

Univerzita Jana Evangelisty Purkyně
v Ústí nad Labem
Přírodovědecká fakulta



Vliv předzpracování obrazu a augmentace dat
na segmentaci rentgenových snímků

DIPLOMOVÁ PRÁCE

Vypracoval: Bc. Milan Gittler

Vedoucí práce: RNDr. Jíří Škvára, Ph.D.

Studijní program: Aplikovaná informatika

ÚSTÍ NAD LABEM 2024

Namísto žlutých stránek vložte digitálně podepsané zadání kvalifikační práce poskytnuté vedoucím katedry.

Zadání musí zaujímat právě dvě strany.

Zadání je nutno vložit jako PDF pomocí některého nástroje, který umožňuje editaci dokumentů (se zachováním elektronického podpisu).

V Linuxe lze například použít příkaz `pdftk`.

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně a použil jen pramenů, které cituji a uvádím v přiloženém seznamu literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona c. 121/2000 Sb., ve znění zákona c. 81/2005 Sb., autorský zákon, zejména se skutečností, že Univerzita Jana Evangelisty Purkyně v Ústí nad Labem má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Jana Evangelisty Purkyně v Ústí nad Labem oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladu, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

V Ústí nad Labem dne 19. dubna 2024

Podpis:

Děkuji vedoucímu práce RNDr. Jiřímu Škvárovi, Ph.D.
za neocenitelné rady a pomoc při tvorbě diplomové práce.

Abstrakt:

Hlavním cílem této diplomové práce je seznámit čtenáře s

Klíčová slova: lorem, ipsum, dolor, sit, amet

IMPACT OF IMAGE PREPROCESSING AND DATA AUGMENTATION ON SEGMENTATION OF X-RAY IMAGES

Abstract:

lorem ipsum dolor sit amet

Keywords: lorem, ipsum, dolor, sit, amet

Obsah

Úvod	13
1. Přehled metod předzpracování obrazu	15
1.1. Klasické metody předzpracování obrazu	15
1.2. Neuronové sítě pro předzpracování obrazu	26
1.3. Obecné postupy augmentace dat	32
2. Placeholder	33
3. Zhodnocení	35
4. Závěr	37
A. Externí přílohy	45

Úvod

1. Přehled metod předzpracování obrazu

V této kapitole se zaměříme na představení klíčových metod předzpracování obrazu, které hrají zásadní roli v procesu analýzy a zpracování rentgenových snímků. Předzpracování obrazu představuje kritický krok v řadě aplikací strojového učení a počítačového vidění, neboť ovlivňuje kvalitu a efektivitu následné analýzy. Metody předzpracování obrazu mohou výrazně zlepšit kvalitu dat a zvýšit přesnost detekce objektů, segmentace obrazu či klasifikace. Specifický výběr těchto technik je klíčový pro zjištění přesnosti a efektivy moderních metod počítačového vidění, včetně strojového učení a hlubokého učení, které jsou stále častěji aplikovány na širokou škálu problémů v oblasti zpracování obrazu. Zejména v kontextu rentgenových snímků může předzpracování pomoci překonat některé běžné výzvy, jako je nízký kontrast, šum, nebo artefakty, které mohou snížit kvalitu obrazu a tím ovlivnit diagnózu nebo automatickou analýzu.[1]

1.1. Klasické metody předzpracování obrazu

Následující sekce se zabývá představením klasických metod a technik, které jsou využívány pro úpravu a zlepšení kvality obrazových dat před jejich dalším zpracováním. Budeme se tedy věnovat různým přístupům filtrace obrazu, metodám redukce šumu či zaostření obrazu a také technikám prahování a binarizace, které jsou fundamentální pro analýzu a interpretaci obrazových dat. Přestože se jedná o metody, které mohou být považovány za základní, jejich správná aplikace a kombinace mohou výrazně zlepšit výslednou kvalitu obrazu a přispět k efektivnějšímu rozpoznávání vzorů a objektů v obrazových datech. Podrobně prozkoumáme každou z těchto metod, přičemž budeme klást důraz na jejich význam pro přípravu dat k dalšímu zpracování.

Vylepšení obrazu

Vylepšení obrazu je klíčovou technikou v předzpracování obrazu, zaměřenou na zlepšení vizuální kvality obrazových dat pro následné zpracování nebo analýzu. Vylepšení obrazu usiluje o zlepšení kontrastu, jasu a ostrosti, aby bylo zajištěno, že obrazová data jsou co nejvíce přístupná pro lidské vnímání nebo automatizované algoritmy. Jedním z klíčů k úspěšnému zlepšení obrazu je výběr vhodné metody a jejích parametrů, které musí být pečlivě nastaveny v závislosti na charakteristikách obrazových dat a konkrétním účelu zpracování.

Ekvalizace histogramu

Ekvalizace histogramu je fundamentální, ale mocná technika, která se používá pro zlepšení kontrastu v obrazových datech, zejména tam, kde původní obraz obsahuje špatně rozlišitelné detaily kvůli nedostatečnému rozsahu intenzit pixelů.

Cílem ekvalizace histogramu je aplikovat transformaci na původní histogram obrazu, $H(i)$, kde i představuje intenzitu pixelů v původním obrazu, tak, aby výsledný histogram měl uniformní rozložení. Tato transformace je založena na kumulativní distribuční funkci (CDF), $CDF(i)$, vy počítané z původního histogramu. $CDF(i)$ představuje součet pravděpodobností všech intenzit pixelů od nejnižší hodnoty až po intenzitu i a slouží jako mapovací funkce pro přiřazení nových intenzit pixelů ve výsledném obraze. Matematicky lze CDF definovat jako $CDF(j) = \sum_{i=0}^j P(i)$, kde $P(j)$ je pravděpodobnost výskytu intenzity j v původním obrazu, což se obvykle určí normalizací histogramu na celkový počet pixelů v obraze. Nová intenzita pixelu i' pro každý pixel s původní intenzitou i ve výsledném obraze je poté určena pomocí normalizované CDF, což zajišťuje, že všechny intenzity jsou rovnoměrně zastoupeny:

$$i' = (L - 1) \cdot CDF(i) \quad (1.1)$$

kde: L = je počet možných úrovní intenzity pixelů (např. $L = 256$ pro 8bitové obrazy) Tímto způsobem transformace zvýší kontrast obrazu tak, že "roztáhne" rozložení intenzit pixelů přes celý dostupný rozsah, což zvýrazní detaily a zlepší vizuální vnímání obrazu. Ekvalizace histogramu tak přináší výrazné zlepšení v oblastech s nízkým kontrastem a umožňuje lepší vizualizaci detailů, což je zásadní pro dalšího zpracování obrazu. [2] [3]

Algoritmus 1 Ekvalizace histogramu

- 1: **Vstup:** Původní obraz Img
 - 2: **Výstup:** Obraz s ekvalizovaným histogramem Img_{eq}
 - 3: Vypočítejte histogram H z Img
 - 4: Vypočítejte kumulativní histogram CH z H
 - 5: Normalizujte CH na rozsah intenzit pixelů obrazu
 - 6: **for** každý pixel p v Img **do**
 - 7: Nastavte $Img_{eq}[p]$ na hodnotu odpovídající $CH[Img[p]]$
 - 8: **end for**
 - 9: **return** Img_{eq}
-

Adaptivní ekvalizace histogramu

Adaptivní ekvalizace histogramu (AHE) představuje pokročilou metodu ekvalizace, která se snaží zlepšit kontrast obrazu lokálně, na rozdíl od globálního přístupu klasické ekvalizace histogramu. AHE algoritmus rozdělí původní obraz na malé, překrývající se bloky, nazývané dlaždice (tiles), a na každou z nich aplikuje ekvalizaci histogramu nezávisle. Tímto způsobem dokáže lépe zachytit lokální kontrastní charakteristiky obrazu a zvýraznit detaily v jednotlivých oblastech. Při překryvu

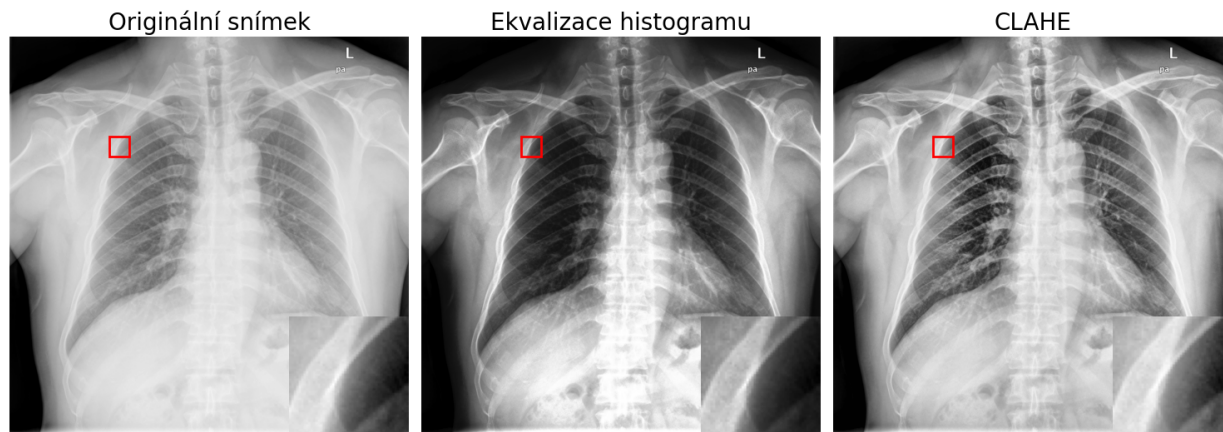
dlaždic se výsledné intenzity pixelů na okrajích vypočítají jako vážený průměr z odpovídajících intenzit získaných z každé příslušné dlaždice, což zajišťuje hladký přechod mezi dlaždicemi. [4]

Adaptivní ekvalizace histogramu s omezením kontrastu

Adaptivní ekvalizace histogramu s omezením kontrastu (CLAHE) byla vyvinuta s cílem předejít problémům spojeným s přílišným zvýrazněním šumu v homogenních oblastech obrazu, které nastává při použití AHE.

Základní myšlenka CLAHE spočívá v rozdělení obrazu na malé, kontextově závislé bloky a aplikování ekvalizace histogramu na každý z těchto bloků. Aby se zabránilo nežádoucímu efektu nadměrného zvýšení kontrastu, aplikuje se na histogram každého bloku proces „osekání“ (clipping), kdy hodnoty histogramu přesahující předem definovaný limit jsou sníženy na tento limit a přebytečné hodnoty jsou rovnoměrně rozděleny mezi ostatní úrovně intenzity. To vede k vytvoření vyváženějšího rozložení intenzit v obrazu a zajišťuje, že zvýšení kontrastu nevede k nežádoucímu zvýraznění šumu.

Jedním z klíčových aspektů CLAHE je výběr „clip limitu“, což je parametr, který omezuje míru zvýšení kontrastu. Tento limit se obvykle definuje jako násobek průměrné hodnoty histogramu a jeho správné nastavení je zásadní pro dosažení optimálního výsledku. [4]



Obrázek 1.1.1.: Ukázka metod ekvalizace histogramu na rentgenovém snímku. CLAHE demonstruje lepší zachování detailů a kontrastu v porovnání s tradiční ekvalizací histogramu.

Unsharp masking

Unsharp masking je technika určená k zlepšení ostrosti obrazu tím, že zvýrazňuje hrany a detaily, které jsou v původním obrazu méně patrné. Klíčovým krokem této metody je vytvoření tzv. masky ostrých detailů, která vznikne odečtením rozmazané verze obrazu od jeho původní

podoby. Rozmazaná verze obrazu se získává například pomocí Gaussova filtru, který je aplikován na původní obraz. Intenzita tohoto rozmazání určuje, jak silně budou hrany a detaily v masce zvýrazněny. Následující rovnice ukazuje získání výsledného, zaostřeného snímku pomocí techniky unsharp masking. Po získání masky ostrých detailů se tato maska přičte zpět k původnímu obrazu. Tento krok zvyšuje kontrast a ostrost obrazu, přičemž zvýrazňuje oblasti s výraznými rozdíly mezi sousedními pixely. [5]

Algoritmus 2 Unsharp Masking

- 1: **Vstup:** Původní obraz I , standardní odchylka Gaussova filtru σ , zesílení λ
 - 2: **Výstup:** Vylepšený obraz I'
 - 3: $G \leftarrow \text{Gauss}(I, \sigma)$ ▷ Vytvoření rozmazané verze obrazu pomocí Gaussova filtru
 - 4: $M \leftarrow I - G$ ▷ Vytvoření masky ostrých detailů odečtením rozmazaného obrazu
 - 5: $I' \leftarrow I + \lambda M$ ▷ Zvýšení ostrosti původního obrazu přičtením masky
-

V tomto procesu:

- σ určuje míru rozmazání při použití Gaussova filtru.
- λ je koeficient zesílení, který kontroluje míru, jakou jsou detaily zvýrazněny při přičítání masky k původnímu obrazu.

Při použití unsharp masking metody může vznikat nežádoucí halo efekt. Tento efekt se projevuje jako nežádoucí světelný okraj kolem kontrastních hran, což může vést k umělému a nepřirozenému vzhledu obrazu. Integrace filtru zachovávajícího hrany (edge-preserving filter) do procesu unsharp masking může být klíčová pro minimalizaci zmíněného efektu. [5]

Unsharp masking metoda je vhodná pro aplikace, kde je žádoucí zlepšit viditelnost detailů a hran. Důležitým aspektem při použití unsharp masking je správný výběr parametrů σ a λ , jelikož tyto hodnoty významně ovlivňují výsledný vzhled obrazu. Příliš vysoké hodnoty mohou vést k nežádoucím artefaktům, jako jsou halo efekty kolem hran, zatímco příliš nízké hodnoty mohou mít za následek nedostatečné zvýraznění detailů.

Gama korekce

Gamma korekce je technika používaná k úpravě jasu obrazových dat. Principem této metody je aplikace nelineární transformace na původní pixelové hodnoty obrazu, čímž se upravuje celková jasová křivka podle gamma funkce. Tato korekce umožňuje efektivněji využít dynamický rozsah zobrazovacích zařízení a zlepšit viditelnost detailů v tmavých i světlých oblastech obrazu bez ztráty informací v ostatních částech obrazového spektra. [6]

Aplikace gamma korekce se obvykle provádí podle vzorce:

$$I' = I^\gamma \tag{1.2}$$

kde: I = původní intenzita pixelu

I' = nová intenzita pixelu po aplikaci gamma korekce

γ = gamma faktor, který určuje míru korekce

Hodnota gamma faktoru menší než 1 zvýší jas tmavších oblastí obrazu, zatímco hodnota větší než 1 ztmaví světlejší oblasti a zvýší kontrast.

Redukce šumu

Redukce šumu ve snímcích je zásadní proces v digitálním zpracování obrazu, který má za cíl odstranit nebo minimalizovat vliv šumu přidaného během akvizice nebo přenosu obrazových dat. Šum může být způsoben různými externími faktory a může výrazně snížit kvalitu a čitelnost snímků, což má negativní dopad na jejich další analýzu a interpretaci.

Šum je obvykle definován jako náhodná variace intenzity obrazových bodů (pixelů), která se projevuje jako viditelné zrnění nebo textura na snímku. Tato náhodná variabilita může být důsledkem základních fyzikálních procesů, jako je povaha fotonu světla nebo tepelná energie v senzorech obrazu. Může vzniknout v okamžiku snímání nebo při přenosu obrazu.

Jednoduchý matematický model, kterým lze vyjádřit přítomnost aditivního šumu v obrazu, můžeme definovat jako:

$$f(x, y) = I(x, y) + N(x, y) \quad (1.3)$$

kde: $f(x, y)$ = je pozorovaná intenzita pixelu se šumem

$I(x, y)$ = je skutečná intenzita pixelu bez šumu

$N(x, y)$ = je hodnota šumu přidaná k pixelu

Redukce šumu je proto klíčovým krokem ve zpracování obrazu, který pomáhá zlepšit celkovou kvalitu snímků tím, že vyhlazuje obraz a odstraňuje nepřesnosti, avšak s opatrností, aby nedošlo k ztrátě důležitých detailů nebo hran v obrazu. Efektivní algoritmy pro redukci šumu se snaží dosáhnout rovnováhy mezi odstraněním šumu a zachováním nebo dokonce zvýrazněním důležitých vlastností obrazu, jako jsou hrany nebo textury. [7, 8]

Typy šumu ve snímcích

Existuje několik typů šumu, které se mohou ve snímcích vyskytovat. Tyto šумы se liší svým původem, charakteristikami a vlivem na kvalitu obrazu. Pro účinnou redukci šumu je klíčové rozpoznat, s jakým typem šumu máme co do činění, aby bylo možné zvolit nejvhodnější metodu pro jeho odstranění. Níže jsou uvedeny některé z nejčastějších typů šumu, které mohou ovlivnit digitální snímky.

- **Gaussovský šum** je pravděpodobně nejrozšířenějším typem šumu v digitálních obrazových systémech, způsobený především elektronickými fluktuacemi v senzoru a obvodech kamery.

Je charakterizován normálním rozdělením intenzit pixelů kolem skutečné hodnoty s určitou střední hodnotou a standardní odchylkou.

$$P(n) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(n-\mu)^2}{2\sigma^2}}$$

Gaussovský šum je obvykle považován za aditivní šum, což znamená, že jeho hodnota je nezávisle přidána k intenzitě každého pixelu. [7]

- **Impulzní šum (Šum soli a pepře)** se projevuje jako náhodné černé nebo bílé (nebo obojí) pixely rozptýlené po celém obrazu. Tento typ šumu může vzniknout kvůli chybám v senzoru nebo při přenosu dat. Na rozdíl od Gaussovského šumu je impulzní šum typicky nelineární a jeho redukce vyžaduje specifické nelineární filtrační techniky. [7]
- **Poissonův šum** je typ šumu, který je často přítomen v obrazových datech získaných z nízkých intenzit světla, jako jsou rentgenové snímky nebo mikroskopické snímky. Poissonův šum je důsledkem náhodného procesu počtu fotonů, které dopadají na senzor, a je charakterizován jako náhodný proces s diskrétním rozdělením pravděpodobnosti. Redukce Poissonova šumu vyžaduje specifické metody, které jsou schopny zacházet s diskrétními daty a zároveň zachovat důležité detaily v obraze. [7] Poissonův šum může být modelován jako:

$$P(n) \sim \frac{e^{-\lambda} \lambda^n}{n!}$$

- **Speckle šum** se řadí do skupiny multiplikativních šumů, které jsou často přítomny v ultrazvukovém a radarovém zobrazování. Tento šum je důsledkem koherence vlnění použitého pro snímání a interferencí mezi rozptýlenými vlnami. Speckle šum může být modelován pomocí Rayleighova nebo Gamma rozdělení v závislosti na povaze obrazových dat. [7]

Typy filtrů pro redukci šumu

Pro efektivní redukci šumu ve snímcích jsou k dispozici různé typy filtrů, které se liší svým principem a účinností v závislosti na konkrétním typu šumu. Níže jsou uvedeny některé z nejčastěji používaných filtrů pro redukci šumu ve snímcích.

- **Lineární filtrování**, také známo jako konvoluční filtrování, aplikuje lineární operátor na skupinu sousedících pixelů v obrazu s cílem vyhladit nebo zvýraznit určité vlastnosti obrazu. Hlavní principem lineárních filtrů je konvoluce obrazu s jádrem filtru či maskou, což je malá matice daného rozměru, která určuje váhy pro každý pixel v okolí. Tyto váhy určují, jakým způsobem okolní pixely ovlivňují hodnotu pixelu výstupního obrazu. Hlavní výhodou lineárních filtrů je jejich jednoduchost a efektivita, což je dělá vhodnými pro rychlé a efektivní zpracování obrazu. Nicméně, lineární filtry mohou způsobovat neostrost hran a detailů ve snímcích. [7, 9]

$$I_{filtered}(x, y) = \sum_{i=-a}^a \sum_{j=-b}^b K(i, j) \cdot I(x - i, y - j) \quad (1.4)$$

kde: $I_{filtered}(x, y)$ = intenzita pixelu na pozici (x, y) ve filtrovaném snímku

$K(i, j)$ = koeficienty kernelu

$I(x - i, y - j)$ = intenzita pixelu v původním snímku v okolí pixelu (x, y)

- **Nelineární filtrování** je alternativní přístup k redukci šumu, který se zaměřuje na zachování hran a detailů v obraze. Nelineární filtry jsou schopny zacházet s nelineárními vztahy mezi pixely a jsou schopny lépe zachovat důležité strukturální vlastnosti obrazu. Nelineární filtry jsou proto obzvláště účinné při redukci impulzního či spekle šumu. [7, 9]

Gaussovský filtr

Gaussovský filtr reprezentuje jeden z nejčastěji používaných lineárních filtrů pro redukci šumu ve snímcích. Tento filtr je založen na aplikaci Gaussova jádra na obraz, což umožňuje odstranit vysokofrekvenční šum. Gaussovský filtr je založen na principu aplikace Gaussovy funkce na vstupní obraz, která počítá vážený průměr okolí každého pixelu, s nejvyšší vahou soustředěnou na samotný pixel a se zmenšujícími se váhami pro sousední pixely na základě jejich prostorové vzdálenosti od středového pixelu. Gaussovský filtr je založen na konvoluci obrazu s Gaussovým jádrem, které je definováno jako:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (1.5)$$

kde: $G(x, y)$ = hodnota Gaussova jádra na pozici (x, y)

σ = standardní odchylka Gaussova jádra

Gaussovský filtr je schopen efektivně redukovat Gaussovský šum a jednou z hlavních výhod je jeho přirozená odezva na vysokofrekvenční oblasti, které potlačuje bez vytváření kroužkových artefaktů, které jsou běžné u low-pass filtrů.

Přestože je Gaussovský filtr velmi účinný při redukci šumu, volba směrodatné odchylky σ je kritická. Příliš vysoká hodnota může vymazat důležité detaily spolu se šumem, zatímco příliš nízká hodnota nemusí šum dostatečně snížit. Navíc, izotropní povaha Gaussovského filtru znamená, že filtr rozmazává ve všech směrech stejnou měrou, což nemusí být žádoucí zejména tam, kde je potřeba zachování hran. [9]

Bilaterální filtr

Bilaterální filtrování je nelineární technika pro vyhlazování obrazu, která zachovává hrany a redukuje šum. Na rozdíl od tradičních lineárních filtrů, které aplikují uniformní konvoluci napříč

Algoritmus 3 Gaussovský filtr

```

1: Vstup: Původní obraz  $I$ , standardní odchylka  $\sigma$ 
2: Výstup: Vyhlazený obraz  $I_{smoothed}$ 
3: Vytvořte Gaussovo jádro  $G$  s rozměry  $n \times n$  a  $\sigma$ 
4: Inicializuj  $I_{smoothed}$  jako prázdný obraz
5: for každý pixel  $p$  v  $I$  do
6:     Aplikujte Gaussovo jádro na okolí pixelu  $p$ 
7:     Nastavte hodnotu pixelu  $p$  v  $I_{smoothed}$  na vážený průměr hodnot v okolí
8: end for
9: return  $I_{smoothed}$ 

```

obrazem, bilaterální filtr kombinuje dva hlavní koncepty: prostorovou blízkost a podobnost intenzity.

Podobně jako Gaussův filtr, bilaterální filtr používá Gaussovu funkci k vážení pixelů založené na jejich prostorové vzdálenosti od centrálního pixelu. Tím pádem, čím jsou pixely blíže k centrálnímu pixelu, tím větší váhu mají. Unikátní pro bilaterální filtr je fakt, že váhy jsou také přizpůsobeny na základě rozdílu intenzit mezi sousedními pixely a centrálním pixelem. Pixely s podobnou intenzitou jako centrální pixel získávají vyšší váhu, což umožňuje zachování ostrých hran a detailů. Pro každý pixel ve zpracovávaném obrazu filtr vyhodnotí jeho sousedy a aplikuje váhy založené na dvou zmíněných kritériích.[10]

Matematický vzorec pro bilaterální filtr aplikovaný na pixel p v obrazu I lze vyjádřit jako:

$$BF[I](p) = \frac{1}{W_p} \sum_{q \in S} G_{\sigma_s}(\|p - q\|) \cdot G_{\sigma_r}(|I_p - I_q|) \cdot I_q \quad (1.6)$$

kde: $BF[I](p)$ = hodnota pixelu po aplikaci bilaterálního filtru

W_p = je normalizační faktor zajišťující, že součet vah je roven jedné

G_{σ_s} = prostorová Gaussova funkce, která snižuje vliv vzdálených pixelů na základě prostorové vzdálenosti $\|p-q\|$ se standardní odchylkou σ_s

G_{σ_r} = rozsahová Gaussova funkce, která redukuje vliv pixelů q na základě rozdílu intenzit $|I_p - I_q|$ se standardní odchylkou σ_r

I_p = hodnota pixelu p

I_q = hodnota pixelu q

$$W_p = \sum_{q \in S} G_{\sigma_s}(\|p - q\|) \cdot G_{\sigma_r}(|I_p - I_q|) \quad (1.7)$$

Na rozdíl od Gaussova filtru, který aplikuje vyhlazování uniformně a může vést k rozmazání ostrých hran, bilaterální filtr nabízí sofistikovanější přístup k redukci šumu. Jeho schopnost

zachovat ostré hrany při současném snížení šumu v homogenních oblastech obrazu činí bilaterální filtr vysoce ceněným nástrojem v pokročilém zpracování obrazu. Avšak, s touto schopností přichází zvýšená výpočetní náročnost a potřeba pečlivého nastavení parametrů σ_s a σ_r , což jsou aspekty, které je třeba zvážit pro dosažení optimálních výsledků. [10]

Wienerův filtr

Wienerův filtr je adaptivní lineární filtr, který je často používán pro redukci aditivního šumu ve snímcích s cílem zachovat důležité detaily a strukturální vlastnosti obrazu. Wienerův filtr funguje na principu minimalizace střední kvadratické chyby (anglicky minimum mean squared error - MMSE)¹ mezi původním obrazem a obrazem se šumem. Matematicky je Wienerův filtr navržen tak, aby našel spektrální rovnováhu mezi inverzním filtrováním signálu a vyhlazováním šumu, s ohledem na výkonová spektra signálu a šumu. Odezva filtru je vypočítána pomocí výkonových spekter, což vede k optimálnímu vyvážení mezi inverzním filtrováním pro obnovu obrazu a potlačením šumu.

$$G(u, v) = \frac{1}{H(u, v)} \left[\frac{|H(u, v)|^2}{|H(u, v)|^2 + \frac{S_\eta}{S_f}} \right] \quad (1.8)$$

kde: $G(u, v)$ = odhad originálního obrazu

$H(u, v)$ = degradační funkce

S_η = výkonové spektrum šumu

S_f = výkonové spektrum originálního obrazu

Tento vztah představuje schopnost Wienerova filtru vyvažovat mezi inverzním filtrováním a vyhlazováním šumu s ohledem na degradační funkci a poměr signálu a šumu $\frac{S_\eta}{S_f}$ ². Filtr je více nakloněn k inverznímu filtrování, kde je magnituda degradační funkce velká vzhledem k šumu (vysoké SNR), a více k vyhlazování v opačném případě. [11]

Mediánový filtr

Medianový filtr je robustní nelineární filtrační technika, který se používá k redukci impulzního šumu (šumu soli a pepře). Medianový filtr funguje tak, že posouvá okno po každém pixelu obrazu, shromažďuje hodnoty pixelů z okolí daného pixelu a nahrazuje středový pixel mediánem těchto hodnot. Okno, které se obvykle používá, je čtvercové (např. 3x3, 5x5) v závislosti na stupni šumu a požadované úrovni zachování detailů. Na rozdíl od průměrových filtrů, které také snižují šum, ale mají tendenci rozmazávat hrany, medianový filtr hrany zachovává. Dále také medianový filtr nezávisí na předpokladu, že šum je specifického typu nebo variance.

¹MSE je metrika, která měří rozdíl mezi odhadovanými a skutečnými hodnotami. V případě Wienerova filtru se minimalizuje rozdíl mezi původním obrazem a obrazem se šumem.

²Poměr signálu a šumu (SNR) udává poměr mezi obrazem a úrovní šumu v daném obraze. Vysoká hodnota SNR

Algoritmus 4 Medianový filtr

```
1: Vstup: Původní obraz  $I$ , velikost okna  $k$ 
2: Výstup: Obraz po redukci šumu  $I_{filtered}$ 
3: Inicializuj  $I_{filtered}$  jako prázdný obraz
4: for každý pixel  $p$  v  $I$  do
5:   okolí  $\leftarrow$  extrahuj_okolí( $I, p, k$ )
6:   medián  $\leftarrow$  vypočti_medián(okolí)
7:   Nastavte hodnotu pixelu  $p$  v  $I_{filtered}$  na medián
8: end for
9: return  $I_{filtered}$ 
```

Přestože je medianový filtr účinný, má svá omezení, zejména při vysokých úrovních šumu nebo když mají obrazy jemné strukturální detaily. Za těchto podmínek může filtr odstranit užitečné informace spolu se šumem. Navíc jeho výkon se může zhoršit, pokud není velikost okna správně zvolena, protože příliš velké okno může odstranit příliš mnoho detailů, zatímco příliš malé okno nemusí efektivně odstranit dostatek šumu.

Filtr ne-lokálních průměrů

Filtr ne-lokálních průměrů, anglicky Non-Local Means (NLM), představuje pokročilou techniku pro odstranění šumu z obrazu (převážně gaussovského šumu), která se vyznačuje výrazným zachováním detailů a hran. Na rozdíl od tradičních metod, které pracují s lokálními okolími pixelů, NLM vyhledává podobné fragmenty (patche) po celém obrazu a využívá jejich informace pro inteligentní vyhlazení cílového pixelu nebo oblasti, čímž se minimalizuje ztráta důležitých vizuálních prvků. NLM můžeme rozdělit na dvě varianty: pixelwise a patchwise NLM.

První varianta, pixelwise NLM, se zaměřuje na každý pixel individuálně. Pro každý pixel p ve vstupním obraze I se vypočítává jeho nová hodnota jako vážený průměr hodnot všech pixelů v obraze. Váhy závisí na podobnosti malých okolí (patchů) okolo každého pixelu a jsou určeny Euklidovskou vzdáleností mezi hodnotami pixelů v těchto okolicích. Matematicky lze pixelwise NLM vyjádřit pomocí vztahu:

$$I_{new}(p) = \frac{\sum_{q \in I} w(p, q) \cdot I(q)}{\sum_{q \in I} w(p, q)} \quad (1.9)$$

kde: $w(p, q)$ = představuje váhu založenou na podobnosti mezi okolím pixelů p a q

$I(q)$ = hodnota pixelu q ve vstupním obraze

Druhá varianta, patchwise NLM, rozšiřuje přístup pixelwise metody tím, že jako základní jednotku pro výpočet váženého průměru bere celé patche namísto jednotlivých pixelů. Tím se zvyšuje

značí, že signál je silnější než šum, což znamená, že obraz má vysokou kvalitu s minimálním množstvím šumu. Naopak nízká hodnota SNR indikuje, že šum je dominantní a signál je obtížně rozeznatelný.

robustnost metody vůči šumu, jelikož se lépe využívá strukturální informace z obrazu. Vztah pro patchwise NLM může být zapsán obdobně, avšak s důrazem na porovnávání celých patchů.

Rozdíl mezi pixelwise a patchwise variantami spočívá v měřítku, na kterém se hodnotí podobnost mezi částmi obrazu. Zatímco pixelwise varianta se soustředí na detailní porovnání na úrovni pixelů, patchwise přístup poskytuje robustnější odhad tím, že zapojuje větší kontext okolí zkoumaného bodu.

Algoritmus NLM, ať už v pixelwise či patchwise variantě, nabízí vynikající schopnost odstraňování šumu při zachování ostrých hran a detailů. Jeho hlavní nevýhodou však zůstává vysoká výpočetní náročnost, zejména v důsledku nutnosti porovnávat velké množství oblastí napříč celým obrazem. To činí aplikaci NLM, především v reálném čase, výzvou. Optimalizace a využití paralelního zpracování však mohou tuto nevýhodu zmírnit. [12]

Lokálně adaptivní mediánový filtr

Lokálně adaptivní mediánový filtr je inovativní metodou redukce šumu, specificky navrženou pro typ šumu speckle. Tento typ šumu je výsledkem interferencí mezi koherentními vlnami odraženými od cíle, což vede k zrnitému vzoru v pořízeném obraze. Tento filtr využívá lokální statistiky pro detekci speckle šumu a jeho nahrazení mediánovou hodnotou z lokálního okolí.

Nejdříve je nutné zjistit celkový součet intenzit pixelů v určitém oblastním okně, které obklopuje centrální pixel. Tento součet je základem pro další výpočty statistik, jako je lokální průměr.

$$S(i, j) = \sum_{n=j-k}^{j+k} \sum_{m=i-k}^{i+k} d(m, n) \quad (1.10)$$

kde: k = poloměr posuvného okna okolo středového pixelu

$d(m, n)$ = hodnota pixelu na pozici (m, n)

Dále se určí průměrná hodnota a lokální standardní odchylka hodnot pixelů v posuvném okně. Lokální průměr pomáhá rozlišit oblasti s vysokým a nízkým šumem tím, že poskytuje základní hodnotu pro porovnání s ostatními pixely. Standardní odchylka je míra variability nebo rozptylu hodnot pixelů kolem lokálního průměru, která pomáhá identifikovat pixely s výrazně odlišnými intenzitami (tj. šum).

$$\mu(i, j) = \frac{S(i, j)}{N(i, j)} \quad (1.11)$$

kde: $S(i, j)$ = součet všech hodnot pixelů v posuvném okně

$N(i, j) = (2k + 1)^2$ = počet pixelů v posuvném okně

$$\sigma(i, j) = \sqrt{\frac{\sum_{m=i-k}^{i+k} \sum_{n=j-k}^{j+k} (d(m, n) - \mu(i, j))^2}{N(i, j)}} \quad (1.12)$$

kde: $\mu(i, j)$ = lokální průměr hodnot pixelů v posuvném okně

Na základě definované dolní a horní hranice pro rozpoznání validních pixelů v rámci posuvného okna se rozhodne, které pixely jsou považovány za validní, a které představují potenciální šum.

$$\begin{aligned} LB(i, j) &= \mu(i, j) - M \cdot \sigma(i, j) \\ UB(i, j) &= \mu(i, j) + M \cdot \sigma(i, j) \end{aligned} \quad (1.13)$$

kde: $\mu(i, j)$ = lokální průměr hodnot pixelů v posuvném okně

$\sigma(i, j)$ = lokální standardní odchylka hodnot pixelů v posuvném okně

M = uživatelem definovaný násobitel pro určení prahu šumu

$$l(m, n) = \begin{cases} 0 & \text{pokud } d(m, n) < LB(i, j) \text{ nebo } d(m, n) > UB(i, j) \\ 1 & \text{pokud } LB(i, j) \leq d(m, n) \leq UB(i, j) \end{cases} \quad (1.14)$$

kde: $LB(i, j)$, $UB(i, j)$ = dolní a horní hranice pro validní hodnoty pixelů

Finální rovnice nahrazuje hodnotu centrálního pixelu, identifikovaného jako šum, mediánovou hodnotou z validních pixelů v okně. Tento proces zajišťuje, že hodnota pixelu reprezentuje jeho skutečné okolí a zlepšuje tak celkovou kvalitu obrazu redukcí šumu.

$$r(i, j) = \text{median } d(m, n) \mid l(m, n) = 1, i - k \leq m, n \leq i + k \quad (1.15)$$

kde: $l(m, n)$ = maska určující, zda je pixel na pozici (m, n) považován za validní nebo za šum

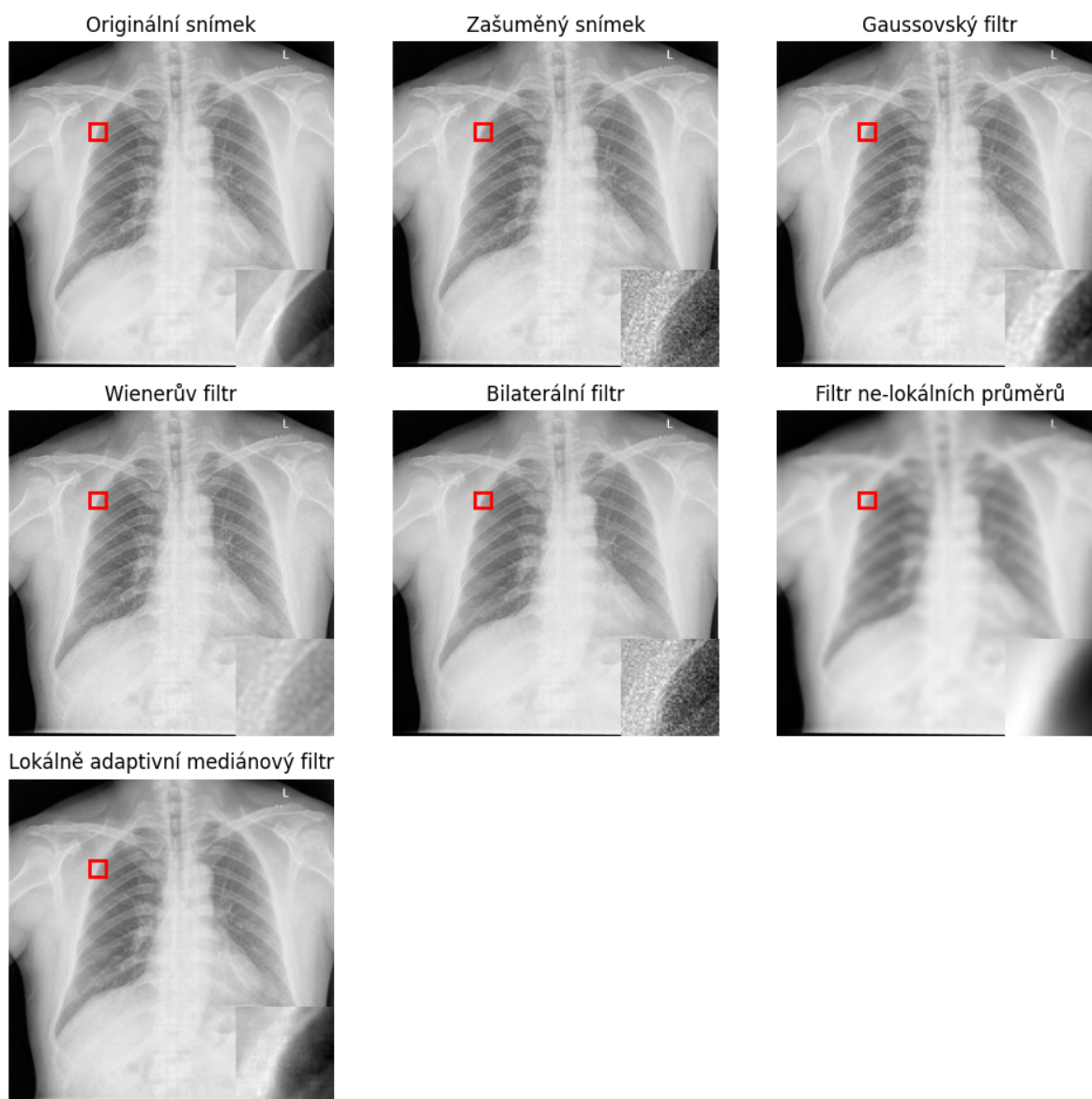
Lokálně adaptivní mediánový filtr se ukazuje jako výrazně efektivní při odstraňování speckle šumu, což je běžný problém v rentgenových a jiných druzích medicínských snímků. Využití lokálních statistik umožňuje filtru selektivně cílit na šumové pixely, zatímco zachovává ostré hrany a důležité detaily, což je klíčové pro zachování diagnostické hodnoty obrazu.

Nicméně, stejně jako většina pokročilých metod zpracování obrazu, i tento filtr má své nevýhody. Především je to jeho výpočetní náročnost, která může být limitující pro aplikace v reálném čase nebo při zpracování velkých datasetů. Navíc, efektivní využití filtru vyžaduje pečlivý výběr parametrů, což může vyžadovat pokročilé znalosti a experimentální ladění pro dosažení optimálních výsledků. [13]

1.2. Neuronové sítě pro předzpracování obrazu

Předchozí sekce byla věnována klasickým metodám předzpracování obrazu, kde byl brán důraz na techniky pro vylepšení obrazu a odstranění či redukci různých typů šumu ze snímků. Tato sekce se bude zabývat pokročilým oborem pro předzpracování obrazu, kterým jsou neuronové sítě.

Neuronové sítě představují významný skok v možnostech zpracování obrazu, především díky jejich schopnosti učit se a generalizovat na základě velkých datových sad. Na rozdíl od klasických



Obrázek 1.1.2.: Snímky demonstrují účinky popsanych filtrů na kvalitu obrazu v porovnání s originálním snímek a snímek po umělém přidání šumu, aby se simulovalo běžné zkreslení vznikající během akvizice obrazu.

metod, které aplikují pevně stanovená pravidla, se neuronové sítě dokáží přizpůsobit na základě konkrétních charakteristik a variací v datech, na kterých jsou trénovány. Tato adaptabilita je klíčová pro zpracování obrazů v dynamických a různorodých podmínkách, což zahrnuje vše od standardního zlepšování obrazu po složité úlohy jako je rozpoznávání objektů nebo segmentace obrazu.

Neuronové sítě, a zvláště konvoluční neuronové sítě (CNN), jsou vysoce efektivní v automatickém učení a rozpoznávání vzorců z obrazových dat. Tyto sítě simulují způsob, jakým lidský mozek zpracovává vizuální informace, což umožňuje extrakci a učení vysokého řádu charakteristik, které jsou následně využity pro klasifikaci nebo predikci. Díky této schopnosti se neuronové sítě stávají základem mnoha pokročilých systémů pro automatizované zpracování obrazu.

Využití neuronových sítí však také přináší výzvy, včetně potřeby velkého množství anotova-

ných trénovacích dat a výpočetně náročných modelů, které mohou vyžadovat specializovanou hardwarovou podporu pro efektivní trénink a nasazení.

Následující sekce bude zaměřena na nejčastěji využívané architektury neuronových sítí pro předzpracování obrazu, s důrazem na konvoluční neuronové sítě (CNN). [14]

Úvod do konvolučních neuronových sítí

Neuronové sítě jsou základním stavebním prvkem moderního strojového učení a mají zásadní význam pro rozvoj oblastí jako je zpracování přirozeného jazyka či obrazu, rozpoznávání objektů v obraze, při analýze signálu, v robotice a mnoha jiných důležitých oborech. Tyto systémy jsou inspirovány strukturou a funkcí lidského mozku a jsou navrženy tak, aby modelovaly složité vzory a vztahy v datech. Strukturálně jsou neuronové sítě složeny z vrstev neuronů, což jsou jednotky zpracovávající informace, které jsou propojeny spojeními, ohodnocenými vahami.

Hlavní komponenty neuronových sítí

Základem neuronové sítě je její architektura, která je definována několika klíčovými komponentami. Tyto komponenty jsou základem pro učení a zpracování dat v síti. V následujícím seznamu jsou jednotlivé komponenty podrobněji popsány:

- **Neuron** Neuron v neuronové síti je elementární jednotka, která simuluje chování neuronu v biologických systémech. Přijímá vstup $x = (x_1, x_2, x_3, \dots, x_n)$, který je zpracován a následně vrácen na výstupu. Tento výstup je vstupem do dalšího neuronu nebo je součástí koncového výstupu ze sítě.

$$z = W \cdot x + b \quad (1.16)$$

kde: W = reprezentuje váhy, jenž jsou parametry určující sílu signálu mezi neurony.

b = bias, což je parametr posunutí, který upravuje výstup neuronu před aplikací aktivační funkce.

- **Vrstvy (Layers)** Vrstvy se skládají z neuronů a jsou základem architektury neuronové sítě. Každá vrstva plní specifickou funkci v procesu učení. Vrstvy mohou být kategorizovány do tří typů:
 - **Vstupní vrstva (Input layer)** Tato vrstva přijímá surová data a připravuje je pro další zpracování v síti. Každý neuron této vrstvy přímo reprezentuje jednu vlastnost vstupního datového vektoru.
 - **Skryté vrstvy (Hidden layers)** Skryté vrstvy jsou mezi vstupní a výstupní vrstvou a jejich úkolem je zpracování signálů přijatých od vstupní vrstvy. Komplexnost a hloubka těchto vrstev jsou klíčové pro schopnost sítě učit se složité vzory.

- **Výstupní vrsta (Output layer)** Poslední výstupní vrstva generuje predikce určené neuronovou sítí. Například v klasifikačních úlohách může tato vrstva používat softmax funkci³.
- **Aktivační funkce (Activation function)** Aktivační funkce je důležitou složkou neuronových sítí, která umožňuje modelování nelineárních vztahů. Funkce, jako je sigmoid, tanh a ReLU (Rectified Linear Unit), jsou běžně používány k výpočtu výstupu z neuronů na základě vstupu a vah. Například, ReLU, která je definována jako $ReLU(x) = \max(0, x)$, je populární pro svou schopnost rychle konvergovat během tréninku a minimalizovat problém mizejících gradientů⁴.
- **Ztrátová funkce (Loss function)** Ztrátová funkce, někdy nazývaná penalizační funkce, poskytuje kvantitativní hodnocení toho, jak daleko jsou predikce neuronové sítě od skutečných hodnot. Během tréninku neuronové sítě je hlavním cílem minimalizace této funkce. Mezi běžné ztrátové funkce se řadí střední kvadratická chyba (MSE) pro regresní úkoly a cross-entropy pro klasifikační úkoly.
- **Optimalizační algoritmus (Optimizer)** Optimalizační algoritmy jsou základem učení neuronových sítí, protože určují, jak se síť učí z dat tím, že iterativně upravují váhy a biasy k minimalizaci ztrátové funkce. Stochastic Gradient Descent (SGD) provádí tuto úpravu tak, že v každém kroku tréninku náhodně vybere vzorek dat a spočítá gradient ztrátové funkce pouze na základě tohoto vzorku: Adam a RMSprop jsou pokročilejší varianty, které se snaží zlepšit konvergenci SGD tím, že upravují rychlost učení pro každý parametr individuálně, což může vést k rychlejšímu a stabilnějšímu učení, zejména v komplexních sítích.

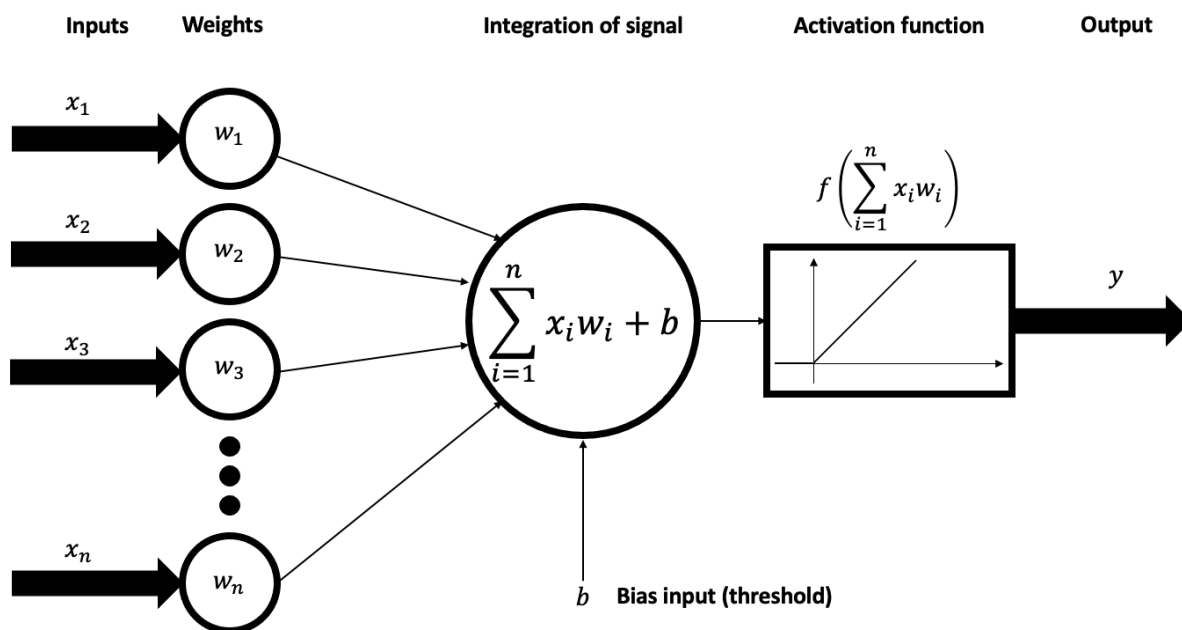
Základní komponenty neuronových sítí hrají zásadní roli ve funkcionalitě vytvořených modelů. Neurony a vrstvy formují strukturu sítě, zatímco aktivační funkce umožňují modelování nelineárních vztahů, což je nezbytné pro zpracování složitých vzorů a funkcí. Ztrátové funkce poskytují metriku pro hodnocení úspěšnosti sítě v předpovídání a jsou klíčové pro navigaci procesu učení. Optimalizační algoritmy řídí tento učící se proces tím, že iterativně upravují interní parametry sítě, aby se minimalizovala celková ztráta a zvýšila přesnost modelu. Každá z těchto komponent společně zajišťuje, že neuronové sítě se mohou učit efektivně, adaptovat se a poskytovat přesné predikce při řešení zadaných úloh. [14, 16]

Základy konvolučního zpracování v CNN

Jak již bylo zmíněno výše, konvoluční neuronové sítě (CNN) jsou specializovaným druhem neuronových sítí, které se používají především pro zpracování strukturovaných dat, jako jsou například obrazová data. CNN jsou známé svou schopností zachytit prostorové a časové závislosti v obraze

³Softmax funkce se často používá jako aktivační funkce v poslední vrstvě neuronové sítě při klasifikačních úlohách. Tato funkce převádí výstupní hodnoty neuronů na pravděpodobnostní rozdělení mezi různými třídami. Matematický vzorec pro softmax funkci je definován následovně: $\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^N e^{z_j}}$

⁴Mizející gradienty jsou problém, kdy gradient ztrátové funkce klesá exponenciálně rychle s každou vrstvou v síti, což způsobuje, že váhy v raných vrstvách se téměř neaktualizují. Tento jev výrazně zpomaluje trénink a může způsobit, že síť nedokáže efektivně učit složité vzory.



Obrázek 1.2.1.: Ilustrace neuronové jednotky, ukazující základní komponenty jako vstup, váhy, bias, aktivační funkci a výstup. (převzato z [15] pod licencí CC BY-SA 4.0)

pomocí aplikace příslušných filtrů. Úloha konvolučních operací je jádrem toho, co činí CNN tak efektivními pro úlohy, jako je rozpoznávání obrazu, klasifikace obrazu a detekce objektů.

Operace konvoluce zahrnuje posunutí jádra (kernelu) přes vstupní data (obvykle obraz) a výpočet skalárního součinu jádra s překrývajícími se částmi obrazu. Výsledek tvoří jednu položku ve výstupní mapě příznaků. Tento proces se opakuje posouváním jádra po všech pozicích vstupní matice. Mezi klíčové aspekty konvoluce patří:

- **Jádro konvoluce** Malé matice používané k detekci specifických prvků, jako jsou hrany nebo textury ve vstupním obraze. Každý filtr je navržen tak, aby zachytil určitý aspekt vstupních dat.
- **Krok jádra** Počet pixelů, o které posouváme filtr po vstupním obraze. To ovlivňuje velikost výstupní mapy prvků.
- **Padding** Existuje několik možností, jak umožnit jádru brát v potaz i hraniční prvky. Jednou z možností je přidání vrstev nul kolem vstupního obrazu. Další možností je použití zrcadlení (mirror padding), kde se okrajové prvky zrcadlově opakují, což pomáhá jádru zahrnout informace z okrajů obrazu bez ztráty dat.
- **Mapa příznaků** Výstup operace konvoluce se nazývá mapa příznaků. Představuje odezvu filtru v každé prostorové poloze. Použitím různých filtrů může síť CNN zachytit širokou škálu aspektů vstupního obrazu.

Dalším důležitým konceptem v CNN je poolování, které se obvykle používá po operaci konvoluce. Nejběžnější formou sdružování je tzv. max pooling, kdy výstupem sdružovacího jádra je maximální hodnota z části mapy, kterou pokrývá. Tato operace snižuje dimenzionalitu mapy příznaků a umožňuje síti být odolnou vůči malým změnám vstupního obrazu.

V hlubokých konvolučních neuronových sítích se vrství několik konvolučních a sdružovacích vrstev za sebou, což síti umožňuje učit se hierarchické rysy. Nižší vrstvy se mohou naučit rozpoznávat jednoduché rysy, jako jsou hrany, zatímco hlubší vrstvy mohou interpretovat složité aspekty, jako jsou objekty.

Konvoluce je základní operací v CNN, která umožňuje extrakci smysluplných rysů ze vstupních dat. Díky vrstvení a kombinaci konvolučních, aktivačních a sdružovacích vrstev se CNN mohou efektivně naučit hierarchii rysů, která je klíčová pro zvládání složitých úloh vizuálního rozpoznávání. Každá složka, od výběru filtrů až po použití poolingů, hraje rozhodující roli při definování schopnosti sítě přesně interpretovat a klasifikovat vstupní data. [14, 16]

Autoenkodéry v oblasti zpracování obrazu

Autoenkodéry jsou samostatnou třídou modelů neuronových sítí určených především k učení efektivního kódování dat. Tyto sítě toho dosahují komprimací vstupu do méně rozměrné reprezentace latentního prostoru a následnou rekonstrukcí výstupu z této komprimované podoby. Tento duální proces kódování a následného dekódování neslouží pouze ke kompresi dat, ale má zásadní význam pro to, aby síť byla nucena upřednostňovat a zachycovat nejvíce informativní rysy dat.

Struktura autoenkodérů

Autoenkodéry se dělí na tři hlavní části:

- **Enkodér** Proces kódování zahrnuje transformaci vstupních dat s vysokou dimenzí do komprimované podoby. Tento krok je klíčový, protože umožňuje modelu naučit se podstatu dat odfiltrováním šumu a redundance. Tím se síť naučí reprezentaci, která zachycuje základní vzory a struktury nezbytné pro rekonstrukci vstupních dat. Enkodér přebírá vstupní vektor x a převede jej na komprimovanou reprezentaci z , často označovaný jako latentní prostor:

$$z = \sigma(W_e \cdot x + b_e) \quad (1.17)$$

kde: W_e = matice vah enkodéru.

b_e = bias vektor enkodéru.

σ = nelineární aktivační funkce.

- **latentní prostor (bottleneck)** Srdcem autoenkodéru je jeho latentní prostor, v němž jsou vstupní data reprezentována ve své nejkomprimovanější podobě. Tato reprezentace poskytuje pohled na data v nižší dimenzi a zdůrazňuje jejich nejdůležitější atributy. Rozměrnost a konfigurace latentního prostoru jsou velmi důležité a ovlivňují, jak efektivně dokáže autoenkodér komprimovat a rekonstruovat data.
- **Dekodér** Proces dekódování se snaží co nejpresněji rekonstruovat původní vstup z komprimované zakódované reprezentace. V této fázi se testuje kvalita naučených rysů a účinnost

kódování. Cílem není dosáhnout dokonalé rekonstrukce, ale spíše vytvořit aproximaci, která zachová nejvýznamnější atributy původních dat. Tato schopnost je užitečná zejména v aplikacích, kde je pochopení základní struktury dat cennější než reprodukce přesné kopie. Dekódovací část rekonstruuje vstupní data x' z latentní reprezentace z .

Autoenkodéry se používají v celé řadě aplikací, které přesahují rámec prosté komprese dat. Například při redukci šumu se naučí ignorovat náhodné odchylky ve vstupních datech a místo toho se soustředí na rekonstrukci smysluplného základního vzoru obrazu. Používají se také při detekci anomálií, kdy se naučí dobře reprodukovat typické případy dat, ale mají problémy s reprodukcí odlehlých hodnot, a tím identifikují anomální data. V oblasti zpracování obrazu se autoenkodéry také využívají pro segmentaci obrazu a extrakci rysů. Tyto schopnosti jsou klíčové pro rozpoznávání objektů a detailní analýzu obrazového obsahu. [17, 18]

1.3. Obecné postupy augmentace dat

Augmentace dat neboli rozšiřování dat je klíčovou technikou v oblasti strojového učení, zejména v oblastech zpracování obrazu a počítačového vidění. Zahrnuje umělé rozšíření velikosti a rozmanitosti datové sady použitím řady transformací, které jsou aplikovány na vstupní data. Tato sekce se bude zabývat konceptem augmentace dat, důvody použití se zdůrazněním na přínos i omezení při aplikaci augmentace a v neposlední řadě budou popsány jedny z nejpoužívanějších technik rozšiřování dat.

Augmentace dat

Přínos a omezení při rozšiřování dat

Přehled metod augmentace dat

2. Placeholder

3. Zhodnocení

4. Závěr

Seznam použitých zdrojů

1. BASAVAPRASAD, Benchamardimath; RAVINDRA, Hegadi. A STUDY ON THE IMPORTANCE OF IMAGE PROCESSING AND ITS APLLICATIONS. *International Journal of Research in Engineering and Technology* [online]. 2014, **03**(15), 155–160 [cit. 2024-03-16]. ISSN 23217308. Dostupné z DOI: [10.15623/ijret.2014.0315029](https://doi.org/10.15623/ijret.2014.0315029).
2. KRIG, Scott. *Computer Vision Metrics*. 1. vyd. California: Apress Berkeley, 2014. ISBN 978-1-4302-5930-5. Dostupné z DOI: [10.1007/978-1-4302-5930-5](https://doi.org/10.1007/978-1-4302-5930-5).
3. *Computer Vision for X-Ray Testing: Imaging, Systems, Image Databases, and Algorithms*. 1. vyd. Cham: Springer International Publishing, 2015. ISBN 978-3-319-20747-6. Dostupné z DOI: [10.1007/978-3-319-20747-6](https://doi.org/10.1007/978-3-319-20747-6).
4. Contrast Limited Adaptive Histogram Equalization. In: *Graphics Gems IV*. 1st Edition. San Francisco: Academic Press, 1994, s. 474–485. ISBN 978-0-12-336155-4.
5. JAYAKUMAR, Dontabhaktuni; PULLARAO, Bandi; PRADEEP, Vanukuru. An Algorithm on Generalized Un Sharp Masking for Sharpness and Contrast of an Exploratory Data Model. *International Journal of Advanced Engineering Research and Science*. 2016, **3**(9), 62–72. ISSN 23496495. Dostupné z DOI: [10.22161/ijaers/3.9.10](https://doi.org/10.22161/ijaers/3.9.10).
6. CHAKI, Jyotismita; DEY, Nilanjan. *A Beginner's Guide to Image Preprocessing Techniques*. 1st Edition. Boca Raton: CRC Press, 2018. ISBN 9780429441134. Dostupné z DOI: [10.1201/9780429441134](https://doi.org/10.1201/9780429441134).
7. FAN, Linwei; ZHANG, Fan; FAN, Hui; ZHANG, Caiming. Brief review of image denoising techniques. *Visual Computing for Industry, Biomedicine, and Art* [online]. 2019, **2**(1) [cit. 2024-03-29]. ISSN 2524-4442. Dostupné z DOI: [10.1186/s42492-019-0016-7](https://doi.org/10.1186/s42492-019-0016-7).
8. HAMBAL, Abdalla Mohamed; PEI, Zhijun; ISHABAILU, Faustini Libent. Image Noise Reduction and Filtering Techniques. *International Journal of Science and Research (IJSR)* [online]. 2017, **6**, 2033–2038 [cit. 2024-03-29]. Dostupné z: <https://www.ijser.net/getabstract.php?paperid=25031706>.
9. MERY, Domingo. X-ray Image Processing. In: *Computer Vision for X-Ray Testing* [online]. 1. vyd. Springer Cham, 2015, s. 109–148 [cit. 2024-03-30]. ISBN 978-3-319-20746-9. Dostupné z: https://link.springer.com/chapter/10.1007/978-3-319-20747-6_4.
10. PARIS, Sylvain; KORNPORST, Pierre; TUMBLIN, Jack; DURAND, Frédo. *Bilateral Filtering: Theory and Applications* [online]. 1. vyd. Foundations a Trends, 2009 [cit. 2024-03-30]. ISBN 978-1-60198-251-3. Dostupné z DOI: [10.1561/06000000020](https://doi.org/10.1561/06000000020).

11. WIENER FILTERS. In: *Advanced Digital Signal Processing and Noise Reduction* [online]. 2. vyd. Wiley, 2005, s. 178–204 [cit. 2024-03-30]. ISBN 0-470-84162-1. Dostupné z: <https://www.roma1.infn.it/exp/cuore/pdfnew/ch06.pdf>.
12. BUADES, Antoni; COLL, Bartomeu; MOREL, Jean-Michel. Non-Local Means Denoising. *Image Processing On Line* [online]. 2011, **1**, 208–212 [cit. 2024-04-04]. ISSN 2105-1232. Dostupné z DOI: [10.5201/ipol.2011.bcm_nlm](https://doi.org/10.5201/ipol.2011.bcm_nlm).
13. QIU, Fang; BERGLUND, Judith; JENSEN, John R.; THAKKAR, Pathik; REN, Dianwei. Speckle Noise Reduction in SAR Imagery Using a Local Adaptive Median Filter [online]. 2013, **41**(3), 244–266 [cit. 2024-04-04]. ISSN 1548-1603. Dostupné z DOI: [10.2747/1548-1603.41.3.244](https://doi.org/10.2747/1548-1603.41.3.244).
14. GHOSH, Anirudha; SUFIAN, Abu; SULTANA, Farhana; CHAKRABARTI, Amlan; DE, Debashis. Fundamental Concepts of Convolutional Neural Network. *Recent Trends and Advances in Artificial Intelligence and Internet of Things* [online]. 2020, 519–567 [cit. 2024-04-18]. ISBN 978-3-030-32643-2. Dostupné z DOI: [10.1007/978-3-030-32644-9_36](https://doi.org/10.1007/978-3-030-32644-9_36).
15. BRUNELLON. *Example of a neural network's neural unit* [online]. 2004. [cit. 2024-04-18]. Dostupné z: https://commons.wikimedia.org/wiki/File:Example_of_a_neural_network%5C%27s_neural_unit.png.
16. TEUWEN, Jonas; MORIAKOV, Nikita. Convolutional neural networks. *Handbook of Medical Image Computing and Computer Assisted Intervention* [online]. 2020, 481–501 [cit. 2024-04-18]. ISBN 9780128161760. Dostupné z DOI: [10.1016/B978-0-12-816176-0.00025-9](https://doi.org/10.1016/B978-0-12-816176-0.00025-9).
17. BANK, Dor; KOENIGSTEIN, Noam; GIRYES, Raja. Autoencoders. *ArXiv*. 2020, **abs/2003.05991**. Dostupné také z: <https://api.semanticscholar.org/CorpusID:260535551>.
18. MICHELUCCI, Umberto. An Introduction to Autoencoders. *ArXiv*. 2022, **abs/2201.03898**. Dostupné také z: <https://api.semanticscholar.org/CorpusID:245853675>.

Seznam obrázků

1.1.1. Ukázka metod ekvalizace histogramu na rentgenovém snímku. CLAHE demonstruje lepší zachování detailů a kontrastu v porovnání s tradiční ekvalizací histogramu.	17
1.1.2. Snímky demonstrují účinky popsaných filtrů na kvalitu obrazu v porovnání s originálním snímek a snímek po umělém přidání šumu, aby se simulovalo běžné zkreslení vznikající během akvizice obrazu.	27
1.2.1. Ilustrace neuronové jednotky, ukazující základní komponenty jako vstup, váhy, bias, aktivační funkci a výstup. (převzato z [15] pod licencí CC BY-SA 4.0)	30

Sazba zdrojových kódů

A. Externí přílohy

Struktura repozitáře je následující:

BostonHousing	vypracovaná regresní úloha a data set Boston Housing
IrisFlowers	vypracovaná klasifikační úloha a data set Iris flowers
Obrázky	adresář s obrázky, které jsou zobrazeny v repozitáři
IntrusionDetection.rar	vypracovaná úloha Intrusion detection s data sety v souboru rar
README.md	jednoduchý popis repozitáře
