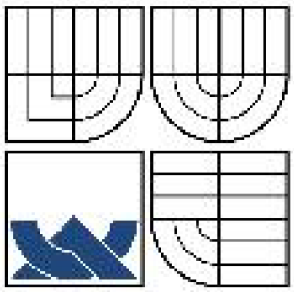
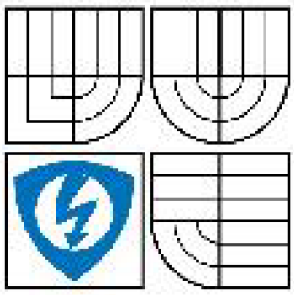


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF CONTROL AND INSTRUMENTATION

KOMPRESE OBRAZU S VYUŽITÍM UMĚLÉ NEURONOVÉ SÍTĚ

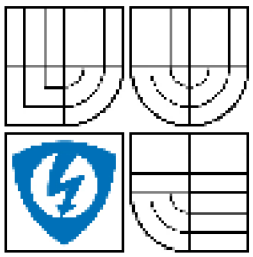
IMAGE COMPRESSION BASED ON ARTIFICIAL NEURAL NETWORK

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — JIŘÍ VONDRÁČEK
AUTHOR

VEDOUCÍ PRÁCE — doc. Ing. VÁCLAV JIRSÍK, CSc.
SUPERVISOR

BRNO 2009



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky a komunikačních technologií

Ústav automatizace a měřicí techniky

Bakalářská práce

bakalářský studijní obor
Automatizační a měřicí technika

Student:	Jiří Vondráček	***ID:***	98456
Ročník:	3	***Akademický rok:***	2008/2009

NÁZEV TÉMATU:

Komprese obrazu s využitím umělé neuronové sítě

POKYNY PRO VYPRACOVÁNÍ:

Prozkoumejte problematiku komprese digitálního obrazu a navrhněte metodu s využitím umělé neuronové sítě. Zvolený algoritmus implementujte ve vybraném vývojovém či simulačním prostředí, zhodnoťte parametry algoritmu a srovnejte účinnost s obecně používanými metodami.

DOPORUČENÁ LITERATURA:

Šíma J., Neruda R. Teoretické otázky neuronových sítí. Praha : MATFYZPRESS, 1996. ISBN 80-85863-18-9.

Richard Nikal: Komprese dat s využitím vrstvených neuronových sítí.

Robert D. Dony: Neural Network Approaches to Image Compression.

Termín zadání:	9.2.2009	***Termín odevzdání:***	1.6.2009

Vedoucí práce: doc. Ing. Václav Jirsík, CSc.

prof. Ing. Pavel Jura, CSc.
Předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práve třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

Abstrakt

Bakalářská práce je zaměřena na kompresi obrazu s využitím umělé neuronové sítě včetně praktické realizace. Jejím cílem je prozkoumat možnosti komprese obrazu umělou neuronovou sítí a vyhodnotit výsledky. V teoretické části je vysvětlen princip umělých neuronových sítí a také jsou zde popsány základní metody komprese obrazu. V praktické části je stručný popis kompresního programu, porovnání různých nastavení a vyhodnocení výsledků.

Klíčová slova

Umělá neuronová síť, Komprese obrazu

Abstract

The thesis is focused on the image compression based on artificial neural network with practical implementation. The objective of this thesis is to explore possibilities of an image compression by artificial neural network and analyze results. In the theoretical part of the work, the fundamentals of artificial neural network are described and basic image compression techniques are explained. In the practical part there is a brief description of the compression program, the comparison of different settings and result evaluation.

Keywords

Artificial neural network, Image compression

VONDRÁČEK, J. Komprese obrazu s využitím umělé neuronové sítě. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 59 s. Vedoucí bakalářské práce doc. Ing. Václav Jirsík, CSc.

Prohlášení

„Prohlašuji, že svou bakalářskou práci na téma Komprese obrazu s využitím umělé neuronové sítě jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.“

V Brně dne: **1. června 2009** ……

podpis autora

Poděkování

Děkuji vedoucímu bakalářské práce doc. Ing. Václavu Jirsíkovi, CSc. a především konzultantovi Ing. Václavu Sáblíkovi za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce.

OBSAH

OBSAH **6**
SEZNAM OBRÁZKŮ **8**
SEZNAM TABULEK **9**
1. ÚVOD **10**
2. KOMPRESE DAT **11**
2.1 Historie komprese dat 11
2.2 Rozdělení komprese dat 11
2.2.1 Bezeztrátové komprese 12
2.2.2 Ztrátové komprese 12
2.3 Vybrané kompresní algoritmy 13
2.3.1 RLE komprese 13
2.3.2 Huffmanovo kódování 14
2.3.3 Ztrátová DCT komprese 15
2.3.4 Ztrátová DWT komprese 16
2.4 Zhodnocení vybraných algoritmů 16
3. NEURONOVÉ SÍTĚ **18**
3.1 Historie neuronových sítí 18
3.2 Komprese neuronovou sítí 19
3.3 Model neuronu 20
3.3.1 Výpočet stavu neuronu 21
3.3.2 Geometrická představa funkce neuronu 23
3.3.3 Funkce neuronu v neuronové síti 24
3.4 Druhy neuronových sítí 26
3.4.1 Cyklická síť 26
3.4.2 Acyklická síť 27
3.4.3 Síť perceptronů 27
3.4.4 Vícevrstvá síť s algoritmem backpropagation 28
3.4.5 Hopfieldova síť 30
3.4.6 RBF síť 32

3.5 Zhodnocení vybraných modelů 33
4. KOMPRESE POMOCÍ NEURONOVÝCH SÍTÍ 34
4.1 Princip komprese 34
4.2 Teoretický rozbor 35
4.2.1 Koncepce komprese obrazu neuronovou sítí 35
4.3 Předpokládané výsledky 36
5. PRAKTICKÁ REALIZACE 37
5.1 Stručný popis programu 37
5.1.1 Hlavní obrazovka 37
5.1.1.1 Hlavní menu 38
5.1.1.2 Ovládací tlačítka 39
5.1.2 Nastavení programu 39
5.2 Vliv konfigurace sítě na kvalitu obrazu 41
5.2.1 Rozměry vzoru 41
5.2.2 Topologie sítě 43
5.2.2.1 Standardní konfigurace, 3 vrstvy, (16-4-16) neuronů 43
5.2.2.2 5 vrstev, (16-32-4-32-16) neuronů 44
5.2.2.3 5 vrstev, (16-32-32-4-16) neuronů 45
5.2.2.4 4 vrstvy, (16-32-4-16) neuronů 46
5.2.3 Kompresní poměr 47
5.2.4 Koeficient učení 48
5.2.5 Volba tréninkového vzoru 50
5.3 Porovnání s jinými metodami komprese 54
6. ZÁVĚR 56
7. LITERATURA 57
8. SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK 58
9. SEZNAM PŘÍLOH 59

SEZNAM OBRÁZKŮ

Obr. 1: Příklad dekódovacího stromu Huffmanova kódování 14
Obr. 2: Pro stejná data může existovat více rovnocenných stromů 15
Obr. 3: DCT komprese [5] 16
Obr. 4: Zjednodušené schéma Neuronu [2] 18
Obr. 5: Komprese pomocí neuronové sítě [1] 20
Obr. 6: Zjednodušený model neuronu 21
Obr. 7: Grafy prahových funkcí [1] 22
Obr. 8: Rozdělení písmen A a B pomocí neuronu [1] 23
Obr. 9: Adaptace vah při chybném rozdělení písmen 24
Obr. 10: Neurony realizující funkci AND a OR [1] 25
Obr. 11: Příklad logického obvodu [1] 26
Obr. 12: Síť perceptronů [1] 27
Obr. 13: Vícevrstvá síť perceptronů [4] 28
Obr. 14: Závislost chyby sítě na čase při učení sítě pomocí algoritmu backpropagation [1] 30
Obr. 15: Topologie Hopfieldovy sítě [1] 31
Obr. 16: Rekonstrukce znaku Hopfieldovou sítí [1] 32
Obr. 17: Nastavování středů RBF [1] 33
Obr. 18: Konfigurace sítě pro kompresi [1] 34
Obr. 19: Návrh sítě pro kompresi obrazu – vícevrstvý perceptron [4] 35
Obr. 20: Hlavní obrazovka aplikace 38
Obr. 21: Konfigurační okno 40

SEZNAM TABULEK

Tabulka 1: RLE komprese 13

1. ÚVOD

Na konci dvacátého století začal široký rozmach informačních technologií. Neustálý rozvoj informačních systémů a zejména internetu klade vzrůstající nároky i na množství dat, která jsou přenášena a skladována. I přes výrazný pokrok v oblasti technologií datových úložišť a přenosových linek je stále potřeba množství dat redukovat. Jednou z možností redukce objemu bez ztráty obsahu je komprimace dat. Existuje celá řada kompresních metod. U některých dochází ke ztrátě dat, ale bývají účinnější, jiné data rekonstruují úplně, zpravidla však na úkor účinnosti. Obě tyto kategorie mohou být založeny na mnoha různých principech.

Tato práce se zabývá využitím umělých neuronových sítí pro účely komprimace. Jedná se o metodu ztrátovou, která však disponuje klíčovými přednostmi pro aplikaci v oblasti dynamické komprese datových toků v reálném čase. Kapitola 2 se podrobněji zabývá problematikou komprese, kapitola 3 shrnuje všeobecný přehled o neuronových sítích a kapitola 4 se zabývá využitím neuronových sítí pro kompresi dat. Závěrečná kapitola 5 shrnuje dosažené výsledky.

2. KOMPRESE DAT

Ve stejné době, kdy lidé začali zaznamenávat události a pořizovat nejrůznější údaje, vznikla také potřeba, nasbíraná data nějakým způsobem ukládat a přenášet. S rostoucím množství uchovávaných informací se objevila také potřeba nějakým způsobem zredukovat jejich objem.

2.1 HISTORIE KOMPRESE DAT

První algoritmy komprese dat sahají relativně daleko do minulosti. Jedním z prvních principů byla například Morseova abeceda, kterou pan Morse vytvořil ve čtyřicátých letech devatenáctého století pro potřebu svého telegrafu. Skutečný rozmach kompresních algoritmů však začal až s rozvojem výpočetní techniky. Mezi průkopníky tohoto oboru patřili zejména pánové Abraham Lempel a Jakob Ziv (LZ komprese) a také pan David A. Huffman (Huffmanovo kódování). V dnešní době je komprese samostatným oborem informační techniky.

Požadavky na účinnou kompresi dat vyplývají především z omezených kapacit jejich úložišť, ale také z potřeby redukovat množství přenášených dat. Ačkoliv v současné době je kapacita úložišť a přenosových linek o několik řádů větší než dříve, stejnou měrou se zvětšuje i objem uchovávané a přenášené informace. Z toho důvodu je komprese dat stále aktuální.

V současné době existuje celá řada nejrůznějších kompresních algoritmů, většinou specificky zaměřených na konkrétní aplikaci. Tyto algoritmy lze dělit podle mnoha různých kritérií. Např. tedy podle oblasti zaměření, ale také podle účinnosti, náročnosti na výpočty, rychlosti dekomprese nebo ztrátovosti (dekomprimovaná data nemusí být přesně stejná). Často bývá plnění jednoho kritéria na úkor jiného. Např. dosažený vysoký kompresní poměr na úkor částečné ztráty informace nebo za cenu velkého množství výpočtů a tím dlouhé doby komprese, případně také dekomprese.

2.2 ROZDĚLENÍ KOMPRESE DAT

Kompresních metod je celá řada a dají se třídit podle mnoha různých hledisek. V zásadě rozlišujeme dva základní druhy komprese. Kompresi

bezeztrátovou a kompresi ztrátovou. Dále je možné dělit jednotlivé druhy kompresí podle rychlosti komprese (případně dekomprese), podle účinnosti nebo podle využití komprese konkrétními aplikacemi.

2.2.1 Bezeztrátové komprese

Bezeztrátová komprese je taková komprese, která umožňuje úplnou zpětnou rekonstrukci dat. To znamená, že pokud provedeme dekompresi dříve zkomprimovaných dat, získáme data naprosto identická s těmi původními. Toto je důležité zejména v případě kdy charakter komprimovaných dat nepřipouští žádnou ztrátu informace. Takový charakter mají například textové dokumenty nebo programy. Nevýhodou takové komprese je nižší kompresní poměr, než jaký nabízejí komprese ztrátové. Algoritmů pro bezeztrátovou kompresi dat je celá řada. Nejjednodušší, avšak nejméně účinná je tzv. RLE komprese, která nahrazuje opakující se znaky informací o počtu opakování. Tato metoda však dává rozumné výsledky pouze pro určitý typ dat – např. obrázky s velkými stejnobarevnými plochami. Mezi nejznámější a nejpoužívanější patří Huffmanovo kódování (frekventovanějším datům jsou přiřazeny kratší bitové sekvence než datům méně frekventovaným). Často používané jsou také tzv. slovníkové algoritmy, které v průběhu komprese vytváří slovník komprimovaných dat, do kterého se při dalším výskytu stejných dat odkazují (např. LZ komprese).

2.2.2 Ztrátové komprese

Ztrátová komprese se snaží rozlišit důležitá data od nepodstatných a ta nepodstatná vypouštět. Pro rozlišení toho, která data jsou důležitá, se využívá zejména nedokonalosti lidských smyslů a záleží tedy na konkrétní aplikaci. Hodí se zejména pro zpracování zvuku a obrazu. Např. u zvuku obvykle není potřeba uchovávat frekvence pro lidské ucho neslyšitelné a proto je můžeme vynechat. Podobně u obrazové informace lze např. body s podobnou barvou považovat za shodné. Vypuštěním nepodstatných informací lze dosáhnout podstatně lepších kompresních poměrů než u kompresí bezeztrátových. Další příklad ztrátové

komprese může být např. průměrování průběhu denních teplot, kdy nás zpětně bude zajímat jen průměrná teplota za celý den, namísto celé série naměřených teplot. Pro další vylepšení kompresního poměru je možné na výsledná komprimovaná data použít ještě některou bezeztrátovou metodu komprese (např. Hoffmanovo kódování). Tato kombinace obou metod se skutečně v praxi často využívá. Používá ji např. ztrátový formát obrázku JPEG.

2.3 VYBRANÉ KOMPRESNÍ ALGORITMY

V této kapitole si na některých algoritmech podrobněji popíšeme co se děje s daty určenými ke kompresi. Jak vzniká výsledný kód a jak jsou zkomprimovaná data uložena.

2.3.1 RLE komprese

Jeden z nejjednodušších kompresních algoritmů je RLE (Run Length Encoding). Komprimuje opakující se znaky do dvojice [Počet znaků, Znak]. Např. sekvence ABBCCC bude zakódována jako 1A2B3C. Nevýhodou je, že pokud se znaky ve vstupních datech neopakují, může být výsledný kód až dvojnásobný. Sekvence ABCD by byla zakódována jako 1A1B1C1D. V praxi se toto kódování hodí jen pro obrázky obsahující velké stejnobarevné plochy.

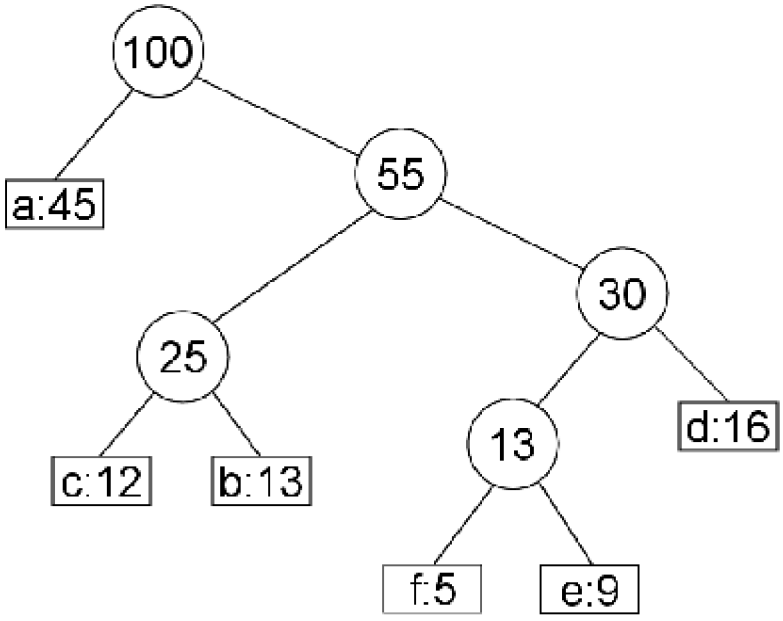
Pro eliminaci tohoto nedostatku vznikla modifikovaná varianta, kde nejvyšší bit v „počtu znaků“ určuje, jestli se jedná o počet opakování následujícího znaku nebo o počet znaků, které se neopakují. Sekvence ABCDABBCCC by byla tedy uložena takto:

příznak	*počet*	*znak*
1	5	ABCDA
0	2	B
0	3	C

Tabulka 1: RLE komprese

2.3.2 Huffmanovo kódování

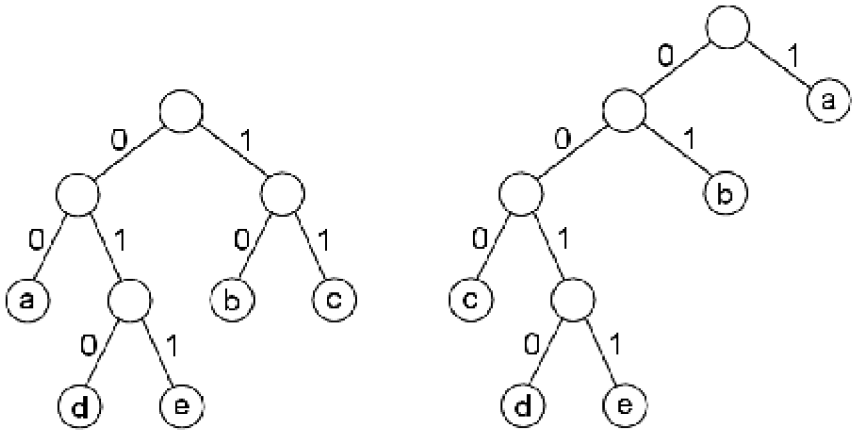
V současné době asi nejznámější a nejpoužívanější algoritmus bezeztrátové komprese. Každému ze znaků vstupních dat je přiřazen bitový řetězec o takovém počtu bitů, který odpovídá četnosti daného znaku ve vstupním souboru. Přesný postup optimálního vytvoření těchto bitových řetězců popsal pan David Hoffman již v roce 1951. Četnosti znaků ve vstupním souboru se nejprve setřídí a dvěma nejméně frekventovaným se přidělí kód (ve skutečnosti bit). Nula znamená jeden znak, jednička znak druhý. Četnost tohoto kódu je součtem četností obou znaků. Tento kód se dle četnosti zatřídí na patřičné místo a celý postup se opakuje tak dlouho, až je celý soubor zahrnut pod jediný kód. Vznikne tak jednoznačný dekódovací strom, kde ke každému znaku „vede“ série nul a jedniček .



Obr. 1: Příklad dekódovacího stromu Huffmanova kódování

Ve skutečnosti mohou nastat také situace, kdy pro stejná data lze navrhnout více různých, ale rovnocenných dekódovacích stromů. Tento stav nejlépe dokumentuje obrázek č. 2.

znak	a	b	c	d	e
četnost	40	20	20	10	10

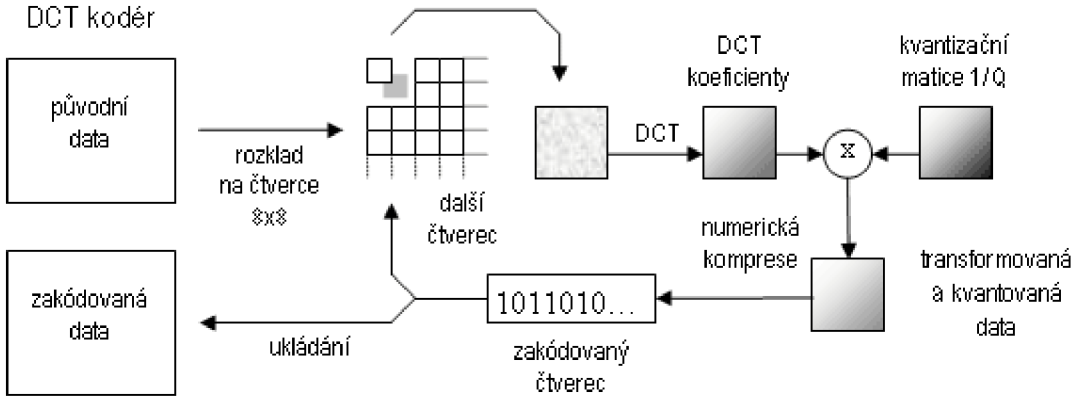


Obr. 2: Pro stejná data může existovat více rovnocenných stromů

2.3.3 Ztrátová DCT komprese

Principem této komprese je vyhledávání podobných obrazových dat. Za tímto účelem se data musí nejprve převést do kmitočtové oblasti. K tomu slouží právě DCT (diskrétní kosinová transformace). DCT je založena na Fourierově transformaci. Jejím nedostatkem je, že vzniklé signálové spektrum neobsahuje informaci o původním umístění v prostoru. Tento nedostatek lze eliminovat rozdělením dat na menší segmenty. Nejčastěji na bloky o velikosti 8x8 pixelů. DCT transformací získáme frekvenční reprezentaci obrazu, kde kmitočet odpovídá úrovni detailů. Dalším krokem je potlačení určitých frekvencí prostřednictvím kvantizační matice. Ta závisí na nastaveném stupni komprese. Posledním krokem je provedení numerické komprese, která zohledňuje vzniklé nulové koeficienty.

Nevýhodou této komprese je segmentace vstupních dat. Protože komprese se provádí pro každý segment zvlášť, může na hranicích segmentů docházet ke zkreslení. Toto zkreslení se projeví hlavně při vyšších kompresních poměrech.



Obr. 3: DCT komprese [5]

2.3.4 Ztrátová DWT komprese

DWT (diskrétní vlnová transformace) komprese je založena na vlnové transformaci. Koncepce vlnové transformace je poměrně nová (90. léta 20. století) a používá se k analýze a zpracování signálů pro celou řadu oborů. DWT odstraňuje problém segmentování DCT komprese. Pracuje s celým obrazem a to tak, že nejprve filtruje celý obraz čtveřicí dekompozičních filtrů. Jeden z těchto filtrů je horní a jeden dolní propust. Filtrace probíhá nejprve po řádcích a vzniklý výsledek se filtruje po sloupcích. Po každé filtraci dochází k decimaci (vypuštění každého druhého vzorku). Výsledkem jsou čtyři obrázky čtvrtinové velikosti, z nichž každý obsahuje jinou informaci. Uvedený postup lze rekurzivně opakovat. Počet opakování se označuje jako hloubka dekompozice. Na konec se podobně jako u DCT komprese aplikuje některá z forem bezeztrátové komprese. Např. numerická komprese nebo Huffmanův kód.

2.4 ZHODNOCENÍ VYBRANÝCH ALGORITMŮ

Do výběru byly zahrnuty čtyři druhy různých kompresí. Dvě bezeztrátové (RLE kódování a Huffmanův kód) a dvě ztrátové (obrazová DCT a obrazová DWT komprese).

Bezeztrátová RLE komprese je nejjednodušší a nejrychlejší. Právě pro tyto její vlastnosti se používala u ve své době populárního grafického formátu PCX. Dnes je však už její praktické využití relativně malé a slouží spíše k demonstračním účelům. Zajímavostí je použití této komprese v nahrávací obrázku Windows 3.11.

Naproti tomu Huffmanův kód je dnes jednou z nejpoužívanějších bezeztrátových kompresí. Nabízí poměrně vysoký kompresní poměr a zároveň není tak výpočetně náročný jako např. aritmetická komprese. Proto nachází uplatnění i v méně výkonných čipech ve fotoaparátech a podobně.

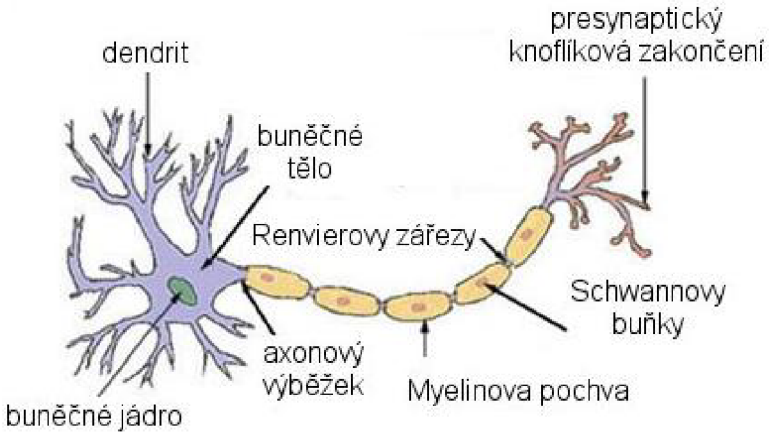
DCT i DWT komprese jsou ztrátové komprese a jsou určeny pro kompresi obrazových dat. DCT komprese je používána např. v grafickém formátu JPG. Oproti DWT kompresi není tak náročná na výpočty, ale zejména při vyšším stupni komprese nabízí horší výsledek zejména díky tzv. blokovým artefaktům (zřetelné přechody mezi segmenty) . Na principu DWT komprese vznikl v roce 2000 formát JPEG2000. Tento formát nabízí oproti standardnímu JPEG zhruba o 20% lepší kompresní poměr [5].

3. NEURONOVÉ SÍTĚ

Jako alternativa ke klasickým sekvenčním algoritmům komprese se objevila možnost využít ke kompresním účelům neuronové sítě. Neuronová síť se snaží napodobit chování lidského mozku. Kompresi dat neuronovou sítí tedy můžeme chápat jako schopnost naučit se komprimovaná data „zapamatovat“ a později opět vyvolat.

3.1 HISTORIE NEURONOVÝCH SÍTÍ

Současně se vznikem prvních elektronických počítačů vznikl i nápad hledat paralelu mezi počítačem a lidským mozkem. Objevily se teorie, podle kterých lidský mozek funguje jako velký počítač, přičemž logická hradla jsou v mozku reprezentovány jednotlivými neurony.



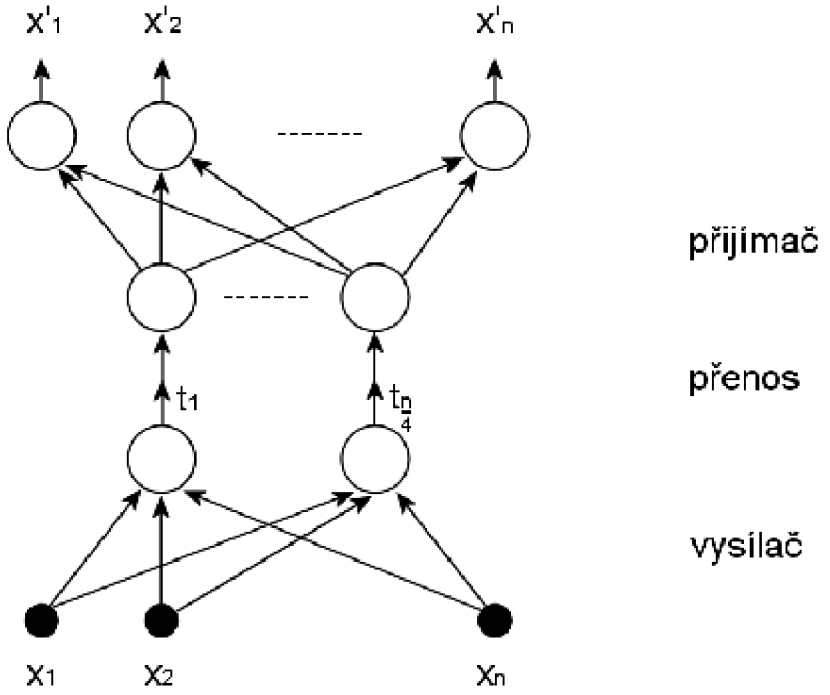
Obr. 4: Zjednodušené schéma Neuronu [2]

S prvním modelem neuronu přišli Warren McCulloch a Walter Pittse již v roce 1943. V roce 1949 vyšla kniha Donalda Hebba ‚The Organization of Behavior‘. Donald Hebb v této knize navrhl pravidla pro učení synapse neuronů (vážené propojení neuronů). V roce 1957 zobecnil původní model neuronu Frank Rosenblatt a nazval ho Perceptron. Ve stejném roce také s jeho využitím sestrojil první neuronový počítač Mark I Perceptron. Tento počítač byl navržen pro rozpoznávání znaků a tato jeho schopnost byla úspěšně prezentována. Přestože se neuropočítače

začaly zprvu úspěšně rozvíjet, přišlo se záhy na to, že záleží také na kvalitě spojení mezi neurony - synapsích a mnoha dalších aspektech. Také proto se v té době a vlastně dodnes nepodařilo sestavit funkční model lidského mozku. Nejen že nejsme schopni strukturu mozku přesně replikovat, ale také neexistuje technologie, která by dokázala simulovat tak obrovský počet neuronů.

3.2 KOMPRESE NEURONOVOU SÍTÍ

Ke kompresi lze využít také neuronové sítě. Síť musí být adaptovaná na data, která má komprimovat. Neuronová síť určená pro kompresi obsahuje ve své struktuře úzké hrdlo, ve kterém se informace šíří nižším datovým tokem. Neuronová síť zde provádí transformaci z více do méně rozměrného prostoru a učením se snaží aproximovat tuto transformaci včetně zpětného uvedení dat do původní podoby s co nejmenší chybou. V úzkém hrdle nedochází ke komprimaci dat, protože matematická transformace dat pouze informaci převede na jiný způsob vyjádření. Tedy data jsou v méně číslech, ale zato tato čísla musí být mnohem přesněji vyjádřena. Skutečná komprimace dat se projeví až v praktické implementaci, kde je použit formát čísla s omezenou délkou 32 nebo 64 bitů.

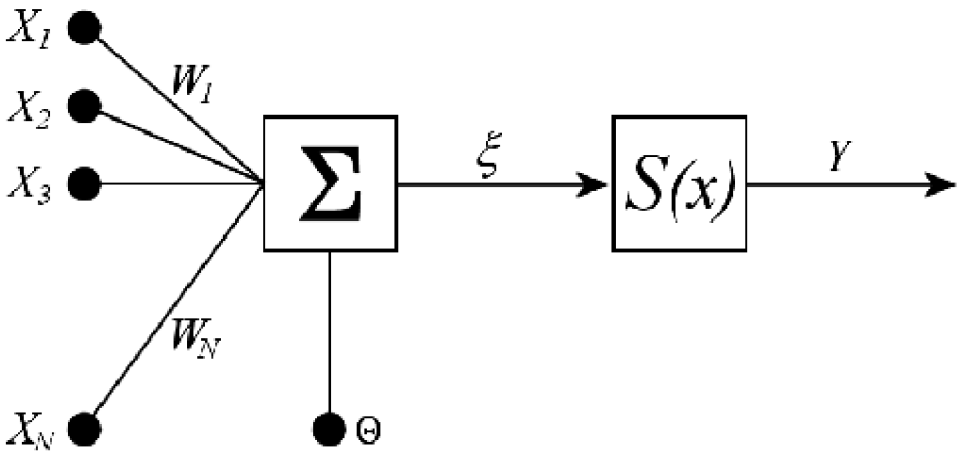


Obr. 5: Komprese pomocí neuronové sítě [1]

Vstupní datům odpovídá určitý stav vnitřní skryté vrstvy. Proto když se síť rozdělí na část vstupní a výstupní získáme kompresor a dekompresor. Počet neuronů ve vstupní a výstupní vrstvě je stejný. Poměr neuronů ve skryté vrstvě potom určuje kompresní poměr dané sítě. Z principu funkce neuronových sítí je zřejmé, že se jedná o ztrátovou kompresi.

3.3 MODEL NEURONU

Základním stavebním prvkem neuronových sítí je model neuronu. Těchto modelů existuje celá řada. Jeden z nejpoužívanějších je model popsaný McCullochem a Pittsem [2].



Obr. 6: Zjednodušený model neuronu

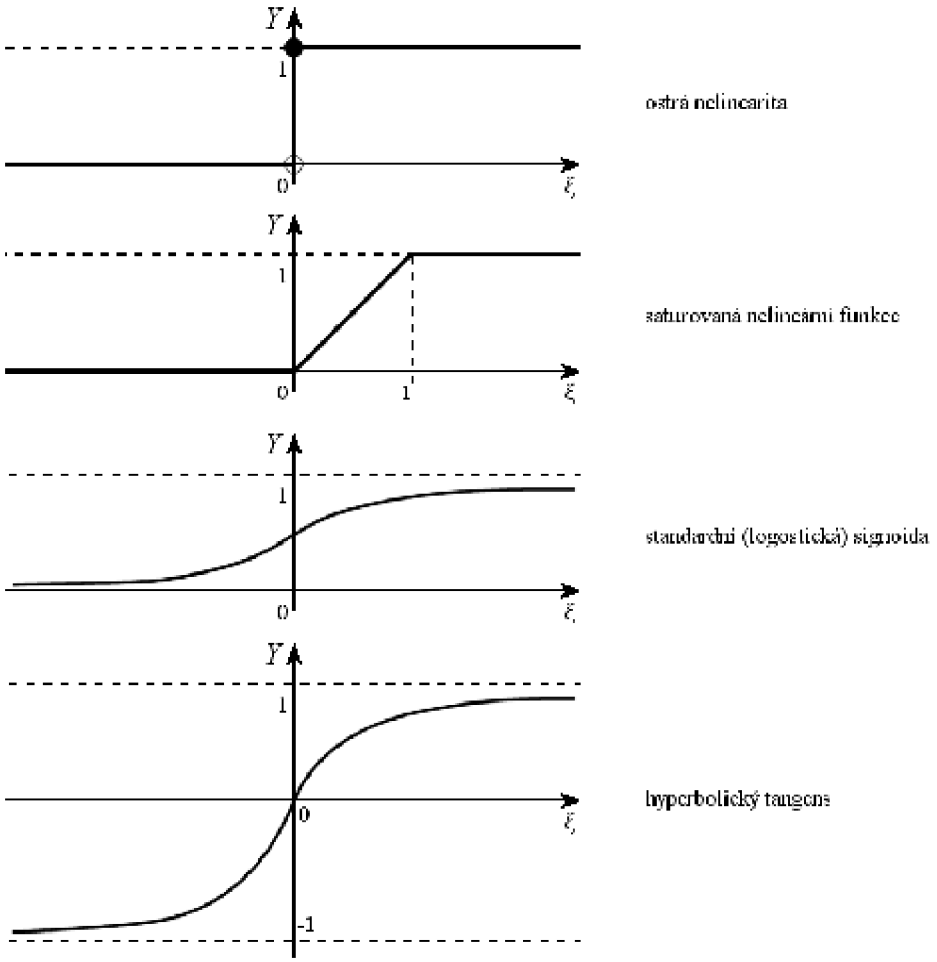
3.3.1 Výpočet stavu neuronu

$$Y = S\left(\sum_{i=1}^{N}\left(w_i x_i\right) - \Theta\right) \qquad (1)$$

kde:

- x_i jsou vstupy neuronu
- w_i jsou synaptické váhy
- Θ je práh
- ξ je potenciál neuronu
- $S(x)$ je přenosová funkce neuronu (někdy aktivační funkce)
- Y je výstup neuronu

Stav neuronu je určen vyhodnocením přenosové funkce. Pokud součet vážených vstupů neuronu přesáhne definovaný práh, je neuron nastaven jako aktivní. Přenosová funkce může být jednoduchá ostrá nelinearita, ale i složitější spojitá funkce.

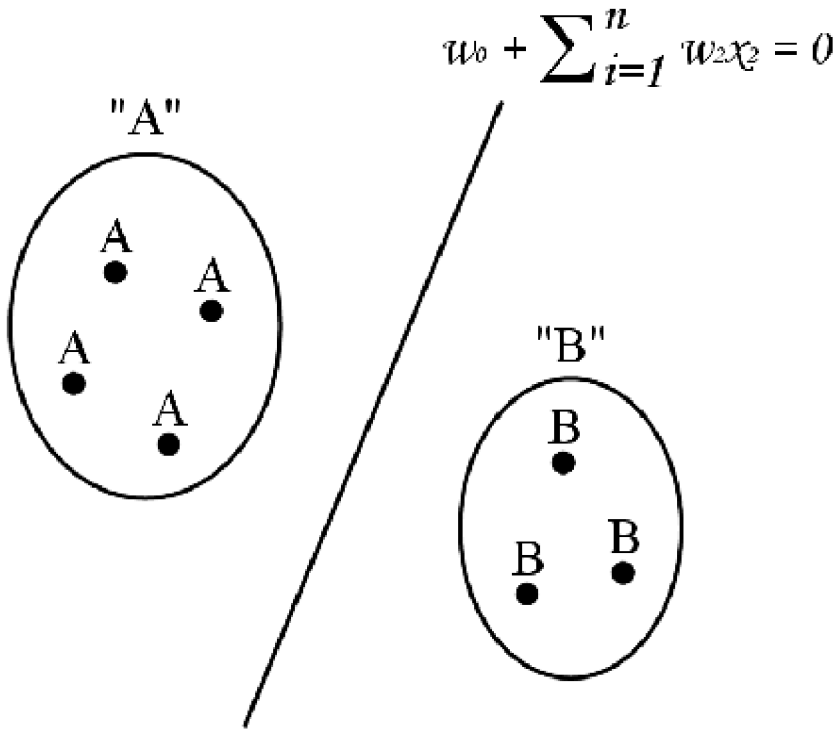


Obr. 7: Grafy prahových funkcí [1]

Takovýto model neuronu bývá označován jako perceptron.

3.3.2 Geometrická představa funkce neuronu

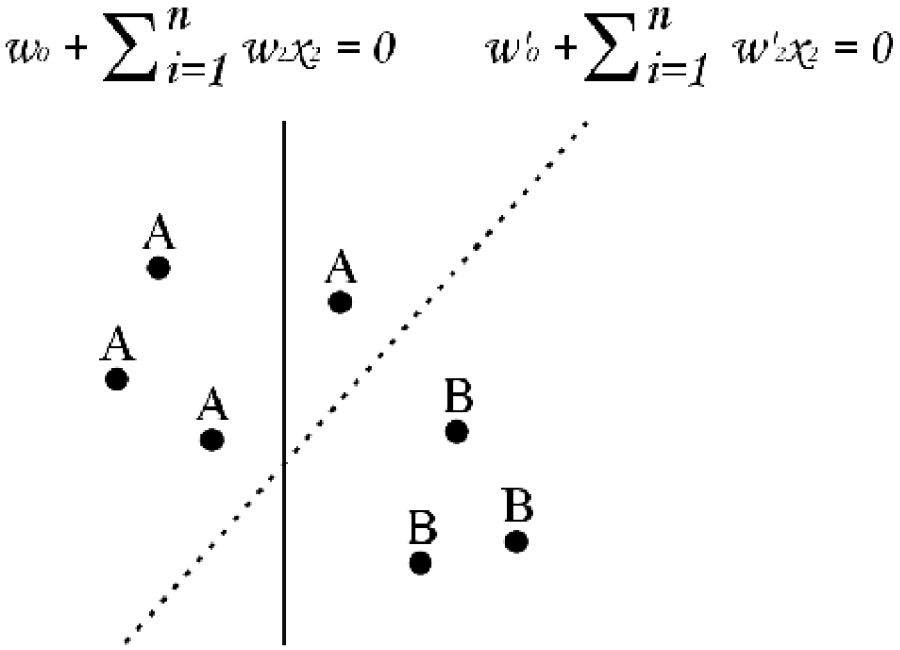
Funkci neuronu lze dobře ukázat na této geometrické představě. Vstup neuronu budeme chápat jako souřadnice stavové roviny, přičemž neuron zde představuje přímku, která tuto rovinu dělí na dvě poloroviny. Umístění této přímky je dáno právě nastavením synaptických vah neuronu. Souřadnice směřující do první poloroviny nastaví neuron do stavu „aktivní“, který nám bude označovat písmeno „A“. Souřadnice druhé poloroviny převedou neuron do stavu „pasivní“, který nám řekne, že se jedná o písmeno „B“.



Obr. 8: Rozdělení písmen A a B pomocí neuronu [1]

Podobně jako neurony v lidském mozku i umělé neurony se musí toto rozlišování nejprve naučit. Správné nastavení vah a tedy polohu přímky rozdělující rovinu zajistí některý s algoritmů výuky sítě. Síť se nejprve učí rozeznávat známá data. V našem případě známe rozdělení písmen „A“ a „B“. Protože známe výsledek, můžeme porovnáváním výstupu neuronu se skutečností postupně adaptovat synaptické váhy tak, aby výstup této skutečnosti odpovídal. Tento stav je znázorněn

na následujícím obrázku. Tečkovanou čarou je znázorněno chování naučeného neuronu.

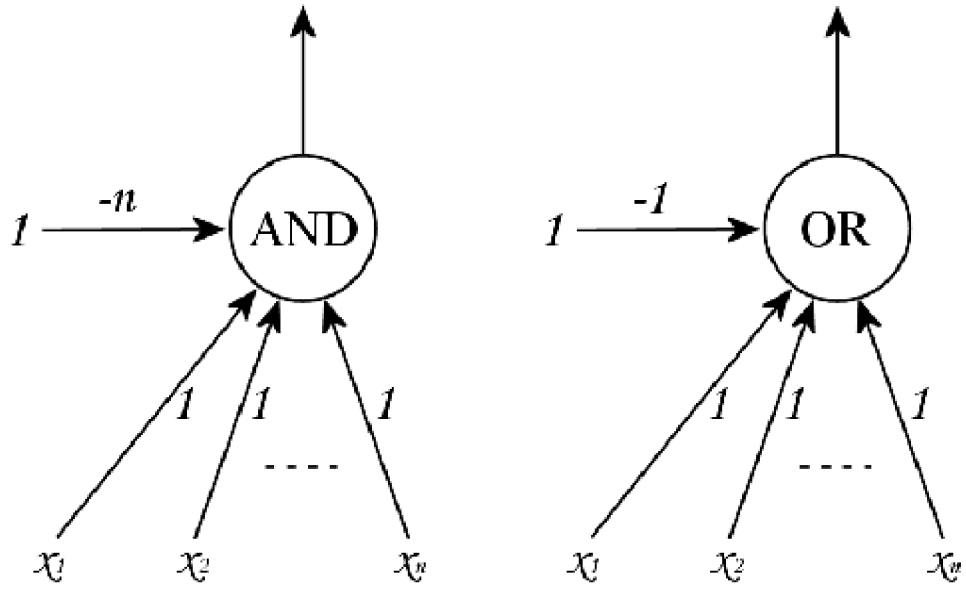


Obr. 9: Adaptace vah při chybném rozdělení písmen

3.3.3 Funkce neuronu v neuronové síti

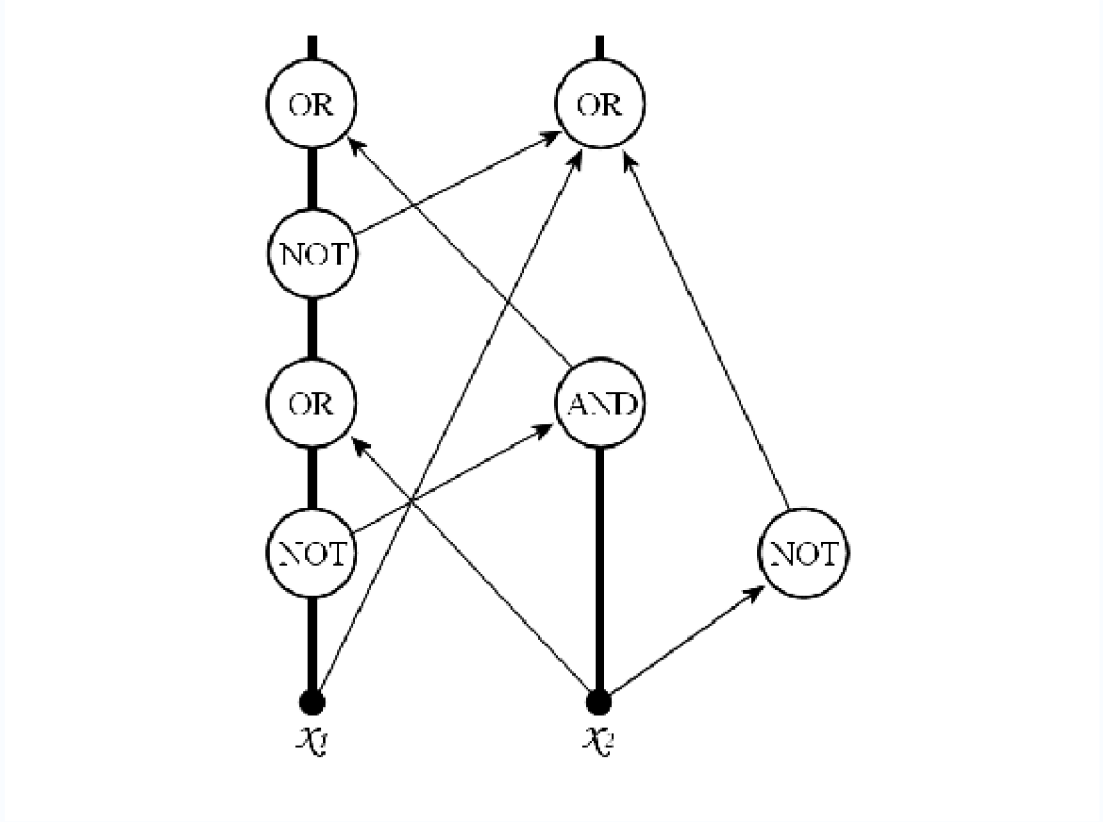
Pokud budeme uvažovat rovinu z předchozí podkapitoly z pohledu třívrstvé neuronové sítě, budou vstupní neurony první vrstvy dělit tuto rovinu na tolik různých polorovin, kolik neuronů bude obsahovat vstupní vrstva (získáme tedy více přímek rozdělující rovinu). Toto rozdělení nám umožní rozpoznávat více písmen než v předchozím přikladu s jedním neuronem. Neurony ve druhé vrstvě mohou k tomuto účelu určit průnik těchto polorovin (realizovat funkci AND) a tím reprezentovat konvexní oblasti. Protože ne vždy je možné požadovanou část vstupního roviny uzavřít do konvexní části, můžeme nekonvexní část získat jako sjednocení konvexních částí roviny. Tuto funkci nám zajistí neurony ve třetí, výstupní vrstvě, které budou realizovat funkci OR. Neurony realizující tyto logické funkce jsou

vyobrazeny na následujícím obrázku. Váhy vstupů jsou nastaveny na 1, aby aktivní binární vstupy odpovídaly číslu 1 na vstupu neuronu. Práh je u neuronu realizující funkci AND nastaven na –n, kde n je počet vstupů neuronu. Tím je zajištěno že neuron bude aktivní právě tehdy, když všechny vstupy neuronu jsou v 1. Práh neuronu realizujícímu funkci OR je nastaven na hodnotu -1. Neuron tedy bude aktivní když bude mít kterýkoliv jeho vstup hodnotu 1.



Obr. 10: Neurony realizující funkci AND a OR [1]

Z těchto neuronů reprezentujících logické funkce lze sestavovat logické obvody. Takové obvody mají strukturu acyklické neuronové sítě a jednotlivá hradla jsou reprezentovány neurony. Příklad logického obvodu je na obrázku č. 10.



Obr. 11: Příklad logického obvodu [1]

3.4 DRUHY NEURONOVÝCH SÍTÍ

Rozeznáváme několik základních druhů, neboli topologií, neuronových sítí. Topologie sítě je určena způsobem propojení jednotlivých neuronů v síti. Pro různé typy aplikací se hodí různé typy neuronových sítí.

V základu můžeme neuronové sítě rozdělit na dva druhy. Cyklické uspořádání neuronů a acyklické uspořádání neuronů.

3.4.1 Cyklická síť

V cyklické topologii neuronové sítě jsou neurony v síti zacykleny tím způsobem, že výstup prvního neuronu je přiveden zpět na jeho vstup. Tento výstup může být připojen buď přímo (tzv. zpětná vazba neuronu), nebo přes libovolný počet

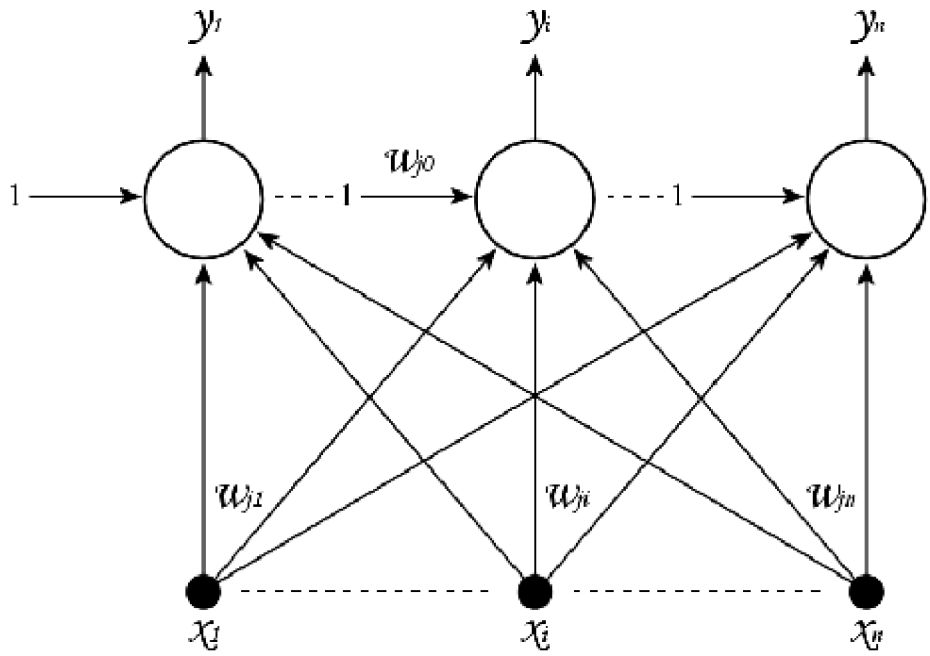
dalších neuronů. Je možné sestrojit také takovou síť, kde výstup každého neuronu v síti bude přiveden na vstup opět každého neuronu v síti.

3.4.2 Acyklická síť

Na rozdíl od cyklické sítě se acyklická síť dá uspořádat do vrstev. Taková topologie potom obsahuje vstupní a výstupní vrstvu a mezi nimi může být libovolný počet dalších, tzv. skrytých vrstev. Výstupy z nižších vrstev tohoto uspořádání jsou přivedeny na neurony v libovolných vyšších vrstvách.

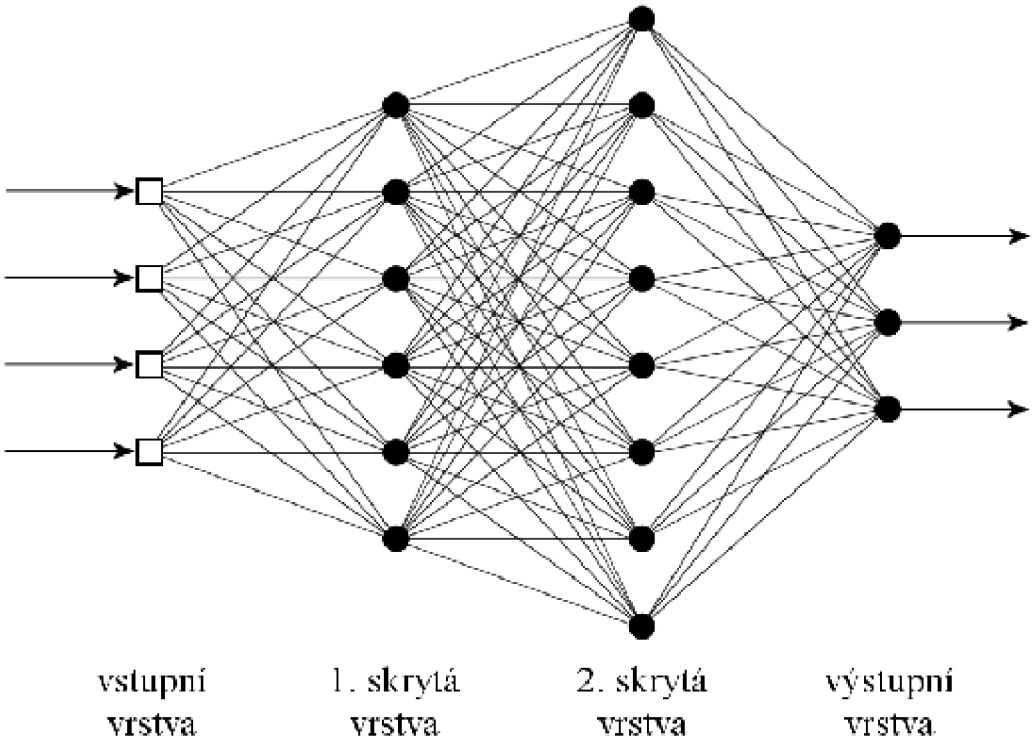
3.4.3 Síť perceptronů

První funkční model neuronové sítě byla síť perceptronů. Jedná se o jednoduchou síť kde výstup z každého z ‚n' vstupních neuronů je přiveden do každého z ‚m' výstupních neuronů.



Obr. 12: Síť perceptronů [1]

V režimu učení se adaptují váhy jednotlivých vstupů tak, aby se síť naučila rozpoznávat tréninková data. Tento model slouží jako základ pro vícevrstvou síť s algoritmem učení backpropagation.



Obr. 13: Vícevrstvá síť perceptronů [4]

3.4.4 Vícevrstvá síť s algoritmem backpropagation

V praxi nejčastěji používaný typ sítě. Algoritmus učení backpropagation (zpětné šíření chyby) existuje ve více různých variantách. Obecně jde o adaptaci vah sítě směrem od chybového výstupu postupně přes všechny vrstvy směrem k počátku (ke vstupu). Učení probíhá postupně ve třech krocích.

1. Do náhodně inicializované sítě se přivede vstup
2. Na základě porovnání požadovaného a skutečného výstupu sítě se určí chyby pro jednotlivé neurony výstupní a vnitřních (skrytých vrstev).
3. Pomocí těchto chyb se adaptují jednotlivé váhy

Chyba sítě se vypočítá jako součet čtverců rozdílu požadovaného a skutečného výstupu všech neuronů a vzorů, adaptace vah je dána parciální derivací této chyby podle váhy [1].

$$E_C = \frac{1}{2}\sum_{k=1}^{p}\sum_{j=1}^{r}\left(d_{kj} - y_{kj}\right)^2 \tag{2}$$

Adaptace vah je dána vztahem

$$w_{ji}^{(t)} = w_{ji}^{(t-1)} + \Delta w_{ji}^{(t)} \tag{3}$$

a $w_{ji}^{(t)}$ se vyjádří jako parciální derivace chyby podle váhy

$$\Delta w_{ji}^{(t)} = \frac{\partial E}{\partial w_{ji}}\left(w_{ji}^{(t-1)}\right) \tag{4}$$

$$\frac{\partial E}{\partial w_{ji}} = \sum_{k=1}^{p}\frac{\partial E_k}{\partial w_{ji}} \tag{5}$$

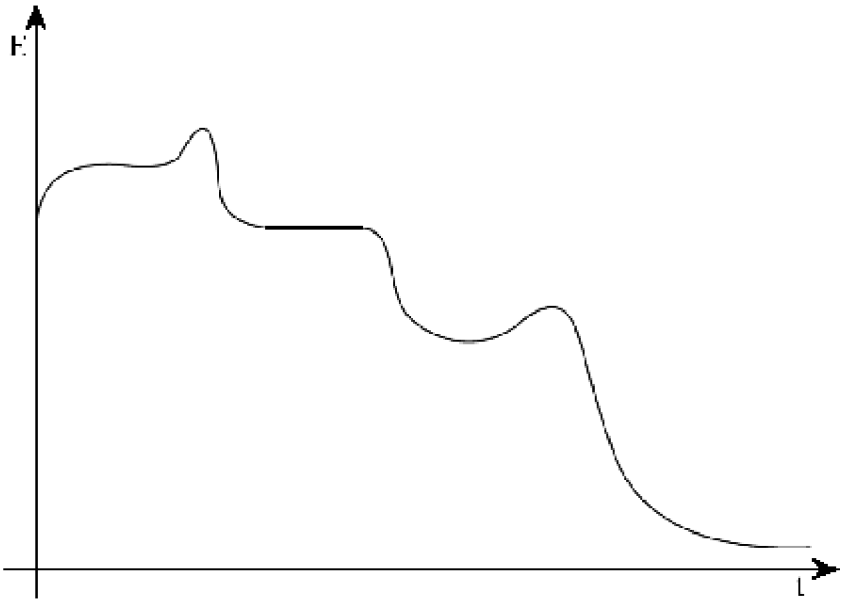
Pro váhy výstupní vrstvy platí

$$\frac{\partial E_C}{\partial w_{ji}^{(L-1)}} = \frac{\partial E_C}{\partial y_i^{(L)}}\frac{\partial y_i^{(L)}}{\partial \xi_i^{(L)}}\frac{\partial \xi_i^{(L)}}{\partial w_{ji}^{(L-1)}} = \left(d_i - y_i\right)\cdot y_i \cdot \left(1 - y_i\right)\cdot x_j = \delta_i x_j \tag{6}$$

Pro váhy mezi neurony skrytých vrstev platí

$$\frac{\partial E_C}{\partial w_{ji}^{(L-N-1)}} = \frac{\partial E_C}{\partial x_i^{(L-N)}}\frac{\partial x_i^{(L-N)}}{\partial \xi_i^{(L-N)}}\frac{\partial \xi_i^{(L-N)}}{\partial w_{ji}^{(L-N-1)}} = \sum_{n=1}^{r}\left(w_{ji}\delta_i\right)\cdot y_i \cdot \left(1 - y_i\right)\cdot x_j \tag{7}$$

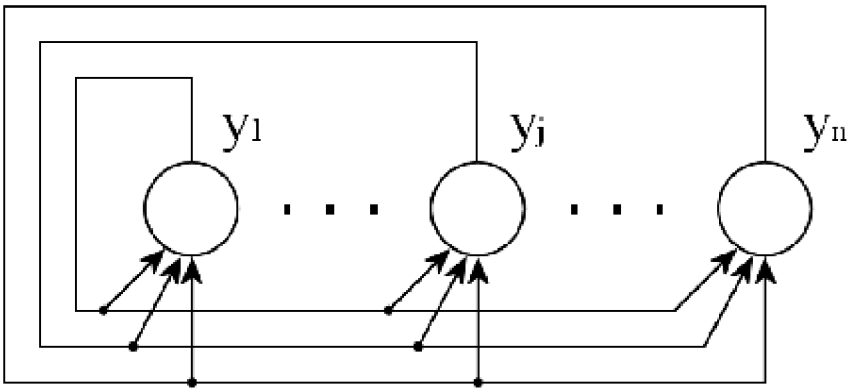
Metoda učení backpropagation dává podstatně lepší výsledky než jednoduchá adaptace vah. Při učení pomocí algoritmu backpropagation může chyba nějakou dobu stagnovat nebo dokonce krátkodobě vzrůst. To je způsobeno tím, že síť při učení konverguje ke globálnímu minimu a cestou prochází i řadou lokálních minim, ze kterých je možné se dostat jen za cenu krátkodobého zvýšení chyby. Proces učení je znázorněn na následujícím grafu.



Obr. 14: Závislost chyby sítě na čase při učení sítě pomocí algoritmu backpropagation [1]

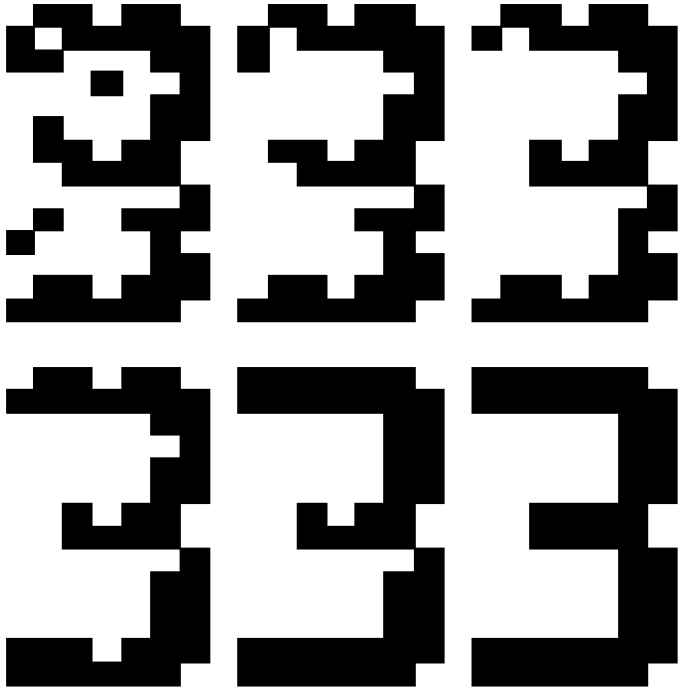
3.4.5 Hopfieldova síť

Hopfieldova síť je organizována jako jednovrstvá cyklická síť. Všechny vstupní neurony jsou zároveň i výstupní. Zároveň jsou na vstup každého neuronu přivedeny výstupy všech ostatních neuronů v síti.



Obr. 15: Topologie Hopfieldovy sítě [1]

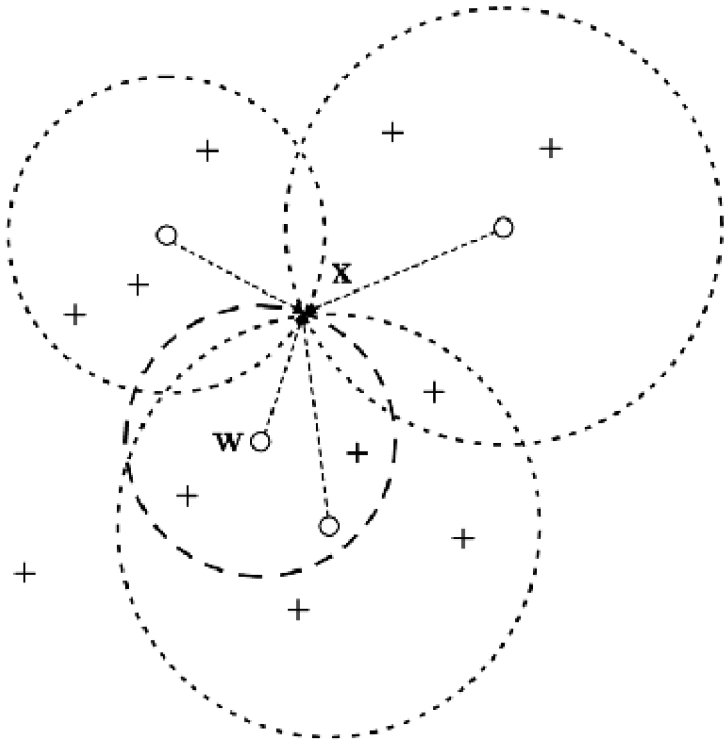
Hopfieldova síť bývá nejčastěji využívána jako asociativní paměť. Typické použití může být například rozpoznávání znaků. Pokud na vstup naučené sítě přivedeme poškozený (zašuměný) obraz znaku tak, že jednotlivé neurony v síti budou přiřazeny rozloženému obrazu, síť v postupných krocích odstraní z obrazu znaku šum a zrekonstruuje původní obraz znaku. Na rozdíl od vícevrstvé perceptronové sítě tedy nedává výsledek okamžitě, ale potřebuje nějaký čas na ustálení výstupu. Příklad takové rekonstrukce je vidět na následujícím obrázku.



Obr. 16: Rekonstrukce znaku Hopfieldovou sítí [1]

3.4.6 RBF síť

Jedná se o jiný, novější typ sítě. Neuron sítě RBF se liší od klasického perceptronu přenosovou funkcí. Neuvažuje se součin vah a vstupů, nýbrž vzdálenost vstupního vektoru od středu. Síť je genericky třívrstvá. Obsahuje vstupní vrstvu, která slouží pouze pro přenos vstupních hodnot. Skrytou vrstvu, která se skládá z RBF (Radial Basis functions – Radiální bazické funkce) jednotek. A nakonec výstupní vrstvu složenou z lineárních jednotek.



Obr. 17: Nastavování středů RBF [1]

RBF síť může dávat lepší výsledky i tam, kde perceptrony selhávají. Bohužel i naopak existují aplikace, ve kterých dávají lepší výsledky perceptronové sítě.

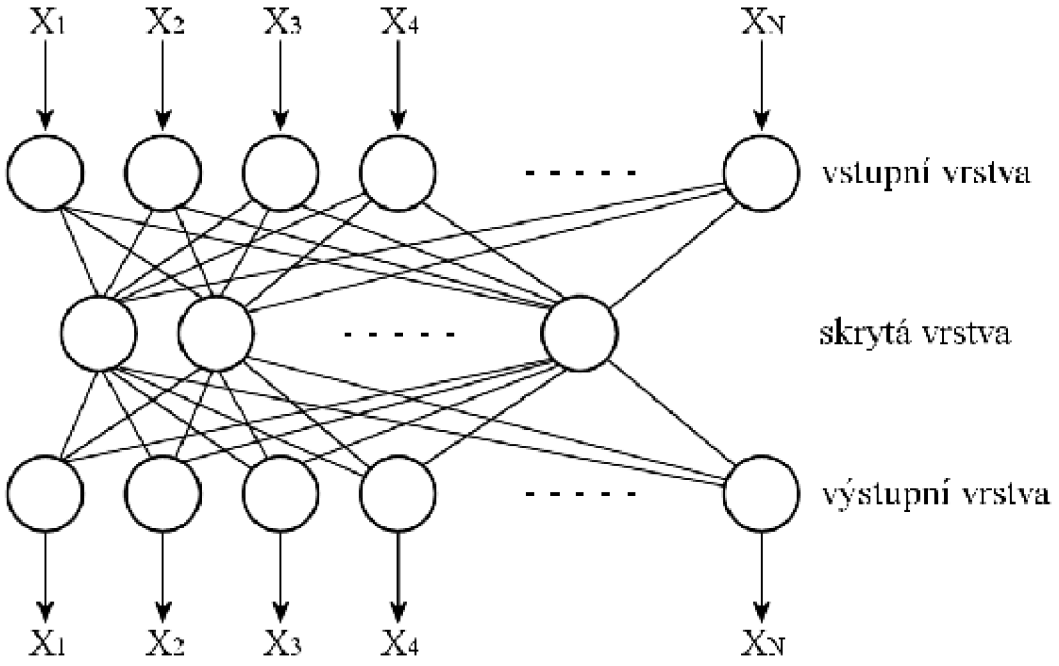
3.5 ZHODNOCENÍ VYBRANÝCH MODELŮ

Jednotlivé modely se od sebe liší především možnostmi využití. Dá se říci, že pro každý druh aplikace se hodí jiný druh a jiná topologie sítě. Pro kompresi dat, které se budeme následně věnovat, se hodí vícevrstvý perceptron a síť RBF. Naopak pro rozpoznávání znaků (aplikace OCR) se zase více hodí sítě inspirované Hopfieldovou sítí, Kohonenova samoorganizační mapa nebo RBF.

4. KOMPRESE POMOCÍ NEURONOVÝCH SÍTÍ

4.1 PRINCIP KOMPRESE

Pro kompresi se výborně hodí vícevrstvý perceptron. Počet vstupních a výstupních neuronů odpovídá vstupním datům. Počet neuronů ve skryté vrstvě potom určuje kompresní poměr. Je zřejmé, že pokud bude počet neuronů ve skryté vrstvě příliš malý, dojde ke ztrátě informace a tím ke ztrátové kompresi. Na druhou stranu, pokud bude počet neuronů příliš velký, bude kompresní poměr nevýhodný. Proto se komprese neuronovou sítí hodí spíš např. právě pro kompresi obrazu a videa, kde malá ztráta nemusí vadit.



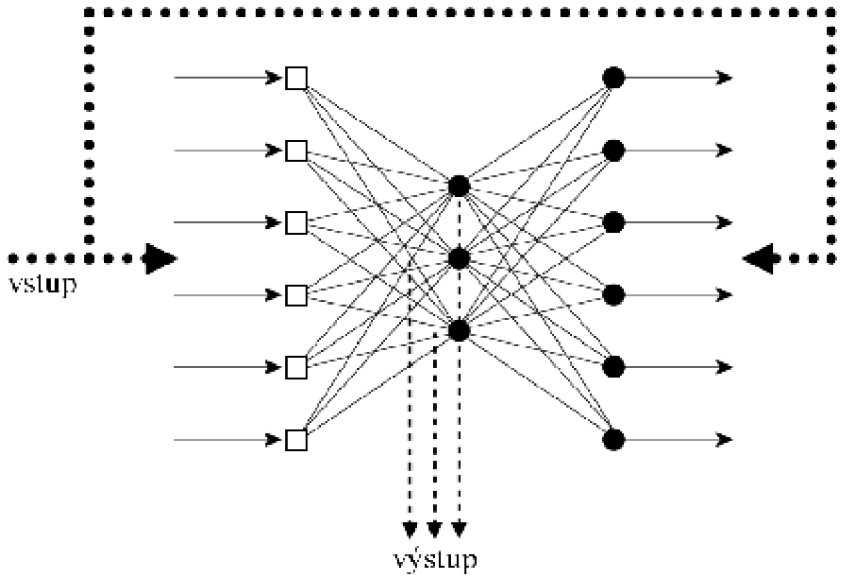
Obr. 18: Konfigurace sítě pro kompresi [1]

4.2 TEORETICKÝ ROZBOR

Jako nejvhodnější pro kompresi obrazu se jeví architektura sítí RBF nebo vícevrstvý perceptron a jejich schopnosti učení.

4.2.1 Koncepce komprese obrazu neuronovou sítí

Obrázek ke kompresi se nejprve rozloží na čtverce 8x8 a ty se potom zpracují neuronovou sítí. Navrhovaná síť bude mít 64 vstupních a 64 výstupních neuronů, kterým se přiřadí jednotlivé pixely ze čtverce 8x8. Skrytá vrstva bude obsahovat 16 neuronů. Zkomprimovaná data tedy budou uložena ve skryté vrstvě, což bude představovat kompresní poměr 4:1. Pokud bychom chtěli dosáhnout vyššího stupně komprese, museli bychom buď zmenšit počet neuronů ve skryté vrstvě, nebo o další neurony rozšířit vrstvy vstupů a výstupů. Ve správně naučené síti jsou data uložená ve skryté vrstvě pouze transformována do jiného vstupního rozměru. K vlastní kompresi dojde až při zaokrouhlení čísel mezi skrytou a výstupní vrstvou. To je způsobeno nedostatečnou přesností formátu reálných čísel.



Obr. 19: Návrh sítě pro kompresi obrazu – vícevrstvý perceptron [4]

Nejprve se síť naučí na tréninkovém obrazu. Po té co dosáhneme nejmenší odchylky od původního obrazu a síť bude tedy naučena, ji rozdělíme (jakoby rozsekneme) na dvě části. Na část kompresní a na část dekompresní.

Kompresní část bude reprezentována vstupní a skrytou vrstvou společně s jejich synapsemi. Pro vlastní kompresi nejprve přeneseme první část obrazu do vstupní vrstvy sítě. Na základě těchto vstupů a vah jednotlivých neuronů vnitřní (skryté) vrstvy sítě nastavíme stav všech neuronů skryté vrstvy. Po té tyto stavy uložíme jako komprimovaná data. Používáme tedy vlastně vnitřní vrstvu jako výstupní. Celý postup budeme opakovat se všemi částmi obrázku. Výsledná komprimovaná data můžeme přenést do dekompresní části sítě.

Dekompresní část bude obsahovat skrytou vrstvu a výstupní neurony. S vnitřní skrytou vrstvu budeme nyní pracovat jako se vstupem sítě a budeme do ní postupně vkládat dříve uložená komprimovaná data. Po výpočtech stavů výstupních neuronů budeme z výstupů postupně odečítat dekomprimované části obrazu a tak rekonstruovat celý obraz.

4.3 PŘEDPOKLÁDANÉ VÝSLEDKY

Ve správně naučené síti bude kvalita výsledného obrazu záviset především na vhodně stanoveném kompresním poměru, tedy na počtu neuronů ve vnitřní skryté vrstvě. Pokud bude tento počet příliš nízký, nebude síť schopna rekonstruovat původní obraz do všech podrobností.

5. PRAKTICKÁ REALIZACE

Jako programovací nástroj byl zvolen program Borland® Delphi® firmy Borland® Software Corporation. Výhodou jsou především široké možnosti tohoto vývojového prostředí a také to, že cílový přeložený kód nevyžaduje pro svůj běh žádné podpůrné knihovny a běží samostatně na běžném počítači s Microsoft® Windows®[1].

5.1 STRUČNÝ POPIS PROGRAMU

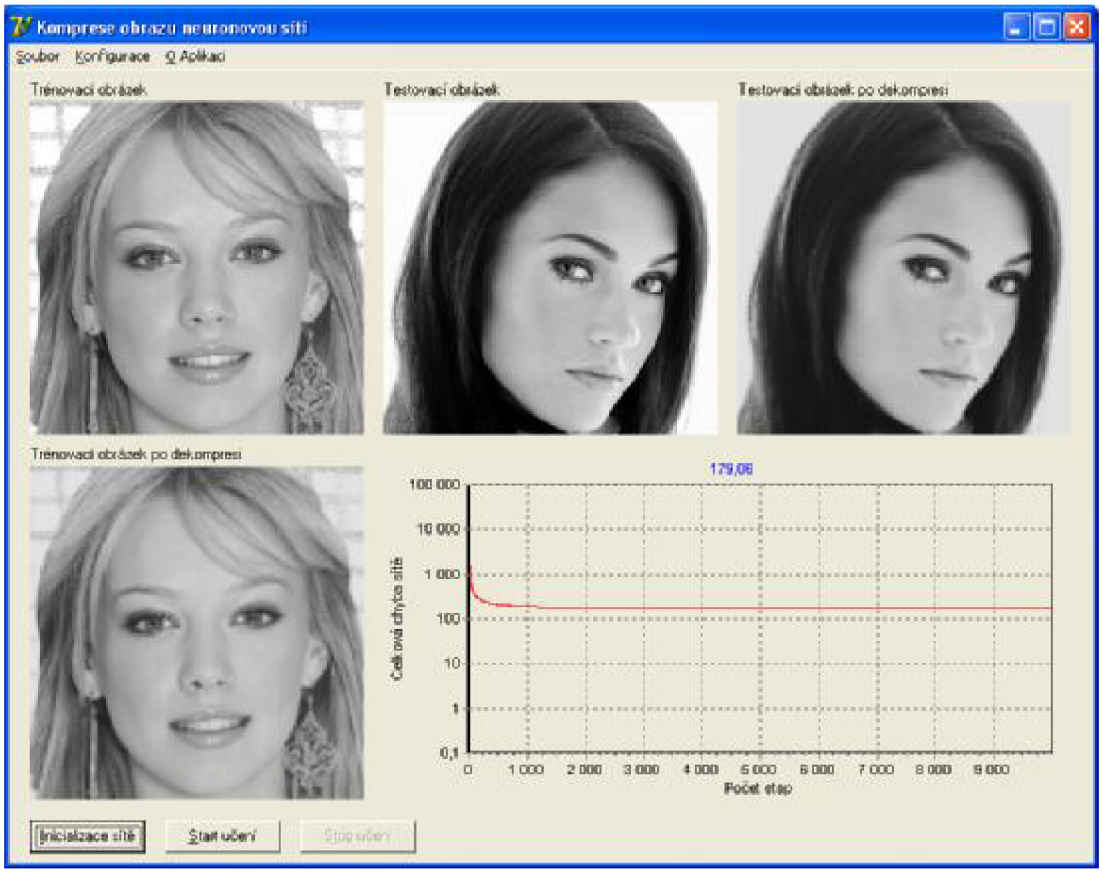
Program slouží k demonstraci využití umělé neuronové sítě, konkrétně vícevrstvého perceptronu, pro kompresi obrazu v odstínech šedi. V nastavení je možné měnit řadu parametrů včetně konfigurace sítě a zkoumat jejich vliv na kvalitu komprese. Program má velmi jednoduché ovládání. Všechny funkce jsou dosažitelné z hlavního menu, nebo přímo tlačítky v hlavní obrazovce. Při spuštění programu jsou automaticky načteny 2 obrázky. Jeden pro učení a jeden pro testování naučené sítě. Program je navržen tak, aby uměl zpracovat černobílé obrázky o rozměrech 256x256 pixelů. Parametry jsou automaticky nastaveny na standardní hodnoty. Kompresi lze ihned spustit tlačítkem „Start učení“ a kdykoliv pozastavit tlačítkem „Stop učení”. V průběhu komprese jsou průběžně zobrazovány oba obrazy po průchodu sítí, tedy procesem komprese a dekomprese, a také graf s celkovou chybou sítě.

5.1.1 Hlavní obrazovka

Na hlavní obrazovce jsou zobrazeny originály trénovacího a testovacího obrázku, dále pak výsledek jejich komprese a dekomprese neuronovou sítí a také graf průběhu učení v čase. V horní části obrazovky se nachází standardní menu s položkami pro načtení tréninkového a testovacího obrázku, pro konfiguraci programu a pro zobrazení stručné informace o aplikaci. V dolní části obrazovky jsou umístěna ovládací tlačítka pro inicializaci sítě, start učení a stop učení.

---

[1] Jmenované ochranné známky a značky jsou majetkem příslušných vlastníků.



Obr. 20: Hlavní obrazovka aplikace

5.1.1.1 Hlavní menu

Nabídka Soubor:

Načíst trénovaní obrázek – načte obrázek určený pro trénování sítě a zároveň provede jeho kompresi a dekompresi.

Načíst testovací obrázek – načte obrázek určený pro testování sítě a zároveň provede jeho kompresi a dekompresi.

Nabídka Konfigurace:

Konfigurace sítě – otevře konfigurační dialogové okno. V tomto okně lze nastavovat všechny modifikovatelné parametry aplikace. Rozměry předkládaných vzorů, topologii sítě, počty neuronů a tím i kompresní poměr, koeficient učení a podmínky pro ukončení výuky neuronové sítě.

Nabídka O Aplikaci:

Otevře dialogové okno se stručnými informacemi o aplikaci.

5.1.1.2 Ovládací tlačítka

Inicializace sítě – nastaví hodnoty vah všech neuronů na náhodné číslo v intervalu <-1;1>. Po té provede výpočet sítě pro oba obrázky a ty následně zobrazí. Graf průběhu učení je vymazán. Toto tlačítko lze použít jen pokud neprobíhá proces učení sítě.

Start učení – zahájí (nebo obnoví pozastavenou) výuku sítě. V průběhu učení jsou průběžně zobrazovány oba výsledné obrázky po průchodu sítí a také graf s celkovou chybou sítě.

Stop učení – zastaví proces výuky sítě. Ve výuce je možno pokračovat opětovným stiskem tlačítka „Start učení".

Při zastaveném učení je možné ověřit stav naučení sítě například načtením jiného testovacího obrázku.

5.1.2 Nastavení programu

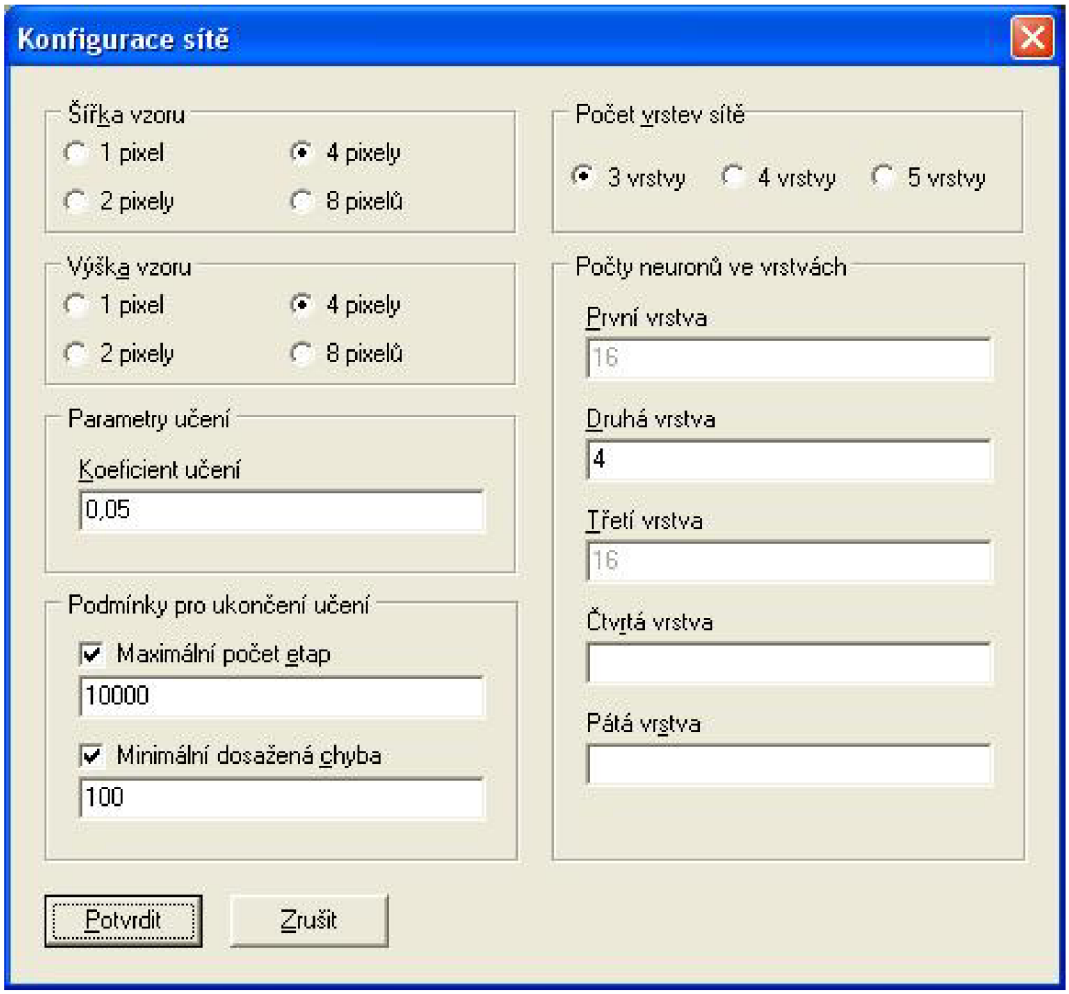
Veškerá nastavení programu se provádí v konfiguračním dialogu přístupném z hlavní nabídky.

Šířka vzoru – lze volit 1, 2, 4 nebo 8 pixelů. Definuje šířku vzorů, které budou předkládány síti při procesu učení. Přímo ovlivňuje také počty neuronů ve vstupní a výstupní vrstvě.

Výška vzoru – lze volit 1, 2, 4 nebo 8 pixelů. Definuje šířku vzorů, které budou předkládány síti při procesu učení. Přímo ovlivňuje také počty neuronů ve vstupní a výstupní vrstvě.

Počet vrstev sítě – lze přepínat mezi třemi, čtyřmi a pěti vrstvami sítě.

Počty neuronů ve vrstvách – definují počty neuronů pro každou vrstvu. Nastavovat lze jen počty neuronů ve skrytých vrstvách. Počet neuronů ve vstupní a výstupní vrstvě je dán počtem pixelů vzoru (tedy součinem šířky a výšky vzoru z konfiguračního dialogu).



Obr. 21: Konfigurační okno

Koeficient učení – určuje rychlost učení neuronové sítě.

Podmínky pro ukončení učení – definuje podmínky, za kterých je proces učení automaticky ukončen.

Maximální počet etap – pokud je aktivována tato volba, proces učení bude ukončen nejpozději při dosažení zadaného počtu etap.

Minimální dosažená chyba – pokud je aktivní tato volba, proces učení bude ukončen nejdéle po poklesu chyby sítě pod zadanou hodnotu.

Nastavení zvolených parametrů je nutné potvrdit tlačítkem „**Potvrdit**“. Tlačítko „**Zrušit**“ slouží k uzavření dialogového okna bez provedení jakýchkoliv změn.

5.2 VLIV KONFIGURACE SÍTĚ NA KVALITU OBRAZU

Na výslednou kvalitu komprimovaného obrazu má vliv mnoho faktorů. Výsledek závisí na konfiguraci sítě, kompresním poměru, koeficientu učení a také na druhu obrazu na který je síť naučena. V následujících podkapitolách jsou uvedeny příklady různých konfigurací a nastavení a jejich vliv na komprimovaný obraz.

V každém příkladu je pro srovnání uveden originální obrázek a potom tentýž obrázek komprimovaný s různými nastaveními testovaného aspektu. Pokud je to pro daný parametr zajímavé, je přiložen i graf průběhu učení neuronové sítě.

5.2.1 Rozměry vzoru

Na vstup neuronové sítě se komprimovaný obraz přenáší po částech, tzv. vzorech. Rozměry těchto částí mohou být voleny různě, avšak jejich volba má značný vliv na výslednou kvalitu. S volbou rozměrů souvisí i počet neuronů ve vstupní a výstupní vrstvě, který musí odpovídat počtu pixelů ve vzoru. V následujících příkladech je použita třívrstvá konfigurace sítě s kompresním poměrem 1:4. U každého obrázku jsou uvedeny rozměry vzoru v pixelech a počty neuronů v jednotlivých vrstvách sítě.

Při stejném počtu neuronů nemá volba rozměrů vliv na průběh učení sítě. Síť se učí přibližně stejně při rozměrech vzoru 4x4 nebo 8x2. Jiný průběh je například při volbě rozměrů sítě 8x8, ale to je způsobeno s tím související změnou topologie sítě, protože počet vstupních a výstupních neuronů musí odpovídat počtu pixelů ve vzoru. Následující obrázky demonstrují vliv nastavení topologie sítě na kvalitu komprese. Síť byla vždy učena desetitisíci etapami s koeficientem učení 0,05. Konkrétní topologie a počty neuronů v jednotlivých vrstvách jsou vždy uvedeny u každého obrázku.

Obr. 22: Originální obrázek

Obr. 23: 4x4 pixely (16-4-16)

Obr. 24: 8x2 pixely (16-4-16)

Obr. 25: 8x8 pixelů (64-16-64)

5.2.2 Topologie sítě

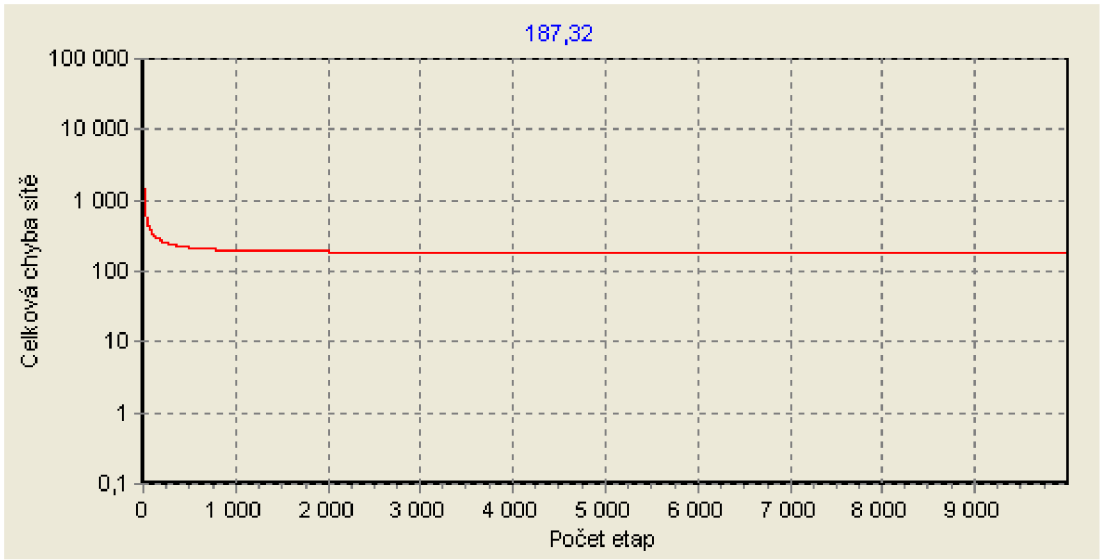
Na následujících obrázcích je zobrazen vliv různých topologií sítě na výslednou kvalitu obrazu a také na celkovou chybu sítě. Z obrázků a grafů rovněž vyplývá, že nízká chyba sítě ještě nezaručuje vyšší optickou kvalitu komprese.

Nalevo je vždy originální obrázek, napravo obrázek po kompresi a dekompresi. V popisku je uveden počet neuronů v jednotlivých vrstvách sítě.

5.2.2.1 Standardní konfigurace, 3 vrstvy, (16-4-16) neuronů

Obr. 26: Originální obrázek

Obr. 27: 4x4 pixely (16-4-16)

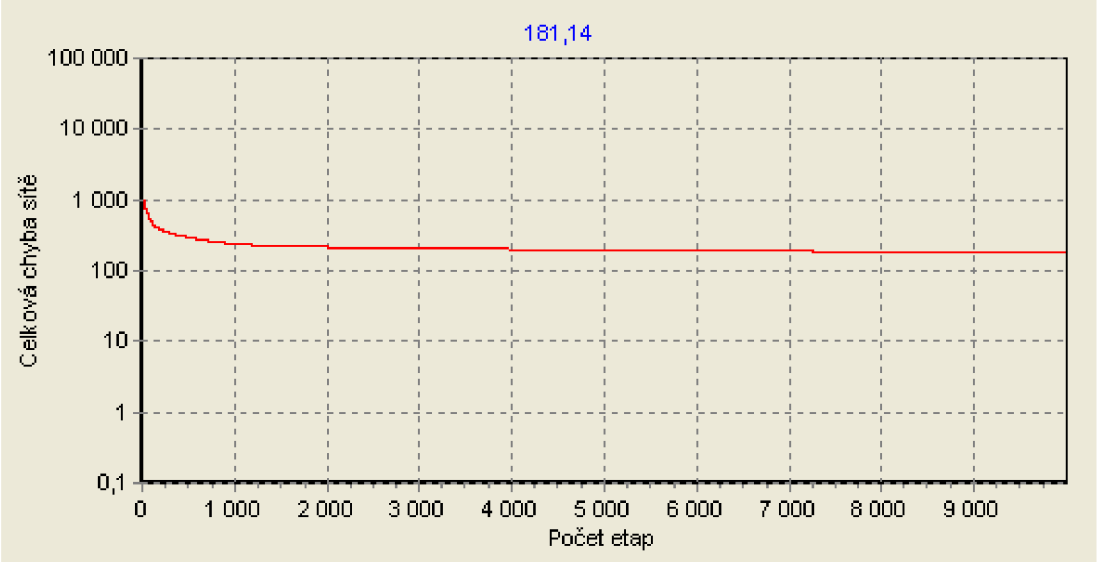


Obr. 28: Průběh učení sítě

5.2.2.2 5 vrstev, (16-32-4-32-16) neuronů

Obr. 29: Originální obrázek

Obr. 30: 4x4 pixely (16-32-4-32-16)

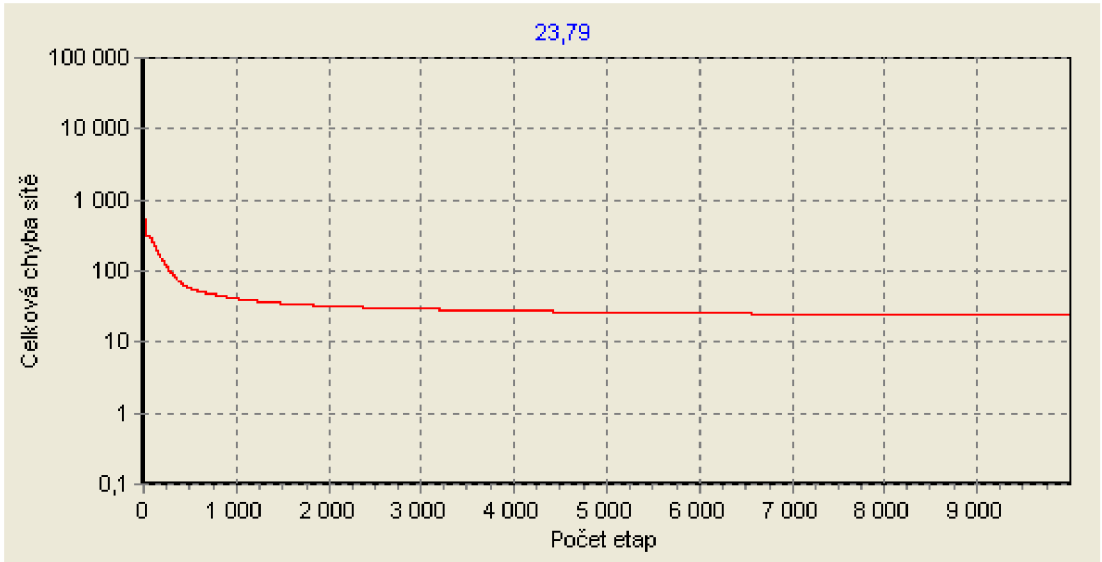


Obr. 31: Průběh učení sítě

5.2.2.3 5 vrstev, (16-32-32-4-16) neuronů

Obr. 32: Originální obrázek

Obr. 33: 4x4 pixely (16-32-32-4-16)

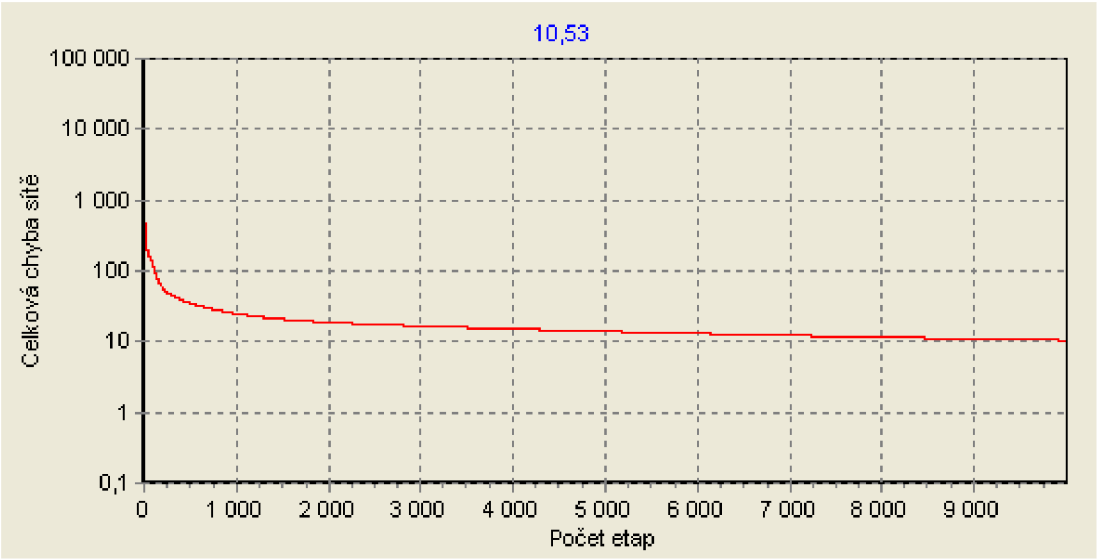


Obr. 34: Průběh učení sítě

5.2.2.4 4 vrstvy, (16-32-4-16) neuronů

Obr. 35: Originální obrázek

Obr. 36: 4x4 pixely (16-32-4-16)



Obr. 37: Průběh učení sítě

5.2.3 Kompresní poměr

Změny kompresního poměru dosáhneme změnou počtu neuronů v „kompresní“ skryté vrstvě. V následujícím srovnání je použito standardní nastavení aplikace s topologií sítě se třemi vrstvami a rozměry vzoru 4x4 pixely. Mění se tedy jen počet neuronů v prostřední vrstvě sítě.

Obr. 38: Originální obrázek

Obr. 39: 4x4 pixely (16-4-16) kompresní poměr 1:4

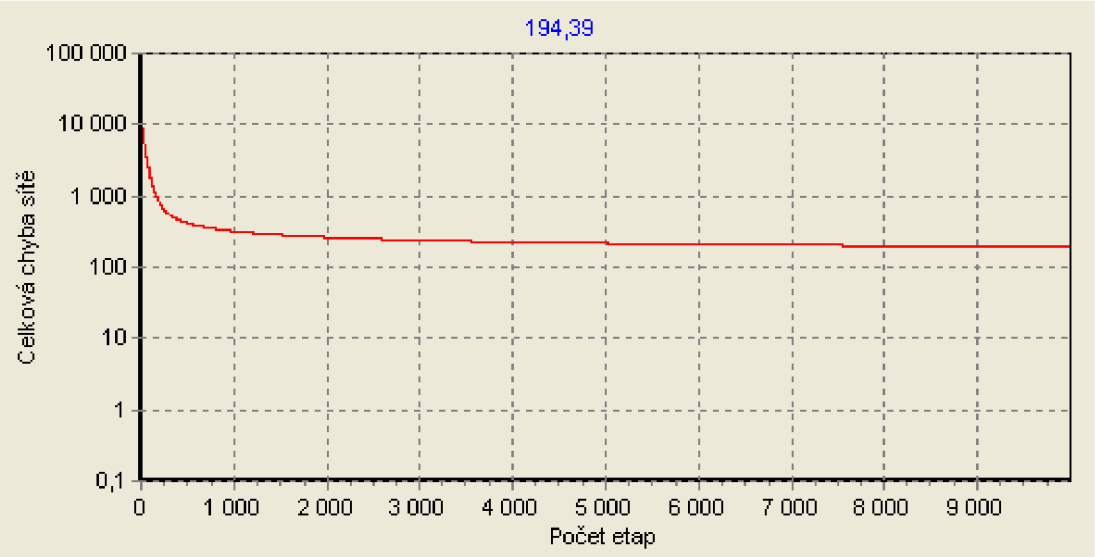
Obr. 40: 4x4 pixely (16-8-16) kompresní poměr 1:2

Obr. 41: 4x4 pixely (16-12-16) kompresní poměr 1:1,5

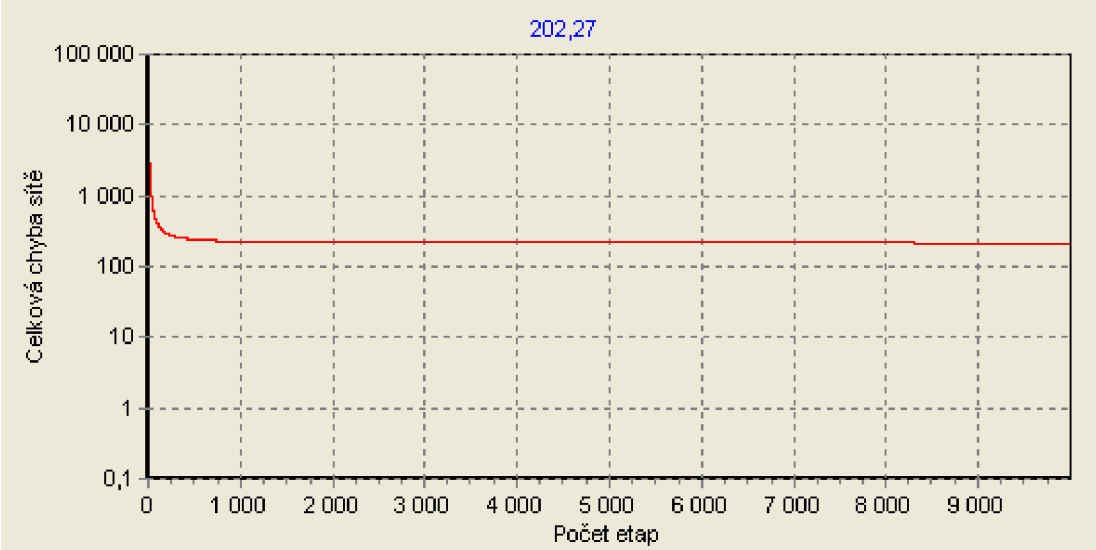
Z obrázků je patrné, že do určitého bodu má kompresní poměr poměrně značný vliv na výslednou kvalitu obrazu. Od jistého poměru však již lidské oko nezaznamená téměř žádný rozdíl. Při praktickém použití by tedy bylo ideální použít právě takový poměr, kdy se již kvalita obrazu významně nezvyšuje.

5.2.4 Koeficient učení

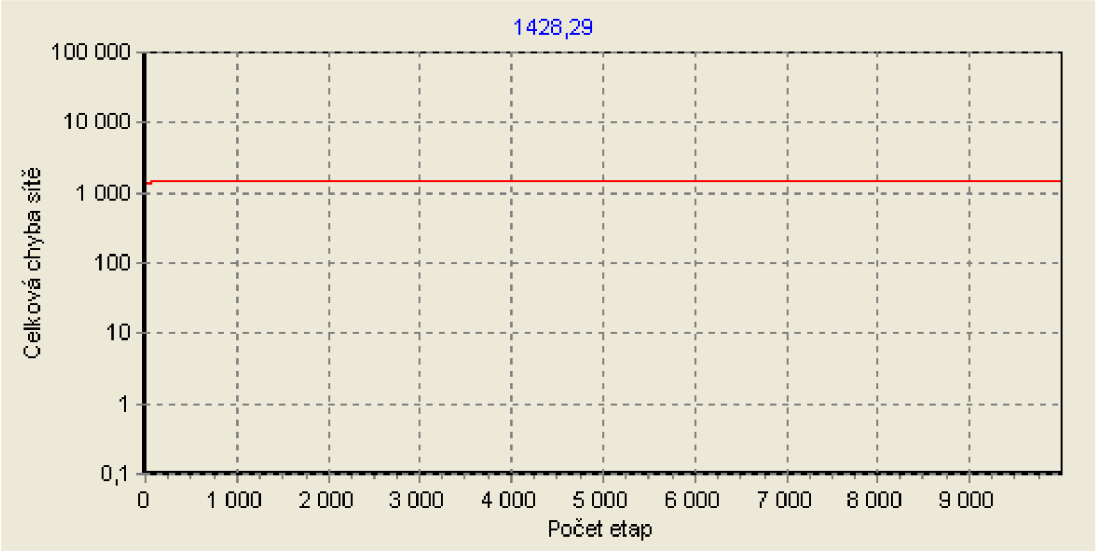
Koeficient učení má vliv na schopnost sítě nalézt globální minimum. Při implicitním nastavení programu se jako optimální jeví hodnota 0,01. Při nevhodně zvoleném koeficientu může učení uvíznout v lokálním minimu nebo chyba sítě vůbec nemusí konvergovat k nule.



Obr. 42: Průběh učení sítě (α=0,01)



Obr. 43: Průběh učení sítě (α=0,05)



Obr. 44: Průběh učení sítě (α=1,00)

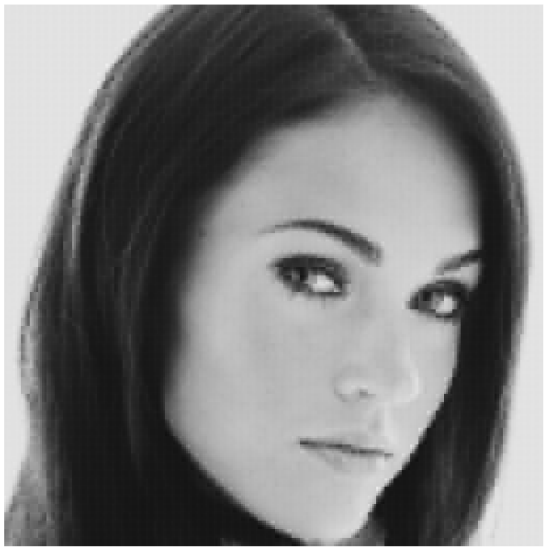
5.2.5 Volba tréninkového vzoru

U komprese neuronovou sítě záleží, na jaké tréninkové vzory je tato síť naučena. Následující skupina obrázků je komprimována sítí naučenou na první obrázek (fotografii tváře).

Obr. 45: Originál trénovacího obrázku

Obr. 46: Trénovaní obrázek

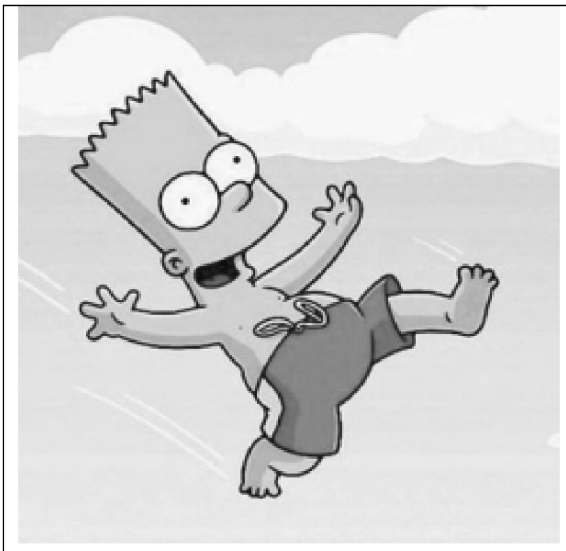
Obr. 47: Originál testovacího obrázku



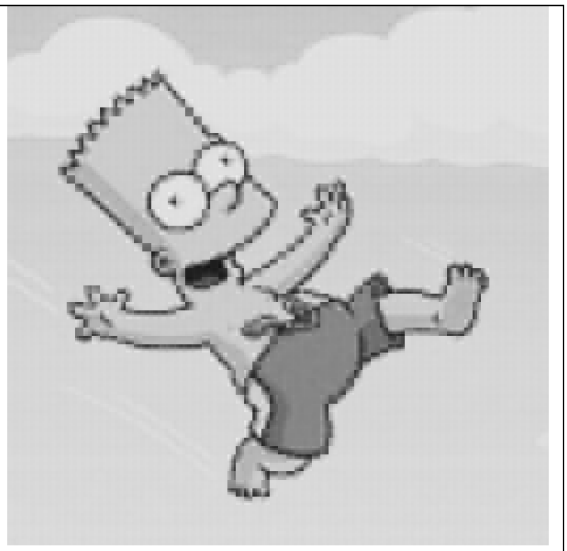
Obr. 48: Testovací obrázek

Obr. 49: Originál testovacího obrázku

Obr. 50: Testovací obrázek

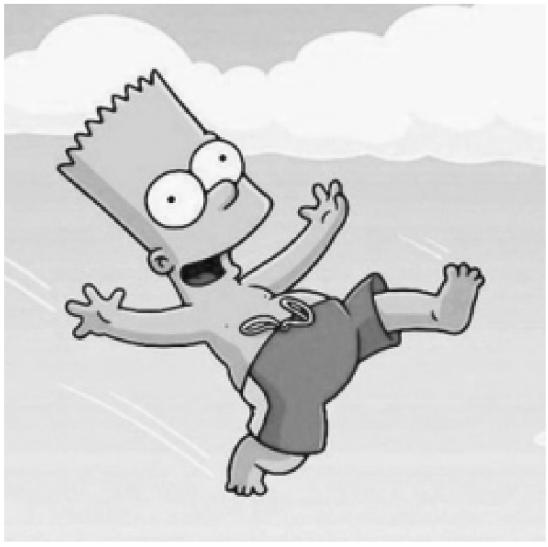


Obr. 51: Originál testovacího obrázku

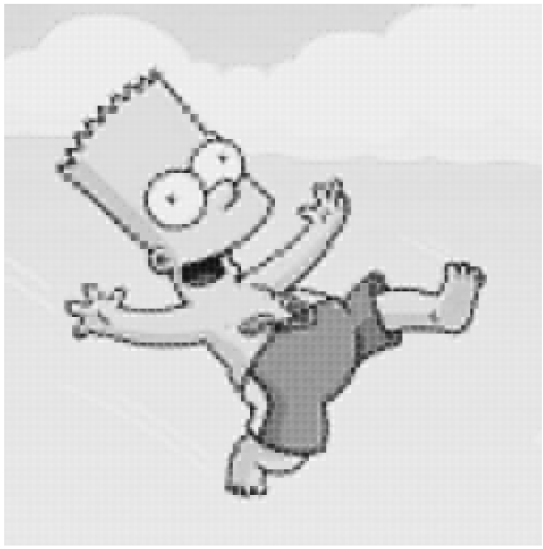


Obr. 52: Testovací obrázek

Tato skupina obrázků je komprimována sítí naučenou na kreslenou postavičku.



Obr. 53: Originál trénovacího obrázku



Obr. 54: Trénovaní obrázek

Obr. 55: Originál testovacího obrázku

Obr. 56: Testovací obrázek

Obr. 57: Originál testovacího obrázku

Obr. 58: Testovací obrázek

Obr. 59: Originál testovacího obrázku

Obr. 60: Testovací obrázek

5.3 POROVNÁNÍ S JINÝMI METODAMI KOMPRESE

Následující obrázky demonstrují kvalitu komprese jednotlivých metod. Pod každým obrázkem je uvedeno o jakou kompresi se jedná a jakou velikost měla výsledná data.

Obr. 61: Originální obrázek
256kB

Obr. 62: 4x4 pixely (16-4-16)
kompresní poměr 1:4
64kB

Obr. 63: JPEG (DCT)
13,5kB

Obr. 64: JPEG2000 (DWT)
7,5kB

Z obrázků je zřejmé, že kvalita ani délka výsledného kódu u obrázku komprimovaného neuronovou sítí není příliš dobrá ve srovnání s ostatními druhy komprese. To znovu potvrzuje skutečnost, že se neuronové sítě příliš nehodí pro kompresi statických obrázků. Výhoda této komprese se objeví především při kompresi pohyblivého obrazu. Naučenou neuronovou síť lze poměrně snadno hardwarově implementovat a zpracovávat tak obrovské objemy dat v reálném čase. Při pohybu obrazu také nebude jeho o něco nižší kvalita pro oko tak zřetelná. Vhodnou aplikací by mohla být zařízení pro přenos obrazu při videokonferencích.

6. ZÁVĚR

Nevýhodou komprese obrazu neuronovou sítí je především skutečnost, že komprimovat s rozumnou kvalitou půjdou pouze obrázky, které budou podobné tréninkovému obrazu, na který bude neuronová síť naučena. Z principu se dá předpokládat, že kvalita výsledného obrazu i tak nebude příliš vysoká. Pokud budeme brát v úvahu i dobu učení, tak dalším nepříjemným aspektem by mohla být doba komprese obrázku, která se dá předpokládat relativně vysoká. Z těchto důvodů proto neuronové sítě nebudou patřit mezi nejvhodnější nástroje pro kompresi statického obrazu.

Pokud se síť naučí na různé obrazy bude schopna s určitou přesností komprimovat i obrazy, na které nebyla naučena. Výpočetní náročnost je pouze při učení. Jakmile je síť již naučena, používá se pouze ve vybavovacím režimu. Architekturu sítě je možné paralelizovat a implementovat do hardwarových zařízení (FPGA) a její výhoda je právě v rychlosti zpracování. Kvalita komprese není nikterak oslňující a ztrátovost je relativně vysoká. Vzhledem k hardwarové implementaci je však možné ji nasadit pro zpracování obrovských toků dat, jako je například přenos videa a podob.

Na základě provedených experimentů se jako nejvýhodnější jeví topologie sítě se třemi vrstvami, šestnácti neurony ve vstupní a výstupní vrstvě a podle požadovaného stupně komprese čtyřmi až dvanácti neurony ve střední skryté vrstvě. Přidávání dalších vrstev do sítě mělo spíše negativní vliv. Kvalita komprese se zlepšovala úměrně se zvyšováním počtu neuronů ve střední vrstvě, avšak přibližně od kompresního poměru 1:2 už byl výsledný obraz natolik dobrý, že další přidávání neuronů nemělo žádný patrný vliv.

Síť se nejrychleji učila v průběhu prvních přibližně tří tisíc etap a učení delší než sedm tisíc etap mělo na chybu sítě už jen zanedbatelný vliv. Na počítači PC s procesorem taktovaným na 2,4GHz trvala doba učení deseti tisíc etap u třívrstvé sítě se třiceti šesti neurony přibližně pět minut.

7. LITERATURA

[1] ŠÍMA J., NERUDA R.: *Teoretické otázky neuronových sítí*. Matfyzpress, Praha 1996

[2] Obrázek neuronu - Wikipedie, URL: http://cs.wikipedia.org/wiki/Soubor:Neuron-cs.jpg [cit. 15.5.2009]

[3] NIKEL R: *Komprese dat s využitím vrstvených neuronových sítí*, Brno: Masarykova univerzita, Fakulta informatiky 1997. Diplomová práce 75s, vedoucí práce Petr Sojka

[4] Robert Dony, R.D., Haykin, S. Neural network approaches to image compression, Proceedings of the IEEE 83 (2) (1995) 288-303

[5] MALÝ, J.: *Srovnání metod pro ztrátovou kompresi obrazu*, Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2006.

8. SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

x_i	vstupy neuronu
y_i	skutečný výstup neuronu
d_i	požadovaný výstup neuronu (tréninková data)
w_{ij}	synaptické váhy
Θ	práh neuronu
ξ	potenciál neuronu
$S(x)$	přenosová funkce neuronu (aktivační funkce)
Y	výstup neuronu
E_C	celková chyba sítě

9. SEZNAM PŘÍLOH

- CD s elektronickou verzí BP a zdrojovým kódem programu pro kompresi obrazu neuronovou sítí.