



SISTEM ZA OBRADU INFORMACIJA O KRETANJU OSOBA UNUTAR PROSTORIJA  
SYSTEM FOR ANALYZING MOVEMENTS OF PERSONS WITHIN ROOMS

Stefan Mičić, *Fakultet tehničkih nauka, Novi Sad*

**Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

**Kratak sadržaj** – U ovom radu predstavljen je sistem za obradu i analizu podataka o kretanjima osoba unutar prostorija. Naglasak je na distribuiranoj i skalabilnoj arhitekturi koja omogućava izvršavanje tih obrada i analiza. U okviru sistema su implementirane obrade koje se tiču broja ljudi u sobi, iskorišćenosti prostora, iskorišćenosti vremenskih perioda i pronalazak najčešćeg šablona kretanja. Pored modula za obradu podataka, u okviru predstavljenog sistema implementiran je i modul za vizuelizaciju podataka dobijenih na osnovu implementiranih obrada. Prilikom implementacije korišćene su tehnologije: Apache Spark, Kafka, HDFS, Flask i VueJS.

**Ključne reči:** Kafka, HDFS, Spark, Veliki skupovi podataka

**Abstract** – In this paper, we present the system for processing and analyzing data about person movement within rooms. The main focus is on a distributed and scalable architecture which enables development of these processing and analysis. There are several data processing algorithms which were implemented regarding number of people inside rooms, space utilization, percentage utilization, calculation of predefined periods of time and finding the most common movement pattern. Apart from data processing, visualization module is also implemented. For implementation Apache Spark, Kafka, HDFS, Flask and VueJS were used.

**Keywords:** Kafka, HDFS, Spark, Big Data

**1. UVOD**

Veliki skup podataka je skup podataka koji sadrži raznovrsne podatke, čija količina se neprestano uvećava i koji se prikupljaju velikom brzinom. U savremenom računarstvu, obrada velike količine podataka predstavlja izazov. Svakog dana nastaju novi skupovi podataka koji mogu biti preuzeti iz različitih izvora kao što su društvene mreže, video platforme i platforme za razmenu poruka. Prilikom obrade ovih skupova podataka, treba obezbediti algoritme koji će služiti za obradu novonastalih podataka i smestiti podatke u skladište.

Za obradu velikih skupova podataka, najčešće nije dovoljno imati samo jedan računar koji će da obavi svu potrebnu obradu i smesti sve podatke u svoju memoriju. Stoga se koristi distribuirana obrada i skladištenje podataka što podrazumeva potrebu za umreženom hardverskom infrastrukturom, odnosno postojanje klastera računara. Klaster računara predstavlja skup računara koji rade zajedno i čine jedan celoviti sistem.

U trenutku pisanja ovog rada, u 2021. godini, prisutna je pandemija virusa Kovid-19. U cilju prevencije daljeg širenja zaraze, nadležni organi apeluju da se u zatvorenim prostorijama okuplja ograničen broj ljudi uz poštovanje unapred definisanih mera zaštite i distanciranja. Naravno, kontrola ljudi koji se ne pridržavaju propisanih mera iziskuje dodatne troškove koje većina preduzeća ne može da priušti. Rešenje ovog problema može ležati baš u obradi velikih skupova podataka i automatizaciji procesa nadgledanja. Na primer, ukoliko se uzme u obzir tržišni centar koji ima stotinu poslovnih prostora za prodaju hrane, obuće, odeće i drugih artikala, može se smatrati da taj tržišni centar ima stotinu interesnih zona sa svojim izvorima podataka.

Jedan od izvora podataka mogu biti sigurnosne kamere, koje većina preduzeća već ima postavljene. Kamere se nalaze u svakom od tih poslovnih prostora i mogu se iskoristiti za davanje informacija o tome gde se koja osoba nalazi u određenoj prostoriji. Da bi se automatizovalo praćenje broja ljudi u prostorijama, potrebno je realizovati sistem koji će vršiti obradu podataka dobijenih od izvora kao što su sigurnosne kamere.

Realizacijom ovakvog sistema prestaje potreba za osobom koja će morati u svakom trenutku da vodi evidenciju koliko je osoba u prostoriji. Deljenjem prostorije na određene regione moguće je dobiti uvid u procenat ljudi koji je posetio datu regiju. S obzirom na to da je dostupna informacija o tačnoj poziciji osobe unutar prostorije, moguće je realizovati algoritam koji će pronalaziti najčešće šablone kretanja osobe kroz prostorije, što može pomoći vlasnicima objekata da bolje razumeju potrebe i kretanje kupaca.

Kako se uz svaku osobu sa kamere dobija i njen identifikacioni broj, moguće je izračunati i koliko osoba u proseku ostane u sobi. Ove, kao i mnoge druge obrade podataka, bile bi omogućene realizovanjem sistema za skladištenje i obradu velike količine podataka o kretanju osoba u prostorijama.

**NAPOMENA:**

Ovaj rad proistekao je iz master rada čiji je mentor bio dr Vladimir Dimitrieski, docent.

## 2. PREGLED TRENUTNOG STANJA U OBLASTI

U radu [1] opisan je sistem implementiran radi praćenja osoba unutar prostorija. U tu svrhu se koriste vektori dobijeni iz okvira izvučenih iz video snimka. Najpre se postave početne tačke na mestima odakle želimo da pratimo objekat. Kada se desi promena u regionu gde je tačka, odnosno kada objekat prođe kroz region, ona se pomera zajedno sa objektom. Ovim postupkom je dobijen vektor kretanja za svaku osobu koja je prošla kroz sobu. Prednost našeg sistema je u tome što nije potrebno da postoje prethodno definisane početne tačke za praćenje objekta, već se koriste detekcije u vidu četvorouglova u kojima se nalaze osobe dobijene od postojećeg modela za detekciju ljudi dobijenog mašinskim učenjem.

U radu [2] korišćena su dva tipa kamera koje omogućavaju identifikaciju osobe i određivanje pozicije osobe u sobi. Zadaci su bili pobuđivanje ovog sistema ako osoba dođe u određeni region, razumevanje potrebe osoba uz pomoć njihovih pravaca kretanja u cilju asistencije i slično. Osnovni nedostatak opisanog sistema je u tome što on nije skalabilan. Kada bi postojala potreba da se sistem upotrebi da paralelno radi u mnogo prostorija, bilo bi teško unaprediti ga za tu svrhu. Taj nedostatak ne postoji u našem sistemu jer je od početka planiran tako da jednostavno podrži rad sa mnoštvom prostorija kao i dodavanje novih izvora podataka.

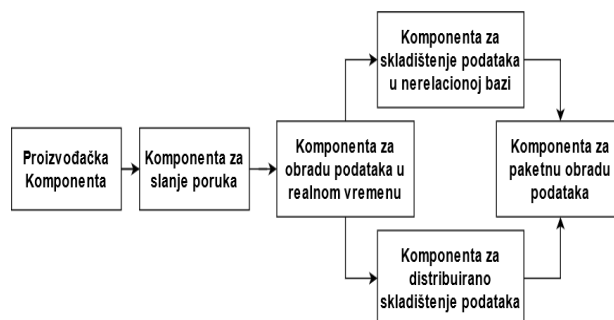
Ono što karakteriše radove [1] i [2] jeste činjenica da nisu zasnovani na upotrebi distribuiranih sistema za obradu podataka prilikom obrade informacija o poziciji osoba unutar prostorija. Sistem opisan u ovom radu donosi poboljšanje u odnosu na prethodno spomenuta rešenja jer predstavlja rešenje koje je nezavisno od načina implementacije sistema kao i od prostorija u kojima se osobe posmatraju. Korišćenjem informacija dobijenih od prethodno obučениh modela za detekciju ljudi, ovaj sistem može da prati ljude kroz koji god oni region u sobi prolazili. Prostorije u kojima se osobe prate mogu imati različite oblike i jedina informacija koja je potrebna jeste zatvorena kontura koja opisuje oblik sobe.

## 3. ARHITEKTURA SISTEMA

U ovom poglavlju biće opisana arhitektura sistema i dato detaljno objašnjenje svih njenih komponenti, kao i veza između njih.

Za potrebe ovog rada posmatramo prostorije kroz koje se svakodnevno kreću osobe i u svakoj od tih prostorija po jednu kameru koja šalje podatke na naš sistem. Prva komponenta koju ovaj sistem treba da ima je **proizvođačka komponenta** koja simulira slanje podataka sa kamera iz više prostorija. Sledeća komponenta je **komponenta za razmenu poruka**. Te poruke treba transformisati i sačuvati u obliku takvom da je kasnije moguće vršiti obrade nad njima te se javlja potreba za **komponentom za obradu podataka u realnom vremenu**. Zatim je potrebno smestiti novonastale podatke u distribuirani sistem datoteka koji može da primi velike količine podataka koje pristižu u realnom vremenu. Za to je potrebna **komponenta za distribuirano skladištenje podataka**. Nakon što su podaci prikupljeni, potrebno je obraditi podatke koji su dospeli u određenom vremenskom periodu i za taj deo sistema je neophodna

komponenta za paketnu obradu podataka. Nakon završetka paketne obrade podataka potrebno je smestiti rezultate obrade u bazu podataka kojoj će klijentska aplikacija moći lako da pristupi u cilju vizuelizacije. Komponenta u kojoj se čuvaju podaci se naziva **komponenta za skladištenje podataka u nerelacionoj bazi podataka**. Nakon definisanja komponenti za obradu i skladištenje javlja se potreba za klijentskom aplikacijom koja se može realizovati kroz dve dodatne komponente sistema: komponenta sa aplikativnim serverom (engl. *Backend*) kao i vizuelizaciona komponenta (engl. *Frontend*). Na slici 1. prikazana je arhitektura dela projektovanog sistema koji se bavi distribuiranom obradom i smeštanjem podataka.



Slika 1: Arhitektura dela projektovanog sistema koji se bavi obradom velike količine podataka

Korišćeni podaci bili su u CSV (engl. *Comma-Separated Values*) formatu te je napravljena Pajton skripta, na slici prikazana kao proizvođačka komponenta, koja je čitala podatke i periodično ih slala sistemu za razmenu poruka. U zavisnosti od vremenske odrednice podatka, proizvođačka komponenta šalje redove iz pročitanoг fajla komponenti za slanje poruka. Strelice između komponenti predstavljaju smer toka podataka unutar sistema. Na primer, vidi se da od proizvođača podaci idu ka komponenti za razmenu poruka.

Platforma *Kafka* [3] je korišćena za realizaciju komponente za slanje poruka. *Kafka* je trenutno među najpopularnijim platformama za razmenu poruka koje se koriste u objavi-pretplati režimu. Ovakav način komunikacije se sastoji od proizvođača, potrošača i teme na koju se potrošači pretplaćuju. U ovom sistemu proizvođače predstavljaju kamere u prostorijama koje šalju poruke sa pozicijama osoba unutar njih. Poruke se dostavljaju potrošaču koji je komponenta za obradu u realnom vremenu, koja će biti objašnjena u narednom pasusu. Kako bi poruka došla od proizvođača do potrošača, potrebna je tema na koju će se potrošač pretplatiti. Za implementaciju sistema je korišćena jedna tema i pomoću nje se vrši prenos svih poruka.

Sledeća komponenta u sistemu je komponenta za obradu podataka u realnom vremenu. Za realizaciju je korišćen *Spark Streaming* [4]. Ova komponenta konzumira podatke koji se nalaze u sistemu za razmenu poruka i ima dva cilja.

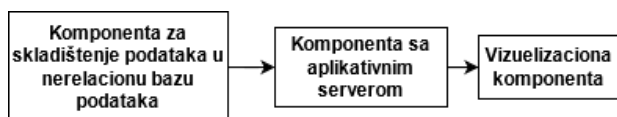
Za implementaciju komponente za distribuirano skladištenje podataka korišćen je HDFS (engl. *Hadoop Distributed File System*) [5]. Kako podaci stižu neprestano iz svake prostorije, jasno je da su količine podataka velike i postoji velika verovatnoća da ih nije

moguće smestiti na jedan računar. Zbog toga se koristi ova komponenta jer služi kao skladište obrađenih podataka u realnom vremenu. Ti podaci predstavljaju ulaz u komponentu koja će vršiti paketnu obradu i praviti obrade čiji će rezultati biti prikazani krajnjim korisnicima.

Najveća i najbitnija komponenta u sistemu je komponenta za paketnu obradu podataka. Ona predstavlja centralni deo sistema i služi za obradu svih uskladištenih podataka. Za potrebu realizacije se koristio *Spark* kao radni okvir za distribuiranu obradu podataka. Uz pomoć *Spark*-a, rezultati obrade ove komponente za paketnu obradu podataka se smeštaju u komponentu za skladištenje podataka u nerelacionoj bazi.

*MongoDB* [6] predstavlja sistem za upravljanje bazom podataka koji je korišćen prilikom realizacije komponente za skladištenje podataka u nerelacionoj bazi podataka. Dužnost komponente za obradu podataka u realnom vremenu je i da ukloni lažne detekcije, što znači da izbriše one osobe čije se lokacije ne nalaze unutar unapred definisanih regiona. Ti unapred definisani regioni se čuvaju pomoću komponente opisane u ovom paragrafu u vidu JSON (engl. *JavaScript Object Notation*) fajla.

Na slici 2. predstavljena je arhitektura aplikacije za prikaz podataka.



Slika 2: Arhitektura aplikacije za vizuelizaciju podataka

Za realizaciju komponente sa aplikativnim serverom korišćen je *Flask* [7]. Kako je ceo sistem za obradu i skladištenje pisan u Pajton programskom jeziku, izabran je *Flask* kao jedan od najpopularnijih radnih okvira za implementaciju veb aplikacija u tom programskom jeziku. Ova komponenta služi kako bi se rezultati obrada u realnom vremenu i paketnih obrada predstavili u formatu koji je pogodan za prikaz korisniku.

Vizuelizaciona komponenta koristi se za prikaz informacija korisniku kako u tabelarnom, tako i putem grafova, realizovana je pomoću *VueJS* [8] tehnologije. *VueJS* je odabran za razvijanje dela aplikacije za vizuelizaciju jer je jednostavan za korišćenje.

#### 4. FUNKCIONALNOSTI SISTEMA

U ovom poglavlju će biti opisane sve vrste obrade podataka koje se izvršavaju u realnom vremenu i prilikom paketne obrade. Takođe, biće objašnjen način na koji aplikacija preuzima te podatke i priprema ih za predstavljanje korisniku.

##### 4.1 Obrada u realnom vremenu

Obrada u realnom vremenu ima dva zadatka. Prvi je formatiranje i uklanjanje svih detekcija koje su lažne, dok je drugi prebrojavanje ljudi koji se nalaze u svakoj prostoriji.

Prilikom realizacije prvog zadatka potrebno je učitati sve regione za svaku prostoriju. Pre provere, dolazi do formatiranja podataka kako bi se lakše proverilo da li je osoba unutar regiona, poređenjem centra detekcija se regionom. Nakon što je odbacivanje lažnih detekcija

izvršeno, prelazi se na brojanje osoba koji se nalaze u prostorijama i rezultat se upisuje u bazu podataka. Izvršavanjem drugog zadatka, vlasnici objekata bi mogli da dobijaju upozorenje kada je u objektu više ljudi nego što je dozvoljeno.

##### 4.2. Paketna obrada

U okviru paketne obrade, realizovano je pet zadataka:

- brojanje ljudi koji se nalazio u nekoj prostoriji za određeni vremenski period,
- računanje procenta iskorišćenosti regiona u određenom vremenskom periodu za određenu prostoriju,
- računanje iskorišćenosti vremenskih intervala za određenu prostoriju,
- pronalazak najčešćeg šablona kretanja unutar određene prostorije i
- računanje prosečnog vreme zadržavanja osobe unutar prostorije u proteklih godinu dana.

Brojanje osoba u svakoj prostoriji u određenom vremenskom periodu pruža informaciju o tome koja prostorija je u kom intervalu bila najaktivnija. Rezultati ove obrade mogu olakšati rad poslovnih objekata tako što bi vlasnici bili obavešteni kada je veliki kapacitet rada potreban, a kad ne.

Računanje procentualne iskorišćenosti prostora unutar određene prostorije predstavlja neki vid proširenja prethodnog zadatka. Nakon što krajnji korisnik sistema pogleda koja osoba je bila aktivna ili neaktivna daje joj se mogućnost da pogleda koliko je svaki od regiona, unutar te prostorije, korišćen u tom vremenskom intervalu. Ako zamislimo prodavnicu koja se bavi prodajom odevnih predmeta, moguće je objekat podeliti na regione koji odgovaraju majicama, jaknama i farmericama, na primer. Korišćenjem ove obrade, dobija se podatak koji deo prostorije je bio najposećeniji u određenom vremenskom intervalu. Takođe, korisnici aplikacije znaće koju prostoriju je potrebno ponovo dezinfikovati jer je veliki broj ljudi boravio u njoj. Na taj način se sprečava potencijalno širenje virusa Kovid-19.

Sledeća obrada je pronalazak uobičajenog šablona kretanja osobe unutar prostorije. Cilj ove obrade je da utvrdi koje prostorije kupci najčešće koriste i time možda pokušaju da ih promenom rasporeda stvari u objektu podstaknu da prođu i kroz ostale delove.

Poslednja obrada koja se vrši jeste određivanje prosečnog vremena zadržavanja osobe u prostoriji. Ona pruža informaciju u kojim prostorijama kupci ne borave dugo, te radnici ne moraju te prostorije da dezinfikuju često jer je kapacitet ljudi manji.

##### 4.3. Aplikacija za prikaz rezultata obrada podataka korisniku

Poslednji deo sistema je aplikacija koja korisniku prikazuje sve prethodno navedene rezultate obrada u formatu koji su razumljivi krajnjem korisniku. Serverski deo aplikacije služi da sve prethodno izračunate obrade preuzme i formatira ih tako da vizuelizaciona komponenta može prikazati korisniku te podatke.

## 5. ZAKLJUČAK

U ovom radu opisan je sistem za obradu podataka o kretanjima ljudi unutar prostorija kao i njegova implementacija. Dat je pregled funkcionalnosti koje su implementirane kao i njihova svrha.

Osnovna prednost korišćenja ovakvog sistema jeste smanjivanje potreba za postojanjem više ljudi koji bi vršili kontrolu posećenosti određenih objekata. Ovaj sistem daje uvid u parametre posećenosti što može olakšati donošenje odluka vlasnicima, a kupcima pruža potrebne informacije vršeći izračunavanje:

- procenta posećenosti unapred definisanih regiona unutar prostorija,
- broja osoba koje su prošle kroz prostoriju u određenom vremenskom intervalu,
- najčešćeg šablona kretanja neke osobe i
- prosečnog vremena zadržavanja osobe u prostoriji.

Sistem je projektovan tako da ga je moguće lako adaptirati i primeniti na objekte sa većim brojem prostorija.

Dalja unapređenja sistema odnose se na usavršavanje algoritama za realizaciju pojedinih tipova obrade podataka. Moguće je dodatno usavršiti algoritam za pronalaženje najčešćeg šablona kretanja upotrebom grafova, umesto samog traženja najučestalijeg niza regiona. Sledeća stvar koju je moguće unaprediti tiče se aplikacije za vizuelizaciju rezultata obrada podataka i uključuje implementiranje vizuelizacije u realnom vremenu umesto da se podaci periodično osvežavaju.

## 6. LITERATURA

[1] Denman, S., Chandran, V., & Sridharan, S. (2007). An adaptive optical flow technique for person tracking systems. *Pattern recognition letters*, 28(10), 1232-1239.

[2] Krumm, J., Harris, S., Meyers, B., Brumitt, B., Hale, M., & Shafer, S. (2000, July). Multi-camera multi-person tracking for easy living. In *Proceedings Third IEEE International Workshop on Visual Surveillance* (pp. 3-10). IEEE.

[3] Garg, N. (2013). Apache kafka. Packt Publishing Ltd.

[4] Chambers, B., & Zaharia, M. (2018). Spark: The definitive guide: Big data processing made simple. " O'Reilly Media, Inc."

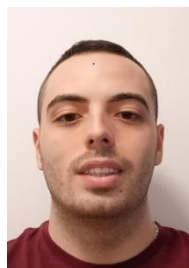
[5] Shvachko, K., Kuang, H., Radia, S., & Chansler, R. (2010, May). The hadoop distributed file system. In 2010 IEEE 26th symposium on mass storage systems and technologies (MSST) (pp. 1-10). Ieee.

[6] Chodorow, K. (2013). MongoDB: the definitive guide: powerful and scalable data storage. " O'Reilly Media, Inc."

[7] Vogel, P., Klooster, T., Andrikopoulos, V., & Lungu, M. (2017, September). A low-effort analytics platform for visualizing evolving Flask-based Python web services. In 2017 IEEE Working Conference on Software Visualization (VISSOFT) (pp. 109-113). IEEE.

[8] Nelson, B. (2018). Getting to Know Vue. js. Berkeley, CA: Apress. <https://doi.org/10.1007/978-1-4842-3781-6>.

### Kratka biografija:



**Stefan Mičić** rođen je 8. juna 1997. godine u Zvorniku, Republika Srpska. Fakultet tehničkih nauka upisao je 2016. na studijskom programu Računarstvo i Automatika. Bečelor rad na usmerenju Inteligentni Sistemi odbranio je 2020. godine. Master rad na usmerenju Inteligentni Sistemi odbranio je 2021. godine.