

Martin Kozmon

Kompresia obrazu pomocou doprednej siete

2. zadanie
Neurónové siete

Úvod

Kompresia obrazu je dlhodobu aktuálna téma, súvisiaca so znižovaním veľkosti obrazových súborov. Okrem zníženia veľkosti obrazového súboru pre jeho uloženie kompresia poskytuje možnosť znižovania veľkosti súboru.

Jednou z metód je využitie doprednej neurónovej siete so spätným šírením. Neurónové siete majú široké využitie v rôznych oblastiach práve vďaka svojim vlastnostiam. Metóda kompresie je založená na tom, že neurónové siete sú schopné spracovať vstupy na jednoduchšie vzorky. V skrytej vrstve je obsiahnutá informácia z vonkajšieho vstupu.

Táto práca sa zaoberá analýzou existujúcich riešení a experimentmi s doprednou neurónovou sieťou pri kompresii obrazových súborov.

Analyza existujúcich riešení

Existujú rôzne prístupy pri využívaní neurónovej siete na kompresiu obrazu, v zásade však existuje niekoľko základných prístupov a modifikácie, ktoré vedú k menšej odchýlke, rýchlejšej konvergencii, zníženiu počtu tréningových vzorov a podobne. Riešenia môžeme rozdeliť nasledovne:

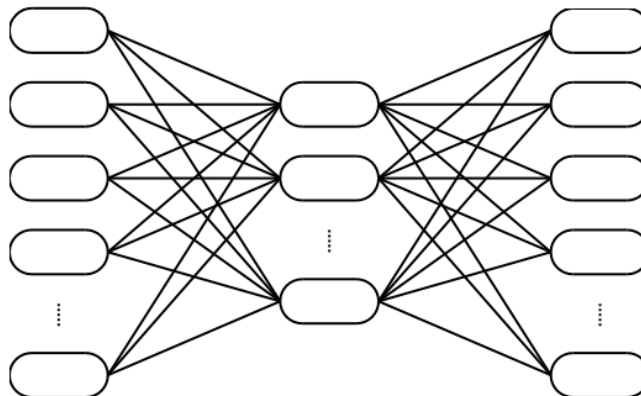
1. dopredná sieť so spätným šírením chyby (backpropagation network) – ktorú využívam aj v tejto práci a ktorej princíp bude popísaný neskôr. V tejto oblasti sa nevyužívajú len backpropagation algoritmy. Iné riešenia[1] využívajú doprednú neurónovú sieť na kompresiu rovnakým spôsobom, ale miesto iteratívneho backpropagation algoritmu využívajú metódu priameho riešenia (direct solution method).
2. Hierarchická sieť so spätnou propagáciou – rozšírená sieť do štruktúrovanej hierarchickej siete pridaním dvoch ďalších skrytých vrstiev, z ktorých jedna slúži ako combiner, ďalšia ako compressor a tretia ako decombiner. Myšlienka tejto štruktúry je založená na využití korelácie pixelov v rámci blokov.
3. Adaptívna sieť so spätnou propagáciou – je navrhovaná tak, že kompresia je vykonávaná vzhľadom na obsah súboru. Teda sa vytvorí skupina sietí s rastúcim počtom skrytých neurónov. Tréningovanie môže prebiehať paralelne, sériovo, alebo v závislosti od aktivity.
4. Prediktívne kódovanie – využíva vlastnosť vysokej korelácie medzi susednými vzorkami, čo zároveň spôsobuje vysoký stupeň redundancie vstupných údajov. Ak by sa znížila redundancia, malo by byť dosiahnuté efektívnejšie kódovanie.

Ďalšie experimenty [2] sa venujú predovšetkým optimálnemu zníženiu počtu tréningových vzorov, keďže pri väčších obrazoch je časová a výpočtová zložitosť značne vysoká. Preto sa využíva redukcia vzorov vylučovaním rovnakých alebo veľmi podobných vzorov.

Nejedná sa pritom len o zníženie časovej náročnosti, ale aj o zníženie redundancie tréningovania.

Dopredná neurónová sieť

Na kompresiu obrazu sa využíva neurónová sieť s tzv. bottleneck architektúrou. To znamená, že vstupná a výstupná vrstva má rovnaký počet neurónov väčší ako je skrytá vrstva. Schéma siete je znázornená na Obr.1. V tomto projekte využívam neurónovú sieť s 64 vstupnými a výstupnými neurónmi a premenlivým počtom neurónov v skrytej vrstve. Počet neurónov v skrytej vrstve určuje kompresný pomer, teda o koľko dokážeme zmenšiť veľkosť súboru oproti originálu. Vstupná aj výstupná vrstva sú úplne prepojené so skrytou vrstvou. Váhy jednotlivých spojení sú určené ako náhodné čísla z malého intervalu (konkrétne $\langle -0.5, 0.5 \rangle$), Používam sigmoidálnu aktivačnú funkciu, pretože podľa naštudovaných existujúcich riešení dosahuje lepšie výsledky ako lineárna, aj keď to nie je potvrdené. Jedna časť experimentu bude zameraná na porovnanie oboch spôsobov. Pre výsledky rátam strednú kvadratickú odchýlku a pomocou nej upravujem váhy jednotlivých spojení. Iterácie sú vykonávané dovtedy, pokiaľ sa nedosiahne maximálny počet cyklov, alebo sa nedosiahne menšia, ako je minimálna požadovaná odchýlka.



Obr. 1 Schéma bottleneck neurónovej siete

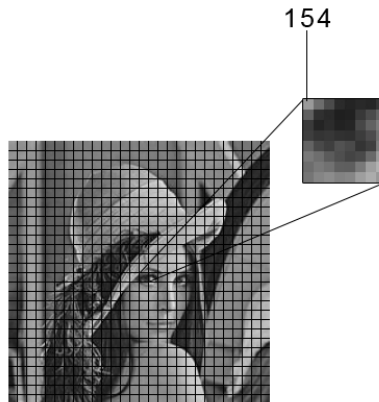
Reprezentácia obrazu

V tomto projekte používam čiernobiely rastrové obrazové súbory. Rastrové obrazové súbory je možné kódovať po jednotlivých pixeloch. Čiernobiely obraz má tú vlastnosť, že všetky tri farebné zložky (r,g,b) majú rovnakú hodnotu, rovnajúcu sa hodnote jasú daného pixelu. Táto hodnota je v rozsahu 0-255. Do neurónovej siete dávame vstupy v rozsahu $\langle 0,1 \rangle$, čo docielime tak, že hodnotu jasú predelíme číslom 255. oto sa nazýva normalizácia obrazu. Je to preto, že neurónové siete pracujú oveľa efektívnejšie s hodnotami v rozsahu $\langle 0,1 \rangle$.

Na testovanie kompresie teda využívam sieť s 64 vstupnými a výstupnými neurónmi. Preto potrebujem tréningové vzory s veľkosťou 64 pixelov. Testovací obraz

teda segmentujem na štvorce veľkosti 8x8 pixelov. Na výstupe ich zase opačným spôsobom skladám dokopy na pôvodný obraz.

Keďže implementácia projektu je v programovacom jazyku Java, na spracovanie obrazu využívam triedu Image a podtriedu BufferedImage, ktorá poskytuje mnohé operácie s obrazovými súbormi. Jednotlivé pixely ukladám do ArrayListu a všetky vzory sú uložené ako Vector.



Obr. 2 Segmentovanie vstupného súboru

Obraz veľkosti 128x128 pixelov môžeme rozdeliť na 256 segmentov, teda 256 trénovacích vzorov po 64 pixelov.

Počas trénovania vypisujem výsledky každých 100 trénovacích epoch, vypočítam priemernú kvadratickú odchýlku a zisťujem, koľko percent pixelov z trénovacej množiny má rovnakú úroveň jasnosti, ako je požadovaná hodnota. Pri nižšom počte neurónov je lepšie zisťovať, koľko percent pixelov sa líši od výsledku o nejakú konkrétnu hodnotu jasnosti, povedzme o plus-mínus dva.

Experimentálne som zistil, že sieť dostáva lepšie výsledky pri nízkych hodnotách rýchlosti učenia menších ako 0.1. Preto v nasledujúcich experimentoch využívam rýchlosť učenia rovnú hodnote 0,05.

Experimenty s kompresiou obrazu

Na experimenty s kompresiou používam obraz Lenna (alebo Lena) zo stránky Signal & Image Processing Institute, Electrical Engineering Department, University of Southern California, ktorý je voľne dostupný na testovanie obrazového spracovania. Tento obraz sa štandardne používa na testovanie kompresie.

Veľkosť obrazu pri väčšine experimentov je 128x128 pixelov, pretože pri väčších obrazoch vzniká veľké množstvo trénovacích vzorov a natrénovanie siete na týchto vzoroch trvá aj niekoľko hodín už pri obraze 256x256 pixelov.

Zmena počtu neurónov v skrytej vrstve

Tieto experimenty sú zamerané na zistenie, ako sa zmení kvalita obrazu na výstupe v závislosti od zmeny počtu neurónov v skrytej vrstve. V existujúcich projektoch sa pracovalo do počtu 20 skrytých neurónov.

1. Neurónová sieť 64 – **64** – 64, rýchlosť učenia 0.05



Výsledok po 10000
trénovacích epochách:

*15,2% pixelov má hodnotu
jasu zhodnú s pôvodným
obrazom. Odchýlka = 1,88*

2. Neurónová sieť 64 – **40** – 64, rýchlosť učenia 0.05



Výsledok po 10000
trénovacích epochách:

*11,75% pixelov má hodnotu
jasu zhodnú s pôvodným
obrazom. Odchýlka = 3,79*

3. Neurónová sieť 64 – **32** – 64, rýchlosť učenia 0.05



Výsledok po 10000
trénovacích epochách:

*10,25% pixelov má hodnotu
jasu zhodnú s pôvodným
obrazom. Odchýlka = 5,84*

4. Neurónová sieť 64 – **20** – 64, rýchlosť učenia 0.05



Výsledok po 10000
trénovacích epochách:

*7,5% pixelov má hodnotu
jasu zhodnú s pôvodným
obrazom. Odchýlka = 14,00*

Peak Signal-to-noise pomer

PSNR je inžiniersky termín používaný na určenie pomeru medzi maximálnou možnou silou signálu a silou šumu, ovplyvňujúcou poškodenie signálu. Pretože signály majú pomerne široký dynamický rozsah, výsledok je určený v mierke logaritmických decibelov. Tento pomer sa používa často na určenie kvality pri stratovej kompresii obrazu a videa. Vypočíta sa nasledovne:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \|I(i,j) - K(i,j)\|^2 \quad PSNR = 10 \cdot \log_{10} \left(\frac{MAX_I^2}{MSE} \right) = 20 \cdot \log_{10} \left(\frac{MAX_I}{\sqrt{MSE}} \right)$$

MSE je priemerná kvadratická odchýlka (mean squared error) pre monochromatické obrazy veľkosti $m \times n$. $I(i,j)$ a $K(i,j)$ sú jasové hodnoty pixelu na príslušných súradniciach pre originálny a komprimovaný obraz. Samotnú hodnotu PSNR určíme podľa druhého vzorca, kde MAX je maximum hodnoty pixelu, pre 8 bitový obraz je to 255.

V medicíne sa pre obrazové súbory požaduje hodnota 40-50 dB. Pre stratovú kompresiu videa a obrazov sa požaduje hodnota 30-50 dB. Podľa existujúcich projektov, ktoré skúmali hodnoty PSNR komprimovaných obrazov pomocou neurónovej siete sa tieto hodnoty pohybovali v rozsahu 26-30 dB.

| Počet neurónov v skrytej vrstve | PSNR (dB) |
|---------------------------------|-----------|
| (Lena) 64 | 32.22 |
| (Lena) 32 | 32.00 |
| (Lena) 20 | 28.54 |

Závislosť od veľkosti vstupného obrazu

V tomto bode skúmam závislosť výsledku kompresie od veľkosti vstupného obrazu. Menší obraz má menší počet pixelov a teda aj menšiu tréningovú množinu. Využívam obraz v rozlíšení 64x64 pixelov.

Tréningový obraz



64 neurónov



PSNR= 32.33 dB

32 neurónov



PSNR= 31.50 dB

20 neurónov



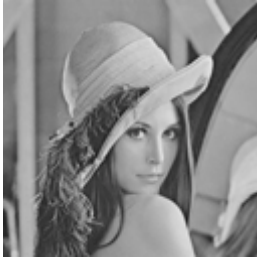
PSNR= 28.94 dB

Výsledné hodnoty PSNR približne zodpovedajú hodnotám pre obraz 128x128.

Zmenšenie segmentov obrazu

Miesto tréningových vzorov 8x8 delím obraz na segmenty 4x4, teda vytváram neurónovú sieť so 16 vstupnými neurónmi a mením počet neurónov v skrytej vrstve.

Trénovací obraz



16 skrytých neurónov



PSNR= 33.217 dB

8 skrytých neurónov

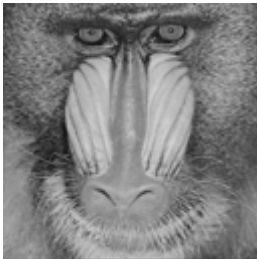


PSNR= 30.727 dB

Prenos súboru po natrénovaní siete na iný vstupný obraz

V predchádzajúcich experimentoch som neurónovú sieť najprv natrénoval pomocou trénujúcej množiny vytvorenej z toho istého obrazu. V tejto časti budem trénovať sieť pomocou iného obrazu ako prenášaného.

Trénovací obraz

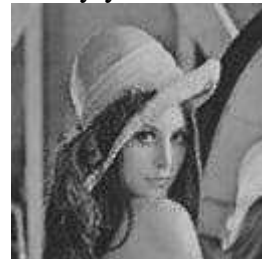


64 skrytých neurónov



PSNR= 30.80 dB

32 skrytých neurónov

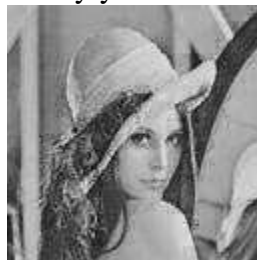


PSNR= 29.15 dB

Trénovací obraz



64 skrytých neurónov



PSNR= 27.67 dB

32 skrytých neurónov



PSNR= 25.18 dB

Trénovanie pomocou zmenšenej verzie obrazu

V tomto experimente budem trénovať sieť na obraze s veľkosťou 64x64 pixelov a po natrénovaní cez ňu nechám prejsť ten istý obraz s veľkosťou 128 x128 pixelov.

Trénovací obraz



64 skrytých neurónov



32 skrytých neurónov



Spracovanie obrazu s rôznym rozdelením jasů

Táto časť analyzuje ako vplýva rozdelenie jasových hodnôt na tréovanie siete. Predpoklad je taký, že testovací vstup, ktorý neobsahuje plynulé prechody, ale má strmšie rozdiely v hodnote jasů medzi susednými pixelmi by mal byť jednoduchší na natréovanie z hľadiska menšieho počtu rôznych jasových hodnôt, ako aj určitej redundancie tréovacích vzorov.

1. Neurónová sieť 64 – 32 – 64, rýchlosť učenia 0.05

Vstupný obraz je zložený len z pixelov čiernej a bielej farby



PSNR= 32.33 dB

Výsledok po 10000 tréovacích epochách:
40,1% úspešnosť

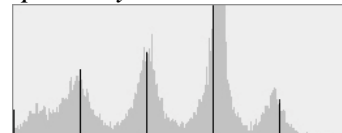


2. Neurónová sieť 64 – 32 – 64, rýchlosť učenia 0.05

Vstupný obraz je zložený len z pixelov s 5 rôznymi jasovými hodnotami



Po 10000 tréovacích epochách: 4,5% pixelov má hodnotu jasů zhodnú s pôvodným obrazom.



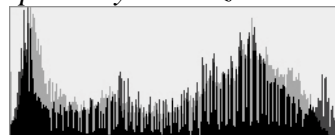
Výsledok je značne horší, ako keď sme pracovali s pôvodným neupraveným obrazom.

3. Neurónová sieť 64 – 32 – 64, rýchlosť učenia 0.05

Vstupný obraz má zvýšený kontrast o hodnotu 100 oproti pôvodnému



Po 10000 trénochách epochách: 3,8% pixelov má hodnotu jasú zhodnú s pôvodným obrazom.

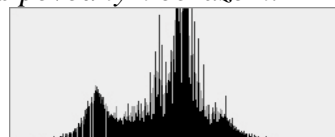


4. Neurónová sieť 64 – 32 – 64, rýchlosť učenia 0.05

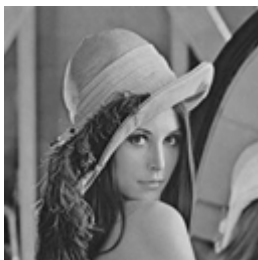
Vstupný obraz má znížený kontrast o hodnotu 100 oproti pôvodnému



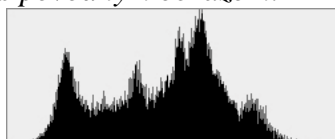
Po 10000 trénochách epochách: 17,8% pixelov má hodnotu jasú zhodnú s pôvodným obrazom.



Pre porovnanie neurónová sieť s rovnakými parametrami pri kompresii pôvodného obrazu aj s porovnávacím histogramom.







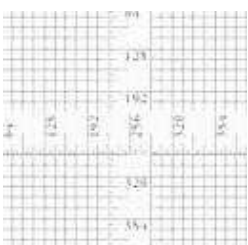
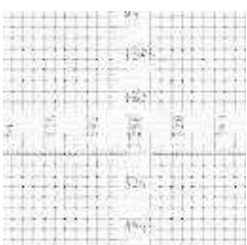


Po 10000 trénochách epochách: 10,25% pixelov má hodnotu jasú zhodnú s pôvodným obrazom.



Redukcia trénochacej množiny

Vytvoril som metódu, pomocou ktorej sa snažím redukovať testovaciu množinu takým spôsobom, že zisťujem priemerný jas každého segmentu a vyhadzujem redundantné segmenty. Pri obraze lena.jpg napríklad dokážeme týmto spôsobom zredukovať počet segmentov z 256 len na 127.

| Neredukovaný | Redukovaný | Počet trénovacích párov | | PSNR (dB) | |
|---|---|-------------------------|------------|--------------|------------|
| | | neredukovaný | redukovaný | neredukovaný | redukovaný |
|  |  | 256 | 127 | 32.00 | 28.68 |
|  |  | 256 | 107 | 30.20 | 28.73 |
|  |  | 256 | 102 | 29.62 | 26.31 |
|  |  | 256 | 67 | 33.21 | 28.08 |

Priebeh tréovania neurónovej siete s 32 skrytými neurónmi

Epocha 20



Epocha 40



Epocha 60



Epocha 80



Epocha 100



Epocha 120



Epocha 140



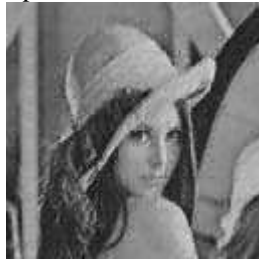
Epocha 160



Epocha 200



Epocha 400



Epocha 600



Epocha 700



Epocha 1200



Epocha 2000



Epocha 3000



Epocha 6000



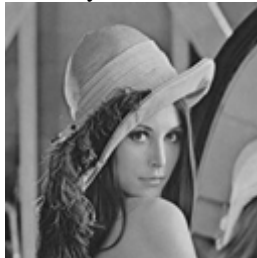
Epocha 7000



Epocha 10000



Pôvodný obraz



Použité zdroje a literatura

[1] S. Kulkarni, B. Verma, M. Blumenstein.: "Image Compression using a Direct Solution Method based Neural Network". Proceedings of the Tenth Australian Joint Conference on Artificial Intelligence, 1997.

[2] S. Anna Durai, and E. Anna Saro "Image Compression with Back-Propagation Neural Network using Cumulative Distribution Function", PROCEEDINGS OF WORLD ACADEMY OF SCIENCE, ENGINEERING AND TECHNOLOGY VOLUME 17 DECEMBER 2006, ISSN 1307-6884