|  |  |
| --- | --- |
| **NASLOV PROJEKTNOG ZADATKA:** | *Automatsko kreiranje popisa za kupovinu pomoću detektora prehrambenih proizvoda* |
| **KOLEGIJ:** | *Primijenjeno strojno učenje* |
| **IME I PREZIME STUDENTA:** | *Anita Ivkov, Viktoria Juranić* |
| **AKADEMSKA GODINA:** | *2024./2025.* |
| **NAPOMENA:** |  |

# UVOD

Automatizacija svakodnevnih zadataka kao što je izrada popisa za kupovinu, predstavlja konkretan doprinos povećanju učinkovitosti u suvremenom načinu života. Klasičan način sastavljanja popisa je vrlo često naporan i podložan ljudskoj pogrešci. Cilj ovog projekta je razviti aplikaciju koja korisniku omogućava brzo, automatsko kreiranje popisa za kupovinu skeniranjem prehrambenih proizvoda pomoću kamere na mobilnom uređaju.

Ovaj projekt se temelji na primjeni dubokog učenja i računalnog vida – konkretno modela za detekciju objekata YOLOv8. Sustav koristi kameru na mobitelu povezanu s računalom putem aplikacije IP Webcam. Korištenjem već istreniranog modela i povezane baze podataka, detektirani artikli automatski se pretvaraju u strukturirani popis.

Ovaj dokument opisuje problem, analizira postojeće pristupe te prikazuje detalje implementacije i evaluira rezultate razvijenog rješenja.

# DEFINIRANJE PROBLEMA

2.1. Postojeći sustavi i metode

Detekcija objekata važan je segment računalnog vida koji ima široku primjenu u industriji, sigurnosti, zdravstvu i trgovini. Neki od najpoznatijih algoritama su:

* R-CNN (Region-based Convolutional Neural Network)
* Fast i Faster R-CNN
* SSD (Single Shot MultiBox Detector)
* YOLO (You Only Look Once)

Svaka od ovih metoda, naravno ima svoje prednosti i mane u pogledu brzine izvođenja, točnosti i složenosti implementacije. U ovom projektu odlučili smo koristiti YOLOv8 zbog visoke učinkovitosti u stvarnom vremenu i jednostavnog treniranja na vlastitim skupovima podataka.

2.2. Treniranje modela

Model je treniran na skupu podataka koji sadrži slike osam klasa prehrambenih proizvoda. Skup podataka sadrži više stotina slika za svaku klasu. Podaci koji se koriste prikupljeni su iz različitih izvora i ručno anotirani pomoću alata Roboflow.

Trening na podatcima je izveden na lokalnom računalu koristeći Ultralytics YOLOv8 biblioteku. Parametri treninga pažljivo su podešeni, a validacija modela rađena je na odvojenom dijelu skupa podataka.

2.3. Izazovi u implementaciji

* Različiti uvjeti osvjetljenja i okruženja prilikom snimanja proizvoda
* Disbalans među klasama (npr. više uzoraka riže nego piva)
* Potreba za vlastitim skupom podataka jer javni skupovi nisu specifični za lokalne proizvode
* Usklađivanje detekcija s bazom proizvoda te filtriranje nepoznatih artikala
* Tijekom implementacije sustava baza podataka, naišli smo na izazov s istovremenim pokušajima pisanja u SQLite bazu (sqlite3.OperationalError: database is locked)

1. PREDLOŽENO RJEŠENJE ZA AUTOMATSKO GENERIRANJE POPISA
   1. Arhitektura sustava

Sustav se sastoji od sljedećih komponenti:

* Detektor objekata (YOLOv8) koji analizira slike iz video streama
* Streamlit aplikacija koja omogućuje unos korisnika i pregled rezultata
* SQLite baze za pohranu podataka
* Mehanizam za odlučivanje koji temeljem pouzdanosti određuje hoće li se detekcija uvrstiti u popis
  1. YOLOv8: Detekcija prehrambenih proizvoda

YOLOv8 trenutno je najnovija inačica poznatog algoritma za detekciju objekata u stvarnom vremenu. U našem slučaju koristi se model treniran na sljedeće klase:

* Banane
* Jaja
* Kikiriki
* Krastavci
* Kruh
* Pivo
* Rajčica
* Riža
  1. Izrada i augmentacija vlastitog skupa podataka

Ukupno u projektu je korišteno 213 originalnih slika, podijeljenih po klasama i okruženju (normalno, kućno, divlje).

Za označavanje bounding boxova korišten je Roboflow, a augmentacija se provodila samo na točno odabranom trenirajućem skupu (70%).

Cilj augmentacije bio je:

* Očuvanje raznolikosti podataka
* Smanjenje overfittinga
* Izbjegavanje umjetnih uzoraka koji bi mogli izazvati bias

Primijenjene su umjerene transformacije (rotacija, šum, zoom), a broj augmentacija po slici bio je ograničen.

* 1. Funkcionalnosti aplikacije

Korisnik treba unijeti svoje korisničko ime, povezuje aplikaciju s IP Webcam aplikacijom za mobitel te započinje detekciju.

Da bi se osiguralo točno brojanje pojedinih instanci (npr. 'rajčica x2'), implementirano je praćenje objekata koje uzima u obzir vrijeme pojavljivanja i nestajanja iz kadra. Korišteni su parametri kao što su pragovi za minimalno vrijeme vidljivosti, tolerancija nestanka i prag resetiranja brojača. Na samom kraju, rezultati se pohranjuju u bazu.

Također, aplikacija prikazuje tablicu s rezultatima detekcija, uključujući naziv artikla, broj ponavljanja i najvišu zabilježenu točnost.

* 1. Implementacija baza podataka

Jedna SQLite baza app.db se koristi za:

* upravljanje korisnicima (tablica users)
* pohranu informacija o proizvodima (tablica foods)
* evidenciju kreiranih popisa (tablica shopping\_lists)

Korišten je Python modul sqlite3 uz kontekst menadžere koji osiguravaju da se pravilno otvaraju i zatvaraju konekcije. Baze se automatski inicijaliziraju prilikom pokretanja aplikacije.

* 1. Praćenje i unaprjeđenje modela

U buduće verzije sustava moguće je dodati:

* praćenje učinka modela kroz vrijeme (log točnosti po sesiji)
* ponovno treniranje modela s novim podacima (aktivno učenje)
* automatsko dodavanje novih klasa proizvoda
* pamćenje pragova detekcije namirnica (thresholdova) koje je korisnik postavio tijekom zadnjeg pokretanja; svaki korisnik bi imao vlastiti zapamćeni prag
* Integraciju mobilne aplikacije za lokalno korištenje.
* Proširenje baze podržanih proizvoda i višejezičnu podršku.
* Razvoj analitičkih izvještaja, poput pregleda najčešće korištenih artikala.
* Povezivanje s platformama za online naručivanje hrane.

1. EVALUCIJA PREDLOŽENOG RJEŠENJA

Model je testiran na testnom skupu od 21 slike. Detekcije su testirane u uvjetima u stvarnom životu s različitim svjetlima te kutovima.

| **Parametar** | **Vrijednost** |
| --- | --- |
| Točnost (mAP) | 88% |
| Vrijeme detekcije | ~0.12 s po slici |
| Broj klasa | 8 |
| Uporaba augmentacija | Samo na train skupu |
| Platforma | Windows 11 + Android |
| Kamera | IP Webcam |

Ovaj sustav se pokazao stabilnim i dovoljno preciznim za stvarnu primjenu.

**4.1 Okruženje i alati**

* OS: Windows 11
* Python 3.11.1
* Biblioteke: Streamlit, OpenCV, Ultralytics, SQLite3, NumPy, contextlib, datetime, collection
  + Streamlit: Za brzo prototipiranje i izgradnju interaktivnog web sučelja
  + OpenCV: Za obradu video streama i manipulaciju slikama
  + Ultralytics (YOLO): Za primjenu modela detekcije objekata
  + SQLite3: Za laganu bazu podataka
  + contextlib: Za osiguravanje pravilnog upravljanja resursima (otvaranje/zatvaranje DB konekcija)
  + collections.defaultdict: Za pojednostavljeno inicijaliziranje rječnika pri brojanju detekcija
* Kamera: Android kamera s aplikacijom IP Webcam
* Treniranje modela izvedeno lokalno (CPU) uz podršku GPU-a za inferenciju

**4.2 Rezultati i analiza**

Prosječna točnost detekcije na testnom setu:

* Jaja: 92%
* Banane: 90%
* Kruh: 88%
* Pivo: 75%

Sustav pouzdano prepoznaje većinu proizvoda, a pragovi po klasama omogućuju dodatnu kontrolu nad lažno pozitivnim detekcijama. Na primjer, pivu je dodijeljen niži prag jer se često pojavljuje u okruženjima sa slabim osvjetljenjem.

**4.3 Ograničenja sustava**

* Detekcija je ovisna o kvaliteti kamere i osvjetljenju
* Trenutno je podržano samo 8 proizvoda
* Ne postoji podrška za prepoznavanje proizvoda u različitim pakiranjima ili pozama
* Nema podrške za uređivanje spremljenih popisa

1. ZAKLJUČAK

U ovom projektu, razvili smo funkcionalan sustav za automatsko prepoznavanje prehrambenih proizvoda i generiranje popisa za kupovinu, koristeći model YOLOv8m i vlastiti skup podataka. Model je pokazao zadovoljavajuću preciznost i brzinu detekcije, a osjetljivost modela je popilično visoka, zbog čega su implementirani pragovi detekcije namirnice unutar aplikacije kojima korisnik može upravljati.

Izrada specifičnog dataseta i primjena kontrolirane augmentacije bili su ključni za postizanje ovih rezultata. Projekt je uspješno integrirao računalni vid, intuitivno korisničko sučelje i bazu podataka, nudeći rješenje s potencijalno značajnom primjenom u svakodnevnom životu.

LITERATURA

[1] Ultralytics YOLOv8. Dostupno na: https://docs.ultralytics.com  
[2] Streamlit Dokumentacija. https://docs.streamlit.io  
[3] S. Raschka, "Python Machine Learning", Packt Publishing, 2015.  
[4] S. Haykin, "Neural Networks: A Comprehensive Foundation", Prentice Hall, 1999.  
[5] Roboflow - anotacijski alat za vizualne skupove podataka. https://roboflow.com

---------------------------------------------------------------------------------------------------------------------------