

Zpracování obrazu pomocí neuronových sítí

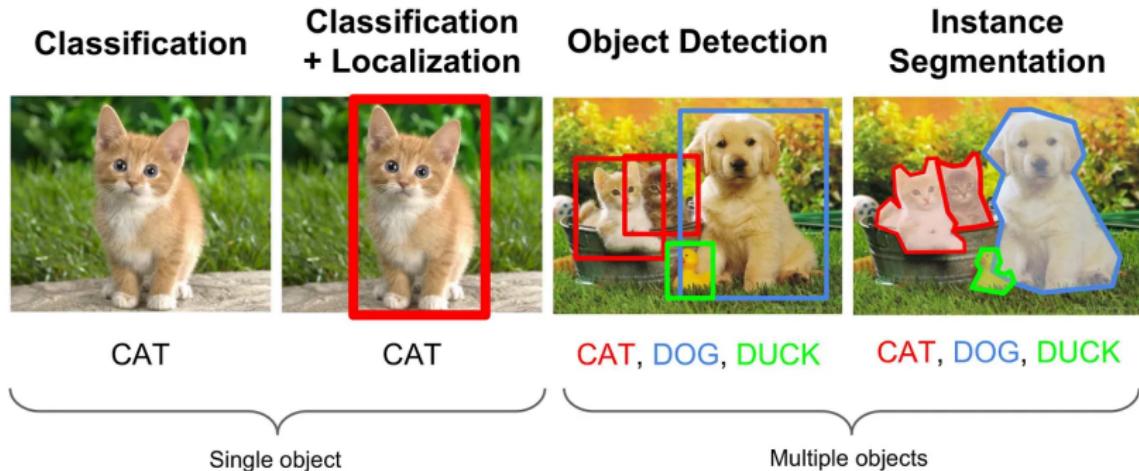
Blok 3: Segmentace, transfer learning a vlastní projekty

[Michal Vašinek]

[25.4.2025]

Základní úlohy zpracování obrazu

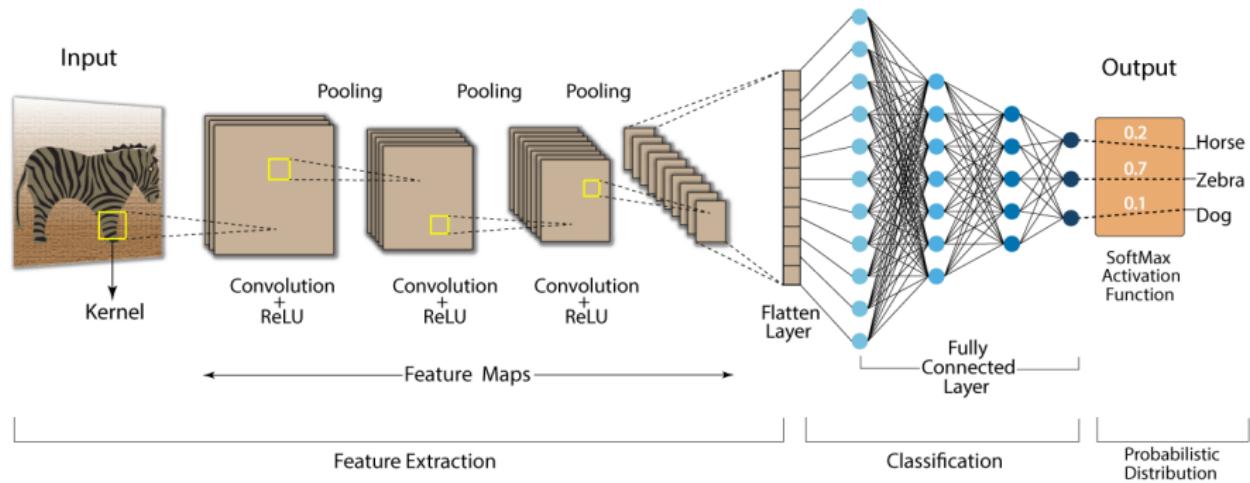
- Klasifikace: určení třídy objektu na obrázku
- Detekce: lokalizace objektu pomocí bounding boxu
- Segmentace: určení pixelové klasifikace (semantická, instance)



Klasifikace obrázků

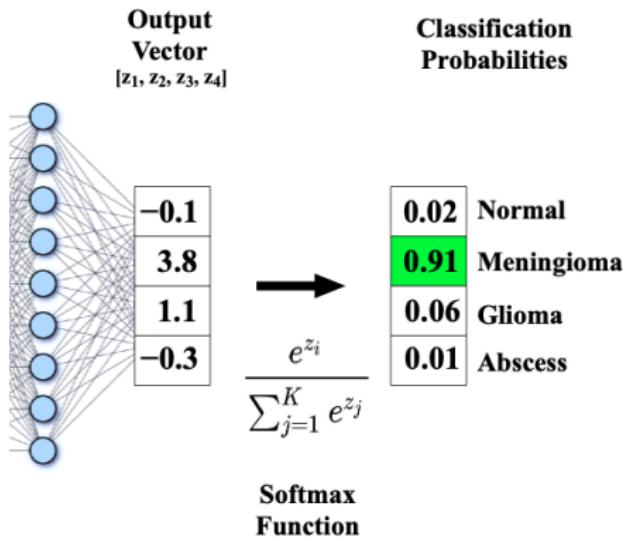
- Klasifikace obrázků: určení třídy objektu
- Datasety ImageNet, CIFAR-10, MNIST
- Modely: VGG, ResNet, Inception

Convolution Neural Network (CNN)



Klasifikace - softmax

- Softmax funkce: převod výstupů plně propojené vrstvy na pravděpodobnosti tříd.
- Vzorec: $p(y_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$
- Použití v poslední vrstvě klasifikačního modelu



Klasifikace - loss funkce

- Klasifikační loss funkce: křížová entropie
- Vzorec: $L(y, \hat{y}) = -\sum_{i=1}^C y_i \log(\hat{y}_i)$
- Použití pro trénink modelu
- Ke každému trénovacímu obrázku přiřazena třída klasifikovaného objektu.
- $y_i = 1$ je skutečná třída, \hat{y}_i je modelem predikovaná pravděpodobnost.

Klasifikace s lokalizací

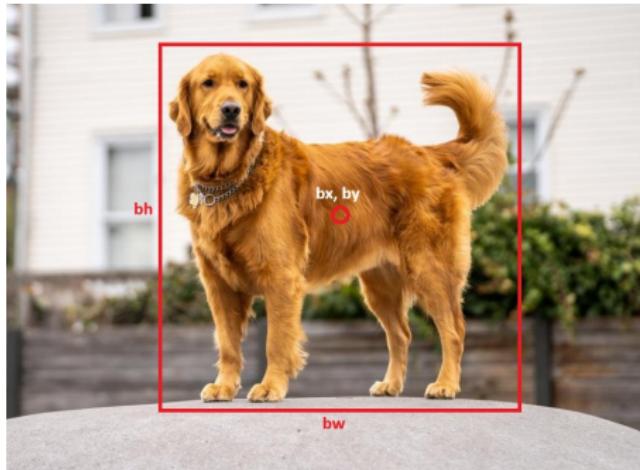
- Klasifikace + lokalizace: určení třídy a pozice objektu
- Použití bounding boxů pro lokalizaci
- 4 dodatečné hodnoty (b_x, b_y, b_h, b_w)
- Výstupem sítě jsou tři skupiny hodnot:
 - Klasifikační výstup (softmax)
 - Koordináty bounding boxu (plně propojená vrstva)
 - Confidence score (pravděpodobnost detekce, sigmoid) - je v obrázku skutečně objekt?

$\underbrace{c}_{\text{confidence}}$ $\underbrace{b_x, b_y, b_w, b_h}_{\text{bounding box}}$ $\underbrace{p_1, p_2, \dots, p_C}_{\text{softmax klasifikace}}$

výstupní vektor: $[c, b_x, b_y, b_w, b_h, p_1, p_2, \dots, p_C]$

Klasifikace s lokalizací - trénovací data

- Kategorie: 1-4 (např. auto, pes, kočka, pozadí)
- Pro každou kategorii máme sadu trénovacích obrázků
- Každý obrázek má přiřazenu třídu a bounding box



trénovací vzorek: [1, b_x , b_y , b_w , b_h , 0, 1, 0]

Klasifikace s lokalizací - trénovací data

- Kategorie: 1-4 (např. auto, pes, kočka, pozadí)
- Pokud obrázek neobsahuje objekt, máme pouze pozadí, v tomto případě nám nezáleží na klasifikaci a lokalizaci.



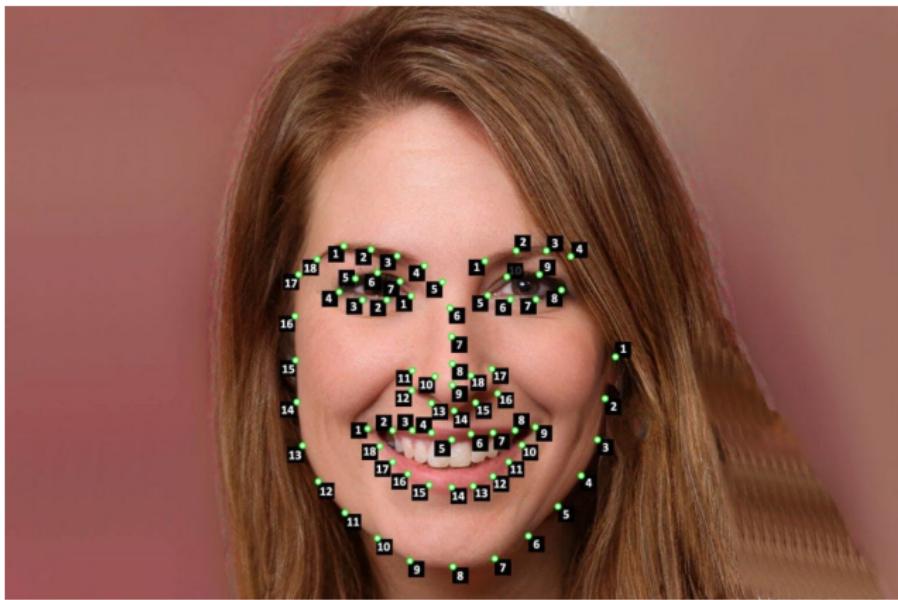
trénovací vzorek: [0, ?, ?, ?, ?, ?, ?, ?, ?]

Klasifikace s lokalizací - loss funkce

- Loss funkce pro klasifikaci: křížová entropie
- Loss funkce se liší pokud je v trénovacím vzorku objekt nebo pozadí
- Pokud je objekt: $L_{y_0=1} = L(\text{třída}) + L(\text{bbox})$
- Pokud je pozadí: $L_{y_0=0} = L(y_0, \hat{y}_0)$

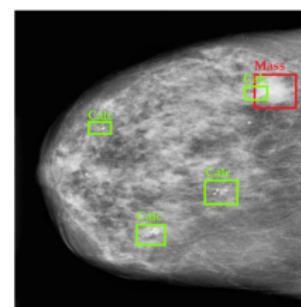
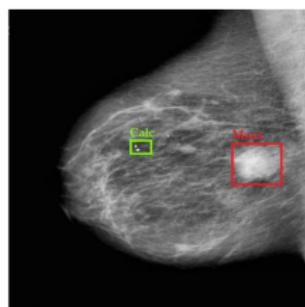
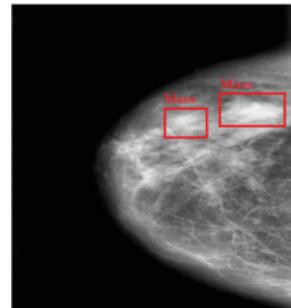
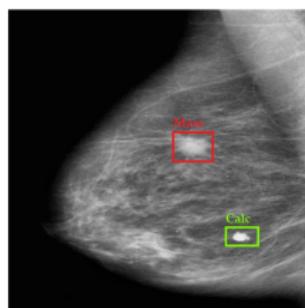
Detekce různých rysů

- Namísto bounding boxu můžeme chtít detektovat různé rysy v obrázcích.
- Např. detekce klíčových bodů (keypoints) na obličeji.
- Hodnoty, které model učíme jsou naší volbou.



Lokalizace a detekce objektů

- Detekce objektů: bounding boxy, klasifikace
- Využití v medicíně: detekce buněk, nádorů



Detekce obrázků

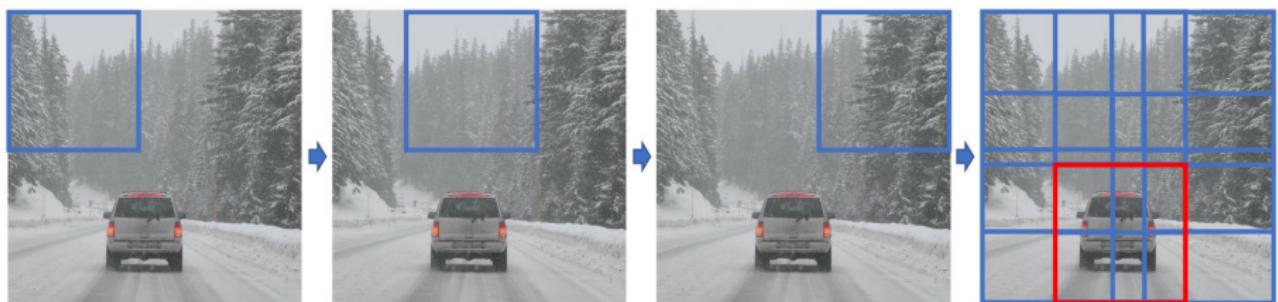


X	y
→	1
1	1
1	1
0	0
0	0

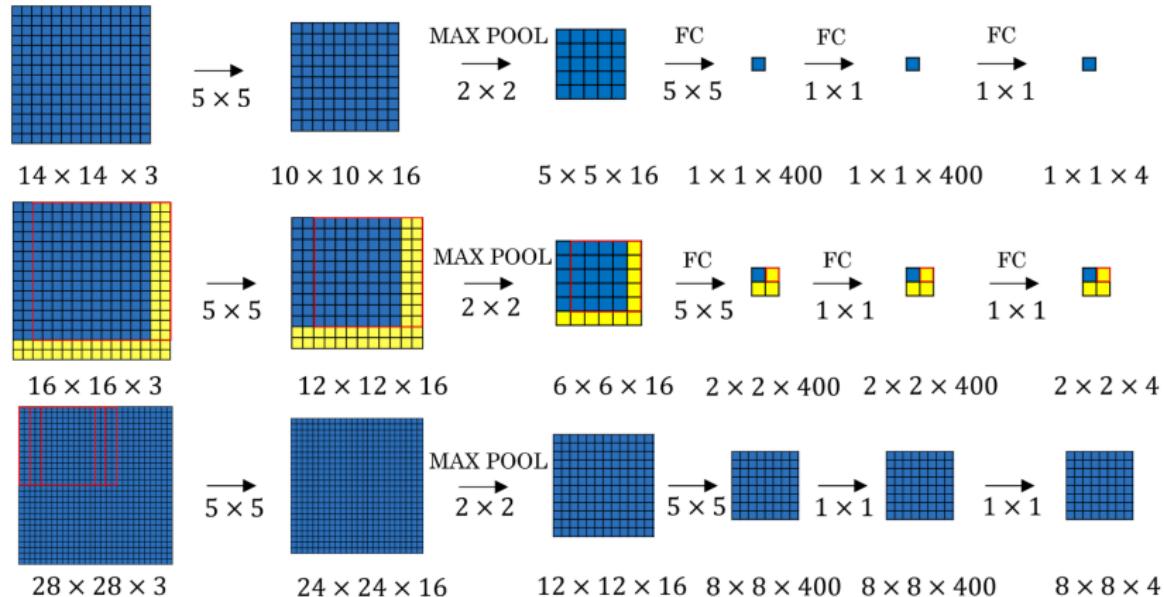
ConvNet → y

Detekce objektů - sliding window

- Sliding window: procházení obrázkem s různými velikostmi oken
- Pro každé okno použijeme convnet pro klasifikaci a nalezení bounding boxu
- Vysoká výpočetní náročnost



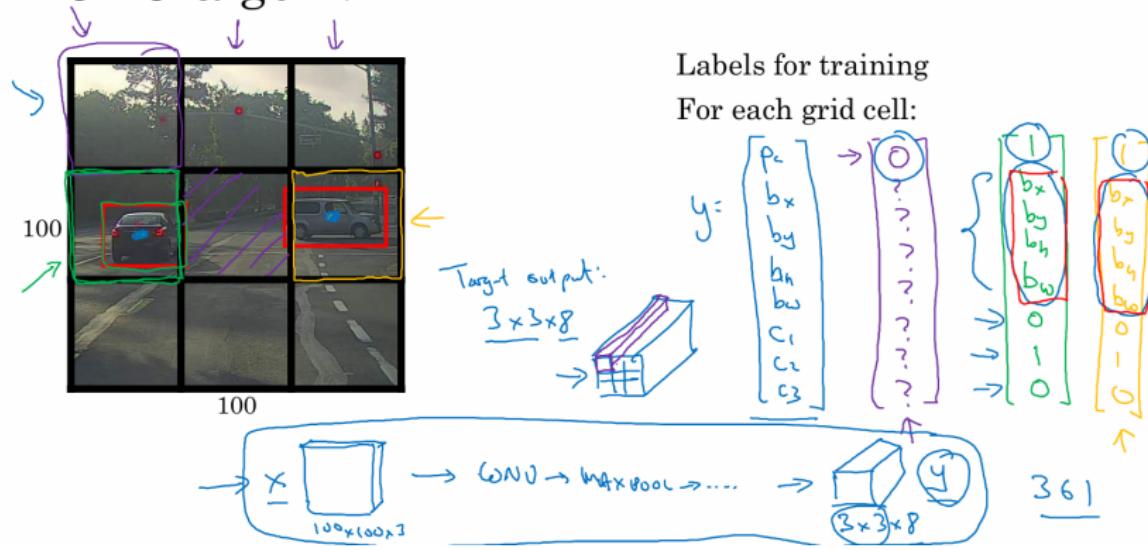
Sliding window pomocí konvoluce



YOLO (You Only Look Once)

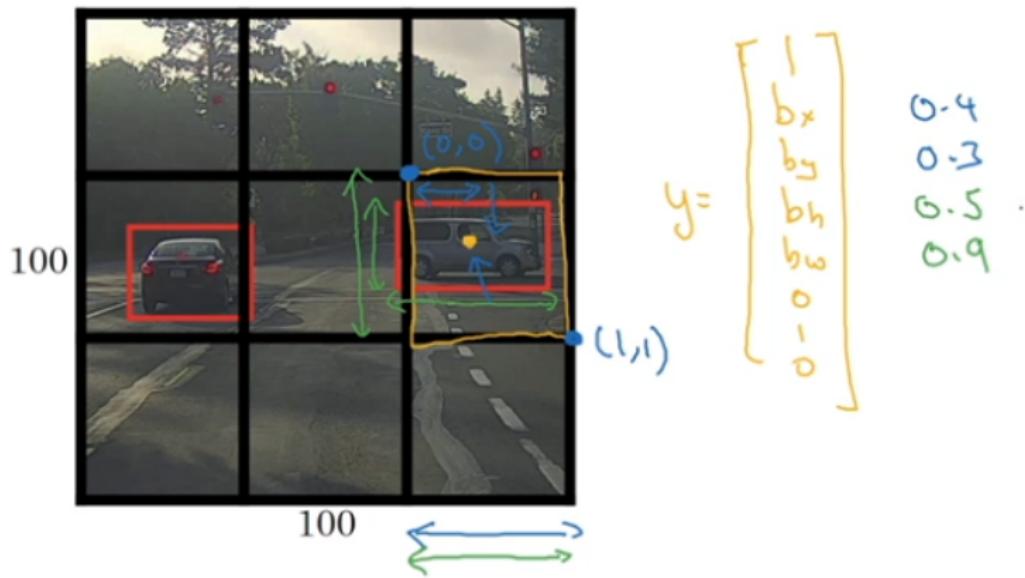
- Rychlá detekce objektů v reálném čase
- Rozdělení obrázku na grid a predikce bounding boxů a tříd
- Využití konvolučních vrstev pro rychlosť

YOLO algorithm



YOLO - specifikace bounding boxu

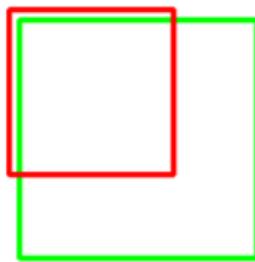
- Střed bounding boxu je vždy v oblasti s jedničkou na první pozici v y a je vždy v rozmezí 0 – 1.
- Výška a šířka bounding boxu se měří relativně vůči velikosti oblasti a může být větší než 1.



YOLO - Vyhodnocení

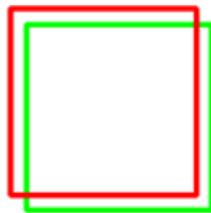
- Využití IoU (Intersection over Union) pro určení shody mezi predikcí a skutečností
- Podíl velikosti průniku a sjednocení bounding boxu skutečného a předpovězeného.

IoU: 0.4034



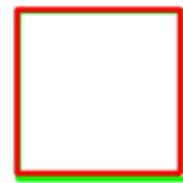
Poor

IoU: 0.7330



Good

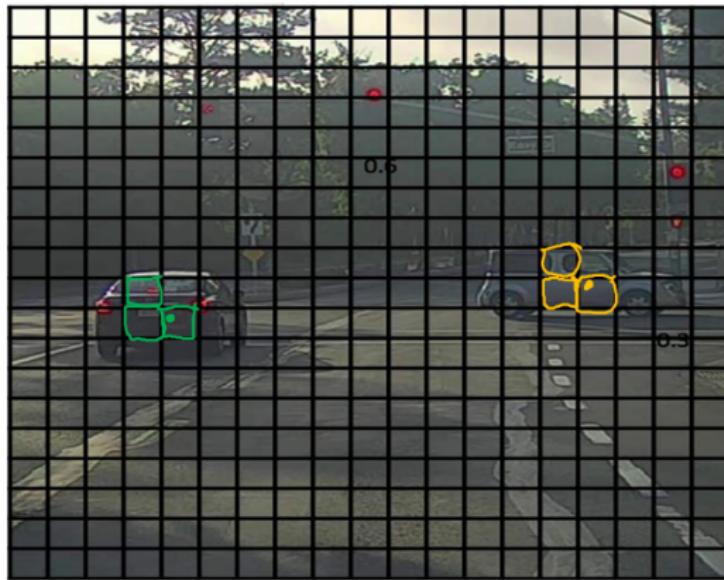
IoU: 0.9264



Excellent

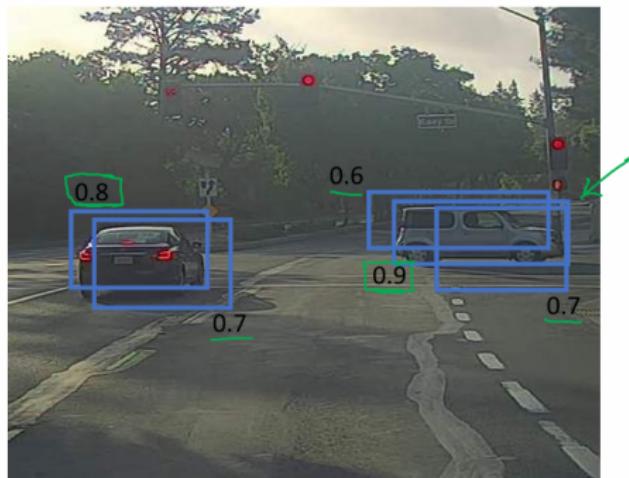
Non-max suppression

Co když se jeden objekt vyskytuje ve více oblastech?



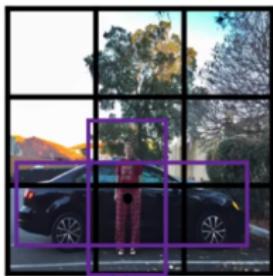
Non-max suppression

- Pro odstranění duplicitních detekcí, kdy je jeden objekt detekován ve více oblastech.
- Nejdříve odebereme všechny bboxy s confidence score menší než prahová hodnota, obvykle 0,6.
- Pro každou zbylou detekci spočítáme IoU s ostatními detekcemi.
- Pokud je IoU mezi dvěma detekcemi větší než prahová hodnota 0,5, ponecháme pouze tu s vyšší confidence score.

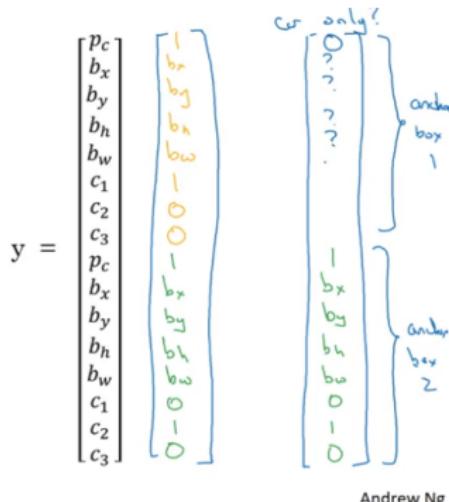
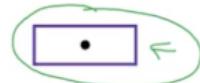


Překrývající se objekty

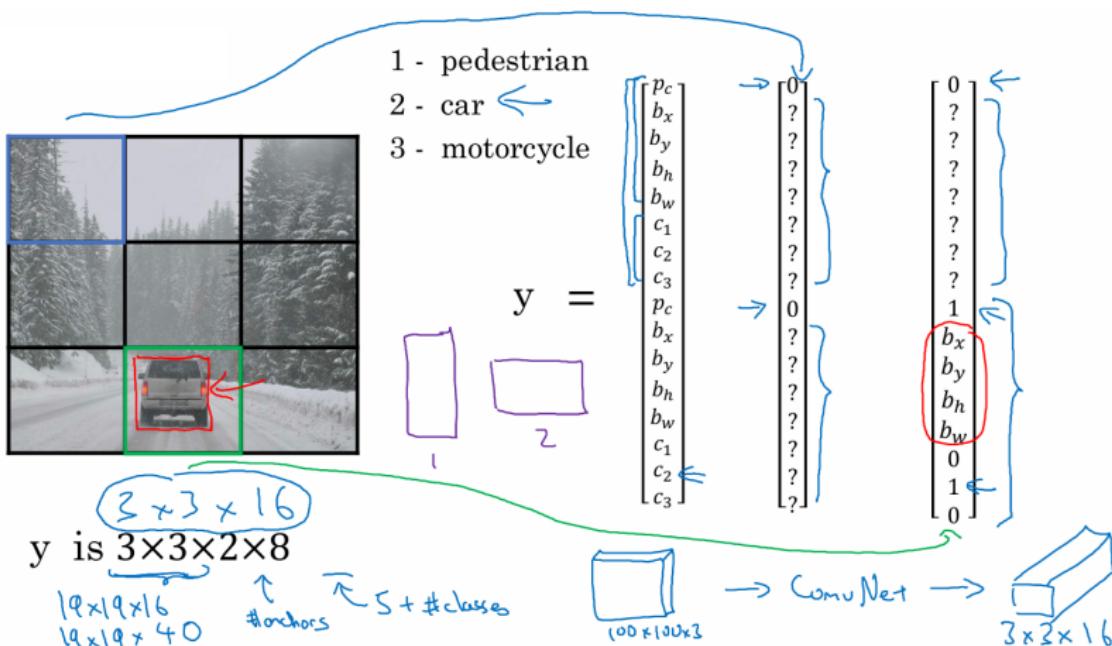
- Co když se objekty překrývají?
- Využití anchor boxes pro detekci různých velikostí objektů
- Anchor boxes: předdefinované bounding boxy různých poměrů stran a velikostí



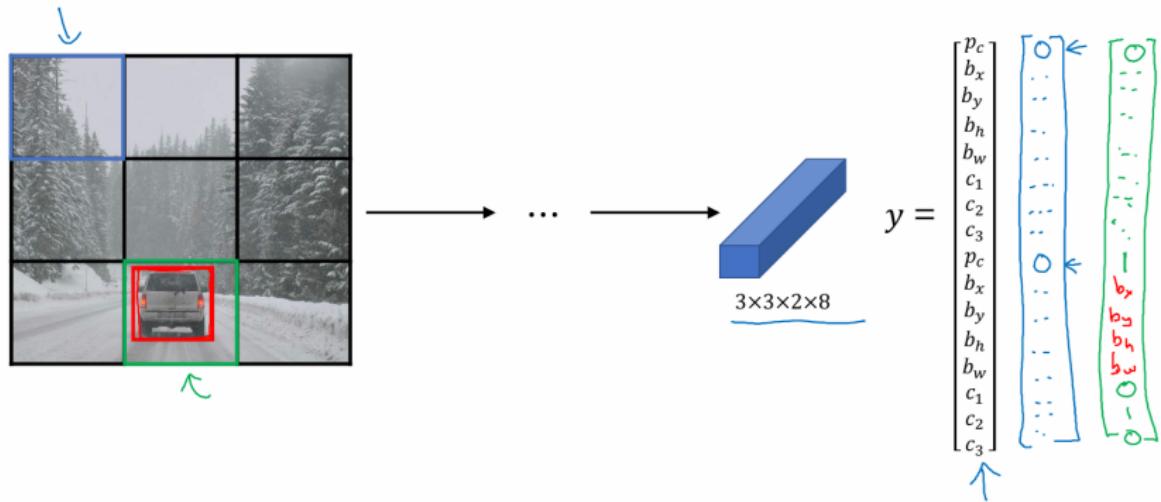
Anchor box 1: Anchor box 2:



YOLO - trénování



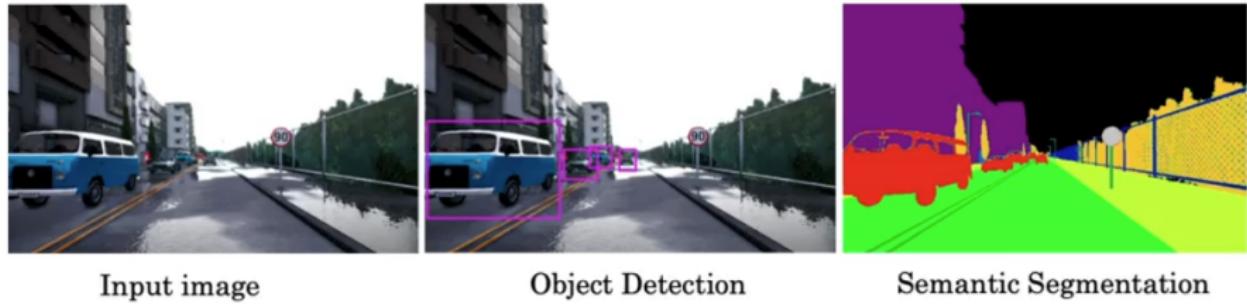
YOLO - testování



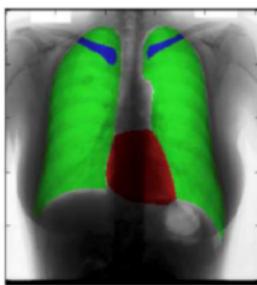
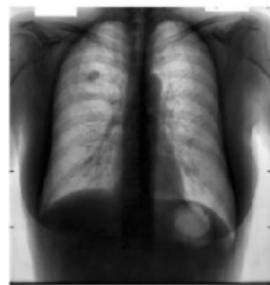
YOLO - postprocessing

- Pro každou oblast obdržíme 2 (nebo více, dle našeho nastavení počtu anchor boxů) bboxů.
- Odstraníme bboxy s nízkým confidence score.
- Pro každou detekovanou třídu objektů, použijeme non-max suppression.
- Výstupem je seznam detekovaných objektů s jejich třídou a bounding boxem.
- Využití v reálném čase (např. sledování objektů, detekce anomalií)

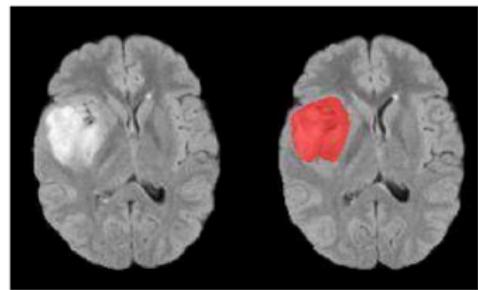
Segmentace obrazu



Segmentace obrazu - aplikace



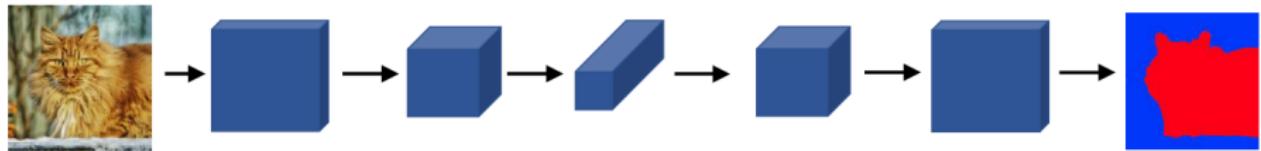
Chest X-Ray



Brain MRI

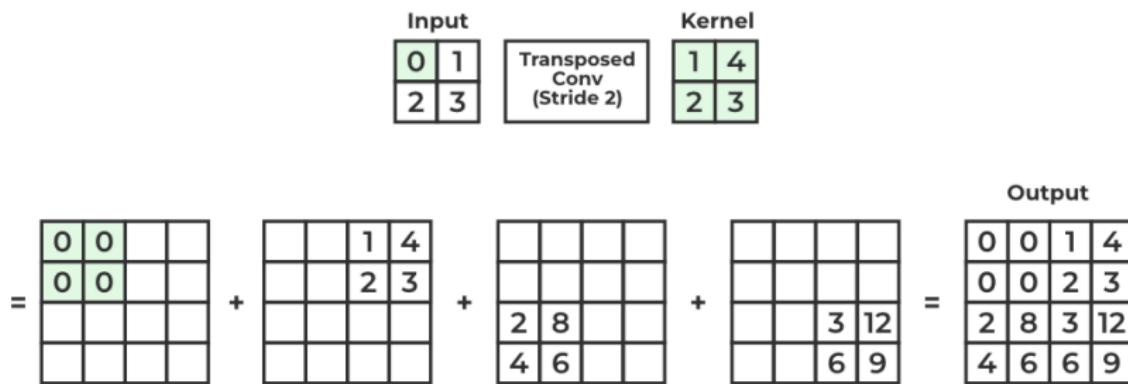
Základní myšlenka - U-Net

- Zakódování informace o třídě objektu
- Dekódování do požadovaného formátu
- Využití konvolučních vrstev pro extrakci rysů

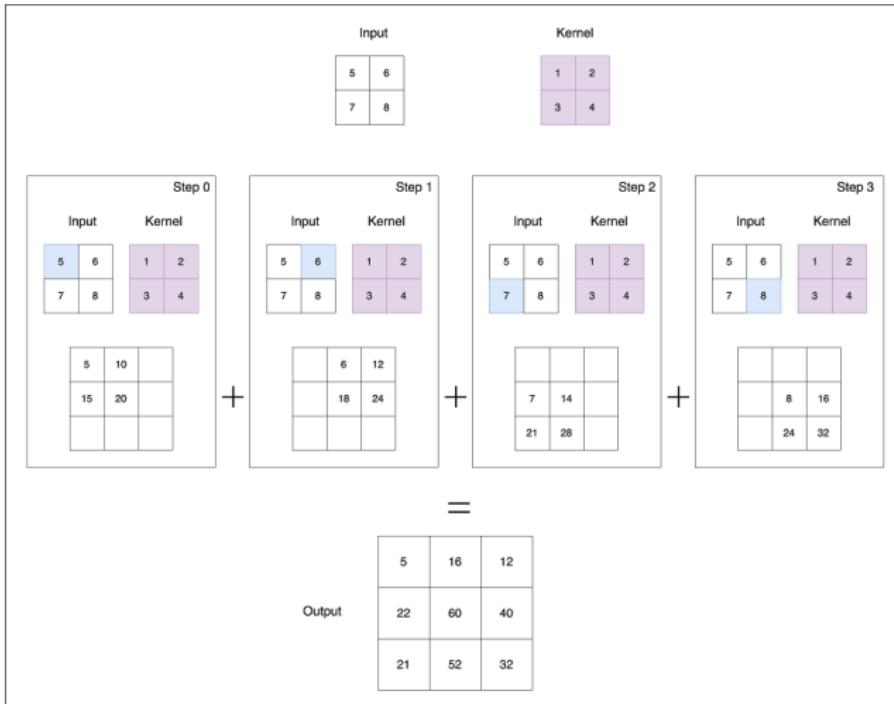


Transponovaná konvoluce

- Transponovaná konvoluce: inverzní operace k běžné konvoluci
- Použití pro dekódování (up-sampling) obrazu
- Vytváříme větší obraz s menším počtem kanálů
- Využití v U-Net architektuře pro dekódování

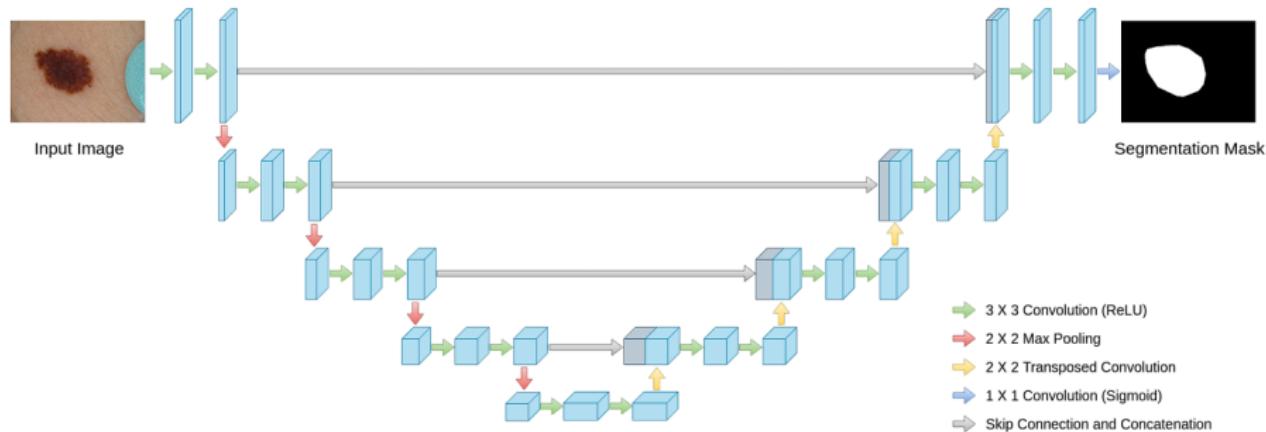


Transponovaná konvoluce - parametry



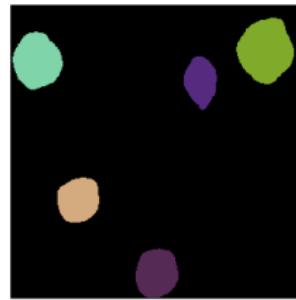
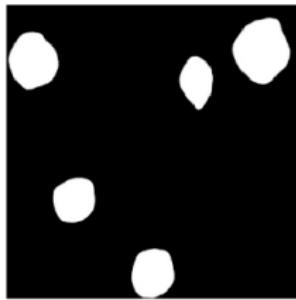
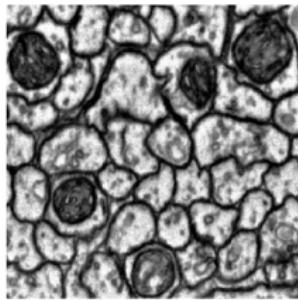
U-Net

- Zakódování - stejné jako při klasifikaci, určení co je v obrazu
- Dekódování - určení, kde se objekty nacházejí - připojujeme informaci o poloze z předchozího zpracování.



U-Net - příprava datasetu

- Dataset: obrázky a masky (binární obrázky s informací o třídě)
- Masky: 0 - pozadí, 1 - objekt, případně další třídy další hodnoty, v grafickém software



U-Net - příprava datasetu

- Rozdělíme dataset na trénovací a testovací data: 80% trénovací, 20% testovací
- Trénovací data: obrázky a masky
- Google Colab - K vyzkoušení