

Rozpoznávání snímků, detekce objektů, moderní trendy

Strojové vidění a zpracování obrazu (BI-SVZ)

Úlohy v oboru počítačového vidění

- Klasifikace obrázků
- Lokalizace objektů
- Detekce objektů
- Sémantická segmentace
- Segmentace instance
- Textový popis obrázků

- Stovky dalších...

Pro začátečníka je těžké tyto obory rozlišit

Nejčastější úlohy v počítačovém vidění

Classification



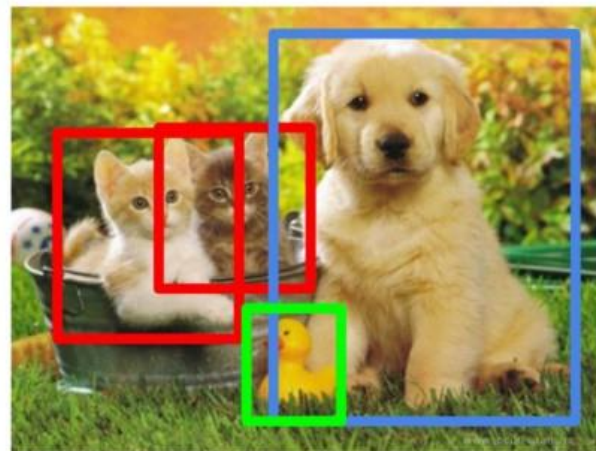
CAT

**Classification
+ Localization**



CAT

Object Detection



CAT, DOG, DUCK

**Instance
Segmentation**



CAT, DOG, DUCK

Single object

Multiple objects

Přístupy k úlohám detekce a rozpoznávání

- Tradiční způsoby
 - Barevné rozpoznávání
 - Tvarové rozpoznávání
 - Šablonové rozpoznávání
 - Výpočty příznaků
 - Klasifikace
- Metody založené na hlubokém učení (deep learning)
- Kombinace předchozích způsobů

Historie rozpoznávání snímků a detekce objektů

- 2001

- První real-time algoritmus pro detekci obličejů od Paul Viola a Michael Jones
- Známý jako Haar Cascades k nalezení v OpenCV

- 2005

- První použitelný algoritmus pro detekci osob od Navneet Dalal and Bill Triggs.
- Známý jako deskriptor Histograms of Oriented Gradients (HOG), k nalezení v OpenCV

- 2012

- Deep learningová síť od autorů Alex Krizhevsky, Ilya Sutskever, a Geoffrey Hinton šokuje svět výhrou v soutěži ImageNet dramatickým zvýšením přesnosti rozpoznávání

- 2015

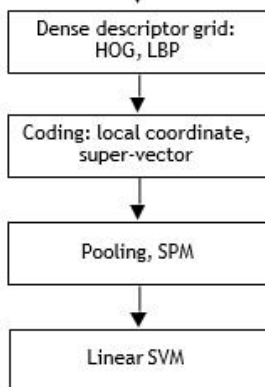
- Deep learning je mainstream, algoritmy překonaly přesnost rozpoznávání lidí
- Přesnost klasifikace snímků převyšuje 95 %

Historie rozpoznávání snímků a detekce objektů

IMAGENET Large Scale Visual Recognition Challenge

Year 2010

NEC-UIUC

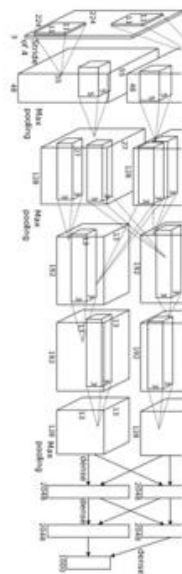


[Lin CVPR 2011]

[Lion image](#) by Swissfrog is
licensed under [CC BY 3.0](#)

Year 2012

SuperVision

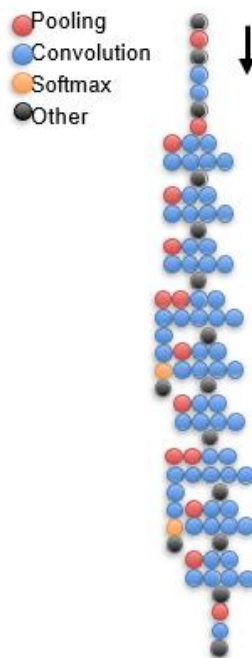


[Krizhevsky NIPS 2012]

Figure copyright Alex Krizhevsky, Ilya
Sutskever, and Geoffrey Hinton, 2012.
Reproduced with permission.

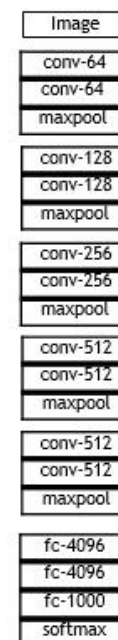
Year 2014

GoogLeNet



[Szegedy arxiv 2014]

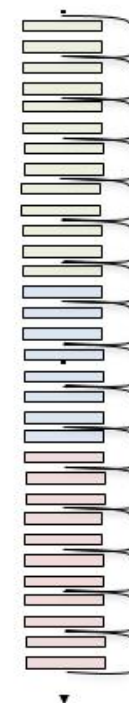
VGG



[Simonyan arxiv 2014]

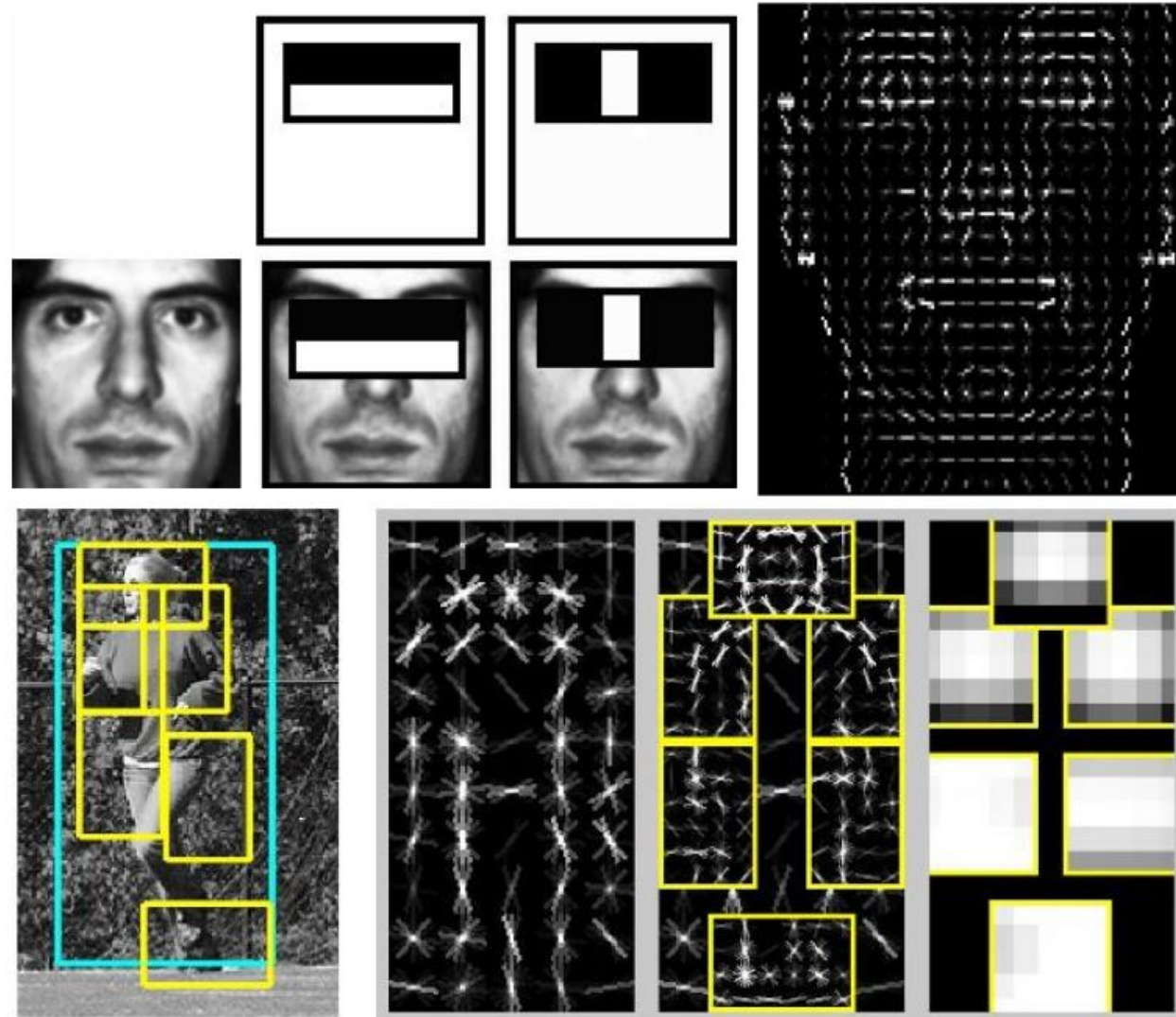
Year 2015

MSRA



[He ICCV 2015]

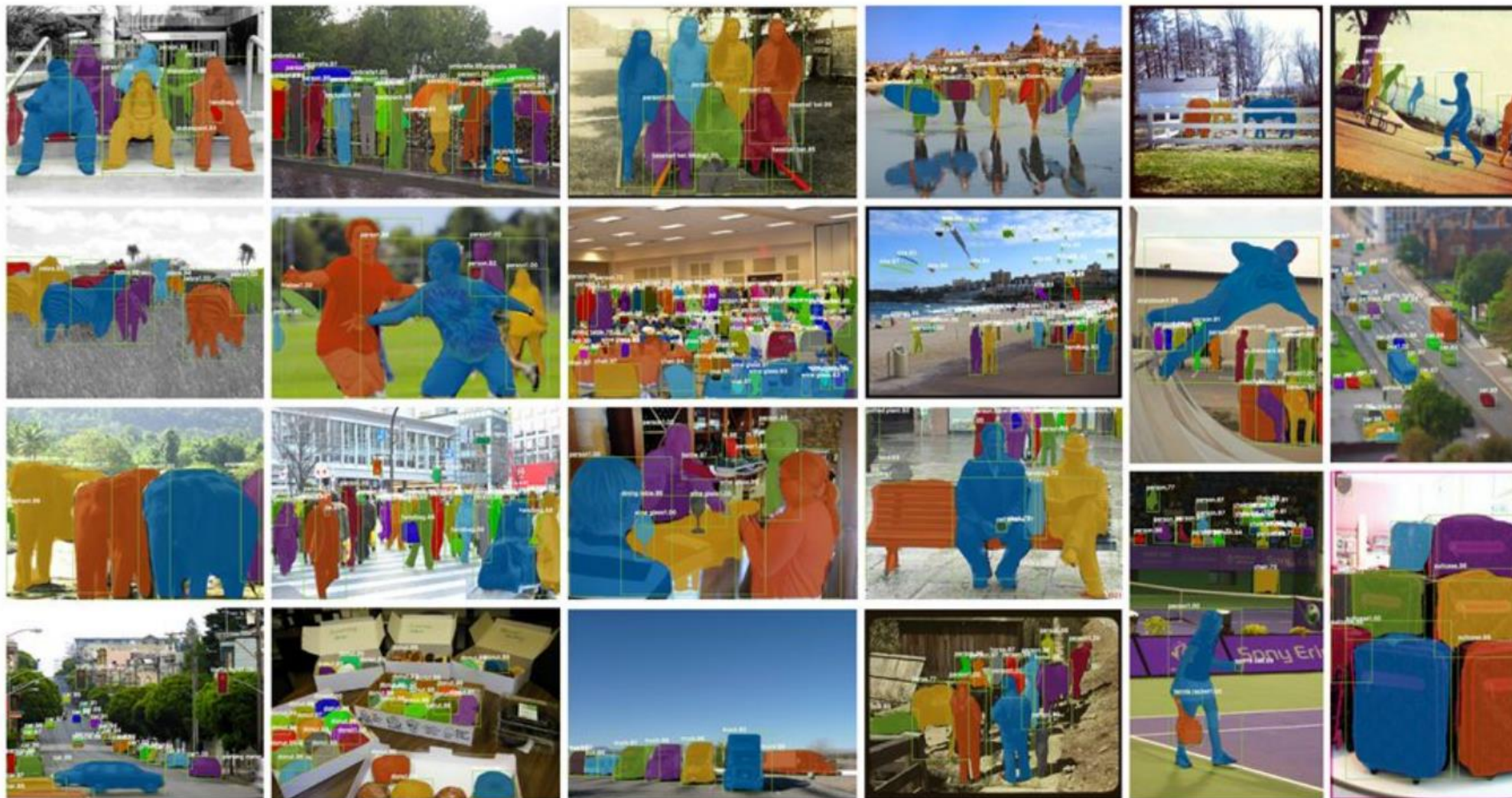
Detekce objektů 2001 – 2007



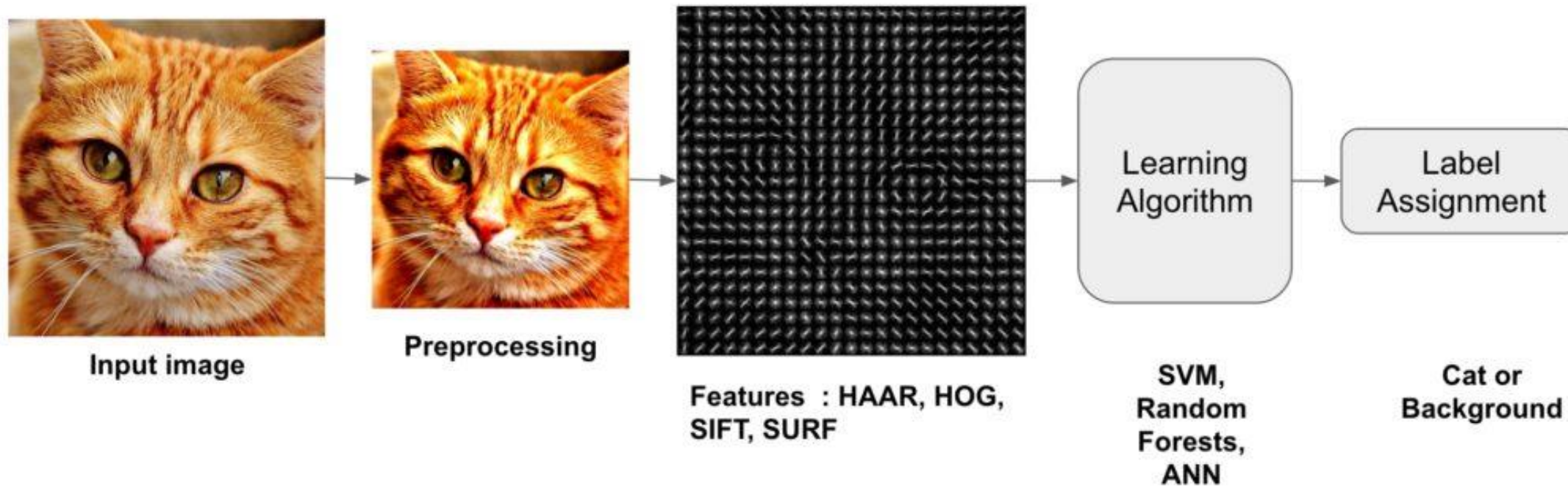
Detekce objektů 2007 – 2012



Detekce objektů nyní



Tradiční techniky počítačového vidění - pipeline



Většina tradičních způsobů respektuje tuto pipeline, deep learningové algoritmy zpravidla přeskakují část s extrakcí příznaků

Rozpoznávání obrazu (klasifikace obrazu)

- Vstupem je obraz
- Výstupem je třída, popisující co daný obraz obsahuje
 - Tedy např.: kočka, pes, auto, člověk
- Algoritmus rozpoznávání je potřeba natrénovat, tím zajistíme rozdílné klasifikace mezi různými třídami
- Komponentu, kterou trénujeme pomocí vstupních dat, nazýváme klasifikátor
- Pokud tedy chceme klasifikovat kočky a psy v obraze, musíme mít stovky (až tisíce) trénovacích vzorků
- Klasifikátor zvládne samozřejmě klasifikovat pouze ty objekty, které předtím viděl
- Pro zjednodušení dále uvažujme pouze binární klasifikátory, kam patří i např. detekce lidí a obličejů

Oblíbený challenge – dog vs mop

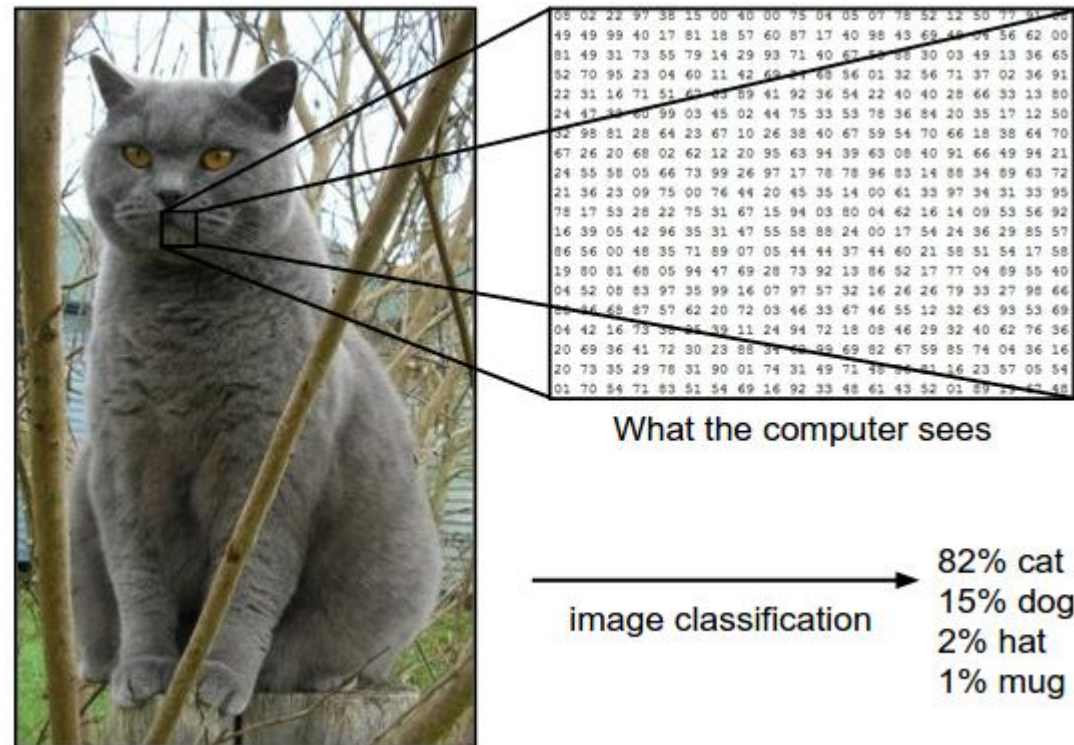


Pipeline rozpoznávání obrazu – předzpracování

- Ve většině případu je vstupní obraz předzpracován, časté operace:
 - Úprava jasu a kontrastu
 - Gamma korekce
 - Ekvalizace histogramu
 - Odečtení průměru obrazu a vydělení standardní odchylkou (normalizace)
 - Převod do různých barevných prostorů (RGB na HSV)
 - Perspektivní transformace
 - Ořez a škálování do předem daných rozměrů (kvůli extrakci příznaků)
- Trik je v tom, že nikdo ve skutečnosti neví, jaké operace předzpracování je potřeba
- Vše vychází z experimentů, kde ověřujeme to, který druh předzpracování poskytuje přesnější výsledky.

Pipeline rozpoznávání obrazu – extrakce příznaků

- Vstupní předzpracovaný obraz má v sobě příliš mnoho informací, které nejsou pro klasifikaci nutné
- Obraz s šířkou 248, výškou 400, v RGB prostoru obsahuje 297 600 čísel



Pipeline rozpoznávání obrazu – extrakce příznaků

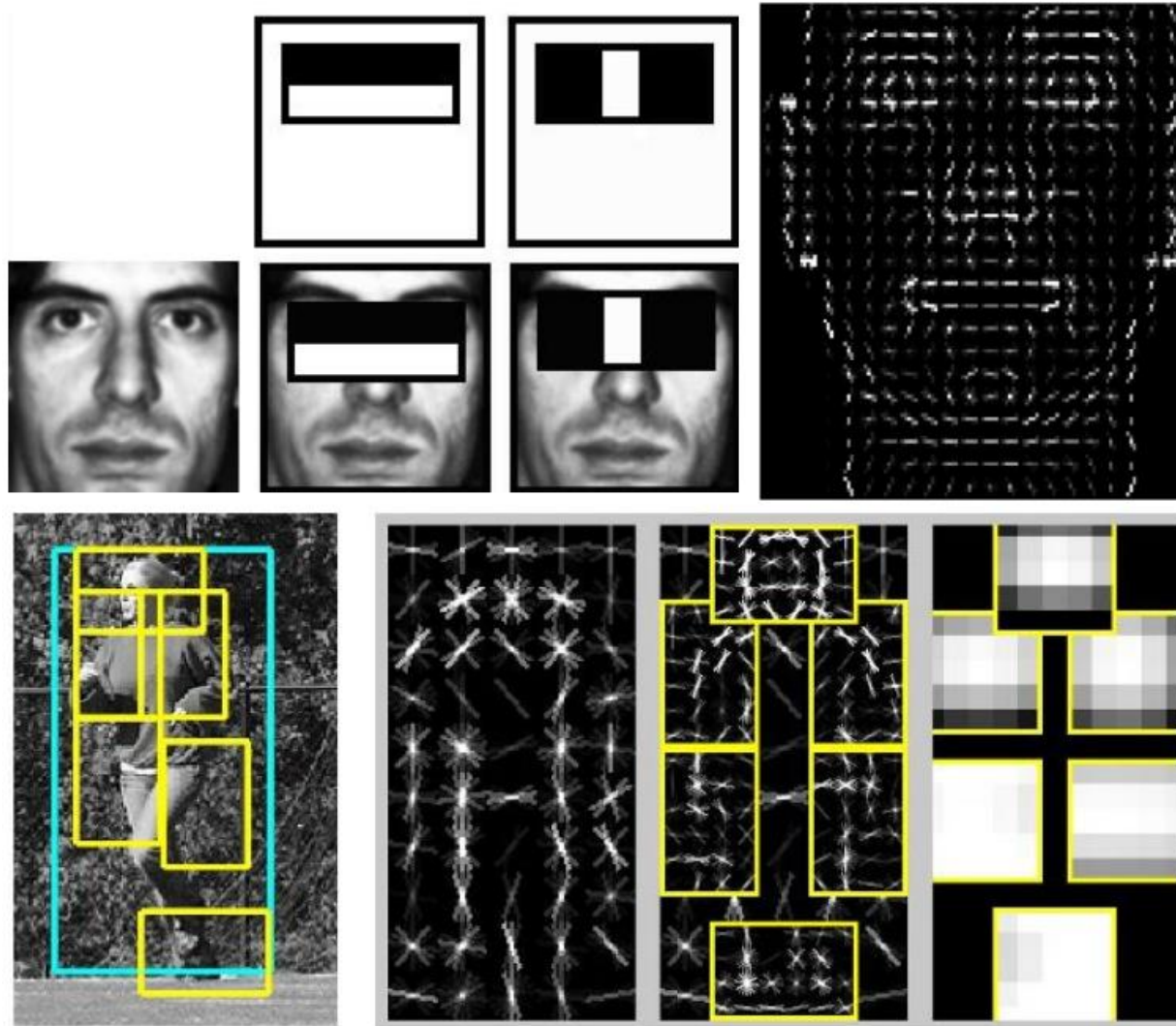
- Proto je prvním krokem v klasifikaci obrázků zjednodušení obrazu extrahováním důležitých informací (příznaků) obsažených v obraze a vynecháním ostatních informací
- Pokud budeme chtít hledat knoflíky na košili v obraze, zjistíme, že máme v okolí knoflíků významně rozdílné hodnoty RGB pixelů
- Nicméně, pomocí spuštění hranového detektoru můžeme obrázek zjednodušit a stále dokážeme snadno rozpoznat kruhový tvar knoflíků. Tedy zachováváme klíčové informace a zahazujeme nepotřebné (RGB hodnoty)



Pipeline rozpoznávání obrazu – extrakce příznaků

- V tradičních přístupech je navrhování extrahovaných příznaků rozhodující pro přesnost algoritmu
- Existují samozřejmě robustnější způsoby k extrakci příznaků, než je samotná hranová detekce, mezi nejznámější patří:
 - [Haar Cascades](#)
 - [Histogram of Oriented Gradients](#)
 - [Harris corners](#)
 - [SIFT](#)
 - [SURF](#)
 - [Local Binary Patterns \(LBP\)](#)
 - Histogramy

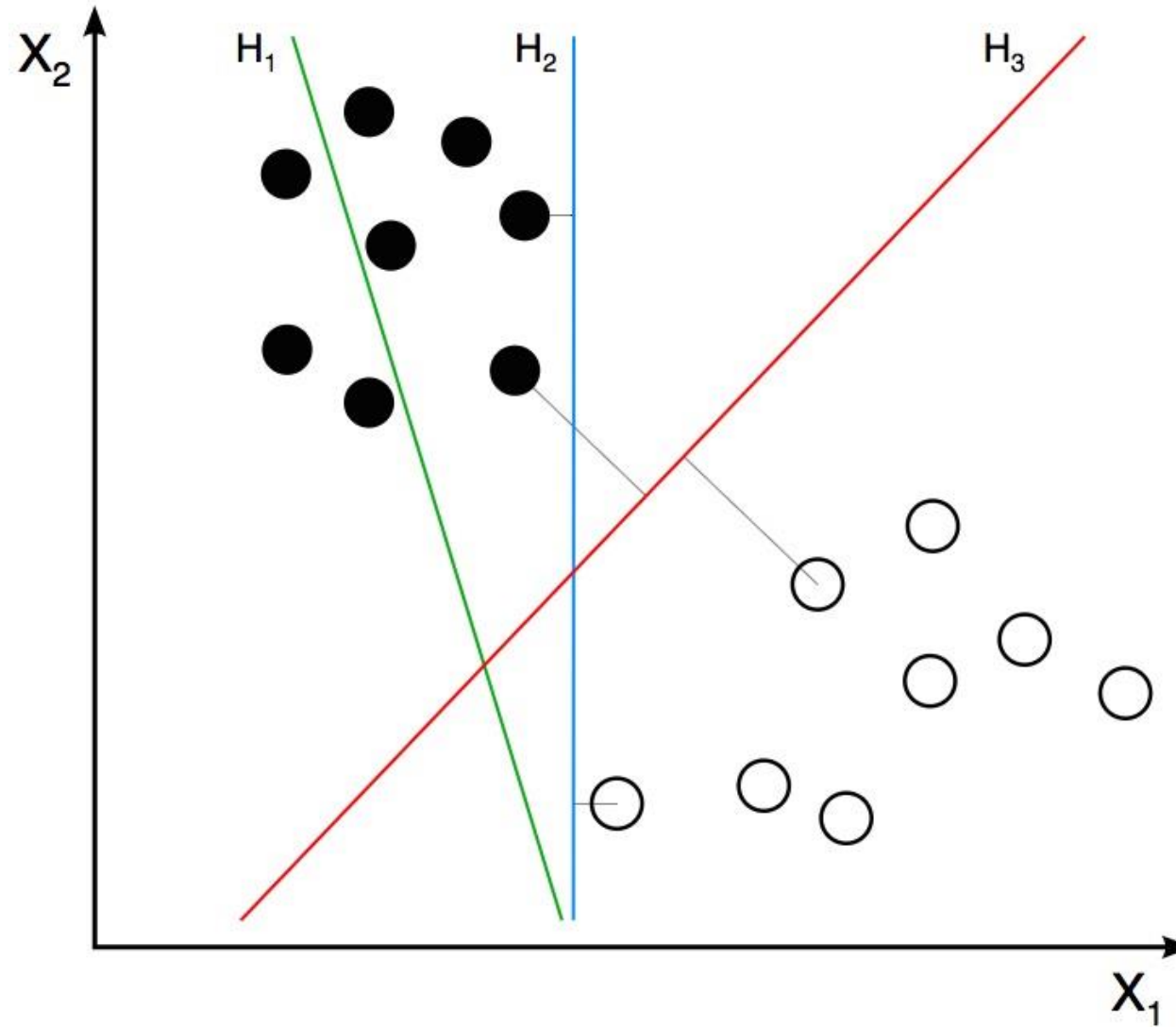
Pipeline rozpoznávání obrazu – extrakce příznaků



Pipeline rozpoznávání obrazu – klasifikátor

- Již víme, jak převést obraz na vektor důležitých příznaků
- Nyní potřebujeme vzít tento vektor, dát ho na vstup klasifikátoru a rozhodnout o jakou třídu objektu se jedná – kočka nebo pes
- Předtím než je klasifikátor použitelný, musíme jej natrénovat na tisících příkladech
- Jak takový klasifikátor funguje, si popíšeme na jednom z donedávna nejpoužívanějších binárních klasifikátorů Support Vector Machines (SVM)
 - SVM se snaží nalézt „ideální“ rovinu, pomocí které rozdělí prostor koček a psů

Pipeline rozpoznávání obrazu – klasifikátor SVM

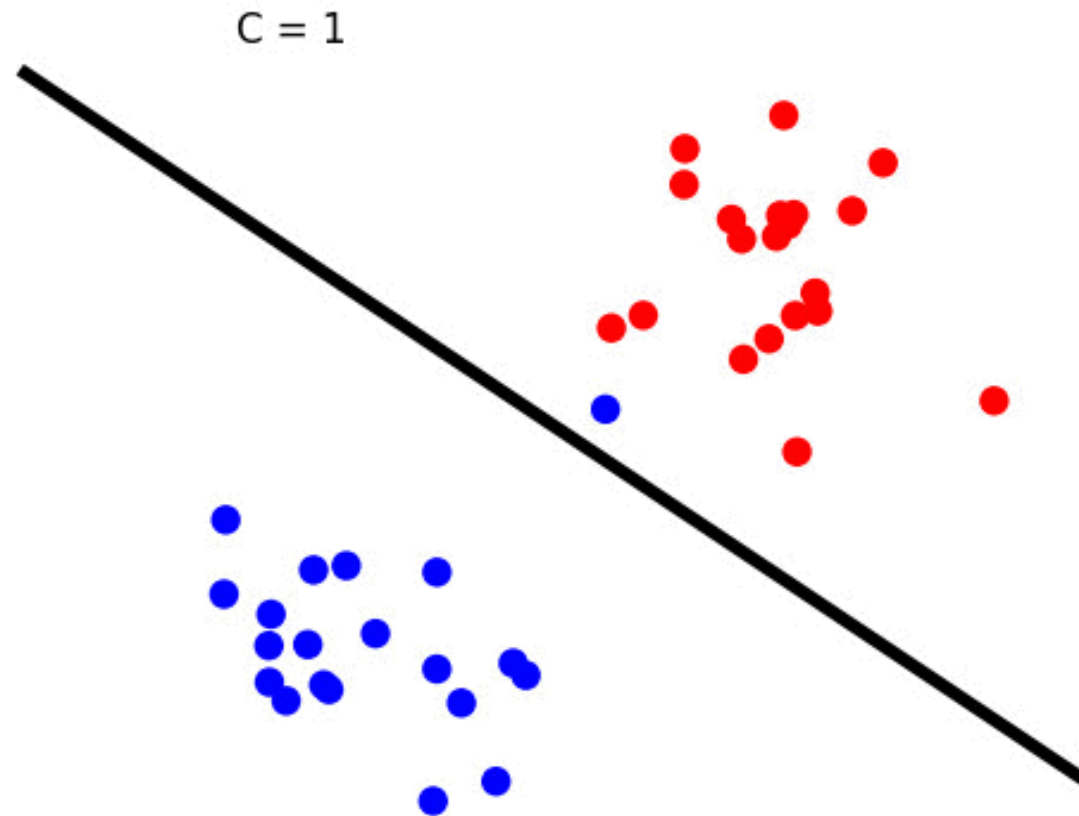


Pipeline rozpoznávání obrazu – klasifikátor

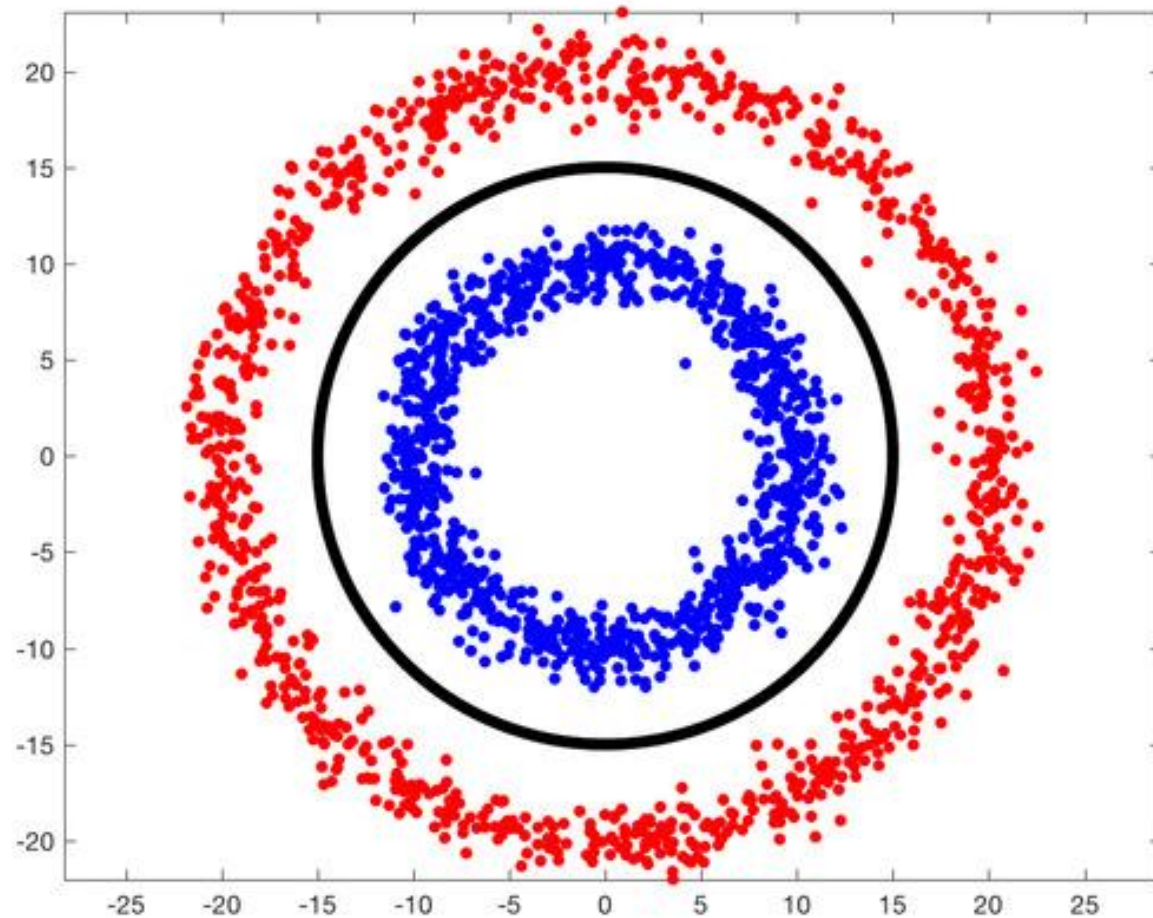
- V předchozím zjednodušeném příkladu máme pouze body ve 2D reprezentující dvě třídy - kočka nebo pes
- Černé tečky patří do třídy kočka, bílé tečky do třídy pes
- Během trénování iterativně hledáme nejlepší přímku rozdělující tyto dvě třídy
- K určení toho, jak si náš klasifikátor vede se využívá ztrátová funkce (loss function)
- Obecně se nepohybujeme pouze v 2D prostoru, ale v tisíce dimenzionálním – ten se ale hůř vizualizuje

Co když třídy nejdou jednoznačně oddělit?

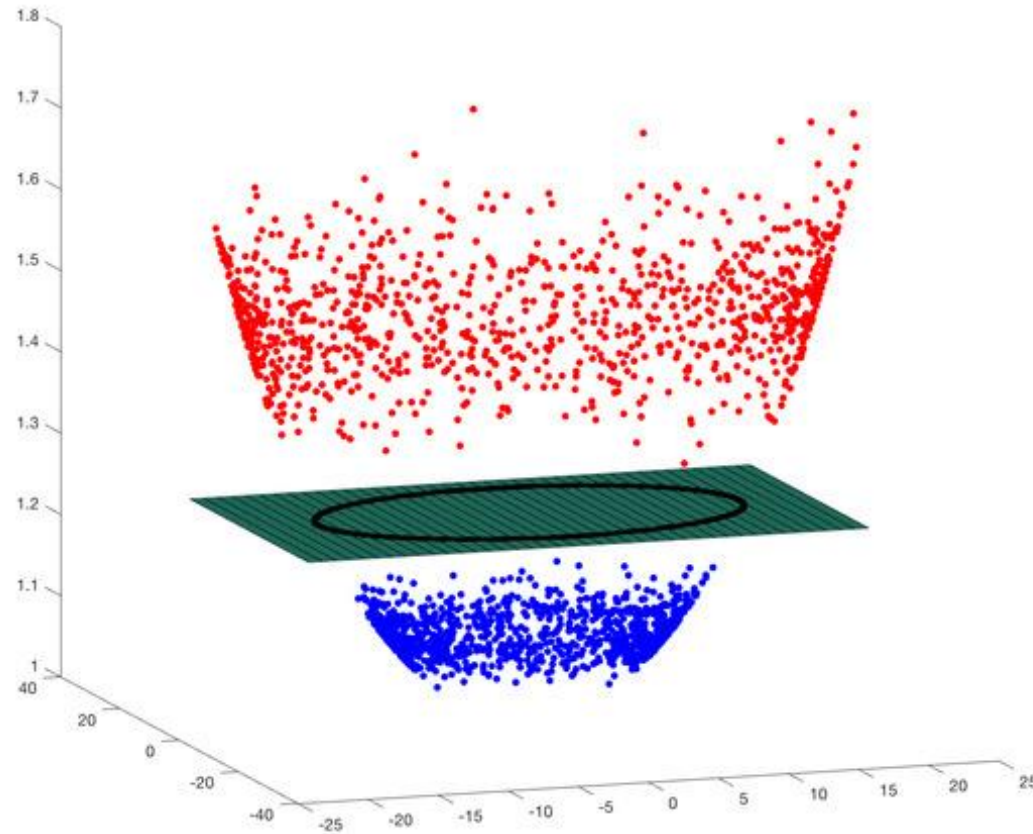
Pipeline rozpoznávání obrazu – klasifikátor



Pipeline rozpoznávání obrazu – klasifikátor



Pipeline rozpoznávání obrazu – klasifikátor



Problémy tradičních technik klasifikace?

Problémy tradičních technik klasifikace

- Rotace
- Barevnost
- Osvětlení
- Škálování
- Doménový posun v datech (concept drift, domain shift)
- **Obecně nedostatečná generalizace**

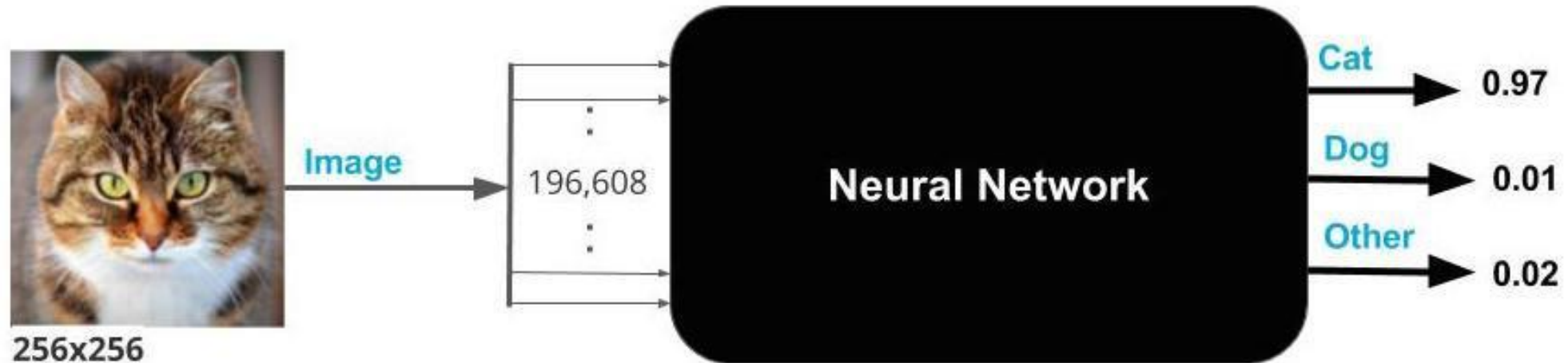
Deep learning, neuronové sítě - pipeline

- Minimální úsilí na předzpracování snímku a využívání black-boxu



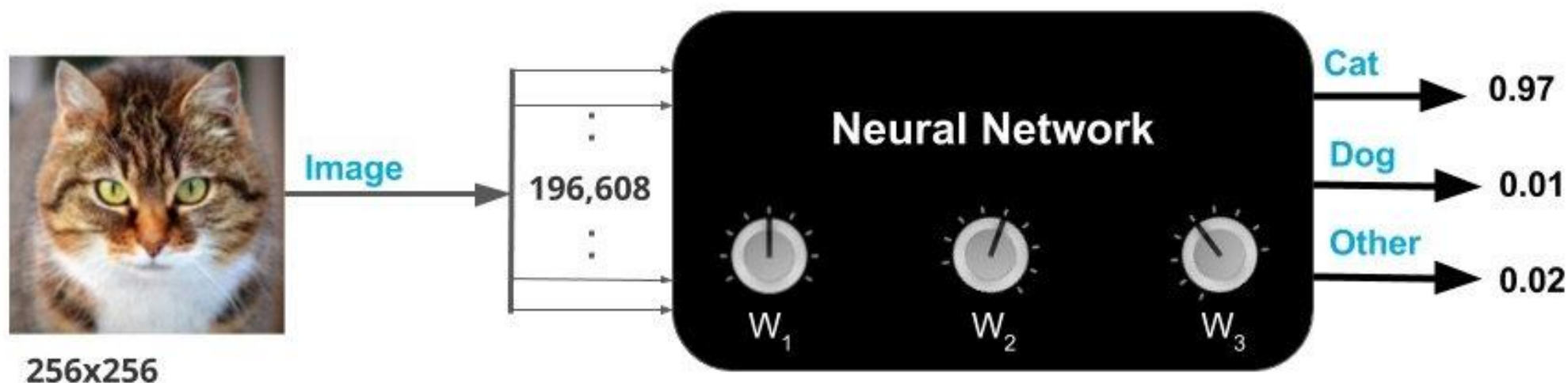
Deep learning, neuronové sítě - pipeline

- Vstupní snímek však musíme převést alespoň do vhodné reprezentace – vektoru fixní délky

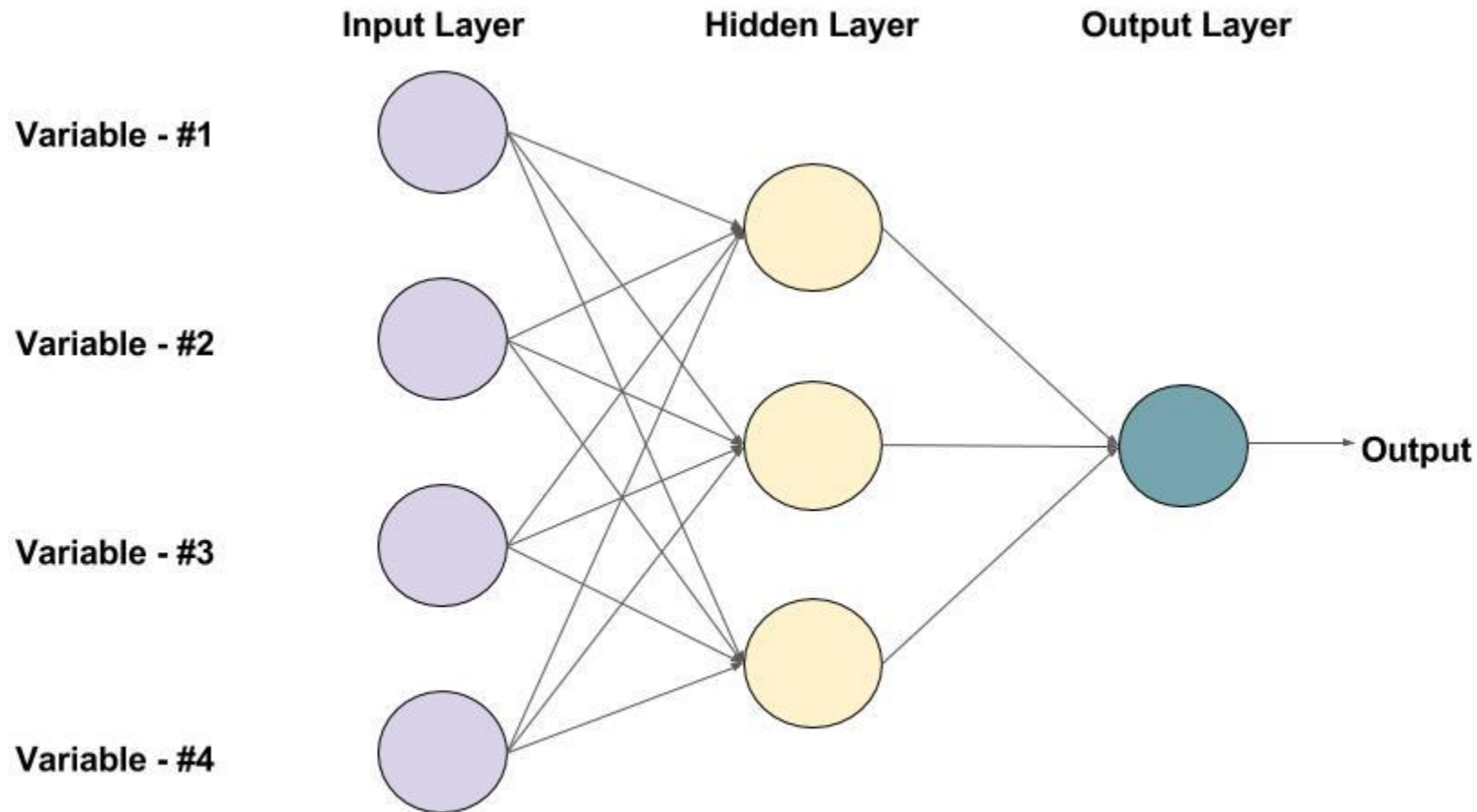


Deep learning, neuronové sítě - pipeline

- Neuronové sítě k predikci využívají miliony vnitřních parametrů (váhy neuronů), které je k dosažení smysluplných výsledků, nutno správně natrénovat



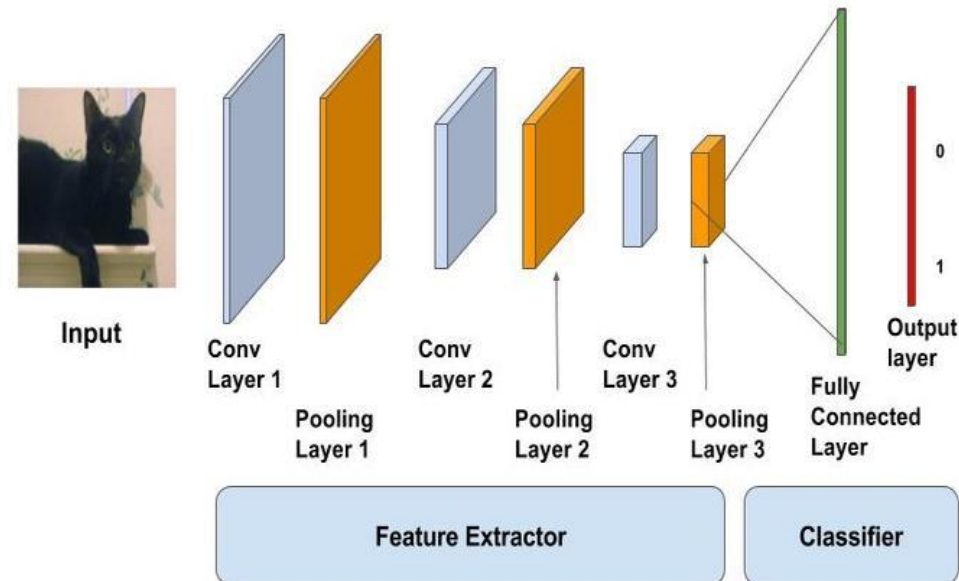
Deep learning, neuronové sítě – Feed-forward NN



An example of a Feed-forward Neural Network with one hidden layer (with 3 neurons)

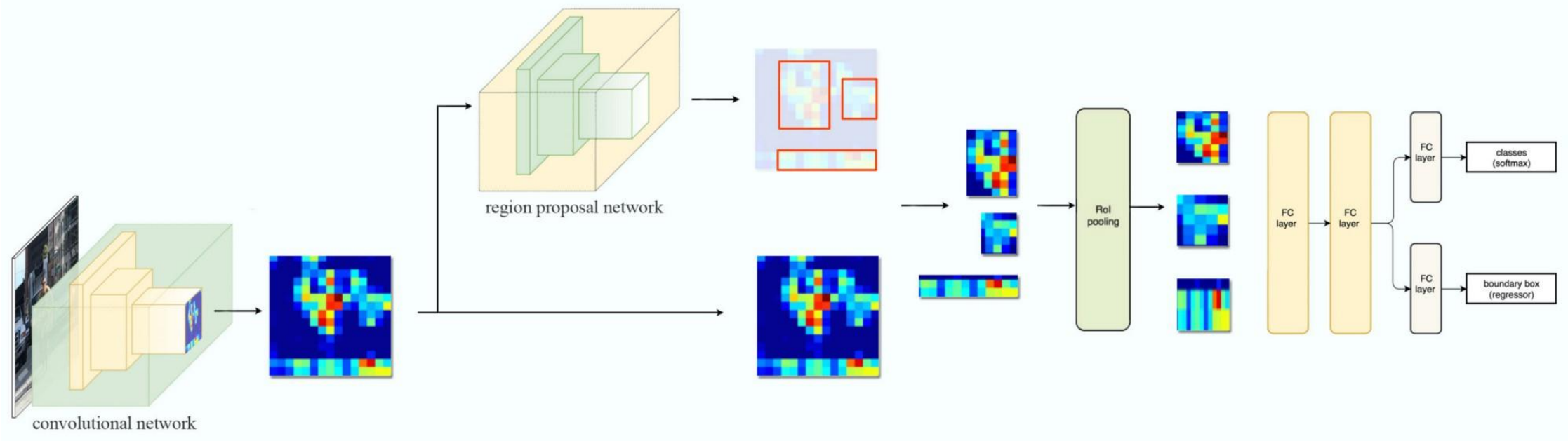
Deep learning, neuronové sítě – CNN

- Na obrázky se však nejčastěji používá Convolution Neural Network místo samotné Feed-forward NN
 - CNN se skládá z konvolučních a pooling prstev, pomocí kterých se snažíme získat vhodné atributy. Ty však již nelze rozumně interpretovat
 - Následně tyto atributy použijeme na vstup Feed-forward NN pro klasifikaci



Deep learning, neuronové sítě – Mask R-CNN

- Existují samozřejmě mnohem složitější architektury, které se nestarají pouze o klasifikaci snímků, ale i o detekci několika objektů, případně získání jejich segmentační masky



Deep learning, neuronové sítě – detekce objektů



CAT? NO
DOG? NO

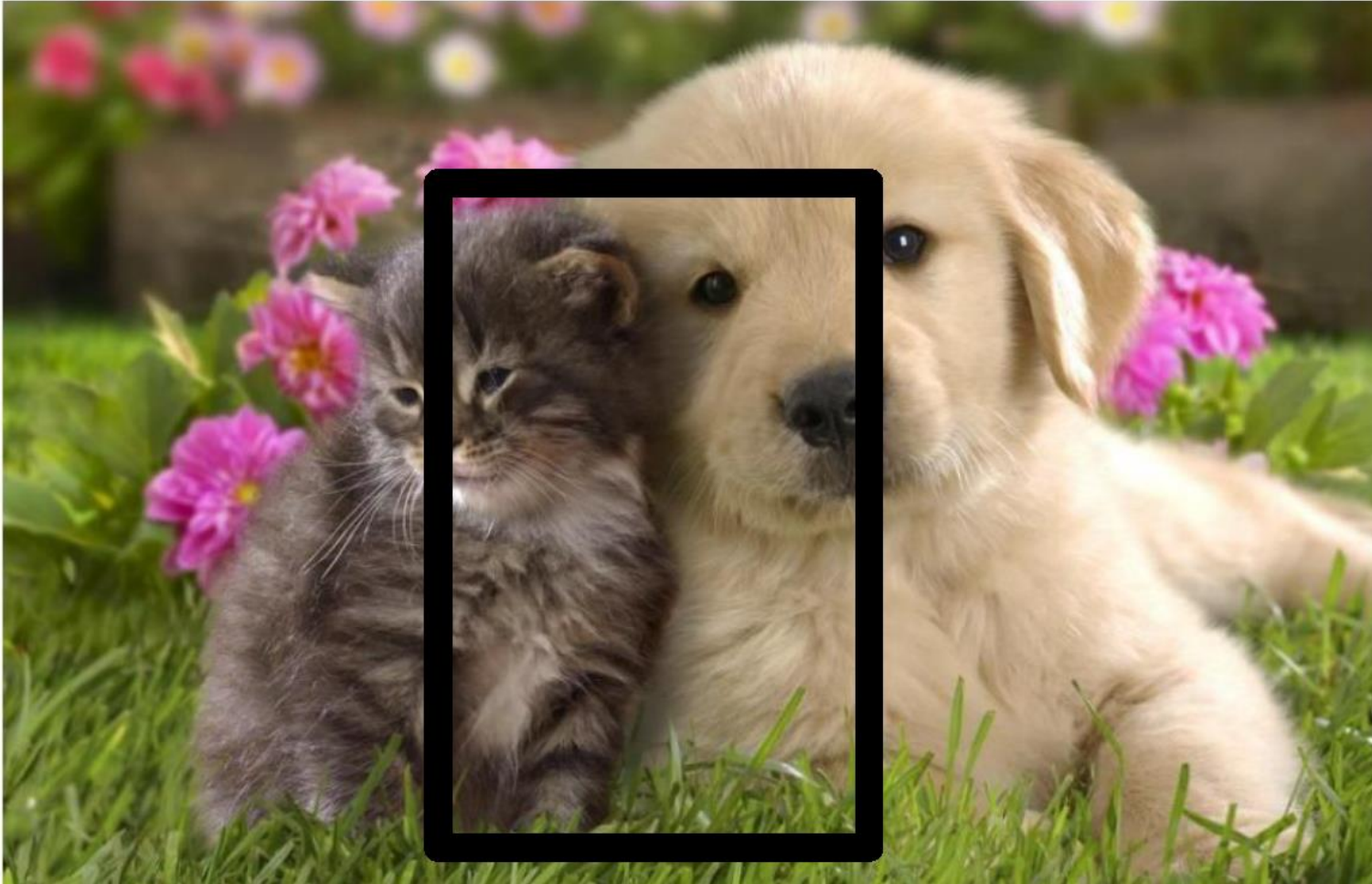
Deep learning, neuronové sítě – detekce objektů



CAT? YES

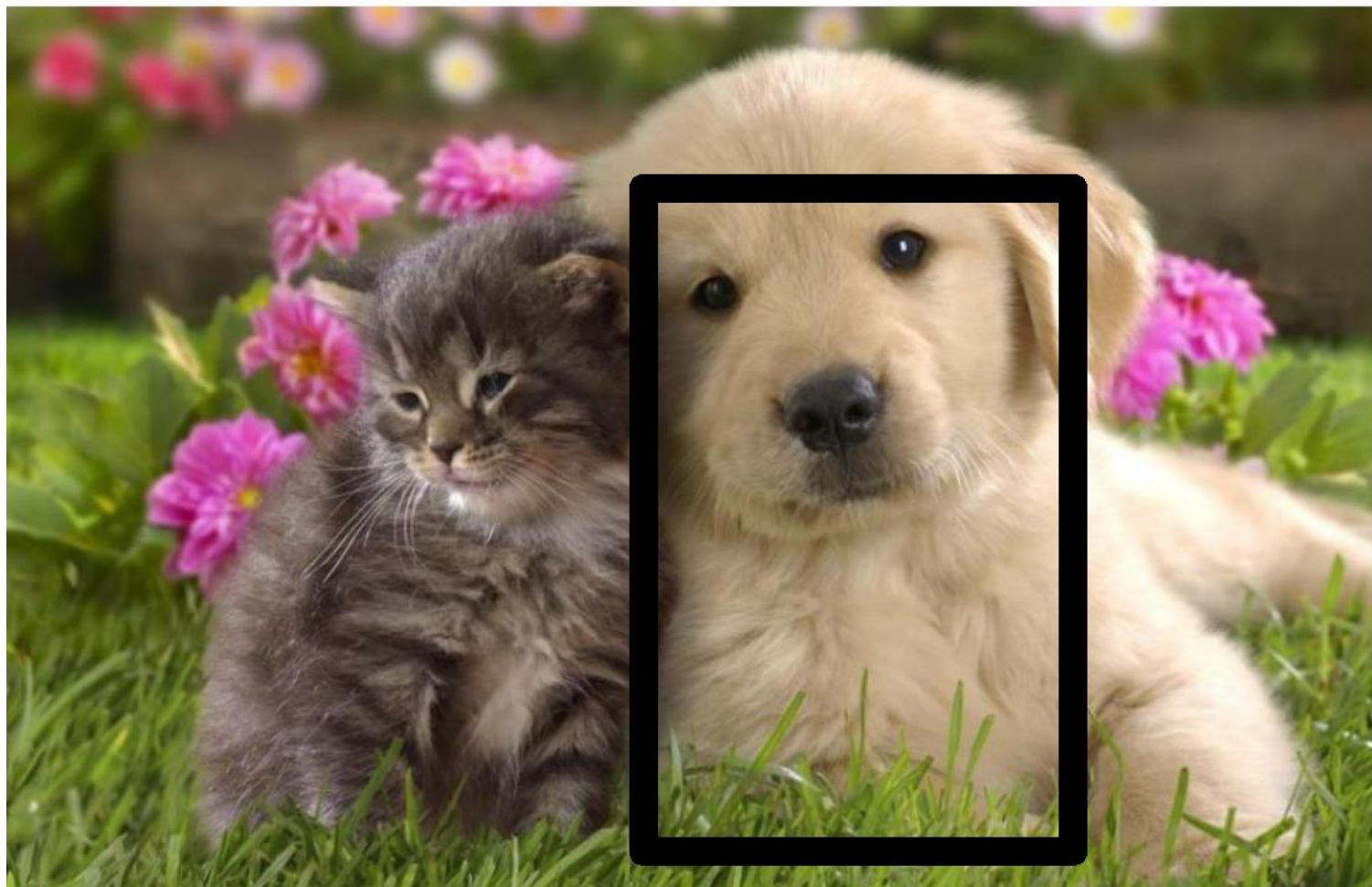
DOG? NO

Deep learning, neuronové sítě – detekce objektů



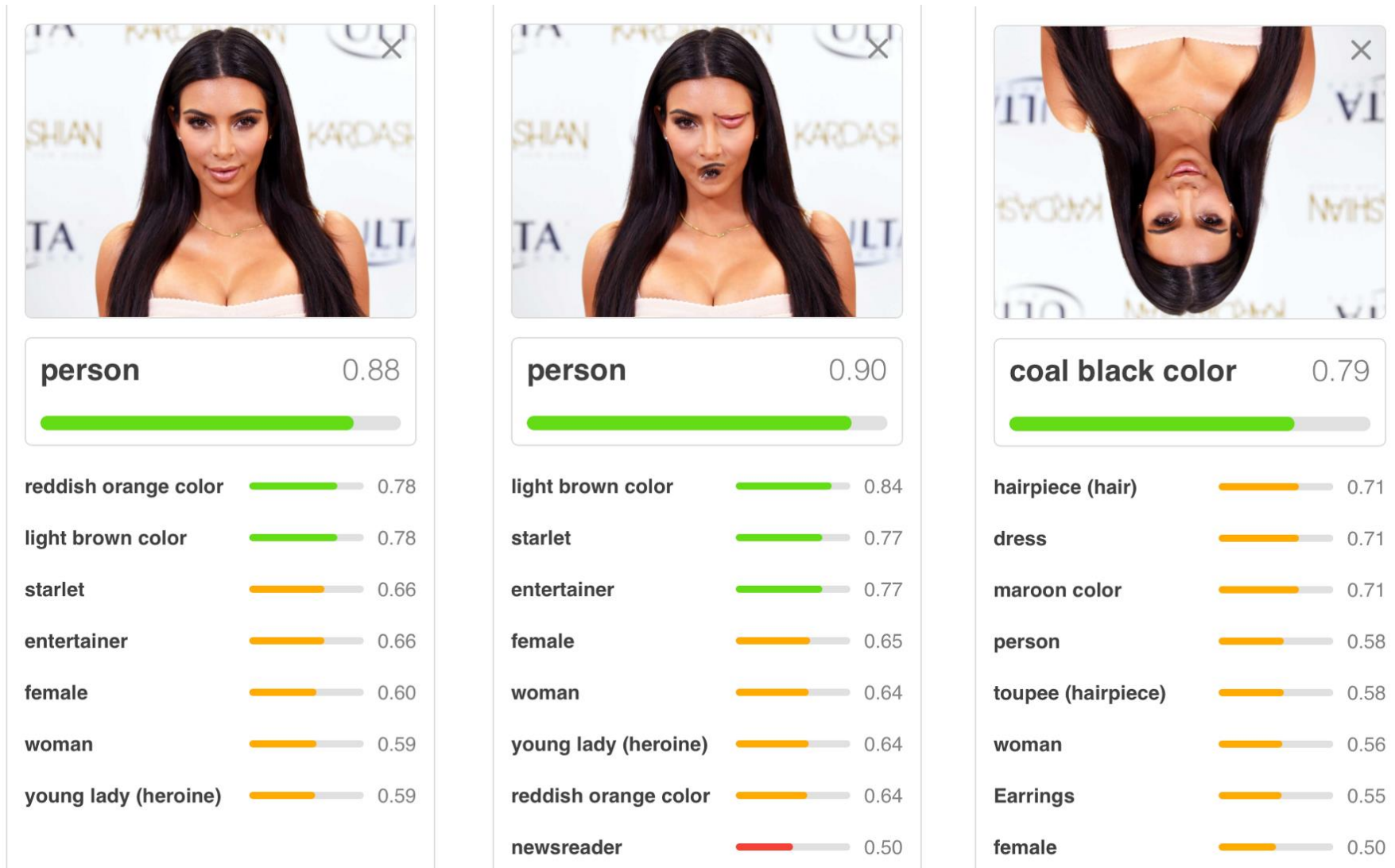
CAT? NO
DOG? NO

Deep learning, neuronové sítě – detekce objektů



CAT? NO
DOG? YES

Problémy neuronových sítí – počítání, lokalizace, rotace



Problémy neuronových sítí – skryté vzory

- [Video](#) – neviditelný pattern



x

“panda”

57.7% confidence

$+ .007 \times$



$\text{sign}(\nabla_x J(\theta, x, y))$

“nematode”

8.2% confidence

$=$



$x +$

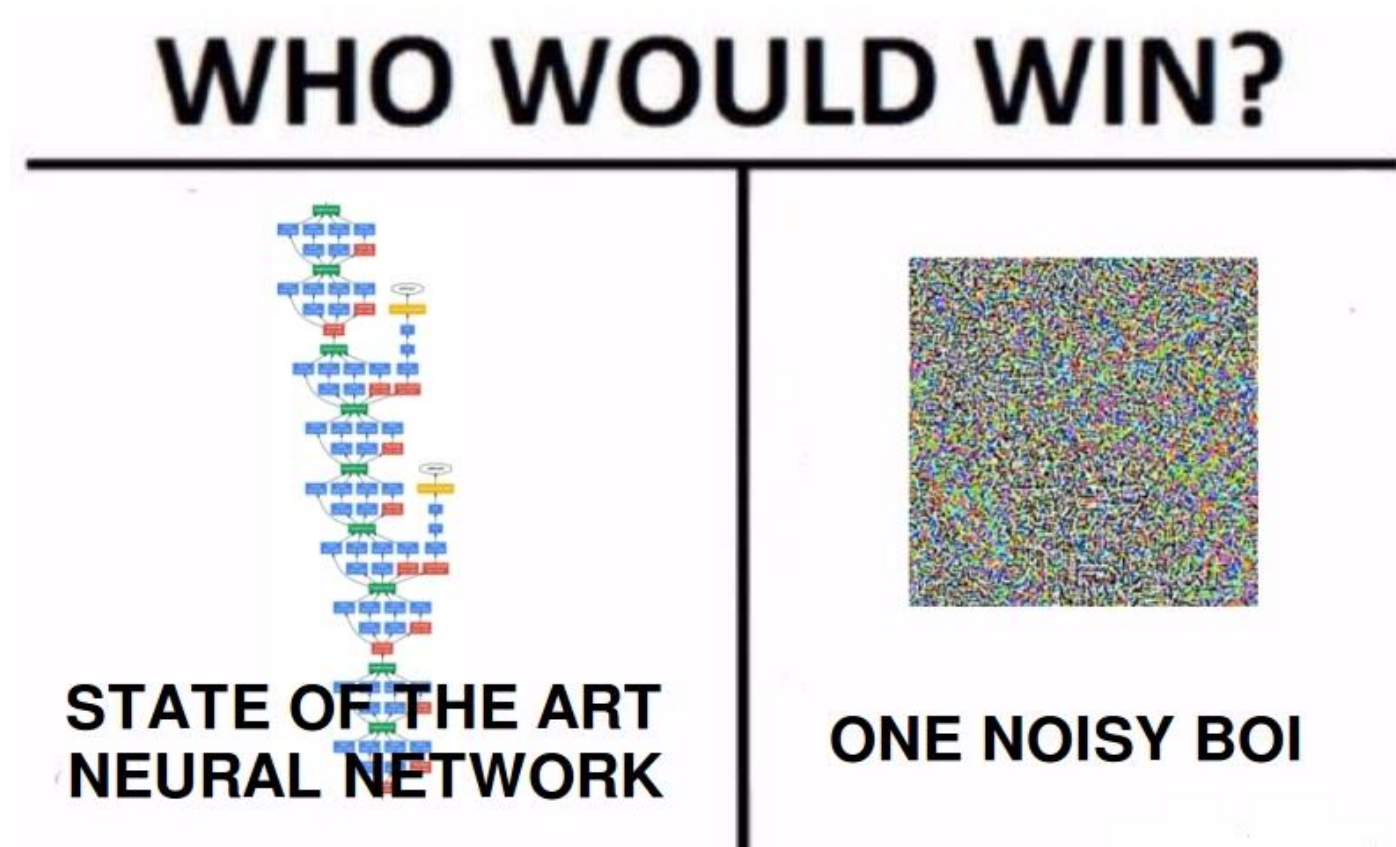
$\epsilon \text{sign}(\nabla_x J(\theta, x, y))$

“gibbon”

99.3 % confidence

Problémy neuronových sítí – skryté vzory

- [Odkaz na blog post](#)



Datasets

2007

Pascal VOC

- 20 tříd
- 11K trénovacích obrázků
- 27K trénovacích objektů

Používán jako standard, nyní již pouze k rychlému otestování nového algoritmu

2013

ImageNet ILSVRC

- 200 tříd
- 476K trénovacích obrázků
- 534K trénovacích objektů

Pascal VOC na steroidech

2015

MS COCO

- 80 Classes
 - 200K trénovacích obrázků
 - 1.5M trénovacích objektů
- Více kategorií v jednotlivých obrazech. Zaměřený spíše na malé objekty

Metriky klasifikace – matice záměn

- Matice záměn sama o sobě není měřítkem přesnosti, ale téměř všechny metriky jsou z ní odvozeny
- Mějme nyní dvě třídy studentů:
 - 0 = nedostali zápočet z BI-SVZ
 - 1 = dostali zápočet z BI-SVZ

		Actual	
		Positives(1)	Negatives(0)
Predicted	Positives(1)	TP	FP
	Negatives(0)	FN	TN

Metriky klasifikace – matice záměn

- **True Positives (TP):** data jsou predikována do třídy 1 a správně patří do třídy 1
- **True Negatives (TN):** data jsou predikována do třídy 0 a správně patří do třídy 0
- **False Positives (FP):** data jsou predikována do třídy 1, ale správně patří do třídy 0
- **False Negatives (FN):** data jsou predikována do třídy 0, ale správně patří do třídy 1

		Actual	
		Positives(1)	Negatives(0)
Predicted	Positives(1)	TP	FP
	Negatives(0)	FN	TN

Metriky klasifikace – matice záměn

- V ideálním případě bychom si přáli, aby náš model měl 0 False Positives a 0 False Negatives
- To bohužel v reálném světě není možné, tedy vždy hledáme kompromis mezi False Positives a False negatives

		Actual	
		Positives(1)	Negatives(0)
Predicted	Positives(1)	TP	FP
	Negatives(0)	FN	TN

Metriky klasifikace – celková správnost (accuracy)

- Poměr správných predikcí modelu proti všem celkovým predikcím
- Slouží jako dobrý ukazatel, když jsou cílové třídy téměř vyvážené

		Actual	
		Positives(1)	Negatives(0)
Predicted	Positives(1)	TP	FP
	Negatives(0)	FN	TN

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}}$$

Metriky klasifikace – přesnost (precision)

- Říká, kolik procent studentů, u kterých jsme predikovali, že dostali zápočet z BI-SVZ, opravdu dostali zápočet z BI-SVZ

		Actual	
		Positives(1)	Negatives(0)
Predicted	Positives(1)	TP	FP
	Negatives(0)	FN	TN

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Metriky klasifikace – senzitivita (sensitivity, recall)

- Říká, kolik procent studentů, kteří skutečně dostali zápočet z BI-SVZ a byli predikováni jako získal zápočet BI-SVZ

		Actual	
		Positives(1)	Negatives(0)
Predicted	Positives(1)	TP	FP
	Negatives(0)	FN	TN

Recall = $\frac{TP}{TP + FN}$

Metriky klasifikace – specificita (specificity)

- Říká, kolik procent studentů, kteří skutečně **NE**dostali zápočet z BI-SVZ a byli predikováni jako **NE**získal zápočet BI-SVZ

		Actual	
		Positives(1)	Negatives(0)
Predicted	Positives(1)	TP	FP
	Negatives(0)	FN	TN

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

Metriky klasifikace – ostatní

- [Odkaz](#)

		True condition			
Total population		Condition positive	Condition negative	$Prevalence = \frac{\Sigma \text{Condition positive}}{\Sigma \text{Total population}}$	$Accuracy (ACC) = \frac{\Sigma \text{True positive} + \Sigma \text{True negative}}{\Sigma \text{Total population}}$
Predicted condition	Predicted condition positive	True positive , Power	False positive , Type I error	Positive predictive value (PPV), Precision = $\frac{\Sigma \text{True positive}}{\Sigma \text{Predicted condition positive}}$	False discovery rate (FDR) = $\frac{\Sigma \text{False positive}}{\Sigma \text{Predicted condition positive}}$
	Predicted condition negative	False negative , Type II error	True negative	False omission rate (FOR) = $\frac{\Sigma \text{False negative}}{\Sigma \text{Predicted condition negative}}$	Negative predictive value (NPV) = $\frac{\Sigma \text{True negative}}{\Sigma \text{Predicted condition negative}}$
		True positive rate (TPR), Recall, Sensitivity, probability of detection $= \frac{\Sigma \text{True positive}}{\Sigma \text{Condition positive}}$	False positive rate (FPR), Fall-out, probability of false alarm $= \frac{\Sigma \text{False positive}}{\Sigma \text{Condition negative}}$	Positive likelihood ratio (LR+) = $\frac{TPR}{FPR}$	Diagnostic odds ratio (DOR) $= \frac{LR+}{LR-}$
		False negative rate (FNR), Miss rate $= \frac{\Sigma \text{False negative}}{\Sigma \text{Condition positive}}$	Specificity (SPC), Selectivity, True negative rate (TNR) $= \frac{\Sigma \text{True negative}}{\Sigma \text{Condition negative}}$	Negative likelihood ratio (LR-) = $\frac{FNR}{TNR}$	
				$F_1 \text{ score} = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}}$	

sensitivity, recall, hit rate, or true positive rate (TPR)

$$\text{TPR} = \frac{\text{TP}}{P} = \frac{\text{TP}}{\text{TP} + \text{FN}} = 1 - \text{FNR}$$

specificity, selectivity or true negative rate (TNR)

$$\text{TNR} = \frac{\text{TN}}{N} = \frac{\text{TN}}{\text{TN} + \text{FP}} = 1 - \text{FPR}$$

precision or positive predictive value (PPV)

$$\text{PPV} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

negative predictive value (NPV)

$$\text{NPV} = \frac{\text{TN}}{\text{TN} + \text{FN}}$$

miss rate or false negative rate (FNR)

$$\text{FNR} = \frac{\text{FN}}{P} = \frac{\text{FN}}{\text{FN} + \text{TP}} = 1 - \text{TPR}$$

fall-out or false positive rate (FPR)

$$\text{FPR} = \frac{\text{FP}}{N} = \frac{\text{FP}}{\text{FP} + \text{TN}} = 1 - \text{TNR}$$

false discovery rate (FDR)

$$\text{FDR} = \frac{\text{FP}}{\text{FP} + \text{TP}} = 1 - \text{PPV}$$

false omission rate (FOR)

$$\text{FOR} = \frac{\text{FN}}{\text{FN} + \text{TN}} = 1 - \text{NPV}$$

accuracy (ACC)

$$\text{ACC} = \frac{\text{TP} + \text{TN}}{P + N} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

F₁ score

is the harmonic mean of precision and sensitivity

$$F_1 = 2 \cdot \frac{\text{PPV} \cdot \text{TPR}}{\text{PPV} + \text{TPR}} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}$$

Matthews correlation coefficient (MCC)

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}}$$

Informedness or Bookmaker Informedness (BM)


$$\text{BM} = \text{TPR} + \text{TNR} - 1$$

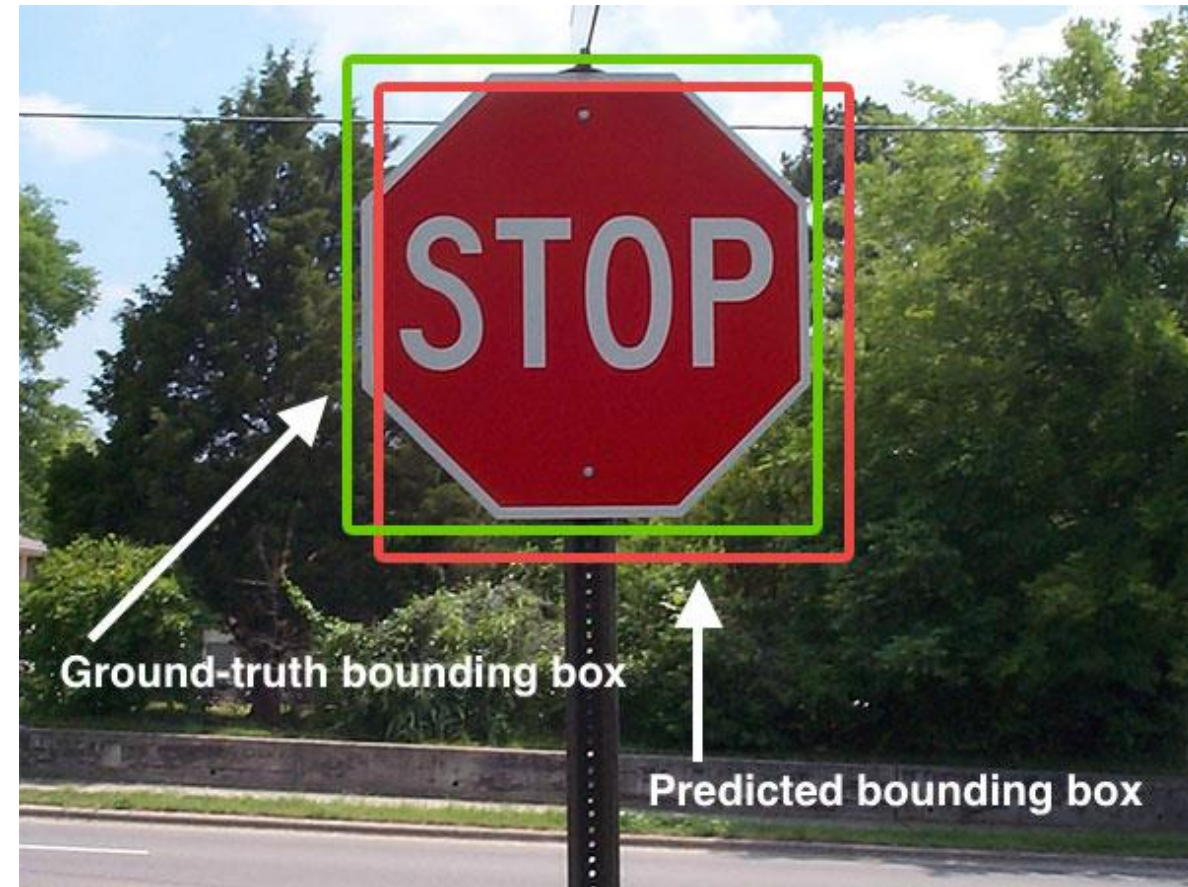
Markedness (MK)

$$\text{MK} = \text{PPV} + \text{NPV} - 1$$

Metrika lokalizace – Intersection over Union (IoU)

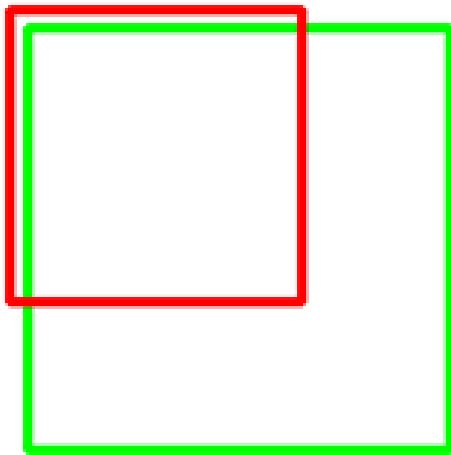
- Vzdálenostní metrika popisující přesnost skutečné lokalizace proti predikci

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$




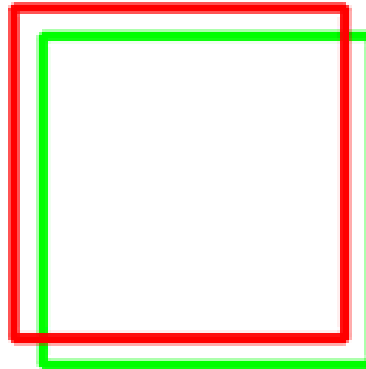
Metrika lokalizace – Intersection over Union (IoU)

IoU: 0.4034



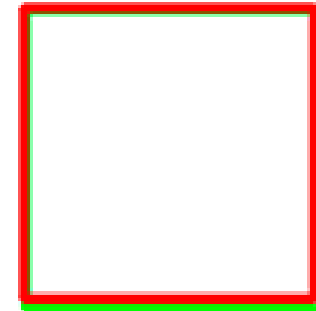
Poor

IoU: 0.7330



Good

IoU: 0.9264



Excellent

Zdroje

- <https://luozm.github.io/cv-tasks>
- <https://www.slideshare.net/Brodmann17/introduction-to-object-detection>
- <https://www.learnopencv.com/image-recognition-and-object-detection-part1/>
- <https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/>
- <https://medium.com/greyatom/performance-metrics-for-classification-problems-in-machine-learning-part-i-b085d432082b>