CSRNet: Dilatovane konvolucione mreže za procenu gustine mase

Seminarski iz predmeta Računarska inteligencija

Autori: Stojanović Mateja, Magenhajm Petar
Mentor: Stefan Kapunac

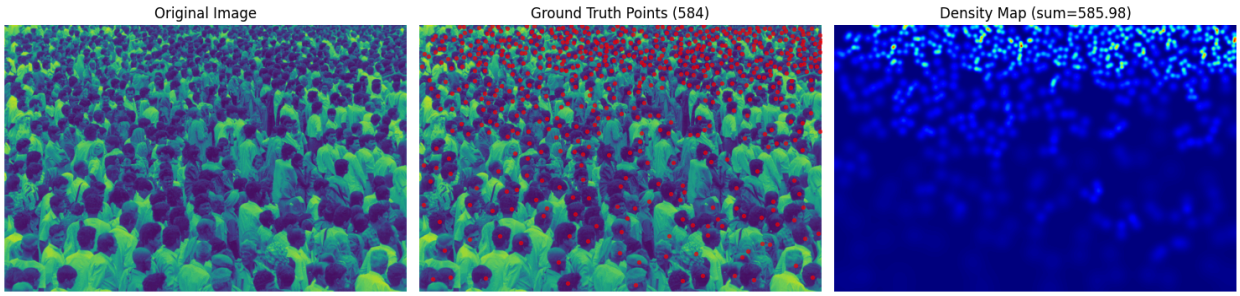
Uvod: **3**
Priprema podataka **3**
Arhitektura Mreže **5**
Front-end (Feature extractor) 5
Back-end (Dilatovane konvolucije) 6
Matematički opis dilatovane konvolucije 6
Struktura back-end dela 7
Treniranje mreže **8**
Rezultati **9**
Ostali pokušaji **10**

Uvod:

U ovom radu predstavljamo proces konstrukcije, treniranja i valuacije, Konvolutivne Neuronske neze CSRNET. Ova neuronska mreža je specijalizovana za prebrojavanje ljudi na nekoj fotografiji. U stvari ova mreza ne radi direktno to vec procenjuje gustinu same mase pa na osnovu toga prebrojavamo ljude. U narednom odeljku ćemo proći kroy proces pripreme podataka.

Priprema podataka:

Koristićemo skup podataka [ShangaiTech Dataset] , koji sadrži A i B podskupove A su raspršenije slike dok su u B gušće, koriste se drugačiji parametari za pripremu podataka za svaki od skupova. Pored samih slika dobili smo i matricu koja odgovara dimenzijama slike gde su istačkane glave pojedinaca (gde se nalazi glava na slici u tu koordinatu je upisano 1 u matricu). Kasnije se kroz za svaku obeleženu tačke u matrici prolazi Gausovim filterom. Šta u stvari radi Gausov filter? Najbolje je videti vizuelno.



Na prvoj slici vidimo originalnu sliku. Na drugoj vidimo originalnu sliku sa istackanim glavama, to je ulazna matrica. Treca slika predstavlja rezultat matrice nakon Gausovog filtera. Vidimo kako je svaka od tačaka zamućena na okolne piksele. Kako određujemo na koliko okolnih piksela ćemo “razliti” naš piksel ćemo u nastavku.

Gausov filter ima dva parametara, kernel size i standardnu devijaciju. Sigma smo određivali na osnovu distanci 3 najbliže komšije. Za ovosmo koristili strukturu KDTree. Sumu tih distance smo pomnožili sa 0,1. Na osnovu njega smo kernel size odredili formulom `kernel_size = 2 * round(3.14 * sigma) + 1` do ovog zaključka je autor originalnog rada došao ekspirementalno. Za skup B je malo drugačije kako su podaci gušći koristimo fiksno sigma koje ima vrednost 15. Ovakva priprema podataka na neki način diskretnu ciljnu promenljivu pretvara u kontinualnu, što neuronskoj mreži omogućava efikasnije treniranje. Rezultujuća mapa gustine predstavljaće našu ciljnu promenljivu i njen zbir će u stvari predstavljati broj ljudi prisutan na slici.

Arhitektura Mreže:

CSRNet (Crowd Spatial Regression Network) predstavlja konvolucionu neuronsku mrežu dizajniranu specifično za problem **procene gustine gužve**. Njena arhitektura se sastoji iz dva glavna dela: **front-end modula** za ekstrakciju karakteristika i **back-end modula** koji koristi **dilatovane konvolucije** radi povećanja receptive field-a bez gubitka prostorne rezolucije. Ovakav dizajn omogućava da mreža uči kako da mapira složene obrasce rasporeda ljudi u slici na gustinsku mapu koja precizno odražava lokalnu gustinu populacije.

Front-end (Feature extractor)

Front-end CSRNet-a baziran je na **VGG-16** arhitekturi, pretreniranom modelu na ImageNet skupu podataka [Simonyan & Zisserman, 2014]. Od originalnog VGG-16 modela koristi se samo prvih deset konvolucionih slojeva (blokovi conv1 do conv4), dok se potpuno povezani slojevi i poslednji max-pooling sloj izostavljaju.

Razlog za ovaj izbor je:

1. **Transfer učenja (transfer learning):** VGG-16, kao model treniran na velikom broju prirodnih slika, uči univerzalne vizuelne osobine (ivice, teksture, oblike) koje su korisne i za zadatak brojanja ljudi.

2. **Očuvanje prostorne informacije:** Uklanjanjem dubokih slojeva i poslednjih pooling operacija, zadržava se prostorna struktura scene, što je ključno za precizno predviđanje gustinske mape.

Nakon prolaska slike kroz ovaj deo mreže, dobijamo **feature map** dimenzija približno 1/8 ulazne slike (zbog pooling operacija u prva četiri bloka).

Back-end (Dilatovane konvolucije)

Drugi deo CSRNet arhitekture, nazvan **back-end**, uvodi **dilatovane (atrous)** konvolucione slojeve. Ideja dilatovane konvolucije je da se poveća receptive field bez povećanja broja parametara i bez smanjenja dimenzije izlazne mape.

Matematički opis dilatovane konvolucije

Za jednodimenzionalni slučaj, dilatovana konvolucija definisana je kao:

Formula dilatovane konvolucije:

$$(y *_{(r)} k)(i) = \Sigma \text{ (od } j = 1 \text{ do } K) [\ y(i + r \cdot j) \cdot k(j)\ ]$$

Gde je:

- **y** — ulazni signal (ili mapa karakteristika)
- **k** — filter konvolucije
- **r** — faktor dilatacije
- **K** — veličina jezgra

U 2D kontekstu (slike), dilatacija „razmiče“ elemente filtera tako da se posmatra šire područje slike. Time se modelu omogućava da istovremeno „vidi“ i lokalne i globalne kontekste — osobina izuzetno važna za razumevanje rasporeda gužve.

Struktura back-end dela

Back-end deo sadrži šest uzastopnih konvolucionih slojeva, svi sa **3×3** jezgrima i **dilatacionim faktorima = 2**. Svaki sloj je praćen ReLU aktivacijom. Nakon poslednjeg sloja, koristi se **1×1 konvolucija** koja projektuje feature map u jednodimenzionalnu gustinsku mapu.

Ukupna arhitektura može se sumirati sledećim nizom slojeva:

Deo	Tip sloja	Kernel	Dilatacija	Broj filtera	Aktivacija
Front-end	VGG-16 (conv1–conv4)	3×3	1	64–512	ReLU
Back-end	Dilated conv ×6	3×3	2	512	ReLU
Output	Conv	1×1	1	1	Linear

Treniranje mreže:

Opis procesa treninga modela

Za potrebe obuke modela **CSRNet**, sproveden je proces treniranja na **ShanghaiTech Part A i B** skupu podataka, koji sadrži fotografije gužvi i odgovarajuće gustinske mape (*density maps*). Cilj modela je da nauči da iz ulazne slike predvidi gustinsku mapu koja predstavlja raspored ljudi u sceni.

Parametri treninga

- **Broj epoha:** 300
- **Veličina batch-a:** 300 slika
- **Learning rate:** 1e-6
- **Optimizator:** SGD (stohastički gradijentni spust) sa momentum-om 0.95 i weight decay-om 5e-4
- **Funkcija gubitka:** MSELoss (Mean Squared Error) – meri razliku između predviđene i stvarne gustinske mape

Proces treninga

U svakoj epohi, model je učitavao batch slika i njihovih gustinskih mapa. Nakon što bi se izračunao izlaz modela, merila se greška preko MSE funkcije gubitka.
Ta greška se koristila za ažuriranje parametara modela metodom **backpropagation** i **SGD optimizacijom**. Tokom treniranja ispisivana je vrednost gubitka po epohi radi praćenja konvergencije.

Rezultati:

Count: 141 Estimate: 167.73 PSNR: 27.17 SSIM: 0.77	Count: 819 Estimate: 679.15 PSNR: 22.75 SSIM: 0.61	Count: 1068 Estimate: 1032.71 PSNR: 22.65 SSIM: 0.72

Ovde dajemo prikaz tri slike koju smo provukli kroz mrežu.
Prvi red predstavlja originalne slike, drugi predstavlja tačne ciljne promenljive, a treći šta je naša mreža predvidela, kao što vidimo predvidjanja neodstupaju previše.

Ostali pokušaji:

Pokušali smo da obogatimo naš model sa još slika koje sadrže veći broj ljudi, ali je generisanje istinitosnih promenljivih, tj. “Zamuljane” matrice trajalo predugo. Najveći problem je bilo računanje sigme, tj. pronalaženje najbližih komšija. Pokušali smo sa korišćenjem FAISS algoritma, koji koristi aproksimacije, ali one nisu bile dovoljno dobre.