic).

3.1 Snemanje lastne zbirke slik

Za potrebe ovrednotenja zanesljivosti razvitega sistema zberemo lastno zbirko slik osmih različnih škatel kosmičev, s katerimi poskusimo posnemati objekte, ki jih pri svojem delu srečuje skladiščni delavec. Zbirka vsebuje testno in galerijsko množico. Testna množica vsebuje približno 100 slik vsake od škatel, pri čemer težimo k temu, da orientacija škatel na slikah čim



Slika 5: Uspešnost verifikacije ob uporabi različnih vrst prednaučenih globokih mrež, prikazana v obliki krivulj ROC. Največjo učinkovitost verifikacije zagotavlja mreža ResNet.

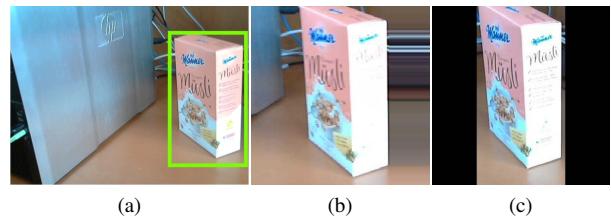
bolj variira. Galerjsko množico poleg šestih slik stranic vsakega od osmih registriranih predmetov sestavlja 1000 umetnih slik vsakega od objektov, ki jih tvorimo po postopku, opisanem v razdelku 2.3.

3.2 Izbira globoke mreže za izračunu značilk

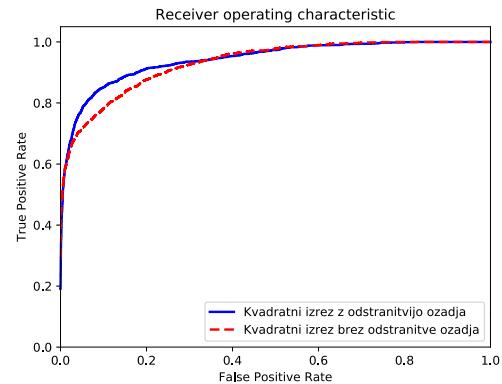
V nizu eksperimentov, opisanem v tem razdelku, preizkusimo uspešnost delovanja pri uporabi različnih prednaučenih globokih mrež, ki smo jih uporabili pri izračunu značilk. Globoke mreže potrebujejo velike količine podatkov za učinkovito učenje, ki je praviloma računsko zahtevno in posledično časovno potratno. Zato se težavam pri učenju mrež izognemo z uporabo mrež, ki so bile pred tem naučene na velikem številu generičnih slik. Osredinimo se na primerjavo šestih prednaučenih mrež iz knjižnice Keras, ki so bile naučene na zbirkni ImageNet. Te mreže so VGG16 in VGG19 [5], Inception [13], Xception [14], ResNet in MobileNet [15]. Rezultati eksperimentov so prikazani na sliki 5 v obliki krivulj ROC in pripadajočih površin pod krivuljami (ang. Area Under Curve, AUC), iz katerih lahko razberemo, da najuspešnejšo verifikacije dosežemo z mrežo ResNet. Zato v vseh nadaljnjih eksperimentih za izračun značilk uporabimo globoko mrežo ResNet.

3.3 Definicija območja pri izračunu značilk

Globoka mreža ResNet, ki jo uporabljam pri izračunu značilk, zahteva na svojem vhodu sliko kvadratne oblike z dimenzijo 229×229 . Detekcijski pravokotnik praviloma ni kvadratne oblike, ampak ima poljubno razmerje med višino in dolžino, ki je odvisno od oblike objekta na sliki. Obravnavamo dve možnosti izvedbe kvadratnega izreza pravokotnega območja detekcije: *i)* kvadratni izrez, kjer ohranimo ozadje in ga po potrebi ekstrapoliramo, če je detekcijsko okno blizu roba slike (slika 6b), *ii)* kvadratni izrez, kjer vrednosti slikovnih elementov zunaj detekcijskega okna nastavimo na določeno konstantno vrednost (črno območje na sliki 6c). Kot vidimo na sliki 7, se izkaže, da ta postopek prinese boljšo učinkovitost preverjanja istovetnosti kot postopek, kjer ohranimo ozadje.



Slika 6: Določitev kvadratnega območja pri izračunu značilk: (a) vhodna slika z detekcijskim oknom, (b) kvadratni izrez brez odstranitve ozadja in interpolacijo manjkajočih vrednosti, (c) kvadratni izrez z odstranitvijo ozadja.



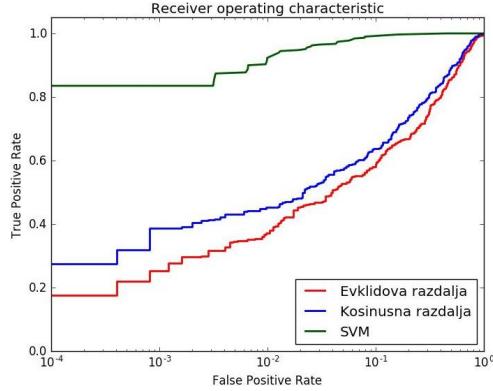
Slika 7: Uspešnost verifikacije glede na uporabljen postopek izreza v obliki krivulj ROC. Odstranitev ozadja iz kvadratnega okna za izračun značilk pripomore k izboljšanju uspešnosti preverjanja istovetnosti.

3.4 Izbira mere podobnosti in razvrščevalnika

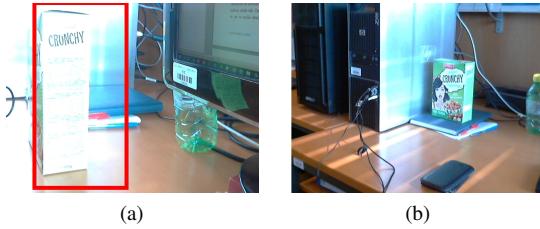
Pri izbiri mere podobnosti v modulu za odločanje poleg iskanja najbližjega soseda iz vsakega razreda z uporabo Evklidove in kosinusne razdalje preizkusimo še merjenje podobnosti z metodo podpornih vektorjev [16] (ang. Support Vector Machine, SVM), s katero, kot je razvidno iz slike 8, dosežemo znatno izboljšanje uspešnosti postopka preverjanja istovetnosti objektov. Odločitveni prag določimo glede na operacijsko točko, ki jo izberemo pri 0,1% napacnih sprejemov (ang. false positives). V tej točki z uporabo metode podpornih vektorjev uspešnost preverjanja istovetnosti preseže 80 % pravilnih sprejemov (ang. true positives).

3.5 Kvalitativno ovrednotenje

Pri ročnem pregledu slik opazimo, da imajo določeni pogoji zajema slik negativen vpliv na uspešnost preverjanja istovetnosti. Slika prikazuje primera, pri katerih je prišlo do napak pri tem postopku. Moteči dejavniki so npr. preosvetljenost slike (slika 9a), prevelika oddaljenost objekta, pri kateri odpove detektor (slika 9b). V primerih, kjer vzrok napak pri preverjanju istovetnosti ni očiten, sklepamo, da je prišlo do napake zaradi pomankljivega učnega gradiva pri učenju detektorja objektov in razvrščevalnika.



Slika 8: Uspešnost preverjanja istovetnosti objektov pri različnih postopkih merjenja podobnosti, prikazan v obliki ROC krivulj. Metoda SVM zagotavlja največjo uspešnost preverjanja istovetnosti med obravnavanimi razvrščevalniki.



Slika 9: Vzroki napak pri preverjanju istovetnosti objektov: (a) napaka zaradi preosvetljenosti slike, (b) detektor ne zazna objekta zaradi prevelike oddaljenosti od kamere

4 SKLEP

V članku smo predstavili podporni sistem navidezne resničnosti za skladiščne delavce, ki smo ga razvili v okviru projekta GOSTOP. Podporni sistem omogoča pohitritev opravil skladiščnih delavcev in lažje preverjanje istovetnosti proizvodov v skladiščih. Opisali smo osnovne značilnosti razvite storitve, orisali njen arhitekturo in funkcionalnost ter eksperimentalno ovrednotili njen delovanje. Pomembna lastnost omenjenega sistema je, da ga je mogoče uporabiti z vsako napravo, ki podpira novejše spletne brskalnike.

ZAHVALA

Predstavljeno raziskovalno delo je bilo delno financirano iz ARRS (raziskovalna programa Metrologija in biometrični sistemi, P2-0250, in Računalniški vid, P2-0214) ter programa GOSTOP (št. pogodbe C3330-16-529000), financiranega z RS-MIZŠ in EU-ESRR.

LITERATURA

- [1] K. Grm *et al.*, “Strengths and weaknesses of deep learning models for face recognition against image degradations,” *IET Biometrics*, zv. 7, št. 1, str. 81–89, 2018.
- [2] K. Grm, S. Dobrišek in V. Štruc, “Deep pair-wise similarity learning for face recognition,” v *4th Int. Conf. on Biometrics and Forensics (IWBFF)*, 2016, str. 1–6.
- [3] R. Girshick, “Fast R-CNN,” v *Proc. of the 2015 IEEE Int. Conf. on Comp. Vis.*, ser. ICCV ’15. Washington, DC, ZDA: IEEE Computer Society, 2015, str. 1440–1448.
- [4] F. Seide in A. Agarwal, “CNTK: Microsoft’s Open-Source Deep-Learning Toolkit,” v *Proc. of the 22nd ACM SIGKDD*, ser. KDD ’16. New York, NY, ZDA: ACM, 2016, str. 2135–2135.
- [5] K. Simonyan in A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” *CoRR*, 2014.
- [6] A. Krizhevsky, I. Sutskever in G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” v *Proc. of the NIPS’ 12 - Volume 1*, ZDA, 2012, str. 1097–1105.
- [7] J. Deng *et al.*, “ImageNet: A large-scale hierarchical image database,” v *Proc. of the IEEE CVPR*, 2009, str. 248–255.
- [8] K. He, X. Zhang, S. Ren in J. Sun, “Deep Residual Learning for Image Recognition,” v *Proc. of the IEEE CVPR*, 2016, str. 770–778.
- [9] J. Preston, “Programska knjižnica ImageMe,” dostopno na: <https://github.com/unwitting/imagine> [15. 10. 2018].
- [10] M. Abadi *et al.*, “TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems,” 2015.
- [11] F. Chollet *et al.*, “Keras,” dostopno na: <https://keras.io> [18. 10. 2018].
- [12] P. Ramachandran in G. Varoquaux, “Mayavi: 3D Visualization of Scientific Data,” *Computing in Science & Engineering*, zv. 13, št. 2, str. 40–51, 2011.
- [13] C. Szegedy *et al.*, “Going Deeper with Convolutions,” v *Proc. of the IEEE CVPR*, 2015, str. 1–9.
- [14] F. Chollet, “Xception: Deep Learning with Depthwise Separable Convolutions,” v *Proc. of the CVPR*, 2017, str. 1800–1807.
- [15] A. G. Howard *et al.*, “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications,” *CoRR*, 2017.
- [16] C. Cortes in V. Vapnik, “Support-Vector Networks,” *Mach. Learn.*, zv. 20, št. 3, str. 273–297, 1995.

Janez Križaj je doktoriral leta 2013 na Fakulteti za elektrotehniko Univerze v Ljubljani. V obdobju 2014–2018 je bil kot raziskovalec dejaven najprej pri dveh podoktorskih projektih, pozneje pa pri projektih v okviru programa GOSTOP. Področje njegovega raziskovanja obsega analizo in obdelavo signalov, biometrične sisteme, razpoznavanje vzorcev, strojno učenje in druga sorodna področja.

Janez Perš je doktoriral leta 2004. Trenutno je zaposlen kot docent v Laboratoriju za strojno inteligenco na Fakulteti za elektrotehniko Univerze v Ljubljani. Področje njegovega znanstvenega dela obsega računalniški vid, robotski vid, vgradne in porazdeljene sisteme ter strojni vid z aplikacijami v športu, morski robotiki in industrijski proizvodnji. Je član IEEE in IAPR ter član uredniškega odbora revije International Journal of Distributed Sensor Networks. Objavil je več kot sto recenziranih znanstvenih člankov.

Simon Dobrišek je doktoriral leta 2001 in je trenutno zaposlen kot izredni profesor na FE UL. Kot ključni raziskovalec je sodeloval pri številnih nacionalnih in EU projektih. Bil je projektni vodja in glavni raziskovalec pri projektih EU FP7 SMART in RESPECT, sopredsednik delavnice Workshop on Privacy Enabled Surveillance OPENSUR 2013, sopredsedajoči lokalnega organizacijskega odbora IEEE FG 2015. Je član IEEE, IAPR in ISCA. Objavil je več kot sto recenziranih znanstvenih člankov.

Vitomir Štruc je doktoriral leta 2010 in je trenutno zaposlen kot izredni profesor na FE UL. Raziskovalno se ukvarja s področjem razpoznavanja vzorcev, biometrije, strojnega učenja, računalniškegavida in drugimi sorodnimi področji. V okviru svojega dela je kot avtor ali soavtor objavil več kot sto člankov. Dr. Štruc je član uredniškega odbora revije IET Biometrics, IET Computer Vision, Signal Processing in IEEE Biometrics Compendium. Je član IEEE, IAPR, Slovenskega društva za jezikovne tehnologije (SDJT) in Slovenskega društva za razpoznavanje vzorcev (SDRV), ki mu trenutno tudi predseduje.