SVEUČILIŠTE U SPLITU
FAKULTET ELEKTROTEHNIKE, STROJARSTVA I BRODOGRADNJE

POSLIJEDIPLOMSKI DOKTORSKI STUDIJ
ELEKTROTEHNIKA I INFORMACIJSKA TEHNOLOGIJA

KVALIFIKACIJSKI ISPIT

PRIMJENA METODA REDUKCIJE KOD OBRADE PODATAKA DETEKTORA CMS

Arijana Burazin Mišura

Split, rujan 2019.

SADRŽAJ

1. UVOD

Razvoj i dostupnost tehnologije omogućio je upotrebu računala i senzora u razne svrhe. Senzorske mreže prikupljaju mnogo različitih vrsta podataka. Generirani podaci rastu ne samo obujmom već i kompleksnošću. Porast dimenzije podataka otežava mogućnost pohrane i analize velikih količina podataka.

Tijekom godina, razvijeni su brojni alati za obradu podataka koji su uglavnom pohranjeni u matricama i vektorima. Linearna algebra – algebra matrica, postala je bitan alat u računalnoj obradi podataka. Međutim, pokazalo se da skalarna i vektorska polja te standardni matrični modeli korišteni za prikaz i obradu imaju brojna ograničenja kada se radi o velikim skupovima složenih podataka. Stoga se ukazala potreba za prilagodljivijim alatima za pohranu i analizu. Između ostalog, novi alati trebali bi olakšati rješavanje dva bitna problema vezana uz obradu velikih skupova podataka: kako različite tipove podataka prikazati što jednostavnijim modelom te kako odrediti srž podataka, u smislu manjeg skupa podataka koji sadrži što više informacije.

Tenzori predstavljaju poopćenje skalara i vektora, dok multilinearna algebra - algebra tenzora višeg reda, predstavlja prirodnu generalizaciju linearne algebre koristeći tenzore kao poopćenje matrica. Razvijeni su alati multilinearne algebre namijenjeni redukciji dimenzije tenzora. Njihovom upotrebom veliki broj varijabli može se zamijeniti manjim brojem koji još uvijek sadrži većinu informacija danih u izvornom skupu podataka.

Matrične i tenzorske dekompozicije postale su vrlo svestran i prilagodljiv alat neophodan u obradi, modeliranju, analizi i razumijevanju podataka, te tako našle primjenu u brojnim područjima uključujući obradu signala, rudarenje podataka, duboko učenje (engl. *Deep learning*) i mnoga druga. Duboko učenje pripada metodama strojnog učenja (engl. *Machine Learning - ML*) koje se temelje na umjetnim neuronskim mrežama (engl. *Artificial Neural Networks - ANN*). Zahvaljujući svojoj sposobnosti reproduciranja širokog spektra problema, ANN se primjenjuju u mnogim disciplinama pa tako i u fizici elementarnih čestica. Razvoj tehnologije i moćnih suvremenih klastera omogućava treniranje i implementaciju dubokih neuronskih mreža s ogromnim brojem parametara. Međutim, njihovu praktičnu primjenu otežavaju troškovi pohrane i računanja što znanstvenici pokušavaju riješiti primjenom tenzorske tehnologije.

Rad je koncipiran na sljedeći način. U drugom poglavlju opisana je obrada senzorskih informacija i dan je opis CERN-ovog eksperimenta kao izvora velike količine podataka koji nameće potrebu za redukcijom dimenzionalnosti. Treće poglavlje daje pregled metoda redukcije dimenzionalnosti baziranih na dekompoziciji matrice, u četvrtom poglavlju predstavljeni su tenzori - temeljni koncept višedimenzionalne algebre, nakon čega slijede metode dekompozicije tenzora. U petom poglavlju ukratko su predstavljene (konvolucijske) neuronske mreže kao potencijalni smjer u traženju inovativnog algoritma za obradu podataka prikupljenih na CMS detektoru CERN-ovog eksperimenta obzirom na sličnost s problemima koji se javljaju u polju računalnog vida. Motivacija za nastavak istraživanja dana je u poglavlju šest, nakon čega slijedi zaključak.

2. OBRADA SENZORSKIH PODATAKA

Tijekom proteklih desetljeća, razvojem računala i njihovom primjenom u različitim područjima, došlo je enormnog porasta volumena generiranih podataka. Dobiveni podaci zajedno s metodama uključenim u njihovu analizu, često su nazivani terminom Veliki podaci (engl. *Big Data*). Također su Veliki podaci ponekad opisani s "3V" modelom (Slika 2.1): volumen (engl. *Volume*) podataka, brzina (engl. *Velocity*) te raznolikost (engl. *Variety*). Pri tom se volumen odnosi na broj promatranja/mjerenja generiranih zapisa, brzina označava frekvenciju dodavanja informacija novog promatranja a ujedno i potrebu za procesiranjem podataka u gotovo realnom vremenu, dok raznolikost opisuje raznovrsnost podataka u smislu dimenzije, značajki ili raspona izvora.

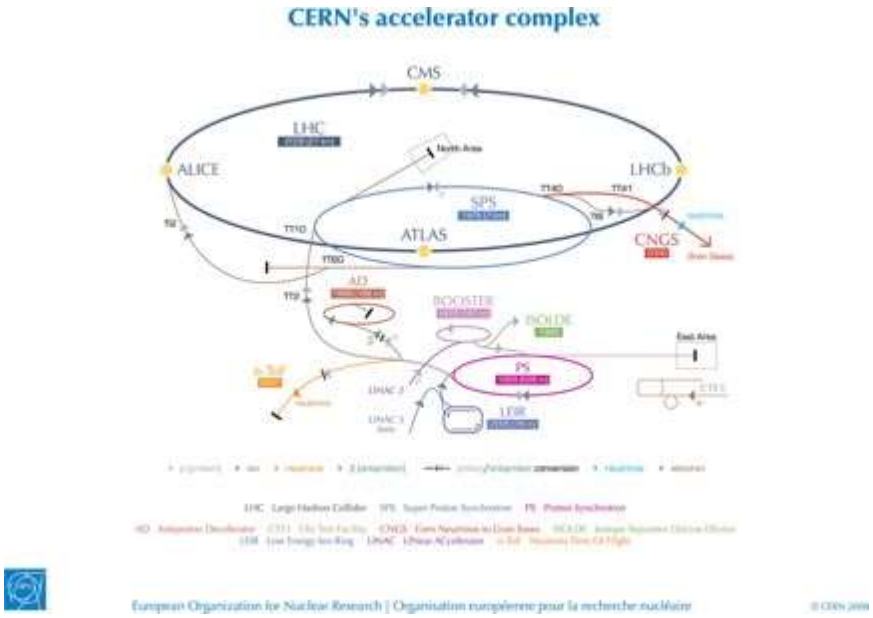
Slika 2.1. 3V model za Velike podatke [1]

Sam pojam, uključujući 3V model, definirao je Laney još 2001. u [2] opisujući izazove koje donosi era velikih količina podataka. Big data, za razliku od tradicionalnih pristupa, obično sadržavaju nestrukturirane podatke (velikog obujma) koji zahtijevaju veću analizu u stvarnom vremenu. Obzirom na veliki potencijal Big data, ne samo znanstvenici, već i razne grane industrije te vladine agencije pokazale su veliki interes za ubrzavanje istraživanja i moguće primjene.

2.1. CERN-ov Veliki hadronski sudarač kao primjer generiranja i obrade velike količine podataka

U organizaciji CERN-a (franc. *Conseil européen pour la recherche nucléaire*) nastao je jedan od najvećih međunarodnih znanstvenih projekata. S ciljem testiranja različitih predviđanja fizike elementarnih čestica, kao rezultat suradnje preko 10 000 znanstvenika i inženjera iz više od 100 država, izgrađen je Veliki Hadronski Sudarač (engl. *Large Hadron Collider - LHC*), najveći svjetski akcelerator čestica. Podaci dobiveni eksperimentom te njihova obrada predstavljaju jedan od najpoznatijih primjera procesiranja velike količine podataka. Obim generiranih podataka (prema [3], samo u studenom 2018. pohranjeno je 15.8 PB podataka) predstavlja veliki izazov za njihovu pohranu i procesiranje.

LHC je smješten na francusko-švicarskoj granici, u obliku prstena na kojem su postavljena četiri detektora (Slika 2.2). Dva su općenite namjene, kompaktni mionski solenoid (engl. *Compact Muon Solenoid*) ili CMS, i ATLAS (engl. *A Toroidal LHC Apparatus*), zatim detektor namijenjen za fiziku b kvarkova, LHCb (engl. *Large Hadron Collider beauty*) i detektor za fiziku teških iona, ALICE (engl. *A Large Ion Collider Experiment*). Već 2012. godine eksperiment je doveo do revolucionarnih saznanja. Preciznije, kao rezultat suradnje na dvama detektorima, ATLAS i CMS, dolazi do otkrića Higgsovog bozona.

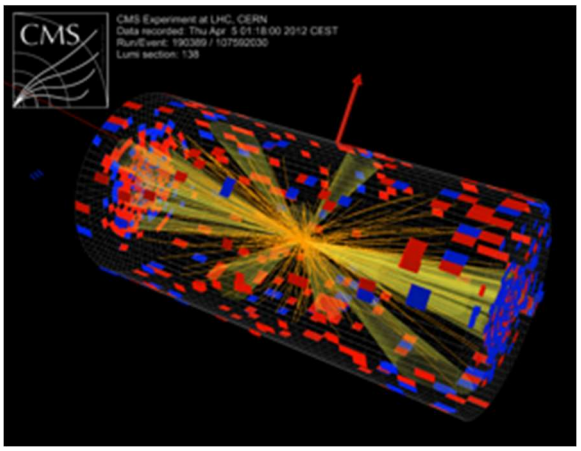


Slika 2.2. Shematski prikaz kompleksa akceleratora u CERN-u [4]

Unutar cijevi akceleratora, grupe (engl. *Bunch*) vrlo zbijenih protona ubrzavaju se u suprotnim smjerovima (skoro do brzine svjetlosti). Protoni su pri tom «pakirani» u zrake koje su stisnute što je više moguće da bi se vjerojatnost sudara povećala. Što je zraka uža i što sadrži više protona luminozitet (engl. *Luminosity*) je veći. Luminozitet predstavlja stopu sudara, a prvoj fazi rada planirani luminozitet bio je $1 \times 10^{34} cm^{-2} s^{-1}$ premda je korišten nešto niži ($0.7 \times 10^{34} cm^{-2} s^{-1}$). Do kraja Run II faze koristio se dvostruko veći luminozitet tj. $2 \times 10^{34} cm^{-2} s^{-1}$. Zraka sadrži slijed od približno 2800 grupa protona, a unutar svake grupe se nalazi otprilike 1.15×10^{11} protona. U središtu detektora, grupe protona iz suprotnih smjerova se presjecaju (engl. *Bunch crossing - BX*) što za posljedicu ima više od jednog proton-proton sudara. Pri tom su zanimljivi sudari visokih energija. Međutim, pojavljuju se dodatne kolizije koje generiraju dodatne čestice, a nisu od direktnog interesa. Taj efekt se naziva gomilanje (engl. *Pile up - PU*). U Run II fazi, broj dodatnih kolizija doseže otprilike 200, u odnosu na prethodnih 50 u Run I fazi.

Kao rezultat sudara nastaju nove čestice koje su vrlo nestabilne te se raspadaju u druge čestice. Točke sudara okružene su slojevima detektora (o čemu će biti riječi u idućem poglavlju) te proizvodi raspada prolazeći kroz njih s njima stupaju u interakciju. Detektori bilježe/registriraju prolaze svake čestice, dok sustav za obradu kodira putanje i energije čestica u svrhu kreiranja slike "događaja sudara". Mjerenjem triju svojstava

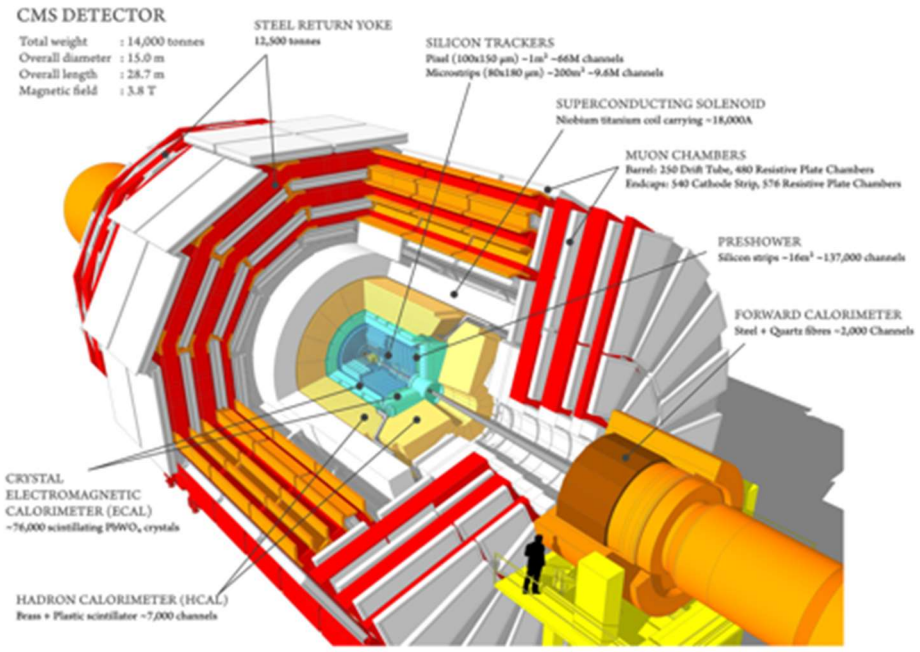
čestica raspada (vrste čestice, njene energije te putanje) dolazi se do saznanja o svojstvima raspadnute početne čestice.
Posljedica efekta gomilanja je da konačno stanje sudara, zabilježeno u detektorima, nisu samo proizvodi raspada «čvrstog» sudara koji nas zanimaju već i rezultati raspada dodatnih sudara. Stoga je potrebna cjelovita analiza podataka da bi se razlikovali sudari koji proizvode čestice od interesa od onih koji proizvode od ranije poznate čestice.



Slika 2.3. Prikaz događaja proton-proton sudara (8TeV) promatran u CMS-u [5]

2.1.1. CMS

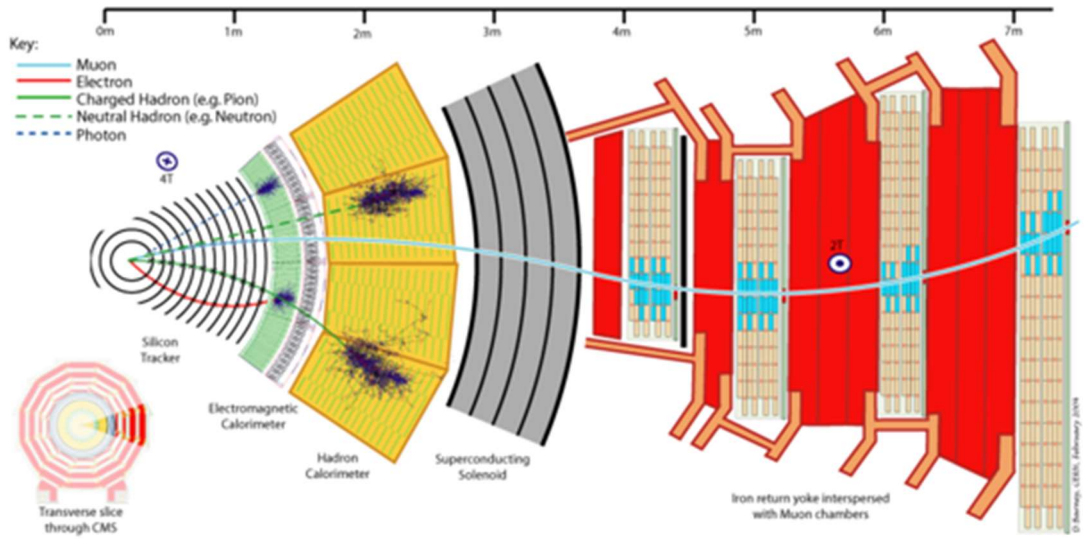
CMS detektor (slika 2.4 i 2.5), jedan je od četiri LHC detektora. To je detektor opće namjene osmišljen da promatra sve nove fizikalne pojave koje bi LHC mogao otkriti. CMS detektira produkte raspadanja različitih čestica nastalim u sudarima protona s protonom, na osnovu čega znanstvenici pokušavaju odrediti koje interakcije sudara rezultiraju kojim česticama. Sastoji se od koncentričnih dijelova oblika cilindra tj. poddetektora unutar kojih se odvija sudar protonskih zraka. Svaki od poddetektora je specijaliziran za mjerenje jednog ili više aspekata čestica koje nastaju, te se raspadaju nakon sudara. Kombinacijom informacija dobivenih iz različitih poddetektora pokušava se dobiti što je moguće potpunija slika o događaju sudara.
Kao što je prikazano na slikama 2.4 i 2.5, središnji detektori su: sustav unutarnjeg praćenja ili tragač (engl. *Inner Tracking system -Tracker*) koji je ujedno i najdublji dio CMS-a, elektromagnetski kalorimetar (engl. *Electromagnetic Calorimetar - ECAL*) te hadronski kalorimetar (engl. *Hadronic Calorimetar - HCAL*) [6,7,8].
Tragač je uređaj za određivanje (praćenje) putanja čestica dok prolaze kroz materijal. Za izradu tragača korišten je silicij koji omogućava preciznu prostornu razlučivost, međutim ne i praktično mjerenje energije. Nabijene čestice prolazom kroz Tragač deponiraju vrlo malu količinu energije. Stoga pri rekonstrukciji putanje čestica Tragač daje preciznu informaciju o položaju, dok korisne informacije o deponiranoj energiji nisu dostupne.



Slika 2.4. CMS detektor [6]

U CMS tragaču broj pogodaka raste linearno s gomilanjem. Visoka granularnost i efikasnost tragača omogućavaju pouzdano spajanje udaraca u staze/tragove. Korištenjem pogodaka zabilježenih u različitim slojevima tragača, moguće je rekonstruirati staze te dobiti njene različite parametre [7]. Nakon rekonstrukcije staze, tragovi su grupirani u točke, od kojih svaka predstavlja sudar proton-proton. Potrebno je razlikovati staze iz različitih vrhova da bi se od cijele grupe sudara mogli izdvojiti sudare koji predstavljaju "događaje od interesa".

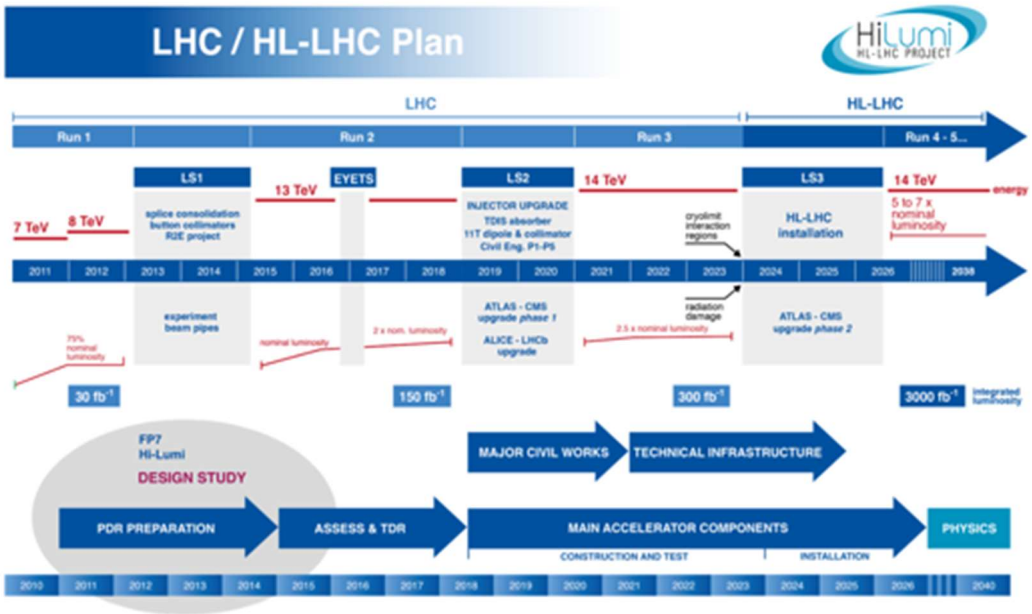
ECAL je sastavljen je od dijela koji se naziva bačva (engl. *Barrel*) i dvaju završnih dijelova koje se nazivaju poklopci (eng. End-caps). Bačva se sastoji od 61200 kristala smještenih u 36 “supermodula”, od kojih svaki teži oko 3 tone i sadrži 1700 kristala Poklopci su smješteni na krajevima bačve i sadrže dodatnih 15000 kristala. U ECAL sloju zaustavljaju se elektroni i fotoni, i mjeri se deponirana energija.
HCAL mjeri energiju koju deponiraju hadroni, čestice građene od kvarkova i gluona (na primjer, protoni, neutroni, pioni i kaoni).
HCAL i ECAL na sličan način vrše rekonstrukcije energije na temelju depozita u kalorimetru. Kako elektromagnetska (hadronska) čestica ulazi u ECAL (HCAL), proizvodi se elektromagnetski (hadronski) pljusak (engl. *Electromagnetic Shower*, *Hadronic Shower*). To se detektira kao signalni impuls koji se digitalizira i može se reći da je pohranjena "3D slika" o rastu i propadanju pljuska u svakom ECAL kristalu (odnosno HCAL tornju). Na osnovu depozita energije u senzorima kalorimetara, dobiva se energija i vrijeme povezano s tim udarcem za svaki kristal (toranj), što se naziva rekonstruirani pogodak (engl. *Hit*).
Čestice slične elektronima, ali s 200 puta većom masom zovu se mioni. Oni se ne zaustavljaju u ni jednom od spomenutih slojeva, te mogu prodrijeti i do nekoliko metara kroz materijal bez interakcije. Stoga se ne mogu detektirati u prethodnim slojevima, te su komore za njihovo otkrivanje (engl. *Muon Chambers*) smještene na samom rubu.



Slika 2.5. Odsječci CMS detektora [3]

2.1.2. Nadogradnja na HL-LHC

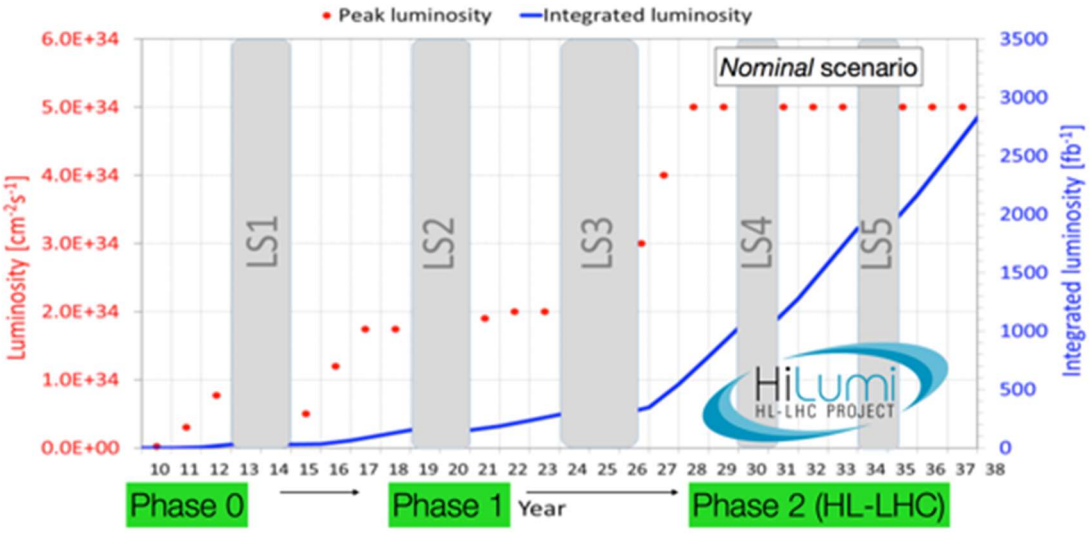
Ukupno razdoblje (planiranog) djelovanja LHC-a podijeljeno je na dvije osnovne faze: LHC i HL-LHC (engl. *High Luminosity - LHC*). One obuhvaćaju «aktivna» razdoblja (Run 1, Run 2,...), između kojih su razdoblja u kojima se vrši nadogradnja korištene tehnologije (slika 2.6). Nadogradnja detektora neophodna je zbog potrošnje materijala, oštećenja koja nastupaju kao rezultat izlaganja visokim dozama radijacije te zbog implementacije novih detektorskih tehnologija.



Slika 2.6. Planirane faze LHC projekta [9]

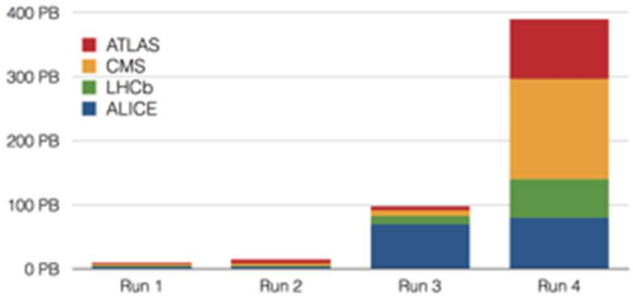
Početkom faze isključenja (engl. *Long Shutdown - LS*) LS3, 2024. godine kreće faza II LHC projekta kada se planira konstantno povećanje luminoziteta (dok se ne dosegne vrijednost od $5 \cdot 10^{34}\mathrm{cm}^{-2}\mathrm{s}^{-1}$, slika 2.7) uz povećanje preciznosti mjerenja.

Očekuje se da će u HL LHC fazi biti prikupljeno 3000 fb^{-1} tijekom razdoblja od narednih desetak godina. Za usporedbu, svi hadronski sudari u svijetu do sada su proizveli ukupni integrirani luminozitet od oko 10 fb^{-1}.



Slika 2.7. Vrijednost luminoziteta i integriranog luminoziteta u različitim fazama LHC projekta [10]

Pod navedenim uvjetima, sudari se događaju frekvencijom od 40MHz i vremenski su odvojeni 25ns, dok je očekivana vrijednost gomilanja između 140 i 200. Stoga veliki broj čestica prolaskom kroz slojeve senzora reagira s materijalom deponirajući energiju što rezultira velikom količinom podataka. Prosječna količina podataka koja se generira odgovara otprilike 70 Terabajta podataka u sekundi. Na slici 2.8 dana je procjena rasta generiranih podataka; prema slici, očekuje se da će količina podataka rasti eksponencijalno. Uz takav rast obima podataka, otežane su i njihova obrada i analiza. Ekstremno povećanje volumena podataka koje se očekuje u eksperimentima Run3 i Run4 primoralo je znanstvenike na promjene u načinu rada detektora.



Slika 2.8. Gruba procjena količine generiranih podataka po fazama [11]

Tehnologija korištena u prethodnim fazama ne može pratiti očekivano povećanje opterećenosti detektora. Stoga HL-LHC zahtijeva naprednije detektore kako bi se osigurala visoka preciznost mjerenja. Promjene u detektorima odvijaju se već u trenutnoj fazi isključenja LS2 tijekom 2019. do 2020. odnosno planirane su za razdoblje od 2023. do 2025. za vrijeme trajanja LS3 faze.

Predložena nadogradnja CMS detektora je novi poklopac na ECAL i prednji dio HCAL kalorimetra. Nova verzija kalorimetra visokog granulacijskog dizajna naziva se

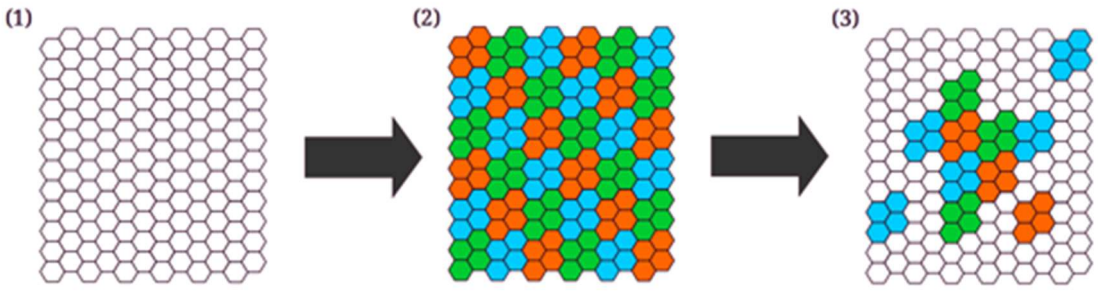
HGCAL (engl. *High Granularity Calorimeter*). Poklopac ima 40 slojeva (podijeljenih na 12 latica svaki), od čega se elektromagnetski dio sastoji od 28 slojeva pločica iz volframa i bakra isprepletenih sa silicijskim senzorima kao aktivnim materijalom, dok hadronski dio ima 12 slojeva isprepletenih sa silicijskim senzorima. Poklopac ukupno sadrži oko 6 milijuna senzora oblika šesterokuta postavljenih na navedenih 40 slojeva.

2.1.3. Selekcija podataka korištenjem sustava okidača

Obzirom na obim generiranih podataka, trenutno nije moguće niti prebaciti/pohraniti sve generirane podatke u svrhu buduće obrade. Stoga je u svaki od detektora ugrađen sustav za analizu u realnom vremenu – takozvani sustav okidača, kojim se detektiraju i pohranjuju za kasniju obradu samo događaji s potencijalno interesantnim podacima. Na taj način reducirana je količina očitanih podataka što omogućuje njihovu ekonomičniju pohranu te daljnju analizu.
Trenutno korišteni sustav okidača organiziran je na dvije razine: L1 okidač (engl. *Level-1 Trigger*) i okidač visokog nivoa (engl. *High Level Trigger - HLT*). L1 okidač donosi odluku o potencijalno interesantnom događaju (npr. čestice s velikom količinom energije). Za podatke spremljene u privremenoj memoriji okidač odabire događaje koji se spremaju, dok su ostali trajno izbrisani. Okidač razine 1 procjenjujući do 512 uvjeta mora donijeti svoju odluku u roku od 4 mikrosekunde (toliko dugo mogu biti pohranjeni podaci u detektorima, čekajući prihvaćanje signala). Proračuni potrebni za donošenje odluke se izvode na brzom hardveru koji se naziva (engl. *Front-End*) detektorska elektronika. L1 okidač očitava cijeli CMS detektor i pri tom stopu očitavanja BX smanjuje s 40 MHz na 100 kHz.
Interesantni podaci propušteni od strane okidača niskog nivoa prosljeđuju se okidačima višeg nivoa, implementiranim na velikim računalnim farmama. Tu se prikupljaju i sinhroniziraju podaci iz svih dijelova detektora provodeći daljnju redukciju sa 100 na 1kHz.

Kako je već navedeno, U HL-LHC fazi, porastom luminoziteta, sudari se događaju frekvencijom od 40MHz, s vremenskim razmacima od 25ns i vrijednošću gomilanja između 140 i 200. Kao rezultat tolikog broja sudara, oslobađa se velika količina energije, što u konačnici generira veliku količinu podataka.

Iako još uvijek nije poznata detaljna arhitektura sustava okidača HG kalorimetra, prvi dio obrade podataka izvest će se na ASIC-u u realnom vremenu, kako bi se smanjila količina podataka koji se šalju na daljnju obradu. Taj drugi dio obrade odnosi se na algoritme rekonstrukcije, poput grupiranja energije i procjene nagomilavanja, a obavit će se u FPGA-ima. Ova obrada dijelit će se između nekoliko slojeva povezanih zajedno brzim optičkim vezama. Rezultat će biti proizvodnja HGCal primitiva (engl. *Trigger Primitive - TP*), koji će se zajedno s primitivama drugih poddetektora koristiti za rekonstrukciju objekata više razine poput elektrona, fotona i drugih. Da bi okidačke podatke bilo moguće prenijeti dostupnim optičkim kanalima zadanog kapaciteta, potreban je faktor smanjenja veličine podataka od oko 20 puta [12]. To se postiže na idući način: senzorske ćelije grupirane su u okidačke ćelije (engl. *Trigger Cells*); samo one okidačke ćelije s najvećim energijama se biraju i šalju na daljnju analizu, kao što je pokazano na slici 2.9.



Slika 2.9. Redukcija podataka koja se provodi u on-line ASIC obradi. HGCal senzorske ćelije (1) grupirane su u okidačke ćelije (2). Odabiru se i prenose samo one s najvećim energijama (3). [12]

Back-end elektronika dekodira dolazne okidačke podatke te provodi daljnju sofisticiranu redukciju korištenjem trodimenzionalnog klasteriranja energija. Algoritam za izvođenje još nije odabran, i to je između ostalog bila motivacija za pisanje rada.

Pored opisanih sustava koji odlučuju u stvarnom vremenu koji podskup podataka detektor treba očitati i arhivirati za offline analizu, DAQ sustav (engl. *Data AcQuisition*) prikuplja podatke iz različitih dijelova detektora, pretvara ih odgovarajući format te trajno pohranjuje.
Uz upotrebu sustava okidača, pri maksimalnom postotku odbačenih događaja (filtriranjem njih 99%), CERN-ov sustav za naprednu pohranu podataka (engl. *CERN Advanced Storage system - CASTOR*) namijenjen trajnom arhiviranju podataka u 2018. godini dosegnuo je 330 PB pohranjenih podataka [3]. Obim i složenost prikupljenih podataka premašuje računalne i financijske resurse CERN-a te je stoga organiziran WLCF (engl. *The Worldwide LHC Computing Grid*), globalna računalna infrastruktura koja pruža računalne resurse za pohranu, distribuciju i analizu podataka generiranih LHC-om.

2.2. Redukcija dimenzionalnosti

Podatke očitane sa senzora/dobivene u eksperimentima je potrebno pohraniti i analizirati, a zbog njihove količine korisno je imati automatizirane metode analize. Uz pomoć njih potrebno je utvrditi važne veze/uzorke među podacima, ustanoviti koje značajke imaju najveći utjecaj, te reducirati veličinu i dimenzionalnost podataka.

2.2.1. Prokletstvo dimenzionalnosti

Prokletstvo dimenzionalnosti (engl. *The course of dimensionality*) odnosi se na razne fenomene koji se javljaju kod pohrane i analize podataka u visoko-dimenzionalnim prostorima što kod manjih dimenzija nije slučaj. Izraz je prvi put upotrebio R.E. Bellman 1961. pri rješavanju problema iz dinamičke optimizacije. On je primijetio sljedeće: mreža jedinične kocke s proredom 1/10 u D dimenzija sadrži 10^D uzoraka što raste eksponecijalno povećanjem dimenzije prostora D. Porastom dimenzionalnosti podaci su rijeđe raspoređeni. Kao poslijedica, moguće je da je udaljenost do najbližeg susjeda slična udaljenosti do najdaljeg. Izgubljen je kontrast u udaljenostima što predstavlja problem u metodama koje zahtijevaju statistički značaj [13,14]. Statističke metode, isto kao i računalne metode namijenjene interpretaciji podataka suočavaju se s brojnim problemima pri manipulaciji s visoko dimenzionalnim podacima. Stoga se

pokazalo neophodnim smanjiti broj ulaznih varijabli da bi neki algoritam za rudarenje podataka mogao biti uspješno primijenjen. Redukciju podataka moguće je primijeniti zbog činjenice da izvorni podaci često sadrže i nepotrebne podatke: ukoliko varijable imaju varijaciju manju od izmjerenog šuma, nebitne su; također, među nekim varijablama postoji korelacija te je moguće/potrebno pronaći novi skup nepovezanih varijabli [15].

2.2.2. Metode redukcije dimenzionalnosti

U računalu su podatkovni objekti pohranjeni kao skup obilježja (npr. vrijeme, koordinate, …). Objekt s *n* obilježja može se promatrati kao element *n*-dimenzionalnog prostora. Redukcija dimenzionalnosti početni *n*-dimenzionalni prostor mapira u niži *k*-dimenzionalni prostor. Kako je ovom operacijom smanjena veličina memorije potrebna za pohranu podataka, također se može smatrati metodom kompresije. Još jedna njena prednost je da se mapiranjem na dvije ili tri dimenzije omogućuje vizualizacija podataka.
Jedna od najraširenijih metoda redukcije dimenzionalnosti - Metoda glavnih komponenti, datira još iz 1901. Osnovna ideja bila je pronaći novi koordinatni sustav u kojem podaci mogu biti izraženi korištenjem manjeg broja varijabli bez neke značajne pogreške. Obzirom na eksplozivni rast generiranih i dostupnih podataka posljednjih godina, uz mogućnost korištenja moćnih računalnih resursa, znanstvenici su predstavili široki raspon metoda koje se bave redukcijom dimenzionalnosti.

Postoje dva osnovna pristupa redukciji dimenzionalnosti:

1. selekcija obilježja: odabir podskupa (više-informativnih) varijabli. U ovom slučaju obilježja koja nisu u korelaciji s izlazom mogu biti u potpunosti izostavljena.

2. transformacija obilježja: kombiniranjem postojećih varijabli vrši se ekstrakcija manjeg broja novih dimenzija-varijabli. Na taj način originalni podaci velike dimenzionalnosti preslikavaju se u novi prostor manje dimenzionalnosti. U novom prostoru svaka značajka je linearna ili nelinearna kombinacija značajki originalnog prostora. Cilj je u novom prostoru zadržati udaljenosti među podatkovnim vektorima. Razvijene su brojne metode: metoda glavnih komponenti, nasumična projekcija, kanonska korelacijska analiza i brojne druge [16].
   Tenzori (poglavlje 4) omogućavaju prirodan način pohrane i prikaza masivnih višedimenzionalnih podataka za čiju se analizu razvijaju nove tehnologije koje bi omogućile efikasnu obradu u što kraćem vremenu. Dekompozicija tenzora na skupove komponentnih matrica i tenzore nižeg reda omogućava otkrivanje skrivenih struktura unutar podataka. Stoga je u ovom radu naglasak stavljen na metode redukcije dimenzionalnosti bazirane na matričnoj (poglavlje 3), odnosno tenzorskoj dekompoziciji (poglavlje 4.4).

Bitno svojstvo bilo koje metode za redukciju dimenzionalnosti je njena stabilnost. Za neku metodu vrijedi da je stabilna ako je za dvije proizvoljne točke (vektora) x_1, x_2 istinita iduća tvrdnja:

$$(1-\varepsilon)\|x_1 - x_2\|_2^2 \leq \|\chi_1 - \chi_2\|_2^2 \leq (1+\varepsilon)\|x_1 - x_2\|_2^2 \quad \forall \varepsilon \epsilon \langle 0,1 \rangle \qquad (2.1)$$

(transformacijski operator T ulazni vektor transformira u vektor značajki $\chi = T(x)$). Interpretacija tvrdnje je da je Euklidska udaljenost u izvornom prostoru relativno sačuvana u izlaznom (transformiranom) prostoru [17].

3. METODE MATRIČNE DEKOMPOZICIJE

Neka su podaci kojima se raspolaže dobiveni kao rezultat *n* razmatranja pri čemu je broj *n* jako velik. Promatrano je *p* različitih obilježja, te su rezultati pohranjeni u matricu tipa *n*×*p*, pri čemu je moguće da su podaci oštećeni šumom. Potrebno je utvrditi vezu među podacima i otkriti eventualne uzorke (engl. *Pattern*) "skrivene" među podacima. Upotrebom matrične faktorizacije reducira se dimenzionalnost. Matrica podataka se projicira u prostor manje dimenzije, gdje "nered" iz viših dimenzija poprima strukturu u kojoj se može utvrditi pravilnost među podacima.

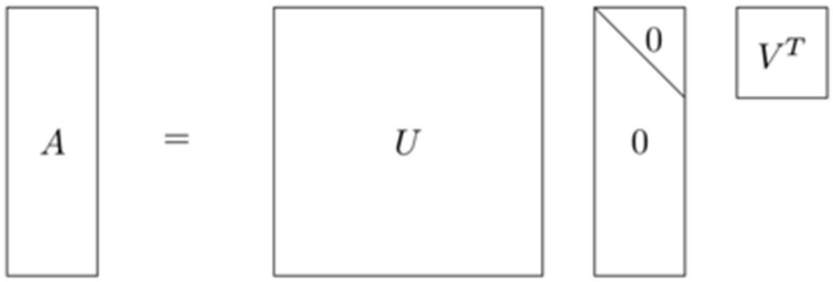
Većina metoda dekompozicije temelji se (ili se u nekom trenutku oslanja) na Dekompoziciju singularnih vrijednosti (engl. *Singular Value Decomposition - SVD*), koja predstavlja jedan od najvažnijih teorema numeričke analize. Stoga pregled metoda počinje upravo detaljnim opisom metode. SVD rastav daje uvid u matematičku strukturu matrice, no činjenica da omogućava aproksimaciju nižeg ranga je razlog za njenu široku primjenu [18,19].

3.1. Dekompozicija singularnih vrijednosti

Ako je $A \in R^{m\times n}$ realna matrica ranga r, onda postoje ortogonalne matrice $U \in R^{m\times m}$ i $V \in R^{n\times n}$ te dijagonalna matrica $\Sigma \in R^{m\times n}$, $\Sigma = diag(\sigma_1, \sigma_2, \dots, \sigma_n)$ gdje je $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$ i $\sigma_i = 0$ za $i \geq r+1$ takve da je

$$A = U\Sigma V^T \tag{3.1}$$

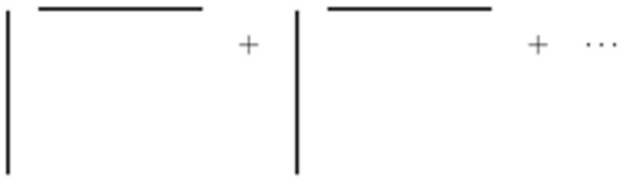
Shematski prikaz dekompozicije dan je na slici 3.1.



Slika 3.1. SVD [19]

Realni brojevi $\sigma_1, \sigma_2, \dots, \sigma_s$ zovu se singularne vrijednosti matrice A, a to su nenegativni kvadratni korijeni *n* svojstvenih vrijednosti matrice AA^T. Matrica Σ naziva se singularnom matricom, $u_1, u_2, \dots, u_m$ lijevim singularnim vektorima, a $v_1, v_2, \dots, v_n$ desnim singularnim vektorima. Ako se U zapiše kao $U = (U_m U_m^{\perp})$, gdje je $U_m \in R^{n\times m}$, a Σ kao $\begin{pmatrix}\Sigma_m \\ 0\end{pmatrix}$ gdje je $\Sigma_m \in R^{m\times n}$, tada se dobije tanki SVD (Slika 3.2):

$$A = U_m \Sigma_m V^T = \sum_{i=1}^{m} u_i \sigma_i v_i^T \tag{3.2}$$

$$A = \sum_{i=1}^{m} \sigma_i u_i v_i^T =$$  $$+ \quad + \quad \cdots$$

Slika 3.2. Tanki SVD [19]

Prvih *r* stupaca ortogonalnih matrica U i V definira ortonormalne svojstvene vektore vezane sa *r* svojstvenih vrijednosti različitih od nule matrica AA^Ti A^TA.

Neka je SVD matrice A definirana jednadžbom $A = U\Sigma V^T$ i neka je $r = \text{rang}(A) \leq p = \min(m, n)$.Tada je $A_k = \sum_{i=1}^{k} u_i \, \sigma_i v_i^T$ za k<r i vrijedi

$$\min_{r(B)=k} \|A - B\|_F^2 = \|A - A_k\|_F^2 = \sigma_{k+1}^2 + \cdots \sigma_p^2 \tag{3.3}$$

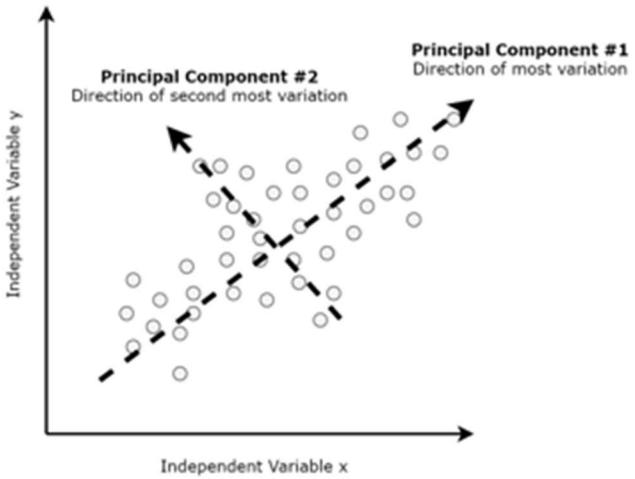
Ova važna tvrdnja pokazuje da je A_k najbolja aproksimacija ranga *k* matrice A (u smislu najmanje kvadratne pogreške), te služi kao osnova za redukciju dimenzionalnosti. Svojstvo dekompozicije $A = \sum_{i=1}^{r} u_i \, \sigma_i v_i^T$ često se koristi kao osnova za redukciju i kompresiju podataka.

Poznata je upotreba SVD-a kod sažimanja slike [20, 21]. Dodavanje vodenog žiga (engl. *Watermarking*) u svrhu zaštite od manipulacije i distribucije korištenjem SVD-a moguće je kako za slike [22], tako i za video [23].

3.2. Metoda glavnih komponenti

Jedna od najstarijih i najviše korištenih metoda za redukciju dimenzionalnosti podataka je Metoda glavnih komponenti (engl. *Principal component analysis - PCA*) [24]. PCA je tehnika multivarijatne analize koju je 1901. prvi put predstavio Pearson. Odredio je pravac (danas poznat kao prva glavna komponenta) koji najbolje aproksimira (u smislu najmanje kvadratne pogreške) skup točaka koje prezentiraju podatke. Neovisno o njemu, metodu je razvijao i Hotelling te ju je predstavio 1933. On je utvrdio da su komponente svojstveni vektori matrice kovarijanci uzoraka. Kao znanstveni alat, PCA se prvi put javlja u časopisu Psychometrica a također je poznata pod nazivima Hotelling transformacija (engl. *Hotelling transform*), Metoda empirijske ortogonalne funkcije (engl. *Empirical Orthogonal Function*) te Karhunen-Loeve transformacija (engl. *Karhunen-Loève Transform*).

PCA se bazira na matematičkom postupku koji pretvara niz (eventualno) koreliranih varijabli u (manji) broj nekoreliranih varijabli zvanih glavne komponente. Prva glavna komponenta je linearna kombinacija s najvećom varijancom. Druga glavna komponenta je linearna kombinacija sa drugom najvećom varijancom, ortogonalna na prvu (Slika 3.3), itd.
Glavnih komponenti ima koliko i početnih varijabli. Obzirom da prvih nekoliko glavnih komponenti mogu pojasniti većinu varijance, ostatak se može zanemariti s minimalnim gubitkom informacija.



Slika 3.3. Prva i druga glavna komponenta dvodimenzionalnog skupa podataka [25]

Ukoliko se svaka n-dimenzionalna točka $\boldsymbol{x_\tau} \in R^n$ predstavi koristeći svih n glavnih komponenti, onda je pogreška $\|x_\tau - \tilde{x}_\tau\| = 0$. Međutim, vrlo mala pogreška postiže se već kod uporabe samo k glavnih komponenti, pri čemu je $k \ll n$.
Glavni smjerovi/osi su svojstveni vektori umnoška $X_t^T X_t$, koji se najlakše dobiju uporabom SVD matrice X_t.

Klasična metoda PCA ima dva bitna nedostatka:

1. podrazumijeva da je veza među varijablama linearna,

2. primjenjiva je samo za numeričke vrijednosti varijabli.

Stoga su razvijene razne metode za prevladavanje ovih ograničenja. PCA se naširoko koristi u obradi slike: za kompresiju [26,27,28]; za određivanje orijentacije objekta [27], te za prepoznavanje slike [29]. Jedan od najpoznatijih primjera upotrebe je sustav za prepoznavanje lica Eigenfaces [30] koji funkcionira na način da projicira slike lica na prostor značajki. Bitne značajke nazivaju se "eigenfaces" zbog činjenice da su one svojstveni vektori (engl. *Eigenvectors*) (osnovne komponente) skupa lica.

U [31] autori su za predstavljanje slika predstavili novu verziju PCA pod nazivom dvo-dimenzionalni PCA - 2DPCA. Bazirana je na matrici slike, matrica kovarianci je konstruirana direktno iz ulazne matrice i svojstveni vektori su izračunati s ciljem izdvajanja značajki. U prepoznavanju lica 2DPCA je pokazao znatne bolje rezultate od standardnog PCA.

Metodu analize nelinearnih glavnih komponenti (engl. *Nonlinear Principal Component Analysis - NLPCA*) moguće je koristiti u slučaju postojanja nelinearnih veza. Za njeno izvođenje potrebna su dva koraka: 1. izabrati razinu analize: definirati prirodu varijabli 2. izvršiti kvantifikaciju: mapirati originalne vrijednosti u odgovarajuće numeričke. Nakon toga se na tako kvantificirane varijable primjenjuje klasični PCA.

3.3. Analiza nezavisnih komponenti

Metoda analize nezavisnih komponenti (engl. *Independent Component Analysis - ICA*) je metoda slična PCA koju su predstavili J. Hérault, C. Jutten i B. Ans osamdesetih godina prošlog stoljeća [32]. Kako je u radu dan samo algoritam, bez teoretskog objašnjenja, te zbog činjenice da je algoritam imao brojna ograničenja, tehnika ICA je ostala nepoznata do 1994. godine kada je predstavljena pod tim nazivom kao novi koncept [33]. ICA se najviše koristi u BSS (engl. *Blind Source Separation - BSS*), području koje se bavi razdvajanjem miješanih signala. Cilj ICA je izvući korisne informacije ili izvorne signale iz podataka (skup izmjerenih signala smjese) i u tu svrhu je korišten u aplikacijama kao što su medicinski signali, biološki testovi, audio signali i mnogim drugim [34]. Međutim, ICA se također smatra algoritmom za smanjenje dimenzionalnosti u slučaju kada ICA može izbrisati ili zadržati pojedini izvor. ICA se smatra proširenjem metode PCA, međutim ona je statistička metoda opće namjene koja linearno transformira originalne podatke na komponente koje su u statističkom smislu maksimalno nezavisne jedna od druge, dok su kod PCA komponente nekorelirane. Za razliku od PCA, ortogonalnost komponenti nije uvjet.

Model prikazan jednadžbom $X = AS$ naziva se ICA model i on je polazište ICA metode. Premda je konačni cilj utvrditi vrijednost nezavisnih komponenti, glavni zadatak te metode je pronalaženje matrice miješanja (engl. *Mixing Matrix*) uz poznavanje mješavina izvornih signala $X(t)$, odnosno odrediti matricu $W = A^{-1}$. ICA model nije uvijek primjenjiv i iduće pretpostavke su obavezne [34]:

1. statistička nezavisnost izvora signala s_i, (dvije varijable y_1 i y_2 su statistički nezavisne ako poznavanje vrijednosti slučajne varijable y_1 ne daje nikakvu informaciju o slučajnoj varijabli y_2, matematičkim zapisom za njihove funkcije gustoće vrijedi $f(y_1, y_2) = f(y_1) \cdot f(y_2)$).

2. međusobno nezavisni izvori nemaju Gaussovu raspodjelu, jer u suprotnom slučaju ICA metoda neće biti primjenjiva

3. matrica miješanja mora biti invertibilna.

Ukoliko su zadovoljeni navedeni uvjeti, matrica miješanja i komponente mogu biti utvrđeni, što je pokazao Comon 1994 [33].

Metode Analiza Nezavisnih Komponenti - ICA, Nenegativna matrična faktorizacija - NMF i Analiza rijetkih komponenti - SPA baziraju se na problemu linearne reprezentacije ili matrične faktorizacije skupa podataka X. Razlika je u dodatnim zahtjevima:

- ukoliko su reci matrice S statistički nezavisni – radi se o analizi nezavisnih komponenti
- ukoliko su elementi matrica X, A i S nenegativni – radi se o nenegativnoj matričnoj faktorizaciji
- ukoliko S sadrži što je više moguće nula – radi se o problemu rijetke reprezentacije

3.4. Nenegativna matrična faktorizacija

Nenegativna matrična faktorizacija (engl. *Nonnegative Matrix Factorization - NMF*) je često korišten alat u analizi visokodimenzionalnih podataka koji iz skupa nenegativnih podatkovnih vektora automatski prikuplja rijetke i bitne značajke. NMF su prvi predstavili Paatero i Tapper pod nazivom pozitivna matrična faktorizacija. Međutim metoda je tek radom autora Lee i Seung-a postala lakše razumljiva i opće prihvaćena [35].

Osnovna metoda za redukciju dimenzionalnosti, PCA, računa skup vektora baze koji će optimalno (u smislu metode najmanjih kvadrata) aproksimirati visoko-dimenzionalne podatke. Kako je broj vektora baze mnogo manji od broja dimenzije, prikaz podataka kao linearne kombinacije vektora baze transformira ih u prostor manje dimenzije. Problem se javlja zbog činjenice da vektori baze mogu imati i pozitivne i negativne komponente, pa su podaci prikazani kao linearna kombinacija ovih vektora s pozitivnim i negativnim koeficijentima. Međutim, u mnogim aplikacijama, negativne komponente su u kontradikciji s fizičkom stvarnošću. U cilju rješavanja navedenog problema, znanstvenici su predložili da vektori baze trebaju biti ograničeni na nenegativne. Tako se NMF bazira na određivanju što bolje aproksimacije nenegativne matrice A u obliku produkta dvaju nenegativnih matrica W i H čiji je rang *k* maksimalno jednak rangu početne matrice. Odabir vrijednosti *k* uvelike utječe na rezultat. Pregled algoritama za implementaciju dan je u [36].

3.5. Analiza rijetkih komponenti

Analiza rijetkih komponenti (engl. *Sparse Component Analysis - SCA*) kao i prethodne dvije metode bavi se modelom $X = AS$, pri čemu je matrica S rijetka matrica.

Osnovne pretpostavke metode [37]: 1. Svi stupci izvorne matrice su nenegativni; 2. Izvorna matrica ima puni redčani rank; 3. Matrica miješanja ima puni stupčani rang i $m \geq n$; 4. Retci matrice izvora i stupci matrice miješanja imaju jediničnu normu; 5. Matrica izvora je rijetka.

Različiti scenariji SCA problema dolaze s različitim rijetkim strukturama. Tipične rijetke strukture raspravljane u SCA literature mogu se podijeliti na:

- lokalno dominantni slučaj: kao dodatak osnovnim pretpostavkama, za zadani red *r* od S, postoji barem jedan jedinstveni stupac *c* takav da vrijedi:

$$s_{i,c}\begin{cases} > 0 \; ako \; je \; i = r \\ = 0 \qquad inače \end{cases}$$

- lokalno latentni slučaj: kao dodatak osnovnim pretpostavkama, za zadani red r od S, postoji barem (n-1) linearno nezavisnih i jedinstvenih stupaca C_r takvih da vrijedi:

$$s_{i,c}\begin{cases} = 0 \;\; ako \; je \; i = r \;\&\; c \in C_r \\ > 0 \;\; ako \; je \; i \neq r \;\&\; c \in C_r \end{cases}$$

- općeniti slučaj, bez ikakvih dodatnih uvjeta

SCA se koristi u rudarenju podataka, medicinskoj dijagnostici [38], BSS.

3.6. Usporedba

Prema [39, 40]:

NMF je odličan izbor za modeliranje nenegativnih podataka kao što su slike, npr. kod prepoznavanja lica daje bolje rezultate u usporedbi s ICA i PCA. Isti rezultati dobiveni su u [41] kod pronalaženja implicitnih struktura skrivenih u web poveznicama gdje se NMF pokazala superiornom u smislu stabilnosti i interpretabilnosti otkrivenih struktura. U slučaju kada podaci nisu dani Gaussovom raspodjelom, ICA daje dobre rezultate u traženju nezavisnih izvora. Premda su se kod prepoznavanja lica metode bazirane na ICA pokazale efikasnijim od onih na PCA [42], PCA ima mnogo širu primjenu; idealan je za utvrđivanje uzoraka i redukciju dimenzionalnosti.

Premda se matrična dekompozicija pokazalo bitnim i korisnim alatom, njen bitni nedostatak je nejedinstvenost. Npr. ukoliko se razmotri dekompozicija ranga matrice M

$$M = AB^T, M \in R^{n\times m}, A \in R^{n\times r}, B^T \in R^{r\times m} \tag{3.4}$$

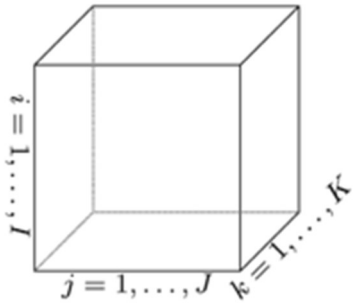
dodavanjem rotacijske matrice i njenog inverza, jednostavno je pokazati da taj rastav nije jedinstven (tkz. rotacijski problem).

$$\widehat{M} = AB^T = ARR^{-1}B^T = (AR)(R^{-1}B^T) = (AR)(BR^{-T})^T = \hat{A}\hat{B}^T \tag{3.5}$$

Jedinstvenost može biti postignuta zadavanjem dodatnih uvjeta, kao što je ortogonalnost kod SVD, dok su tenzorske dekompozicije jedinstvene pod mnogo blažim uvjetima [43].

4. MULTILINEARNA ALGEBRA – ALGEBRA TENZORA

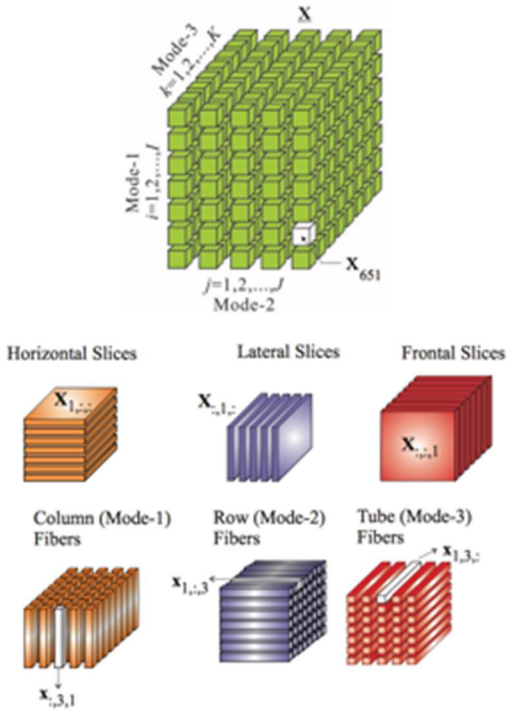
Tenzor je višedimenzionalno polje. Formalnija definicija glasi: tenzor *N*-tog reda je element tenzorskog produkta od *N* vektorskih prostora, od kojih svaki posjeduje vlastiti koordinatni sustav. Neformalno, tenzor se može smatrati višedimenzionalnom kockom (najčešće) realnih podataka u kojoj svaka dimenzija odgovara drugom obilježju ili mjerenju (Slika 4.1).



Slika 4.1. Tenzor 3. reda $\boldsymbol{\mathcal{X}}^{I \times J \times K}$ *[44]*

Neki standardi pri označavanju: vektori (tenzori reda jedan) se označavaju podebljanim malim slovima ili italic velikim npr. **a** ili *A*. Matrice (tenzori reda dva) se označavaju podebljanim velikim slovima npr. **A**. Tenzori višeg reda se označavaju podebljanim Euler script slovima npr. $\boldsymbol{\mathcal{A}}$. Skalari se označavaju malim slovima npr. a. Element tenzora $\boldsymbol{\mathcal{A}} \in \mathrm{R}^{\mathrm{I}_1 \times \mathrm{I}_2 \times \ldots \times \mathrm{I}_\mathrm{N}}$ označava se s $\boldsymbol{\mathcal{A}}_{\mathrm{i}_1 \ldots \mathrm{i}_\mathrm{n} \ldots \mathrm{i}_\mathrm{N}}$ odnosno $\mathrm{a}_{\mathrm{i}_1 \ldots \mathrm{i}_\mathrm{n} \ldots \mathrm{i}_\mathrm{N}}$. Za vektor **v** njegov element se označava s v_i, gdje je i = 1, ..., m dok je a_{ij} element matrice **A**. Red tenzora označava broj njegovih modova ili dimenzija (što može uključivati prostor, vrijeme, frekvenciju, pokuse, klase, rječnike). Može se reći i da je red tenzora broj indeksa potrebnih za indeksiranje njegovih elemenata. Pojam "veličina" označava koliki je broj vrijednosti koje neki indeks može poprimiti.

Manipuliranje tenzorima često zahtjeva njihovo reformatiranje, a najjednostavniji način je fiksiranje (nekog broja) indeksa, dok se upotrebom dvotočke označavaju svi elementi pojedinog moda, tj. smjera. Na taj način nastaju podtenzori, što su za matrice bili retci i stupci. Od posebnog su interesa tenzorska vlakna (engl. *Fiber*), jednodimenzionalni fragmenti tenzora dobiveni fiksiranjem svih indeksa osim jednoga. Tenzorska vlakna su analogani višeg reda matričnih redaka i stupaca. Matrični stupac je analogan tenzorskog vlakna moda 1, a matrični redak je tenzorsko vlakno moda 2.
Tenzori trećeg reda imaju stupčana, retčana i poprečna tenzorska vlakna. Označavaju se redom $\mathbf{x}_{:\mathrm{jk}}$, $\mathbf{x}_{\mathrm{i:k}}$, $\mathbf{x}_{\mathrm{ij:}}$ (slika 4.2).
Tenzorski odsječci (engl. *Slice*) su dvodimenzionalni fragmenti tenzora, definirani fiksiranjem svih osim dvaju indeksa tenzora.
Tenzori trećeg reda imaju vodoravne, bočne i frontalne odsječke označene s $\mathbf{X}_{\mathrm{i::}}$, $\mathbf{X}_{:\mathrm{j}:}$, $\mathbf{X}_{::\mathrm{k}}$ (slika 4.2). *k*-ti frontalni odsječak $\boldsymbol{X}_{::\mathrm{k}}$ tenzora trećeg reda alternativno je moguće označiti s $\boldsymbol{X}_k$.



Slika 4.2. Vlakna i odsječci tenzora 3.reda [18]

Za tenzor se kaže da je kockast ako su svi njegovi modovi iste dimenzije, tj. $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I \times \ldots \times I}$. Tenzor je supersimetričan ako njegovi elementi ostaju isti pri bilo kojoj permutaciji njegovih indeksa. Tenzor može bit i (parcijalno) simetričan u 2 ili više modova. Npr. tenzor trećeg reda je simetričan u modovima 1 i 2 ako su svi njegovi frontalni odresci simetrični.

Tenzor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times \ldots \times I_n}$ je dijagonalan ako vrijedi da je $x_{i_1 \ldots i_N} \neq 0$ samo za $i_1 = \cdots = i_N$.

Tenzorska norma analogan je matrične Frobeniusove norme i za tenzor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times \ldots \times I_n}$ se definira kao korijen sume kvadrata svih njegovih elemenata, odnosno

$$\| \boldsymbol{\mathcal{X}} \| = \sqrt{\sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \ldots \cdot \sum_{i_N=1}^{I_N} x_{i_1 i_2 \ldots i_N}^2} \tag{4.1}$$

U radu s tenzorima koriste se razne vrste produkata tenzora, matrica i vektora.
Unutarnji (ili skalarni) produkt dvaju tenzora istog reda $\mathcal{X}, \mathcal{Y} \in \mathbb{R}^{I_1 \times \ldots \times I_n}$ definira se kao

$$\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \ldots \cdot \sum_{i_N=1}^{I_N} x_{i_1 i_2 \ldots i_N} \, y_{i_1 i_2 \ldots i_N} \tag{4.2}$$

Za dva tenzora čiji je skalarni produkt jednak 0 se kaže da su međusobno ortogonalni.

Vanjski produkt dvaju vektora $\boldsymbol{x} = (x_1, x_2, \ldots, x_m)$ i $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)$ definira se kao matrica $\mathbf{A}$ tipa $m \times n$ čiji su elementi dobiveni množenjem svakog elementa vektora $\boldsymbol{x}$ sa svakim elementom vektora $\boldsymbol{y}$:

$$\boldsymbol{x} \circ \boldsymbol{y} = \boldsymbol{A} = \begin{bmatrix} x_1y_1 & x_1y_2 & \cdots & x_1y_n \\ x_2y_1 & x_2y_2 & \cdots & x_2y_n \\ \vdots & \vdots & \ddots & \vdots \\ x_my_1 & x_my_2 & \cdots & x_my_n \end{bmatrix} \tag{4.3}$$

odnosno u indeksnom zapisu $(x \circ y)_{ij} = x_i y_j$. Vanjski produkt dvaju vektora jednak je matričnom produktu xy^T.

Rezultat vanjskog produkta N vektora $\boldsymbol{a}^{(n)} \in \mathbb{R}^{I_n}, n \in \{1, \ldots, N\}$ je tenzor

$$\underline{\mathbf{Z}} = \boldsymbol{a}^{(1)} \circ \boldsymbol{a}^{(2)} \circ \ldots \circ \boldsymbol{a}^{(N)} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_n} \tag{4.4}$$

s elementima

$$z_{i_1 i_2 \ldots i_N} = a_{i_1}^{(1)} \cdot a_{i_2}^{(2)} \cdot \ldots \cdot a_{i_N}^{(N)} \tag{4.5}$$

Rezultat vanjskog produkta dvaju tenzora (različitog reda) $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times \ldots \times I_n}$, $\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{J_1 \times \ldots \times J_m}$ je tenzor

$$\underline{\mathbf{Z}} = \underline{\boldsymbol{\mathcal{X}}} \circ \underline{\boldsymbol{\mathcal{Y}}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times J_N \times J_1 \times J_2 \times \ldots \times J_M} \tag{4.6}$$

čiji su elementi dani s

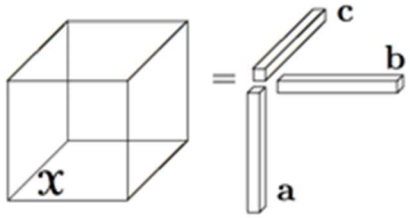
$$z_{i_1 i_2 \ldots i_N j_1 j_2 \ldots j_M} = x_{i_1 i_2 \ldots i_N} \cdot y_{j_1 j_2 \ldots j_M} \tag{4.7}$$

Rezultirajući tenzor sadrži sve moguće kombinacije produkata dvaju ulaznih tenzora.

Za tenzor koji se može prikazati kao vanjski produkt vektora se kaže da je tenzor ranga jedan:

$$\boldsymbol{\mathcal{X}} = \mathbf{a}^{(1)} \circ \mathbf{a}^{(2)} \circ \ldots \circ \mathbf{a}^{(N)} \tag{4.8}$$

Vizualizacija tenzora reda 3 ranga 1 dana je na slici 4.3.
Pojam ranga tenzora razlikuje se od ranga matrice. Rang matrice je broj njenih linearno nezavisnih stupaca (ili redaka), odnosno dimenzija slike ako se matrica promatra kao linearni operator između vektorskih prostora. Rang tenzora definira se kao najmanji broj tenzora ranga jedan čija je suma jednaka početnom tenzoru (ovo je jedna od definicija ranga tenzora). Kruskal je u [45] pokazao da rang tenzora s realnim vrijednostima nije isti u ovisnosti da li se promatra tenzor nad vektorskim prostorom $\mathbb{R}$ ili $\mathbb{C}$. Još jedna razlika je da za slučaj tenzora ne postoji jednostavan algoritam za određivanje ranga već je to NP-problem [46].

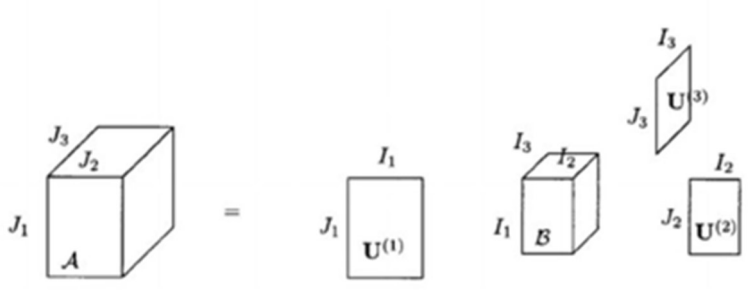


Slika 4.3. Tenzor reda 3 ranga 1 $\boldsymbol{\mathcal{X}} = \boldsymbol{a} \circ \boldsymbol{b} \circ \boldsymbol{c}$ [47]

Mod-n produkt tenzora $\boldsymbol{\mathcal{A}} \in R^{I_1 \times I_2 \times \ldots \times I_N}$ i matrice $\mathbf{U} \in R^{J_n \times I_n}$ označen s $\boldsymbol{\mathcal{A}} \times_n \boldsymbol{U}$ je $(I_1 \times I_2 \times \ldots \times I_{n-1} \times J_n \times I_{n+1} \times \ldots \times I_N)$ tenzor definiran sa

$$(\boldsymbol{\mathcal{A}} \times_n \boldsymbol{U})_{i_1 i_2 \ldots j_n \ldots i_N} = \sum_{i_n} a_{i_1 i_2 \ldots i_n \ldots i_N} U_{j_n i_n} \tag{4.9}$$

za sve vrijednosti indeksa (slika 4.4).



Slika 4.4. Vizualizacija n-mode množenja: tenzor reda 3 $\mathcal{B} \in C^{I_1 \times I_2 \times I_3}$ *pomnožen je s matricama* $\boldsymbol{U}^{(1)} \in C^{J_1 \times I_1}$, $\boldsymbol{U}^{(2)} \in C^{J_2 \times I_2}$, $\boldsymbol{U}^{(3)} \in C^{J_3 \times I_3}$ *[48]*

U radu s tenzorima potrebno je poznavanje određenih matričnih produkata čije definicije slijede.

Kroneckerov produkt dviju matrica $\boldsymbol{A} \in R^{I \times J}$ i $\boldsymbol{B} \in R^{T \times R}$ označava se $\boldsymbol{A} \otimes \boldsymbol{B} \in R^{IT \times JR}$ i definira s

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & a_{12}\mathbf{B} & \cdots & a_{1J}\mathbf{B} \\ a_{21}\mathbf{B} & a_{22}\mathbf{B} & \cdots & a_{2J}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1}\mathbf{B} & a_{I2}\mathbf{B} & \cdots & a_{IJ}\mathbf{B} \end{bmatrix}$$
$$= \begin{bmatrix} \boldsymbol{a_1} \otimes \boldsymbol{b_1} & \boldsymbol{a_1} \otimes \boldsymbol{b_2} & \boldsymbol{a_1} \otimes \boldsymbol{b_3} & \cdots & \boldsymbol{a_J} \otimes \boldsymbol{b_{R-1}} & \boldsymbol{a_J} \otimes \boldsymbol{b_R} \end{bmatrix} \tag{4.10}$$

Hadamardov produkt definira se za matrice istog tipa, označava s $\boldsymbol{A} \circledast \boldsymbol{B}$ a dobije se umnoškom elemenata na istim pozicijama.

Khatri-Rao produkt definiran je za matrice s jednakim brojem stupaca na idući način

$$\mathbf{A} \odot \mathbf{B} = \begin{bmatrix} \boldsymbol{a_1} \otimes \boldsymbol{b_1} & \boldsymbol{a_2} \otimes \boldsymbol{b_2} & \cdots & \boldsymbol{a_J} \otimes \boldsymbol{b_J} \end{bmatrix}$$
$$= \begin{bmatrix} vec(\boldsymbol{b_1 a_1^T}) & vec(\boldsymbol{b_2 a_2^T}) & \cdots vec(\boldsymbol{b_J a_J^T}) \end{bmatrix} \in \mathbb{R}^{\boldsymbol{IT \times J}}. \tag{4.11}$$

Više detalja o tenzorima može se naći u [49, 44].

4.1. Tenzorizacija

Vektori i matrice nastali su kao rezultat potrebe za prezentacijom segmenta rezultata skalarnog mjerenja i mjerenja na mreži. Za razliku od njih, tenzori su prvobitno predstavljeni kao matematički objekti korisni u analizi podataka. Tenzorizacija je postupak kreiranja podatkovnog tenzora od izvornih podataka niže dimenzionalnosti. Autori u [50] predlažu sljedeću taksonomiju kod generiranja tenzora:

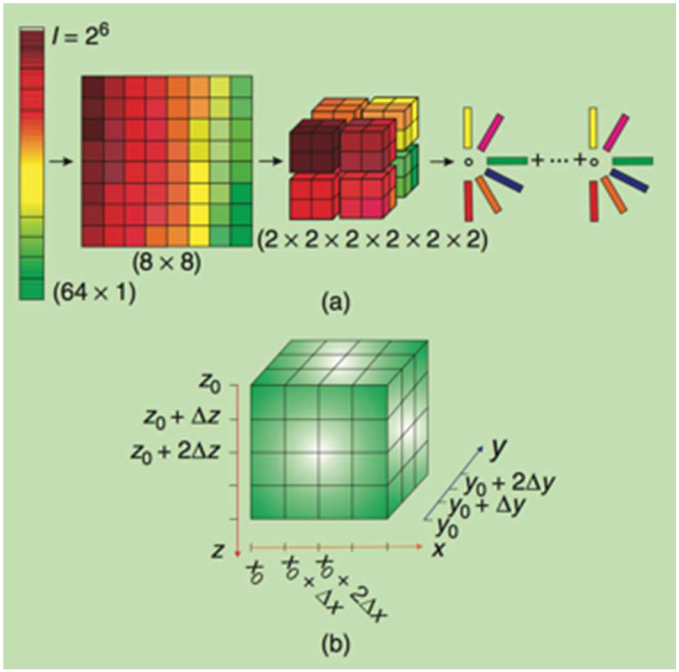
1. Preslagivanje podatkovnih struktura nižih dimenzija: vektori i matrice velikih dimenzija daju se lako tenzorizirati te zatim komprimirati uporabom dekompozicije tenzora (ukoliko dozvoljava aproksimaciju tenzorom nižeg ranga), kao što je pokazano na slici 4.5 a). Kod obrade senzorskih polja,

kombiniranjem podataka iz identičnih podpolja na prirodan način se generiraju tenzori.

2. Matematička konstrukcija: uporabom odgovarajućih matematičkih alata, više u [50]

3. Dizajn eksperimenta: višeznačni podaci mogu se na prirodan način složiti u tenzor; na primjer, u bežičnim komunikacijama komponente signala (vremenska, prostorna, spektralna, itd.) odgovaraju redu tenzora

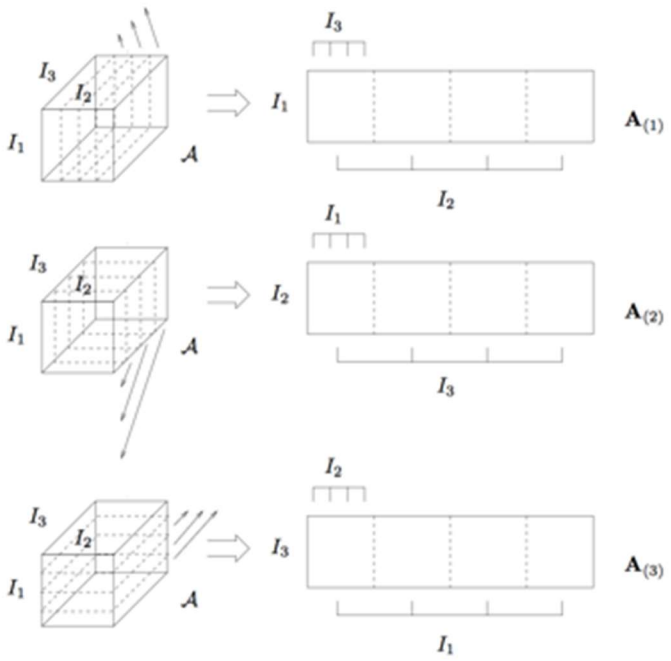
4. Prirodni tenzorski podaci: neki izvori podataka su lako generirani kao tenzori (npr. RGB slike, video, …). U znanstvenom računanju ponekad je potrebno procijeniti diskretiziranu multivarijantnu funkciju što je prirodni tenzor, slika 4.5 b)



Slika 4.5. a) Tenzorizacija vektora ili matrice b) Formiranje tenzora diskretizacijom funkcije f(x,y,z) [50]

4.2. Matrizacija i vektorizacija

Ponekad je korisno tenzor prikazati u obliku matrice. Matrizacija, također poznata kao razvlačenje (engl. *Unfolding*) ili poravnavanje (engl. *Flattening*) je postupak slaganja elemenata tenzora u matricu. Mode-n matrizacija tenzora $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times \ldots \times I_n}$ je postupak kojim se mode-n stupci (odsječci) raspoređuju u matricu, a označava se s $\boldsymbol{\mathcal{X}}_{(n)}$ (Slika 4.6). (Među različitim autorima ne postoji jedinstven pristup o redoslijedu, međutim permutacija stupaca nije bitna, već samo konzistentnost u provođenju kalkulacija.)



Slika 4.6. Razvlačenje $I_1 \times I_2 \times I_3$ tenzora $\boldsymbol{\mathcal{A}}$ u $I_1 \times I_2 I_3$ matricu $\boldsymbol{\mathcal{A}}_{(1)}$, u $I_2 \times I_3 I_1$ matricu $\boldsymbol{\mathcal{A}}_{(2)}$ i u $I_3 \times I_1 I_2$ matricu $\boldsymbol{\mathcal{A}}_{(3)}$ pri čemu je $I_1 \times I_2 \times I_3 = 4$ [51]

Više autora predložilo je generalizaciju pojmova singularnih vrijednosti i vektora, te svojstvenih vrijednosti i vektora sa matrica na tenzore višeg reda, međutim ni jedan pristup nije uspio sačuvati sva svojstva matričnih analogana. Jedan od pristupa dan je u [52].

U nekim slučajevima, pogodnije je matrice i tenzore prikazati u obliku vektora. Vektorizacija matrice je linearna transformacija koja matricu pretvara u vektor na način da sve stupce matrice svrstava jedan ispod drugog. Vektorizacija tenzora definira se kao vektorizacija pripadne mod 1 poravnate matrice.

4.3.Dekompozicija tenzora

Broj elemenata u tenzoru *N*-tog reda tipa $100 \times 100 \times ... \times 100$ za *N>41* premašuje broj atoma u svemiru, te je jasno da rad s tenzorima višeg reda zahtjeva eksponencijalno povećanje u memorijskim i računalnim resursima. U rijetkim slučajevima će biti moguće pohraniti sve elemente tenzora višeg reda eksplicitno. Stoga su razvijeni različiti načini redukcije podataka. U početku su navedene metode primijenjene na tenzorima na način da su podaci iz tenzora reformatirani u matrice i obrađeni na klasičan način. Međutim, pokazalo se da je samo uporabom dekompozicije tenzora višeg reda moguće razviti sofisticirane modele koji sačuvaju i prepoznaju višestruke interakcije i veze. Tenzorska dekompozicija omogućuje prirodno poopćenje nekih metoda korištenih u obradi signala, poput kanonske korelacije i tehnika potprostora, separacije signala, linearne regresije, ekstrakcije i klasifikacije značajki. Dva koncepta multilinearne algebre predstavljaju bitan alat kod redukcije dimenzije u obradi signala višeg reda. To su multilinearno poopćenje dekompozicije singularne vrijednosti i najbolja aproksimacija (u smislu najmanjih kvadrata) danog tenzora tenzorom nižeg ranga.
Ideja dekompozicije tenzora potječe još iz 1927. kada je Hitchcock u [53] iznio prve verzije prikaza tenzora u obliku sume tenzora ranga jedan, dok je 1944. Cattell u [54]

iznio ideju o multiway modelu. Njihove ideje nisu privukle pažnju, te dekompozicija tenzora nije bila interesantna znanstvenicima sve do šezdesetih godina, kada Tucker objavljuje radove o faktorskoj analizi trodimenzionalnih matrica [55,56]. Dok je većina radova bila objavljena u psihometrijskim časopisima, prvi primjer upotrebe dekompozicije tenzora, i to u kemometriji, dali su Appellof i Davidson u [57]. Obzirom da nije postojao opće prihvaćen dogovor o notaciji, ideje su uglavnom bile izražene na način najprihvaćeniji samom autoru što je naravno otežavalo razumijevanje istih. Pored toga, nije bila jasna mogućnost šire primjene tenzora. Tek oko 2000. godine nekoliko grupa znanstvenika radeći neovisno jedna od druge, popularizirala je tenzorski framework i omogućile njegov prijelaz s psihometrije na računalnu numeričku analizu. Bili su to u Nizozemskoj De Lathauwer, De Moore i Vanderwalle, u SAD/Kanadi Vasilescu i Terzopoulos, u SAD-u još dvije grupe znanstvenika Kolda i Bader, te Zhang i Golub, te u Izraelu Shashua i Levin [58]. Moglo bi se reći da je novu tenzorsku eru pokrenuo DeLathauwer, tražeći bolje riješenje od klasičnog ICA pri rješavanju BSS problema. Pojedini znanstvenici naglasak su stavili na teoretsko razmatranje tenzora i razvoj novih metoda dekompozicije, dok su drugi dali konkretne primjere upotrebe tenzora i tenzorskih dekompozicija. Šire promatrano, sa stanovišta analize podataka, pokazali su da tenzorske dekompozicije imaju veliki raspon primjene. Obzirom da je većina podataka koje je potrebno analizirati rezultat više različitih čimbenika, po svom karakteru su prikladni za multimodalnu analizu generiranog tenzora podataka. Popularnost (upotrebe) dekompozicije tenzora stalno raste, te se zadnjih godina s područja psihometrije i kemometrije proširila i na brojna druga: obradu signala, računalni vid, rudarenje podataka, neuroznanost te mnoge druge. Tenzorske dekompozicije mogu se koristiti za klasificiranje bilo kojih objekata ili njihovih atributa, dok bi tipičan primjer upotrebe bio obrada slika, ili konkretnije temeljem slika prepoznavanje lica [59] ili prepoznavanje ljudskih kretnji.

Razvijeni su i softverski paketi koji omogućavaju rad s tenzorima. To su samostalni softverski paketi: SPLATT - Parallel Sparse Tensor Decomposition [60], CADABRA [61], dok sva tri najpoznatiji matematička softvera, MATLAB, Mathematica i Maple također imaju ugrađenu podršku za rad s tenzorima. Među najviše korištene spadaju Tensorlab [62] i Tensor Toolbox [47] namjenjeni korištenju unutar MATLAB-a.

U slučaju tenzora reda 2, većina metoda za sažimanje svodi se na reduciranu dekompoziciju singularnih vrijednosti, ali se situacija znatno razlikuje za tenzore višeg reda. Korištenjem navedenih tenzorskih definicija SVD (dekompozicija singularnih vrijednosti) se za kvadratnu matricu reda 2 da izraziti u obliku [63]:

$$\mathbf{M} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} \sigma_{11} & 0 \\ 0 & \sigma_{22} \end{bmatrix} \begin{bmatrix} f & g \\ h & i \end{bmatrix} = \sigma_{11} \begin{bmatrix} a \\ c \end{bmatrix} \circ \begin{bmatrix} f \\ g \end{bmatrix} + \sigma_{22} \begin{bmatrix} b \\ d \end{bmatrix} \circ \begin{bmatrix} h \\ i \end{bmatrix} = \mathbf{U}_1 \mathbf{\Sigma} \mathbf{U}_2^{\boldsymbol{T}}$$
$$= \begin{bmatrix} \mathbf{u}_1^{(1)} & \mathbf{u}_1^{(2)} \end{bmatrix} \begin{bmatrix} \sigma_{11} & 0 \\ 0 & \sigma_{22} \end{bmatrix} \begin{bmatrix} \mathbf{u}_2^{(1)} & \mathbf{u}_2^{(2)} \end{bmatrix}^{\boldsymbol{T}} = \sum_{i=1}^{R=2} \sum_{j=1}^{R=2} \sigma_{ij}\, \mathbf{u}_1^{(i)} \circ \mathbf{u}_2^{(j)} \qquad (4.12)$$

Kao generalizacija dekompozicije singularnih vrijednosti predstavljene su najšire prihvaćene kanonska poliadična dekompozicija (CPD) te Tuckerova dekompozicija ili multilinearni SVD (MLSVD). Premda obje generaliziraju SVD, CP razlaže tenzor u sumu minimalnog broja tenzora ranga jedan, dok je MLSVD povezana sa skupom tenzora ranga *n*. De Lathauwer je u [64] predstavio metodu koja objedinjuje dvije navedene dekompozicije i nazvao je Blok terminska dekompozicija (engl. *Block Term Decomposition - BTD*).

Obzirom da je trodimenzionalni slučaj široko primjenjiv, a uz to dovoljan za mnoge potrebe zbog jednostavnijeg izračuna a i percepcije, većina metoda pojašnjena je upravo na primjerima tenzora reda tri.

4.3.1. Kanonska poliadična dekompozicija

Kanonska poliadična dekompozicija (engl. *Canonical Polyadic - CP*) koja je još poznata pod nazivima kanonska dekompozicija (engl. *Canonical Decomposition - CANDECOMP*) te paralelna faktorska dekompozicija (engl. *Parallel Factor Decomposition - PARAFAC*) predstavlja generalizaciju metode PCA. CP dekompozicija bazira se na ideji da se tenzor prikaže u obliku sume tenzora ranga 1, čime je tenzor aproksimiram tenzorom ranga *r*. Više je autora (u različitim razdobljima) koji su neovisno jedan o drugom radili na razvoju navedene ideje, a predložene nazive dekompozicije zajedno s autorima (i razdobljem) mogu se vidjeti u Tabeli 4.1.

Tabela 4.1: Neki od brojnih naziva CP dekompozicije [44]

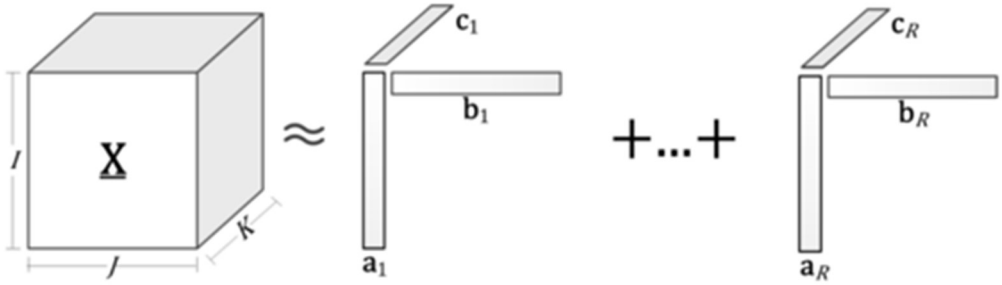
Naziv	Autor
Polyadic Form of a Tensor	Hitchcock, 1927
PARAFAC (Parallel Factors)	Harshman, 1970
CANDECOMP or CAND (Canonical decomposition)	Carroll and Chang, 1970
Topographic Components Model	Mocks, 1988
CP (CANDECOMP/PARAFAC	Kiers, 2000

Gore navedena ideja primijenjena na tenzor $\mathrm{X} \in \mathrm{R}^{\mathrm{I}\times\mathrm{J}\times K}$ reda 3, odnosno CP dekompozicija zapisuje se

$$\underline{\mathbf{X}} \approx \underline{\hat{\mathbf{X}}} = \sum_{r=1}^{R} \boldsymbol{a_r} \circ \boldsymbol{b_r} \circ \boldsymbol{c_r} \tag{4.13}$$

pri čemu je R pozitivan cijeli broj, $a_r \in R^I, b_r \in R^J$ i $c_r \in R^K$ za $r \in \{1, \dots, R\}$ (slika 4.7). Na razini elementa CPD model se može zapisati kao

$$\boldsymbol{\mathcal{X}}_{i,j,k} \approx \sum_{r=1}^{R} a_{ir} b_{jr} c_{kr} \tag{4.14}$$



Slika 4.7. CP dekompozicija za tenzor reda 3 [65]

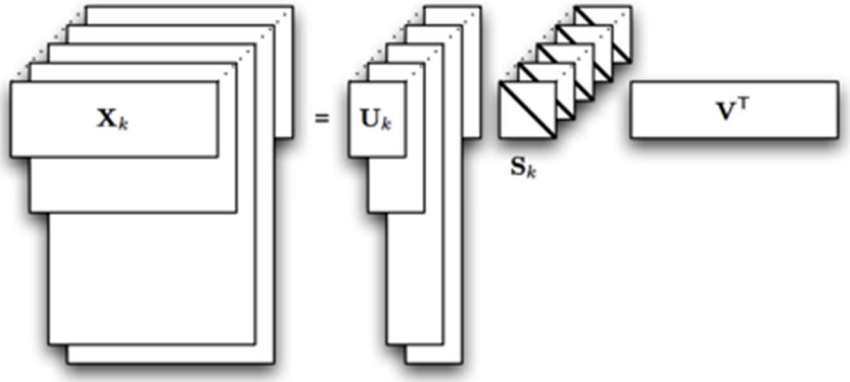
Vektori iz svakog od modova mogu se grupirati u matrice koje se nazivaju faktorskim matricama $\boldsymbol{A} = [\boldsymbol{a_1}, \boldsymbol{a_2}, \dots, \boldsymbol{a_R}]$, $\boldsymbol{B} = [\boldsymbol{b_1}, \boldsymbol{b_2}, \dots, \boldsymbol{b_R}]$ i $\boldsymbol{C} = [\boldsymbol{c_1}, \boldsymbol{c_2}, \dots, \boldsymbol{c_R}]$ dok se

vektori nazivaju faktorima. Korištenjem produkta tenzora i matrice CPD model se može zapisati kao

$$\underline{\mathbf{X}} \approx \underline{\hat{\mathbf{X}}} = \underline{\mathbf{I}} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} \tag{4.15}$$

pri čemu je $\mathbf{I} \in \mathrm{R}^{\mathrm{R}\times\mathrm{R}\times\mathrm{R}}$ dijagonalni tenzor reda 3 sa jedinicama na dijagonali.

Ukoliko je broj komponenti rastava u egzaktnoj (nije u pitanju aproksimacija, već jednakost) CP dekompoziciji jednak rangu tenzora, takvu dekompozicij zove se dekompozicija ranga. Zanimljivo svojstvo tenzora višeg reda je da je njihova dekompozicija ranga često jedinstvena, što nije slučaj kod matričnih dekompozicija.
S ciljem prilagodbe osnovnog modela nekim karakterističnim zahtjevima, znanstvenici su s vremenom razvili verzije CP/PARAFAC modela: PARAFAC2, pomaknuti PARAFAC (engl. *Shifted PARAFAC - S-PARAFAC*), konvolutivni PARAFAC (engl. *convolutive PARAFAC - cPARAFAC*) te Paralelni Faktori s Linearnom Ovisnošću (engl. *Parallel Factors with Linear Dependency - PARALIND*). PARAFAC2 [66, 67] model je manje restriktivan u odnosu na osnovni zahvaljujući čemu može biti primijenjen na skup matrica s promjenjivom veličinom u jednom modu (Slika 4.8). Stoga PARAFAC2 također rješava problem modeliranja trodimenzionalnih polja s odrescima različitih dimenzija.



Slika 4.8. Ilustracija PARAFAC2 dekompozicije [44]

S-PARAFAC [66, 68] je modifikacija koja rješava problem pomaknutih faktora koji nastaje kod sekvencijalnih podataka (poput vremenskih nizova, digitaliziranih slika) ukoliko profili latentnih faktora pomiču poziciju gore ili dolje slijedom mjerenja: takvi pomaci narušavaju multilinearnost te modeli standardnih faktora / komponenti više ne vrijede. cPARAFAC [68, 69] predstavlja generalizaciju nenegativne matrične faktorske dekonvolucije (NMFD) primijenjene na višedimenzionalne spektralne podatke. U analizi podataka problem koji se često pojavljuje je da rang komponentnih matrica nije jednak (engl. *Rank deficiency*). U takvom slučaju PARALIND model omogućuje modeliranje uvođenjem matrice zavisnosti (ili interakcije) među komponentnim matricama [68, 70].
Najčešće korišteni algoritam za implementaciju CP dekompozicije je algoritam izmjeničnih najmanjih kvadrata (engl. *Alternating Least Squares - ALS*). Suma kvadrata razlike između tenzora i modela minimizira se:

$$\min_{A,B,C} \|\boldsymbol{\mathcal{X}} - \sum_{r=1}^{R} \boldsymbol{a_r} \circ \boldsymbol{b_r} \circ \boldsymbol{c_r}\|_F^2 \tag{4.16}$$

fiksirajući dvije od tri faktorske matrice što problem svodi na linearnu metodu najmanjih kvadrata za treću faktorsku matricu.

Ostali popularni algoritmi su: Jennrichov algoritam (primjenjiv u slučaju kad su A, B i C linearno nezavisni tj. punog ranga), te metoda tenzorske snage (ukoliko su faktorske matrice jednake, a svi a_r ortogonalni) [više u 43]. Implementacija dekompozicije porastom reda tenzora postaje računalno sve zahtjevnija te su predstavljeni različiti pristupi s ciljem rješavanja problema. U [71] autori umjesto direktne faktorizacije tenzora višeg reda predlažu dekompoziciju poravnatog tenzora nižeg reda (npr. 3). Na temelju procijenjenog tenzora, generira se strukturirani Kruskalov tenzor, iste dimenzije kao i podatkovni tenzor. Konačno rješenje sada je jednostavno utvrditi koristeći brze algoritme za strukturirani CPD. Pregled i usporedba algoritama baziranih na PARAFAC modelu dana je u [72, 73].
Posljednjih godina, CP dekompozicija koristi se u različitim područjima: u obradi signala [73, 74, 75], u telekomunikacijama [76], u obradi senzorskih polja za procjenu smjera dolaska (engl. *Direction Of Arrival - DOA*) [77], u rudarenju podataka [78].

4.3.2. Tuckerova dekompozicija

Tuckerova dekompozicija predstavlja općenitiji model koji uključuje CP kao poseban slučaj. Smatra se generalizacijom metoda PCA i SVD za tenzore višeg reda. Kao i CP dekompozicija, može se pronaći pod različitim nazivima, a pregled je dan u Tabeli 4.2.

Tabela 4.2: Različiti nazivi Tuckerove dekompozicije [44]

Naziv	Autor
Three-mode factor analysis (3MFA/Tucker3)	Tucker
Three-mode principal component analysis (3MPCA)	Kroonenberg and De Leeuw
N-mode principal components analysis	Kapteyn
Higher-order SVD (HOSVD)	De Lathauwer

Ova dekompozicija tenzor razlaže u jezgreni tenzor pomnožen s matricom duž svakog moda, što se u trodimenzionalnom slučaju može zapisati

$$\boldsymbol{\mathcal{X}} \approx \boldsymbol{\mathcal{G}} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} = \sum_{p=1}^{P} \sum_{q=1}^{Q} \sum_{r=1}^{R} \boldsymbol{g}_{pqr}\, \boldsymbol{a}_p \circ \boldsymbol{b}_q \circ \boldsymbol{c}_r = [\![\boldsymbol{\mathcal{G}}; \mathbf{A}, \mathbf{B}, \mathbf{C}]\!]. \tag{4.17}$$

gdje je G jezgreni tenzor čiji elementi pokazuju stupanj interakcije između komponenti a A, B i C su faktorske matrice (koje su obično ortogonalne). Ovaj model se još naziva Tucker3 model. Ukoliko su P, Q, R manji od I, J, K jezgreni tenzor se može smatrati komprimiranom verzijom početnog tenzora.

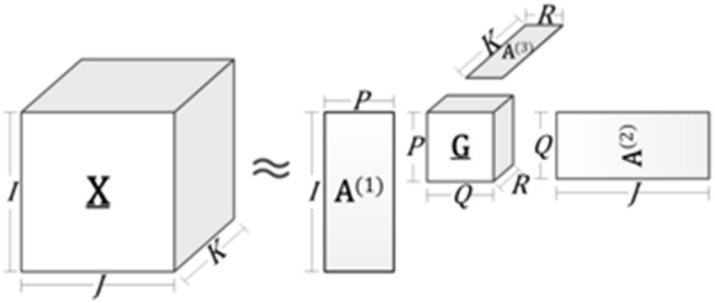
Analogijom se dobije poopćenje za tenzor $\boldsymbol{\mathcal{X}} \in R^{I_1 \times I_2 \times \ldots \times I_N}$ reda N kao

$$\underline{\boldsymbol{\mathcal{X}}} \approx \widehat{\underline{\boldsymbol{\mathcal{X}}}} = \underline{\mathbf{G}} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \cdots \times_N \mathbf{A}^{(N)} \tag{4.18}$$

gdje je $\underline{\mathbf{G}} \in \mathbb{R}^{J_1 \times J_2 \times \cdots \times J_N}$, a $\mathbf{A}^{(n)} \in \mathbb{R}^{I_n \times J_N}$.

U Tuckerovom modelu, dimenzije (komponenti) jezgrenog tenzora određuju broj faktora u modovima 1, 2 i 3. U slučaju kada je broj faktora manji od dimenzije izvornog tenzora u pripadnom modu jezgreni tenzor može se smatrati komprimiranom verzijom izvornog tenzora. Kako se Tucker model često koristi za kompresiju i smanjenje dimenzionalnosti, obično je jezgra manje dimenzije od izvornog tenzora. Ilustracija Tucker3 modela dana je na slici 4.9 gdje je vidljivo da elementi jezgre određuju razinu sprege između faktora u pojedinim modovima [65].

Varijanta Tucker modela u kojoj se jedna od faktorskih matrica fiksira kao jedinična naziva se Tucker2 model. Slično se dobiva i Tucker1 model, tako da se dvije od tri faktorske matrice postave u jediničnu. Ideja na kojoj se temelji Tucker1 model je da se višedimenzionalno podaci prikažu u obliku matrice na koju je onda primijenjena SVD.



Slika 4.9. Tucker 3-model [65]

Veća fleksibilnost Tucker modela posljedica je slobode u izboru elemenata jezgre, što donosi više mogućnosti u praktičnim primjenama. Glavni nedostatak fleksibilnosti je što se za razliku od CPD modela u ovom slučaju ne mogu jedinstveno odrediti faktorske matrice. U ovisnosti o primjeni i tipu podataka, jedinstvenost se pokušava dobiti uvođenjem uvjeta za faktorske matrice i jezgru (nenegativnost, rijetkost ili ortogonalnost faktorskih matrica, određena struktura jezgre).
Obzirom da je Tuckerova dekompozicija predstavljena, te kao takva najviše korištena u psihometriji, način razmatranja i korištena terminologija poprilično je različita od onog potrebnog npr. u obradi signala.
Korištenjem SVD terminologije, te upotrebom oznaka koje predstavljaju prirodno proširenje onih upotrjebljenih kod matrica autori su u [48] pokazali da je ona multilinearna generalizacija SVD-a (HOSVD iz Tabele 4.2, slika 4.10). Prema [48], svaki $I_1 \times I_2 \times \ldots \times I_N$ tenzor $\boldsymbol{\mathcal{A}}$ da se napisati u obliku produkta

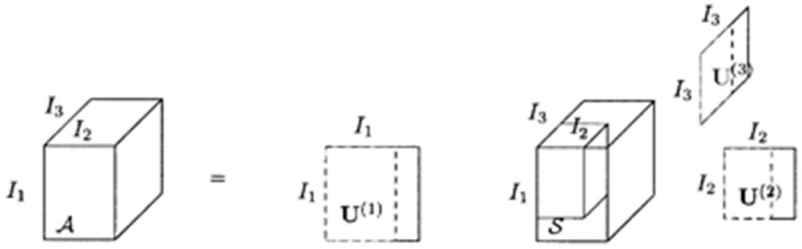
$$\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{S}} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \cdots \times_N \mathbf{U}^{(N)} \tag{4.19}$$

pri čemu je $\boldsymbol{U}^{(n)} = \left(\boldsymbol{U}_1^{(n)} \boldsymbol{U}_2^{(n)} \ldots \boldsymbol{U}_{I_n}^{(n)}\right)$ unitarna matrica tipa $I_n \times I_n$, $\boldsymbol{\mathcal{S}}$ je $I_1 \times I_2 \times \ldots \times I_N$ tenzor (istih dimenzija kao i početni!) čiji su podtenzori $\boldsymbol{\mathcal{S}}_{i_n} = \alpha$ dobiveni fiksiranjem n-tog indeksa na α. Navedeni podtenzori imaju svojstva:

1) međusobna ortogonalnost: dva podtenzora $\boldsymbol{\mathcal{S}}_{i_n} = \alpha$ i $\boldsymbol{\mathcal{S}}_{i_n} = \beta$ su ortogonalna za sve moguće vrijednosti od n, α, β uz uvjet $\alpha \neq \beta$ ako vrijedi $\langle \boldsymbol{\mathcal{S}}_{i_n=\alpha}, \boldsymbol{\mathcal{S}}_{i_n=\beta} \rangle = 0$
2) poredak: $\left\| \boldsymbol{\mathcal{S}}_{i_n=1} \right\| \geq \left\| \boldsymbol{\mathcal{S}}_{i_n=2} \right\| \geq \cdots \left\| \boldsymbol{\mathcal{S}}_{i_n=I_n} \right\| \geq 0$ za sve moguće vrijednosti n.

Frobeniusove norme $\left\|\boldsymbol{S}_{i_n=i}\right\|$ simbolički navedene s $\sigma_i^{(n)}$ su n -mod singularne vrijednosti od $\boldsymbol{\mathcal{A}}$ dok je vektor $\boldsymbol{U_i^{(n)}}$ n-mod singularni vektor.

De Lathauwer i Vandewalle su u [66] razmotrili primjenu Tuckerove dekompozicije u obradi signala dok je jedna od poznatijih primjena dekompozicije u Tensorfaces [63], aplikaciji s područja računalnog vida.



Slika 4.10. Vizualizacija HOSVD za tenzor 3. reda [48]

Probleme koji se javljaju kod obrade slika mogu se svrstati u različite kategorije a jedna od njih bila bi fuzija slike. To je proces koji kombinira podatke dobivene s dvije ili više slika iste scene na način da rezultirajuća slika bude prilagođena ili daljnjoj obradi (segmentaciji, izdvajanje značajki, prepoznavanju ciljeva) ili ljudskoj i strojnoj percepciji. U [79] autori se predstavili metodu fuzije baziranu na HOSVD-u; umjesto cijelog tenzora obrađuju se podijeljeni podtenzori na osnovu čega se vrši fuzija.

4.3.3. Tenzorska vlak dekompozicija

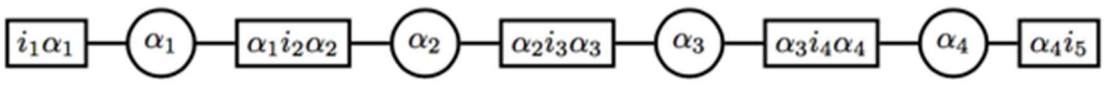
Predstavljena CP dekompozicija ima bitne nedostatke: njen izračun je NP-problem, a pored toga aproksimacija fiksnim kanonskim rangom u nekim slučajevima nije izvediva [80, 81]. Alternativa je bila Tuckerova dekompozicija međutim ona je pogodna samo za tenzore malih dimenzija. Kod Tuckerove dekompozicije rastom dimenzije jezgre javlja se problem pohrane $r_1 \times \cdots \times r_d$ jezgrenih tenzora. Stoga su znanstvenici tražili takvu dekompoziciju gdje bi se izbjegla potreba za eksponencijalno rastućim memorijskim zahtjevima. Oseledets je u [80] predložio Tenzorsku vlak dekompoziciju (engl. *Tensor Train - TT*) idućeg oblika

$$A(i_1, i_2, \cdots i_d) = G_1(i_1)G_2(i_2)\cdots G_d(i_d) \tag{4.20}$$

pri čemu su $\boldsymbol{G}_k(i_k)$ matrice tipa $r_{k-1} \times r_k$ pri čemu je $r_0 = r_d = 1$. U indeksnom obliku:

$$\boldsymbol{A}(i_1, \cdots i_d) = \sum_{\alpha_0,\cdots,\alpha_{d-1},\alpha_d} \boldsymbol{G}_1(\alpha_0, i_1, \alpha_1)\boldsymbol{G}_2(\alpha_1, i_2, \alpha_2)\cdots \boldsymbol{G}_d(\alpha_{d-1}, i_d, \alpha_d) \tag{4.21}$$

Tenzorski vagoni su $G_1, G_2, \ldots G_d$ pri čemu svaka dva susjeda imaju zajednički indeks zbrajanja. Indeksi zbrajanja α_k kreću se od 1 do r_k (r_k je rang kompresije) i nazivaju se pomoćnim indeksima, za razliku od indeksa $i_k$koji se nazivaju prostorni indeksi. Grafički prikaz, tj. mreža dekompozicije, dana je na slici 4.11 .



Slika 4.11. Mreža TT za d=5 [80]

Mreža sadržava dvije različite vrste čvorova. Oni u pravokutnicima sadrže prostorne indekse (tj. indekse izvornog tenzora) i neke pomoćne indekse dok krugovi sadrže samo pomoćne indekse i predstavljaju vezu. Ako je pomoćni indeks prisutan u dvije jezgre, oni se povezuju. Da bi se dobio element tenzora na određenoj poziciji, potrebno je pomnožiti sve tenzore u pravokutnicima i zatim izvršiti zbrajanje preko svih pomoćnih indeksa. Obzirom da prikaz mreže izgleda upravo kao vlak s kolicima i vezama između njih, to objašnjava porijeklo samog naziva dekompozicije.

TT dekompozicija sve više nalazi primjenu u raznim područjima. U [82] autori su pokazali kako je prebacivanjem matrica težina potpuno povezanih slojeva neuronskih mreža u TT format moguće znatno reducirati broj parametara bez smanjenja efikasnosti samih slojeva.

4.3.4. Ostale dekompozicije

U [83] dana je multilinearna verzija metode analize glavnih komponenti, pod nazivom multilinearni PCA (engl. *Multilinear - PCA*) gdje ulazni podaci ne moraju biti u formi vektora već također u obliku tenzora višeg reda. Kako je multilinearni PCA proširenje osnovnog modela PCA, predstavljene su i definicije pojmova svojstvene vrijednosti i svojstveni vektori. Predstavljeni model moguće je primijeniti u raznim aplikacijama, a kao primjer uporabe naveden je problem prepoznavanja hoda korištenjem novog prikaza pod nazivom EigenTensorGait.

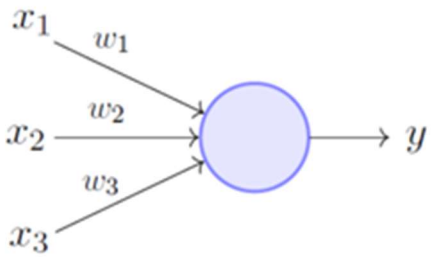
Klasične metode PCA i ICA često se koriste u prepoznavanju lica i daju dobre rezultate ukoliko je identitet osobe jedina stavka koja se mijenja. Međutim njihova efikasnost drastično opada u slučaju promjene dodatnih čimbenika, poput osvjetljenja, ili točke gledišta. Multilinearna verzija analize nezavisnih komponenti dana u [84] primjenom statistike višeg reda uspješno tretira dodatne uvjete.
Više autora predstavilo je svoje verzije ortogonalne dekompozicije tenzora, od čega je većina metoda varijacija ALS. Kolda u [85] u ovisnosti o dvije različite definicije ortogonalnosti istražuje ortogonalnu dekompoziciju tenzora.

5. NEURONSKE MREŽE

Umjetne neuronske mreže (*ANN*) motivirane su biološkim neuronskim mrežama te predstavljaju njihovu digitalnu imitaciju. Mogu se definirati kao skup međusobno povezanih jednostavnih procesnih elemenata (jedinica, čvorova), a čija se funkcionalnost temelji na biološkom neuronu i koji služe distribuiranoj paralelnoj obradi podataka.

Često se koriste za raspoznavanje uzoraka te se mogu smatrati nelinearnim klasifikatorom koji dijeli ulazni vektorski prostor uzoraka u klase koje imaju nelinearne granice [86]. Pokazale su se iznimno učinkovite u rješavanju širokog raspona zadataka (analizi slike, govora, podataka, signala), osiguravajući pri tom velika dostignuća po pitanju točnosti. Međutim, brojni su i nedostaci ANN-a, poput zahtjeva za ogromnom količinom promatranih podataka, velikih računalnih zahtjeva, njihove značajke je teško interpretirati te nije moguće matematički/teoretski potpuno objasniti kada će raditi i zašto.

Obzirom na veliku brzinu obrade podataka ljudskog mozga te mogućnost učenja iz iskustva, oduvijek je postojala težnja za ostvarivanjem umjetne inteligencije nalik na ljudsku. Prve pokušaje računalne imitacije rada ljudskog mozga, odnosno ljudske neuronske mreže, napravili su McCulloch i Pitts još 1940-ih godina kada su kao rezultat istraživanja neurofizičkih karakteristika živih bića objavili model pod nazivom Threshold Logic Unit (TLU). TLU model "oponašao" je biološki neuron na idući način: ulazni signali su numeričke vrijednosti, jakost sinapse izražena je težinskim faktorom *w* koji se pomnoženi sa signalima sumiraju u tijelu stanice; ukoliko je dobiveni iznos iznad definiranog praga, neuron daje izlazni signal. Inspiriran radom svojih prethodnika, Frank Rosenblatt je desetak godina kasnije predstavio klasični model perceptrona – najjednostavniji oblik umjetnog neurona. Njega su kasnije nastavili razvijati Minsky i Papert i upravo je njihov model prihvaćen pod nazivom perceptron. (Slika 5.1).

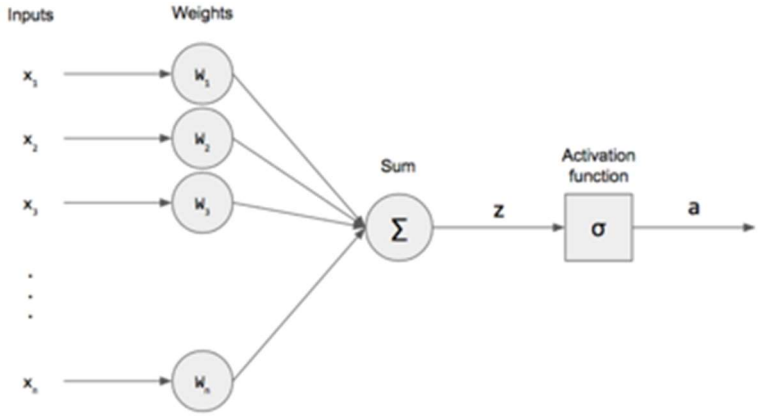


Slika 5.1. Model perceptrona (Minsky i Papert) [87]

Umjetni neuron (engl. *Artificial neuron*) koji se danas koristi kao osnovni građevni blok neuronskih mreža (Slika 5.2), ponešto se razlikuje od modela perceptrona. Osnovna razlika je u aktivacijskog funkciji.

Ulazi u neuron (koji ujedno mogu biti i izlazi prethodnih neurona) označeni su s x_i, $i = 1 \dots n$, dok su pripadajuće težine, koje predstavljaju važnost pojedine ulazne varijable označene s w_i, $i = 1 \dots n$. Težine su parametri koje mreža može sama naučiti. Ukoliko se sumi produkata ulaza s pripadnim težinama doda vrijednost praga (engl. *Bias*) dobije se vrijednost $z = \sum_{i=1}^{n} w_i x_i + b$. Radi pojednostavljenja, navedena suma

može se zapisati uporabom skalarnog produkta, a ukoliko se definira $x_0 = 1$ te vrijednost praga ugradi u vektor težina na način da se definira $w_0 = b$, gornja vrijednost se računa kao $z = wx$.

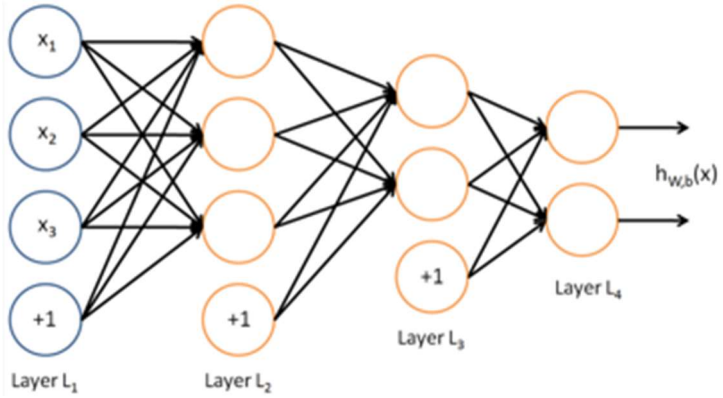


Slika 5.2. Umjetni neuron [87]

Za razliku od perceptrona, koji za izlaz može imati samo vrijednosti 0 ili 1, upotrebom aktivacijske funkcije omogućeno je da izlaz može biti (bilo koji) realan broj. Najčešće korištene aktivacijske funkcije su: korak funkcija, signum, linearna, sigmoidna, hiperbolično-tangentna. Prednost uglađenih funkcija je da male promjene u ulaznim podacima rezultiraju s malim promjenama u dobivenoj vrijednosti. Aktivacijska funkcija zatim se primijeni na prethodno izračunatu vrijednost z, tj. računa se $a = \sigma(z) = \sigma(wx)$ što predstavlja izlaznu vrijednost neurona.

5.1. Arhitektura i princip rada neuronskih mreža

Neuronske mreže sastoje se od slojeva, i u ovisnosti o njihovom broju se dijele na: dvoslojne, koje sadrže samo ulazni i izlazni sloj (poput perceptrona), višeslojne (pored ulaznog i izlaznog postoji i više skrivenih slojeva, slika 5.3), te duboke mreže (sadrže i nekoliko stotina slojeva). Broj slojeva mreže naziva se dubinom modela.



Slika 5.3. Primjer NN s dva skrivena sloja [88]

Složenost podataka koje je potrebno obraditi određuje broj čvorova u ulaznom sloju, dok broj izlaznih čvorova ovisi o broju klasa koje neuronska mreža treba predvidjeti. Najveći problem leži u određivanju broja i složenosti skrivenih slojeva.

Unaprijedne (engl. *Feedforward*) neuronske mreže su takve kod kojih čvorovi idućeg sloja primaju izlaz iz prethodnog sloja. Za razliku od njih, rekurentne neuronske mreže (engl. *Recurrent neural networks - RNN*) dozvoljavaju upotrebu petlji.

Rad NN se odvija u dvjema fazama: faze učenja (treniranja) i faze obrade podataka. Pri tom se razlikuje učenje s nadzorom (engl. *Supervised learning*) kod kojeg je dan skup primjera gdje su poznate vrijednosti izlaznih varijabli od učenja bez nadzora (engl. *Unsupervised learning*) gdje je rezultat obrade *a priori* nepoznat. Učenje mreže se provodi sve dok nije postignuta odgovarajuća točnost. Valjanost aproksimacije mjeri se funkcijom cijene (engl. *Cost function*) $c(w; x, y) = \frac{1}{n}\sum_{i=o}^{n}\|y - \hat{y}\|^2$ koja predstavlja (prosječni) ukupni gubitak.

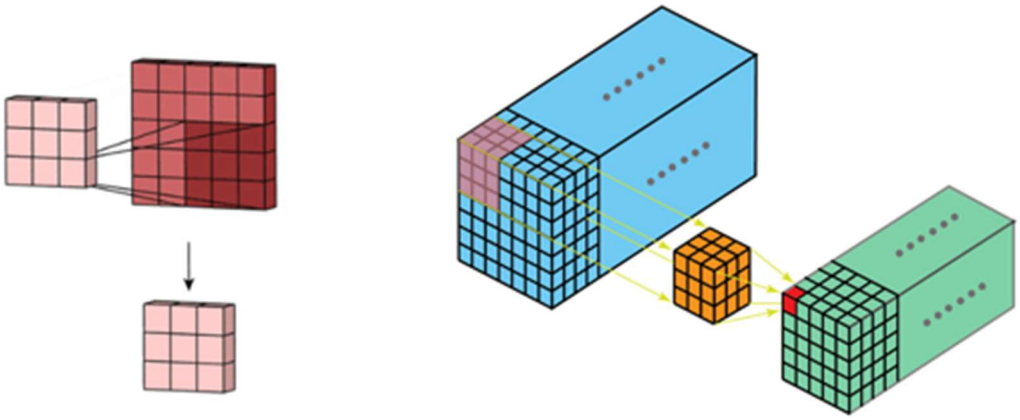
Način na koji neuronska mreža "uči" je da računa ukupni gubitak i pokušava ga minimizirati, odnosno smanjuje vrijednost funkcije troška podešavanjem parametara – što su kod NN težine. Učenje se provodi kroz niz iteracija koje se sastoje od propagacije unaprijed i propagacije pogreške unazad, pri čemu se parametri ažuriraju u svakoj iteraciji. Algoritam propagacije unaprijed u ovisnosti o ulaznoj vrijednosti x generira rezultat $\hat{y}$. Nakon toga računa vrijednost funkcije cijene, te primjenjuje algoritam propagacije pogreške unazad. Korištenjem metode gradijentnog spusta (engl. *Gradient descent*) se u svakoj iteraciji parametri modela ažuriraju u smjeru pada funkcije cijene.

5.2. Konvolucija

Konvolucija je jedna od najvažnijih operacija u obradi signala. Premda se operacija konvolucije obično prvo definira u kontinuiranom slučaju (za dvije funkcije s realnim domenama koja kao rezultat daje modificiranu verziju jedne od dviju ulaznih funkcija) naš je fokus stavljen na uporabu diskretne verzije konvolucije koja se koristi u (konvolucijskim) neuronskim mrežama. Za operaciju konvolucije najčešće se koristi oznaka $*$, a diskretna konvolucija definira se na idući način:

- u jednodimenzionalnom slučaju
$$h(x) * f(x) = \sum_{i=-n}^{n} h(i) f(x - i) \tag{5.1}$$
- u dvodimenzionalnom slučaju
$$h(x, y) * f(x, y) = \sum_{j=-N}^{N}\sum_{i=-N}^{N} h(i, j) f(x - i, y - j) \tag{5.2}$$
- te analogno u trodimenzionalnom slučaju
$$h(x, y, z) * f(x, y, z) = \sum_{k=-N}^{N}\sum_{j=-N}^{N}\sum_{i=-N}^{N} h(i, j, k) f(x - i, y - j, z - k) \tag{5.3}$$

Operacija konvolucije ima svojstvo komutativnosti, asocijativnosti, te distributivnosti u odnosu na zbrajanje. Vizualizacija djelovanja konvolucije u 2-dimenzionalnom i 3-dimenzionalnom slučaju prikazana je na slici 5.4.



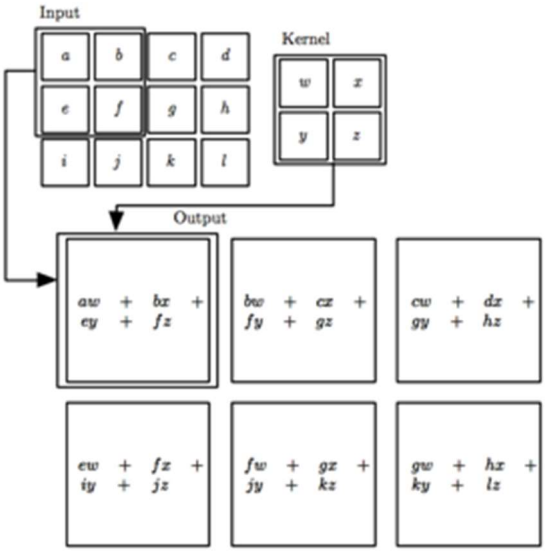
Slika 5.4. Vizualizacija konvolucije a)2-D slučaj. Filter se može kretati u dva smjera, i rezultat je dvodimenzionalan skup podataka b)3-D slučaj. Trodimenzionalni filter kreće se u sva tri smjera i rezultat je trodimenzionalan [89]

5.2.1. Upotreba konvolucija u neuronskim mrežama

Konvolucije korištene u umjetnim neuronskim mrežama služe za ekstrakciju značajki (niskog) nivoa iz ulaznih podataka. Njihovo bitno svojstvo je da čuvaju prostorne veze među ulaznim podacima.

U terminologiji neuronskih mreža, prvi argument (funkcija) konvolucije naziva se ulazom (najčešće slika), drugi argument je filter, odnosno kernel dok se dobiveni rezultat naziva mapa značajki. Kod neuronskih mreža, ulazni podaci su obično prikazani u obliku višedimenzionalnog polja podataka, dok je filter također višedimenzionalno polje parametara te se i jedni i drugi mogu smatrati tenzorima.

Diskretna konvolucija definirana formulom (5.2) može se promatrati kao poseban tip matričnog množenja, što je u dvodimenzionalnom slučaju prikazano na slici 5.5.



Slika 5.5. Primjer izračuna 2D konvolucije [90]

Ukoliko se na filter gleda kao na predložak, s njim se klizi po slici tražeći lokaciju gdje se predložak poklapa sa sličnim vrijednostima na slici.

Detaljan opis načina djelovanja filtera je idući [91]: ukoliko se na primjer uzme kernel $K = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$, te ako se na slici na lokaciji (x,y) nalazi horizontalni rub (u tom slučaju vrijednosti piksela na pozicijama (x+1,y) i (x-1,y) su znatno drugačije od onih u promatranoj lokaciji), može se očekivati veliku vrijednost rezultata konvolucije.

Ako se za mjeru sličnosti uzme Euklidska udaljenost - sumu kvadrata razlika između vrijednosti na predlošku i na slici, jednostavno je pokazati da povećanje rezultata dobivenog konvolucijom slike i filtera rezultira smanjenjem izračunate udaljenosti. Stoga, u gore navedenom primjeru, velika dobivena vrijednost rezultata konvolucije ukazuje na postojanje horizontalnog ruba.

Transponiranjem matrice K moguće je detektirati vertikalne rubove, dok bi detekcija rubova koji leže pod drugim kutovima zahtjevala upotrebu nekog drugog kernela. U dubljim slojevima neuronske mreže moguće je implementirati složenije uzorke, npr. grupe rubova koji formiraju određeni oblik. Udruživanjem složenijih uzoraka u još dubljim slojevima mreže omogućuje prepoznavanje određenih tipova objekata.

Navedeno svojstvo asocijativnosti konvolucije omogućuje poboljšanje efikasnosti primjene uzastopnih konvolucija u određenim slučajevima. Naprimjer, u slučaju da se želi izgladiti neku sliku i dobiveni rezultat derivirati. To se može napraviti konvolviranjem slike s Gaussovim filterom te zatim konvolviranjem s filterom za derivaciju. Alternativni pristup bio bi (zahvaljujući svojstvu asocijativnosti) izvršiti konvoluciju derivativnog filtera s Gaussovim čiji rezultat je filter nazvan DOG (engl. *Difference of Gaussian*) koji se zatim konvolvira sa slikom. DOG filter može biti unaprijed izračunat, čime se uštedi na jednoj konvoluciji [92].

5.2.2. Separabilne konvolucije

Ukoliko je K veličina (retka ili stupca) konvolucijskog kernela, izvođenje konvolucije zahtjeva K^2 operacija. Proces konvolucije može biti znatno ubrzan ukoliko je filtriranje moguće izvršiti prvo jednodimenzionalnom horizontalnom konvolucijom za kojom slijedi jednodimenzionalna vertikalna konvolucija, što bi ukupan broj procesa svelo na 2K. Konvolucijski kernel za koji je moguće izvesti navedeno se kaže da je separabilan. Može se pokazati da je dvodimenzionalni kernel K koji može biti zamijenjen slijednim izvođenjem horizontalnog h te vertikalnog kernela v , moguće prikazati kao vektorski produkt tih dvaju kernela.

Separabilne konvolucije mogu se podijeliti na prostorno separabilne i dubinski separabilne. Prostorno separabilne konvolucije ime su dobile radi činjenice da se bavi prostornim dimenzijama slike i kernela: širinom i visinom. Jedna od poznatih prostorno separabilnih konvolucija je Sobel kernel, korišten za detekciju rubova:

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \times \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \tag{5.4}$$

Dubinski separabilne konvolucije pored prostornih dimenzija se bave i dimenzijom

dubine, tj. brojem kanala što bi u slučaju RGB slike bilo 3. Kao i prostorne, dijele kernel u dva odvojena kernela koji onda obavljaju dvije konvolucije: dubinsku i točkastu.

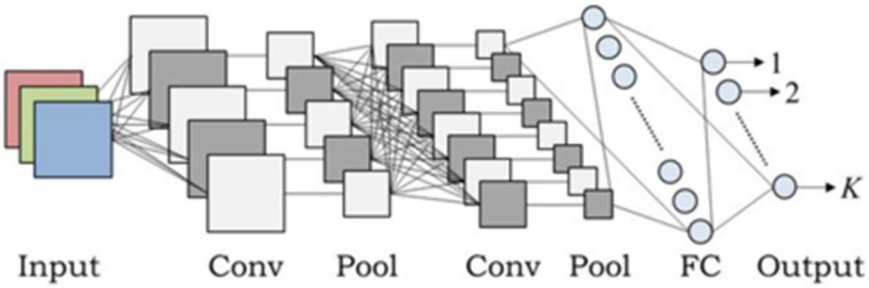
Povećanje efikasnosti zahvaljujući separabilnosti bitno je svojstvo (pojedinih) kernela te je potrebno utvrditi da li je neki kernel separabilan. Direktna metoda je promatrati 2D kernel kao 2D matricu, uzeti njenu dekompoziciju singularnih vrijednosti $K = \sum_i \sigma_i u_i v_i^T$. Kernel je separabilan ukoliko je prva singularna vrijednost σ_0 različita od 0 i u tom slučaju $\sqrt{\sigma_0} u_0$ i $\sqrt{\sigma_0} v_0^T$ predstavljaju vertikalni i horizontalni kernel rastava [92].

5.3. Konvolucijske neuronske mreže

Pokazalo se da klasične neuronske mreže ne daju dobre rezultate u slučaju kada su ulazni podaci slike. Predloženi model konvolucijskih neuronskih mreža (engl. *Convolutional Neural Networks - CNN*) se upotrebom odgovarajućih filtera uspješno prilagođava prostornim i vremenskim ovisnostima slikovnih (video) podataka. Stoga je razvoj konvolucijskih mreža doveo je do velikog napretka u rješavanju brojnih problema područja računalnog vida premda polako pronalaze primjenu i u drugim područjima. Klasifikacija slika, lokalizacija i detekcija objekata neke su od zadaća vizualnog raspoznavanja koje nalaze primjenu u brojnim disciplinama, pogotovo u medicini, robotici, upravljanju i nadzoru.
Konvolucijske mreže organizirane su u više slojeva pri čemu je uobičajeno da su prvi slojevi odgovorni za određivanje značajki niskog nivoa, poput rubova. U idućim slojevima obrađuju se značajke višeg nivoa što u konačnici dovodi do potpunog razumijevanja slike.
Dijelovi konvolucijske mreže su:

- Ulazni sloj koji predstavlja ulaznu sliku
- Konvolucijski sloj, koji se najčešće sastoji od dva dijela, u prvom dijelu na prethodni sloj djeluje s unaprijed određenim brojem filtera namjenjenih prepoznavanju točno određenih značajki; nakon toga, u drugom dijelu, na rezultat prvog dijela se djeluje neka aktivacijska funkcija (najučinkovitijom se pokazala ReLU funkcija, više u [91])
- Sloj sažimanja (engl. *Pooling*) koji djeluje na prostorne dimenzije aktivacijske mape, pri čemu se ne mijenja dubina aktivacijske mape. Na taj način je izvršena redukcija podataka. Razlikuju se maksimalno i prosječno sažimanje. Maksimalno sažimanje pored redukcije podataka služi i kao mehanizam za smanjenje šuma. Konvolucijski sloj i sloj sažimanja zajedno čine *i*-ti sloj konvolucijske mreže.
- Potpuno povezani (engl. *Fully connected*) slojevi kod kojih su neuroni između dva susjedna sloja svi međusobno povezani; kao izlaz daju vektor rezultata.

Primjer CNN arhitekture s više konvolucijskih slojeva dan je na slici 5.6.



Slika 5.6. Primjer CNN arhitekture [93]

Hiperparametri modela koje je potrebno definirati u svakom konvolucijskom sloju su:

- broj filtera
- veličina filtera: definira se prostornim dimenzijama filtera, uglavnom se koriste kvadratni filteri malih dimenzija.
- dopunjavanje (engl. *Padding*) je nadopunjavanje originalne mape značajki nulama oko rubova.
- korak (engl. *Stride*) predstavlja broji piksela za koji se pomiče horizontalno i vertikalno pri konvoluciji

Dostupnost velikih količina podataka, te razvoj hardvera omogućili su razvoj CNN arhitektura koje postižu odlične rezultate. Pregled danas najpoznatijih korištenih arhitektura, poput AlexNet, Le-Net5 dan je u [94].

6. NASTAVAK ISTRAŽIVANJA

Temeljna motivacija za daljnje istraživanje je potencijalna primjena redukcije dimenzionalosti korištenjem navedenih tehnologija u obradi podataka iz događaja u CMS detektoru.

Dosada korišteni algoritmi za detekciju i praćenje čestica pokazali su se učinkovitima i s prihvatljivom latencijom. Međutim, obzirom na (očekivano) ekstremno povećanje volumena podataka koje će detektori čestica morati analizirati (i isfiltrirati) da bi utvrdili postojanje događaja od interesa u HL-LHC fazi projekta, potrebno je pored senzorske tehnologije unaprijediti i korištenu metodologiju.

Metode strojnog učenja korištene su u različite svrhe pri obradi podataka u LHC-u (npr. za rekonstrukciju sirovih podataka detektora u fizikalne objekte), a vrlo efikasnim pokazali su se sofisticirani algoritmi strojnog učenja korišteni u okidačima. Mnoge nove publikacije vezane za CMS/LHC baziraju se na primjeni tehnika strojnog učenja dok je pod okriljem CERN-a pokrenut projekt HEP.TrkX koji istražuje mogućnosti primjene metoda strojnog učenja na HL-LHC rekonstrukciju staza [95].

Sličnost s problemima segmentacije i označavanja slika te praćenjem objekata, koji su uspješno rješavani metodama dubokog učenja, motivirala je znanstvenike da navedene metode primjene na rješavanje nekih LHC problema.

Prvi problem u implementaciji neuronskih mreža u LHC projektu su ograničeni resursi. U idućem poglavlju dani su primjeri upotrebe neuronskih mreža u slučaju limitiranih računalnih resursa, te moguće prilagodbe. Nakon toga predstavljeni su primjeri upotrebe konvolucijskih neuronskih mreža kod rješavanja određenih problema u fizici čestica: prepoznavanje mlazova i klasifikator događaja.

6.1. Primjena neuronskih mreža na uređajima s ograničenim resursima

Premda efikasnost ANN uvelike ovisi o kapacitetu/količini dostupnih računalnih resursa, u određenim primjenama (upravo kao i u slučaju LHC senzora), ANN se moraju koristiti na uređajima s ograničenim resursima poput pametnih telefona, ili kao u našem slučaju FPGA-ova. U [96] autori su istražili implementaciju neuronskih mreža na FPGA-ovima, te ispitali korištenje resursa i latenciju za različite arhitekture NN te različiti izbor hiperparametara. Upotreba ML u hardveru korištenom za selekciju u realnom vremenu baziranom na FPGA-ovima je ograničena. Razlozi za to su u prvom redu složenost implementacije istih, te FPGA-ovi zahtjevi za resursima. Stoga je bitno minimizirati korištene modele i ubrzati postupak zaključivanja. Za izradu modela stojnog učenja na FPGA-ovima razvijen je kompajler baziran na sintezi visoke razine (engl. *High-Level-Sintesis*) zvanoj hls4ml što je omogućilo smanjenje vremena potrebnog za razvoj softvera.

Alternativni pristup prilagodbi neuronskih mreža ograničenim resursima bio bi smanjiti i ubrzati mrežu, ali bez većeg gubitka točnosti. Predložene su razne metode, poput korištenja separabilnih konvolucija, smanjenja broja parametara, te tehnike kvantizacije mreže. Binarne neuronske mreže (engl. *Binary/Binarized Neural Networks - BNN*) su nedavno definirane i predstavljene [97-101]. BNN su duboke neuronske mreže koje za

težine i aktivaciju koriste binarne vrijednosti (+1 i -1). Deterministička binarizacija težina može se izvršiti korištenjem funkcije

$$w_b = \begin{cases} +1 & ako\ je\ w \geq 0 \\ -1 & inače \end{cases} \tag{6.1}$$

pri čemu je je w_b binarizirana vrijednost težine, dok je w prava (realna) vrijednost težine. Autori su u [101] dali stohastičku verziju koja je u njihovim testiranjima pokazala bolje rezultate:

$$w_b = \begin{cases} +1 & s\ vjerojatnošću\ p = \sigma(w) \\ -1 & s\ vjerojatnošću\ 1 - p \end{cases} \tag{6.2}$$

gdje je

$$\sigma(x) = clip\left(\frac{x+1}{2}, 0{,}1\right) = \max(0, \min\left(1, \frac{x+1}{2}\right)) \tag{6.3}$$

Većina dosad ponuđenih modela BNN radi jednostavnije implementacije koristi deterističku verziju. Zahvaljujući ograničenju vrijednosti težina (i aktivacija) na +1 i -1, BNN-ovi mogu izvoditi izračune pomoću operacija na bitovima, što smanjuje vrijeme izvršavanja. Pri tom je vrijednost -1 kodirana kao 0, +1 kao 1. Veličine modela BNN-a mnogo su manje od modela NN koje rade s punom preciznošću. Također su dobri kandidati za implementaciju dubokog učenja na FPGA-ovima i (pogotovo) ASIC-ovima zbog njihove efikasnosti u izvođenju operacija na bitovima. U prvim verzijama BNN-a uočen je problem kod propagacije pogreške unazad i metode stohastičkog gradijentnog spusta. One se ne mogu izravno primijeniti jer se težine ne mogu ažurirati u malim koracima te je bilo potrebno razviti alternativne pristupe, više u [99]. Većina pristupa zahtjeva očuvanje realnih vrijednosti težina u nekoj od faza rada neuronske mreže.

Nedostatak BNN je lošija točnost u usporedbi s rezultatima dobivenim korištenjem realnih vrijednosti. U [99] dan je pregled točnosti BNN na različitim skupovima podataka.

U [100] autori su predložili tenzorizaciju korištenih filtera u BNN, te njihovu parametrizaciju uporabom matrične ili tenzorske dekompozicije. Dobiveni rezultati pokazuju poboljšanje u točnosti treniranih modela uz očuvanje prednosti koje obično nude binarne mreže: zadržana brzina izvođenja (do 58 puta) i ušteda prostora (do 32 puta).

6.2. Primjer upotrebe CNN-a u obradi LHC podataka

U LHC-u, kao rezultat raspada čestice na visokoenergetskih kvarkova i gluona, nastaju mlazovi čestica (engl. *Jet*) čija rekonstrukcija, klasifikacija i razlikovanje od pozadinskih (mlazova) čestica predstavlja temeljni izazov.

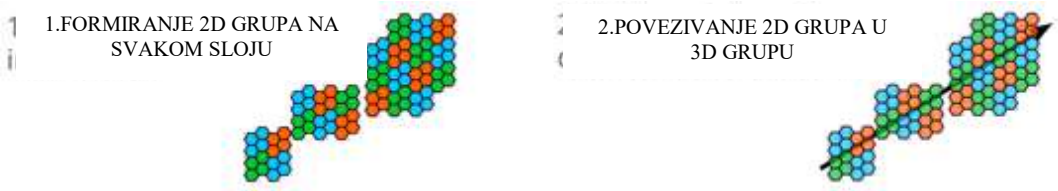
Upotreba ANN u fizici čestica ima dugu povijest, međutim tek je razvoj dubokih konvolucijskih mreža omogućio njihovu primjenu na sirovim podacima tretirajući

mlazove čestica poput slika, bez postavljanja ulaznih značajki baziranih na znanjima iz područja.
Inspirirani tehnikama računalnog vida (engl. *Computer Vision - CV*) korištenim kod prepoznavanja lica, autori su u [103] predstavili Jet Images - algoritam namijenjen prepoznavanju mlazova. Pri tom nisu izravno koristili fizičkim svojstva čestica, već se oslanjali na informacije na niskoj razini. To se odnosi na činjenicu da su mlazovi čestica prikazani kao slike s energetskim depozitima čestica unutar mlaza koje služe kao intenziteti piksela. Evolucija metode predstavljena je u [104] u kojima su autori za utvrđivanje i označavanje mlazova zajedno s tehnikama računalnog vida koristili konvolucijske neuronske mreže. U nekim istraživanjima, autori su predložili klasifikator događaja koji za ulaz koristi sirove podatke (engl. *Raw data*) CMS detektora. Detaljnije, klasifikator baziran na konvolucijskoj neuronskoj mreži testiran je na 2012 CMS Simulated Open Data [105], koji koristi najveći stupanj simulacije detektora. Pristup rekonstrukciji staze u tragaču korištenjem konvolucijskih neuronskih mreža pri HL-LHC uvjetima dali su autori u [106].

6.3. Mogući smjerovi daljnjeg istraživanja

Kako je već navedeno u poglavlju 2.1.3, još nije definirana detaljna arhitektura sustava okidača HG kalorimetra, a samim tim ni algoritam rekonstrukcije tragova čestica.
Nadograđena verzija HGCAL-a sadržavat će kalorimetar za uzimanje uzoraka (engl. *Sampling Calorimeter*) na bazi silicija visoke razlučivosti, uz upotrebu silicijskih senzorskih modula koji omogućuju finu segmentaciju na detektorskoj ravnini. Visoka granularnost smanjit će broj pogrešno identificiranih tragova čestica. Da bi se postiglo što
efikasnije prekrivanje detektorskog sloja koriste se senzorski moduli nastali grupiranjem heksagonalnih senzorskih ćelija. Kao uvod u temu doktorata, obavljeno je preliminarno istraživanje o mogućnostima i efikasnosti prekrivanja detektorske ravnine senzorskim modulima različitih oblika [107,108].

Postojeće arhitekture za detekciju zanimljivih regija u detektoru temeljene su na obradi u dvije dimenzije kako je prikazano na slici 6.1. Senzorske ćelije s najvećim energijama su grupirane unutar svakog sloja čime nastaju 2D klasteri (engl. *Clusters*) korišteni za izračunavanje centroidne pozicije i informacije o energiji na svakom sloju. Upotrebljene su dvije temeljne koordinate podataka na sloju, a zatim se rezultati 2D obrade povezuju rekonstruiranjem treće dimenzije [109]. 3D klasteri mogu se klasificirati primjerice kao EM kandidati za elektron ili foton.



Slika 6.1. Postojeća strategija za detektiranje zanimljivih regija [109]

Potencijalna strategija primjene tenzora u nadograđenom okidaču temelji se na korištenju čitavog volumena podataka u detektoru, odnosno istraživanje mogućnosti direktne 3D obrade i redukcije podataka. Naime, kako je već navedeno u radu, kombiniranjem podataka iz identičnih senzorskih podpolja na prirodan način se

generiraju tenzori. U našem slučaju, podpolja za obradu bili bi slojevi detektora. Oni predstavljaju tenzorske odsječke.

Jedna strategija zasnivala bi se na korištenju cjelokupnih senzorskih podataka, bez ikakve prethodne selekcije na bazi najvećih energija, znači tenzor bi sadržavao vrijednosti svih očitanih energija, pozicije senzorskih ćelija predstavljale bi prva dva moda dok bi treći mod bio redni broj sloja u tenzoru. Nakon toga izvršili bi redukciju podataka aproksimirajući početni tenzor s tenzorom nižeg ranga.

Druga moguća strategija bila bi prepoznavanje 3D oblika (karakterističnog za oblik rasapa neke čestice). Ukoliko se tenzor definira na prethodni način ali umjesto vrijednosti deponiranih energija stavi vrijednost 1 ili 0 u ovisnosti o tome da li je senzorska ćelija detektirala depozit energije ili ne, dobije se binarni tenzor. Kako se svaki 3D oblik može također prikazati kao binarni tenzor, potrebno je utvrditi da li postoji „poklapanje“ između tenzora senzorskih podataka i tenzora oblika.

Još jedan potencijalni smjer za daljnje istraživanje jest mogućnost razvoja konvolucijske/binarne umjetne neuronske mreže koja prepoznaje zanimljive 3D regije. Naglasak može biti na realizaciji mreže postupkom tenzorske dekompozicije kako bi se olakšala višedimenzionalna obrada i reducirala dimenzionalnost. U prethodnom poglavlju pokazano je da je primjena umjetne inteligencije postala jako popularna na CERN-u zadnjih godina te su dani primjeri raznih 2D modela klasifikacije koji su se istraživali/primjenjivali. Postavlja se pitanje da li se neki od tih modela može proširiti na 3D problem analize te kako razviti model HGCAL 3D događaja.

7. ZAKLJUČAK

Senzorske mreže generiraju velike količine kompleksnih podataka čija obrada i pohrana predstavljaju veliki izazov. Izvorni podaci često sadrže i nepotrebne podatke (šumove, redundantne podatke). Uz pomoć automatiziranih metoda analize, moguće je utvrditi važne veze među podacima, i uz pomoć toga reducirati veličinu i dimenzionalnost generiranih podataka. Razvijene su brojne metode redukcije a u radu je dan pristup redukciji dimenzionalnosti baziran na matičnim i tenzorskim dekompozicijama koje su postale standardan alat za smanjenje dimenzionalnosti. Dan je pregled najpopularnijih metoda dekompozicije, te su navedeni primjeri uporabe.

Podaci dobiveni očitavanjem senzora LHC eksperimenta dani su kao primjer generiranja i obrade velike količine podataka. U fazi HL-LHC eksperimenta kao rezultat povećanja luminoziteta i poboljšanja korištene tehnologije očekuje se ekstremno povećanje volumena generiranih podataka. Uvjeti pod kojima se odvija eksperiment, brzina priljeva podataka te ograničeni računalni resursi dostupni primarnoj selekciji podataka dodatno otežavaju njihovo procesiranje. Stoga njihovo obrada zahtjeva interdisciplinarni pristup koji uključuje primjenu najnovijih metoda.

Metode strojnog učenja, posebno neuronske mreže, već se koriste u različitim fazama eksperimenta. Premda su visoki računalni zahtjevi koje imaju neuronske mreže u neskladu su s ograničenim raspoloživim resursima, dani su primjeri implementacije neuronskih mreža na uređajima s ograničenim resursima (FPGA) što ukazuje na mogućnost primjene istih u obradi HL-LHC podataka. Binarne neuronske mreže koriste samo vrijednosti 1 i -1, te stoga obradu izvode pomoću operacija na bitovima. Efikasnost u izvođenju operacija, veličina modela i mogućnost jednostavne implementacije na FPGA-ovima i ASIC-ovima čine ih bitnim kandidatima za primjenu u obradi HL-LHC senzorskih podataka.

Matrične i tenzorske dekompozicije predstavljene u prvom dijelu rada moguće je koristiti na više načina: 1. u predobradi podataka 2. kao alat za ubrzanje neuronskih mreža. Primjena tenzorskog računa u okidaču omogućila bi korištenje čitavog volumena podataka u detektoru, odnosno istraživanje mogućnosti direktne 3D obrade i redukcije podataka.

Neuronske mreže, posebno binarne, u kombinaciji s tenzorskom tehnologijom predstavljaju potencijalni alat za direktno istraživanje 3D senzorskih podataka.

REFERENCE

[1] D. Sindol (2013): “Big Data Basics - Part 1 - Introduction to Big Data“, “, [on-line materijal], preuzeto u travnju 2019, link: www.mssqltips.com/sqlservertip/3132/big-data-basics--part-1--introduction-to-big-data/

[2] D. Laney (2001): “Application Delivery Strategies“, [on-line materijal], preuzeto u travnju 2019, link: blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf

[3] CERN službena web-stranica:“LHC: pushing computing to the limits“, [on-line materijal], preuzeto u lipnju 2019, link: home.cern/news/news/computing/lhc-pushing-computing-limits

[4] CERN službena web-stranica: Christiane Lefèvre (2008): “The CERN accelerator complex“, [on-line materijal], preuzeto u lipnju 2019, link: cds.cern.ch/record/1260465

[5] CERN službena web-stranica: Maria Chamizo Llatas, Lucas Taylor: “New world record - first pp collisions at 8 TeV“, [on-line materijal], preuzeto u svibnju 2019, link: cms.web.cern.ch/news/new-world-record-first-pp-collisions-8-tev

[6] CERN službena web-stranica: “About CMS“, [on-line materijal], preuzeto u svibnju 2019, link: cms.cern/detector

[7] Manfred Jeitler (2008): “Trigger systems at LHC experiments“, Institute of High Energy Physics of the Austrian Academy of Sciences.

[8] „The CMS experiment at the CERN LHC“, the CMS Collaboration at al 2008 JINST 3 S08004

[9] HL-LHC Industry, Project schedule, [on-line materijal], preuzeto u kolovozu 2019, link: project-hl-lhc-industry.web.cern.ch/content/project-schedule

[10] Thorben Quast: “CMS High-Granularity Calorimeter Upgrade HGCAL “, [on-line materijal], preuzeto u svibnju 2019, link: twiki.cern.ch/twiki/pub/CLIC/TerascaleDW19/CMS_HGCal_TerascaleWS2019_final.pdf

[11] Krzewicki M.: HL-LHC Computing, [on-line materijal], preuzeto u kolovozu 2019, link:indico.cern.ch/event/315626/contributions/729452/attachments/605642/833484/ECFA-HLLHC-Aix-Les-Bains-Computing-Krzewicki.pdf

[12] Jean-Baptiste Sauvan (2016): “Concepts and design of the CMS high granularity calorimeter Level-1 trigger“, CERN, CH-1211 Geneva 23, Switzerland

[13] “Curse of dimensionality“, [on-line materijal], preuzeto u svibnju 2019, link: en.wikipedia.org/wiki/Curse_of_dimensionality

[14] Beyer K., Goldstein J., Ramakrishnan R., Shaft U. (1999) When Is “Nearest Neighbor“ Meaningful?. In: Beeri C., Buneman P. (eds) Database Theory — ICDT’99. ICDT 1999. Lecture Notes in Computer Science, vol 1540. Springer, Berlin, Heidelberg

[15] Miguel A. Carreira-Perpinan (1997): “A Review of Dimension Reduction Techniques“, Dept. of Computer Science University of Sheffield

[16] Kezić,I. (2005): “Analiza i implementacija algoritma za smanjenje dimenzionalnosti dekompozicijom na singularne vrijednosti“, Fakultet Elektrotehnike i Računarstva, Sveučilište u Zagrebu

[17] C.O.S. Sorzano, J. Vargas, A. Pascual Montano: “A survey of dimensionality reduction techniques“, Natl. Centre for Biotechnology (CSIC) Campus Univ. Autónoma, Madrid, Spain

[18] Ivančević Z.,Jelić S. (2009): “Primjena SVD rastava matrice u dohvatu informacija i kompresiji slike“, Odjel za matematiku, Sveučilište Josipa Jurja Strossmayera u Osijeku

[19] Zirdum I. (2017): “Prepoznavanje lica pomoću tenzorske dekompozicije singularnih vrijednosti“, Matematički odsjek , Prirodoslovno–matematički fakultet, Sveučilište u Zagrebu

[20] Harry C. Andrews, C. Patterson III (1976): “Singular Value Decomposition (SVD) Image Coding“. Communications, IEEE Transactions on. 24. 425 - 432. 10.1109/TCOM.1976.1093309.

[21] H.S. Prasantha, H.L. Shashidhara, Balasubramanya Murthy, Kannamedi (2008): “Image compression using SVD“. 143-145. 10.1109/ICCIMA.2007.386.

[22] V. Santhi, Thangavelu, Arunkumar (2009): “DWT – SVD Combined Full Band Watermarking Technique for Color Images in YUV color Space“, International Journal of Computer Theory and Engineering. 1. 10.7763/IJCTE.2009.V1.68.

[23] Rajab Lama, Khatib Tahani, Al-Haj Ali (2009): “Video Watermarking Algorithms Using the SVD Transform“, European Journal of Scientific Research ISSN. 30. 1450-216.

[24] Herve Abdi, Lynne J. Williams (2010): “Principal component analysis“, John Wiley & Sons, Inc. WIREs Comp Stat 2010 2 433–459

[25] Jillur Quddus (2019): “Principal Component Analysis—Unsupervised Learning Model“

[26] Mudrová, M & Procházka, Aleš. (2019). “Principal component analysis in image processing“

[27] S. C. Ng (2016): “Principal Component Analysis to Reduce Dimension on Digital Image“, School of Information Technology, SEGi University, Malaysia, 8th International Conference on Advances in Information Technology, IAIT2016, 19-22 December 2016, Macau, China

[28] Wilmar Hernandez, Alfredo Mendez (2018): “Application of Principal Component Analysis to Image Compression“

[29] J. Ashok, Dr. E. G. Rajan (2010): “Principal Component Analysis Based Image Recognition“

[30] Matthew A. Turk, Alex P. Pentland: “Face Recognition Using Eigenfaces, Vision and Modeling Group“, The Media Laboratory Massachusetts Institute of Technology

[31] Jian Yang, D. Zhang, A. F. Frangi and Jing-yu Yang (2004): “Two-dimensional PCA: a new approach to appearance-based face representation and recognition," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, no. 1, pp.131-137, doi: 10.1109/TPAMI.2004.1261097

[32] Alaa Tharwat (2018): “Independent component analysis: An introduction“, Faculty of Computer Science and Engineering, Frankfurt University of Applied Sciences

[33] Pierre Comon (2003): “Independent component analysis, A new concept?“, Thomson-Sintra, Parc Sophia Antipolis, BP 138, F-06561 Valbonne Cedex, France

[34] Ganesh R. Naik, Dinesh K. Kumar (2009): “An Overview of Independent Component Analysis and Its Applications“, School of Electrical and Computer Engineering RMIT University, Australia

[35] Joel A. Tropp: Literature Survey: “Non-Negative Matrix Factorization“, Institute for Computational Engineering and Sciences, The University of Texas at Austin

[36] Ngoc-Diep Ho (2008): “Nonnegative matrix factorization algorithms and applications“, Université Catholique De Louvain, École Polytechnique De Louvain, Département D’ingénierie Mathématique

[37] Mujahid N. Syed (2017): “Big Data Blind Separation“, Department of Systems Engineering, King Fahd University of Petroleum & Minerals

[38] Pando Georgiev, Fabian Theis, Andrzej Cichocki , and Hovagim Bakardjian: “Sparse Component Analysis: A New Tool for Data Mining“

[39] Xuyang Lu, Qijia Jiang, Boying Meng (2013): “Comparison of Three Different Matrix Factorization Techniques for Unsupervised Machine Learning“,

[40] Menaka Rajapakse, Lnnce Wyse: “NMF vs ICA for Face Recognition“, Institute for Infncomm Research Singapore

[41] N. F. Chikhi, B. Rothenburger, N. Aussenac-Gilles (2007): “A Comparison of Dimensionality Reduction Techniques for Web Structure Mining“, IEEE/WIC/ACM International Conference on Web Intelligence (WI'07), Fremont, CA

[42] M. S. Bartlett, J. R. Movellan, T. J. Sejnowski (2002): “Face recognition by independent component analysis“, in IEEE Transactions on Neural Networks, vol. 13, no. 6

[43] Stephan Rabanser, Oleksandr Shchur, Stephan Günnemann (2017): “Introduction to Tensor Decompositions and their Applications in Machine Learning“, Department of Informatics Technical University of Munich

[44] Tamara G. Kolda, Brett W. Bader (2009): “Tensor Decompositions and Applications“, Sandia National Laboratories

[45] Joseph B Kruskal (1983): “Statement of some current results about three-way arrays.“

[46] Johan Håstad (1989): “Tensor rank is NP-complete“, Royal Institute of Technology, Stockholm 70, Sweden

[47] Brett W. Bader, Tamara G. Kolda and others (2015): MATLAB Tensor Toolbox Version 2.6

[48] De Lathauwer, Lieven & De Moor, Bart. (2000): “A Multi-Linear Singular Value Decomposition“, Society for Industrial and Applied Mathematics. 21. 1253-1278. 10.1137/S0895479896305696.

[49] Pierre Comon (2014): “Tensors: a Brief Introduction“, IEEE Signal Processing Magazine, Institute of Electrical and Electronics Engineers,

[50] Andrzej Cichocki, Danilo P. Mandic, Anh Huy Phan, Cesar F. Caiafa, Guoxu Zhou, Qibin Zhao, Lieven De Lathauwer (2015): “Tensor Decompositions for Signal Processing Applications“

[51] L. de Lathauwer, Bart De Moor, Joss Vandewalle (2000): “On the best rank-1 and rank-(Rl, R2,..., Rn) approximation of higher order tensors. “ SIAM J. Matrix Anal. Appl.. 21. 132-1342.

[52] Lek-Heng Lim: “Singular values and eigenvalues of tensors: a variational approach“, Stanford University Institute for Computational and Mathematical Engineering

[53] Frank L. Hitchcock: “The expression of a tensor or a polyadic as a sum of products“

[54] Cattell, R.B. Psychometrika (1944) 9: 267. [on-line materijal], preuzeto u travnju 2019, link:doi.org/10.1007/BF02288739

[55] Tucker, L.R. Psychometrika (1966) 31: 279. [on-line materijal], preuzeto u travnju 2019, link:doi.org/10.1007/BF02289464

[56] Tucker, L. R., Harris, C. W. (1963): "Implications of factor analysis of three way matrices for measurements of change: Problems in measuring change", Madison University of Wisconsin Press.

[57] C. J. Appellof, E. R. Davidson (1981): "Strategies for analyzing data from video fluorometric monitoring of liquid chromatographic effluents", Department of Chemistry, University of Washington.

[58] Alex Vasilescu: "What is all the current fuss about data tensor decomposition and analysis? Data analysis and historical perspective", [on-line materijal], preuzeto u travnju 2019, link: www.quora.com/What-is-all-the-current-fuss-about-data-tensor-decomposition-and-analysis-Data-analysis-and-historical-perspective

[59] L De Lathauwer: "Signal processing based on multilinear algebra", Ph.D. dissertation, KU Leuven, Leuven, 1997.

[60] Karypis Lab, "SPLATT - Parallel Sparse Tensor Decomposition", [software], preuzeto u lipnju 2019, link: glaros.dtc.umn.edu/gkhome/splatt/overview

[61] GitHub, "Cadabra2", [on-line materijal], preuzeto u svibnju 2019, link: github.com/kpeeters/cadabra2

[62] Vervliet N., Debals O., Sorber L., Van Barel M. and De Lathauwer L. (2016): "Tensorlab 3.0", [on-line materijal], preuzeto u travnju 2019, link: www.tensorlab.net

[63] M. Alex O. Vasilescu, Demetri Terzopoulos (2002): "Multilinear Analysis of Image Ensembles: Tensor Faces", Courant Institute, New York University, USA Department of Computer Science, University of Toronto, Canada

[64] Lieven De Lathauwer, Joos Vandewalle: "Dimensionality Reduction in ICA and Rank-(R1, R2, . . . , RN) Reduction in Multilinear Algebra"

[65] Ante Jukić: "Dekompozicije tenzora i primjena u izdvajanju značajki", Zavod za laserska i atomska istraživanja i razvoj, Institut Ruđer Bošković

[66] Evrim Acar, Bulent Yener (2008): "Unsupervised Multiway Data Analysis_ A Literature Survey", Computer Science Department, Rensselaer Polytechnic Institute, Troy, NY, USA

[67] Harshman, R. A. (1972): "PARAFAC2: Mathematical and technical notes", UCLA Working Papers in Phonetics, 22, 30-44. (University Microfilms, Ann Arbor, Michigan, No. 10,085).

[68] Harshman, R. A., Hong, S., Lundy, M.E. (2003): “Shifted factor analysis - Part I: Models and properties. “

[69] M. Mørup and M. N. Schmidt: “Sparse non-negative tensor 2d deconvolution (SNTF2D) for multichannel time-frequency “

[70] R. Bro, R. A. Harshman, N. D. Sidiropoulos (2009): “Modeling multi-way data with linearly dependent loadings. “

[71] A. Phan, P. Tichavský, A. Cichocki (2013): “CANDECOMP/PARAFAC Decomposition of High-Order Tensors Through Tensor Reshaping“, in IEEE Transactions on Signal Processing, vol. 61, no. 19, pp. 4847-4860

[72] Giorgio Tomasi, Rasmus Bro (2006): “A comparison of algorithms for fitting the PARAFAC model“, University of Copenhagen

[73] Alex Pereira Da Silva (2016): “Tensor techniques for signal processing: algorithms for Canonical Polyadic decomposition“, Signal and Image processing, Université Grenoble Alpes

[74] P. Comon, C. Jutten (2010): “Handbook of blind source separation, independent component analysis and applications“. Academic Press, Oxford UK, Burlington USA.

[75] Muti D., Bourennane S. (2007): “Survey on tensor signal algebraic filtering“, Signal Processing, 87(2), 237–249.doi:10.1016/j.sigpro.2005.12.016

[76] A. L. F. de Almeida, G. Favier and J. C. M. Mota (2007): “Parafac-based unified tensor modeling for wireless communication systems with application to blind multiuser equaliza-tion“, Signal Processing, vol. 87, no. 2, pages 337–351,

[77] Souleymen Sahnoun, Pierre Comon (2015): “Joint Source Estimation and Localization“, IEEE Transactions on Signal Processing, Institute of Electrical and Electronics Engineers, 63 (10), pp.2485-2495.

[78] B. Savas (2008): “Algorithms in Data Mining using Matrix and Tensor Methods“, PhD thesis, Linköping Univ. Tech.,

[79] Junli Liang, Yang He, Ding Liu, & Xianju Zeng. (2012): “Image Fusion Using Higher Order Singular Value Decomposition“, IEEE Transactions on Image Processing, 21(5), 2898–2909.doi:10.1109/tip.2012.2183140,

[80] Oseledets Ivan (2011): “Tensor-Train Decomposition“, SIAM J. Scientific Computing. 33. 2295-2317. 10.1137/090752286.

[81] Vin De Silva1, Lek-Heng Lim: “Tensor rank and the ill-posedness of the best low-rank approximation problem“

[82] Novikov, A., Podoprikhin, D., Osokin, A., & Vetrov, D.P. (2015): “Tensorizing Neural Networks“, NIPS.

[83] Lu Haiping, Plataniotis Konstantinos, N. Venetsanopoulos Anastasios (2006): "Multilinear Principal Component Analysis of Tensor Objects for Recognition", Proceedings - International Conference on Pattern Recognition. 2. 776-779. 10.1109/ICPR.2006.837.

[84] M. Alex O. Vasilescu, Demetri Terzopoulos (2005): "Multilinear Independent Components Analysis"

[85] Tamara G. Kolda (2001): "Orthogonal tensor decompositions", Computational Science and Mathematics Research Department, Sandia National Laboratories

[86] Prof. dr. sc. Sven Lončarić: "Neuronske mreže: Uvod", Fakultet elektrotehnike i računarstva

[87] ZENVA: Mohit Deshpande (2017): "Perceptrons: The First Neural Networks", preuzeto u srpnju 2019, link:pythonmachinelearning.pro/perceptrons-the-first-neural-networks/

[88] UFLDL Tutorial, [on-line materijal], preuzeto u lipnju 2019, link: ufldl.stanford.edu/tutorial/

[89] K. Bai (2019): "A Comprehensive Introduction to Different Types of Convolutions in Deep Learning" [on-line materijal], preuzeto u lipnju 2019, link: towardsdatascience.com/a-comprehensive-introduction-to-different-types-of-convolutions-in-deep-learning-669281e58215

[90] I. Goodfellow, Y. Bengio i A. Courville: "Deep Learning", MIT Press, 2016

[91] Wu, J. (2017): "Introduction to Convolutional Neural Networks"

[92] R. Szeliski (2010): "Computer Vision: Algorithms and Applications" ,draft c 2010 Springer

[93] A. Hidaka, T. Kurita (2017): "Consecutive Dimensionality Reduction by Canonical Correlation Analysis for Visualization of Convolutional Neural Networks", DOI: 10.5687/sss.2017.160, Proceedings of the ISCIE International Symposium on Stochastic Systems Theory and its Applications

[94] R. Karim (2019): "Illustrated: 10 CNN Architectures", [on-line materijal], preuzeto u lipnju 2019, link: towardsdatascience.com/illustrated-10-cnn-architectures-95d78ace614d

[95] Paolo Calafiura i dr. (2016): "HEP advanced tracking algorithms with cross-cutting applications" (Project HEP.TrkX)

[96] Duarte Javier i dr. (2018): "Fast inference of deep neural networks in FPGAs for particle physics"

[97] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Binarized neural networks" in Advances in Neural Information Processing Systems 29(D. D. Lee, M. Sugiyama, U. V.Luxburg, I. Guyon, and R. Garnett, eds.), pp. 4107–4115, Curran Associates, Inc., 2016.

[98] Tadej Murovič, Andrej Trost (2019): "Massively parallel combinational binary neural networks for edge processing", Elektrotehniski Vestnik/Electrotechnical Review. 86. 47-53.

[99] Simons T., Lee D.:"A Review of Binarized Neural Networks", Electronics 2019, 8(6), 661; doi.org/10.3390/electronics8060661

[100] Adrian Bulat, Jean Kossaifi, Georgios Tzimiropoulos, Maja Pantic (2019): "Matrix and tensor decompositions for training binary neural networks", Samsung AI Center, Cambridge, United Kingdom

[101] M. Courbariaux, Y. Bengio, and J.-P. David. Binaryconnect: "Training deep neural networks with binary weights during propagations", In NIPS, 2015.

[102] A. Bulat and G. Tzimiropoulos: "Binarized convolutional landmark localizers for human pose estimation and face alignment with limited resources", In ICCV, 2017.

[103] J. Cogan, M. Kagan, E. Strauss and A. Schwarztman: "Jet-Images: Computer Vision Inspired Techniques for Jet Tagging", JHEP 02 (2015) 118 [1407.5675].

[104] Luke de Oliveira i dr. (2017): "Jet-Images – Deep Learning Edition", [on-line materijal], preuzeto u lipnju 2019, link: doi.org/10.1007/JHEP07(2016)069

[105] CMS collaboration (2018): 2018 CMS data preservation, re-use and open access policy, CERN Open Data Portal

[106] Adriano Di Florio1, Felice Pantaleo, Antonio Carta: "Convolutional Neural Network for Track Seed Filtering at the CMS High-Level Trigger"

[107] Prvan M,. Ožegović, J., Burazin Mišura. A. (2019): "On Calculating the Packing Efficiency for Embedding Hexagonal and Dodecagonal Sensors in a Circular Container", Mathematical problems in engineering, 2019, 1-16 doi:10.1155/2019/9624751

[108] Prvan M,. Ožegović, J., Burazin Mišura. A. (2019): "A Review of Embedding Hexagonal Cells in the Circular and Hexagonal Region of Interest", . International Journal of Advanced Computer Science and Applications. 10. 10.14569/IJACSA.2019.0100747.

[109] Lindsey Gray: "Concepts and Design of the CMS High Granularity Calorimeter Level 1 Trigger", [poster], preuzeto u lipnju 2019, link: indico.cern.ch/event/432527/contributions/1071750/attachments/1322473/1983846/Poster-HGCalTrigger.pdf

8. POPIS OZNAKA I KRATICA

ALICE	A Large Ion Collider Experiment
ALS	Alternating Least Squares
ANN	Artificial Neural Network
ASIC	Application-Specific Integrated Circuit
ATLAS	A Toroidal LHC Apparatus
BNN	Binary Neural Networks
BSS	Blind Source Separation
CASTOR	CERN Advanced Storage system
CERN	Conseil européen pour la recherche nucléaire
CMS	Compact Muon Solenoid
CNN	Convolutional Neural Network
CP	Canonical Polyadic
CV	Computer Vision
CANDECOMP	Canonical Decomposition
DAQ	Data AcQuisition
DOA	Direction Of Arrival
ECAL	Electromagnetic Calorimetar
engl.	engleski
FPGA	Field Programmable Gate Arrays
HCAL	Hadronic Calorimetar
HGCAL	High Granularity Calorimeter
HLT	High Level Trigger
HL-LHC	High Luminosity LHC
HOSVD	Higher-order SVD
ICA	Independent Component Analysis
LHC	Large Hadron Collider
LHCb	Large Hadron Collider beauty
ML	Machine Learning
NLPCA	Nonlinear Principal Component Analysis
NMF	Nonnegative Matrix Factorization
PARAFAC	Parallel Factor Decomposition
PARALIND	Parallel Factors with Linear Dependency
PCA	Principal component analysis
PU	Pile Up
RNN	Recurrent neural networks
ROI	Region Of Interest
SCA	Sparse Component Analysis
SVD	Singular Value Decomposition
TLU	Threshold Logic Unit
TP	Trigger Primitive
TT	Tensor Train
WLCF	The Worldwide LHC Computing Grid