

Praktikum zo strojového učenia a umelej inteligencie na vizuálnych dátach

Andrej Lúčny

Katedra aplikovanej informatiky FMFI UK

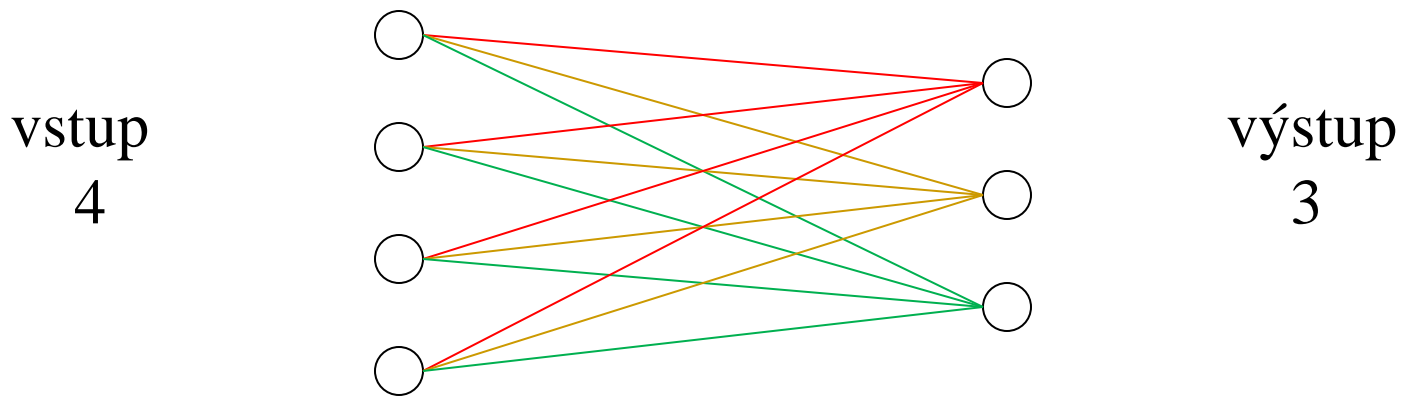
lucny@fmph.uniba.sk

http://dai.fmph.uniba.sk/w/Andrej_Lucny

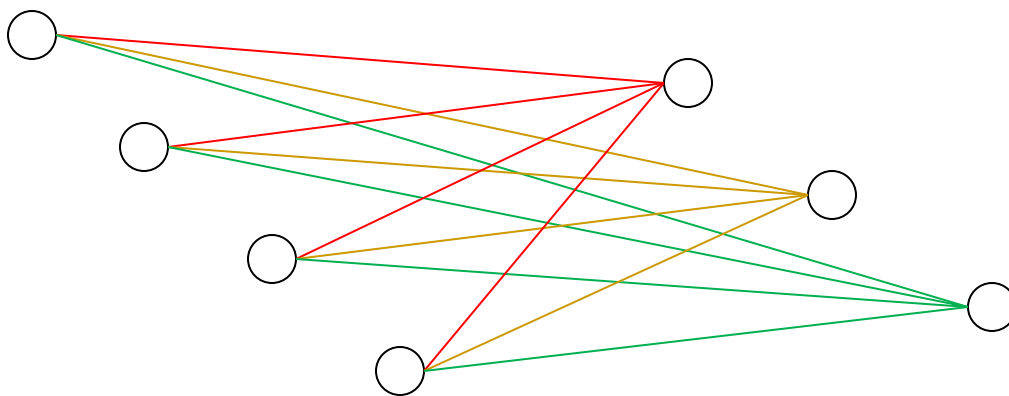
www.agentspace.org/praktikum

plne prepojenú vrstvu je možné implementovať pomocou bloku konvolučných vrstiev s kernelom 1x1.

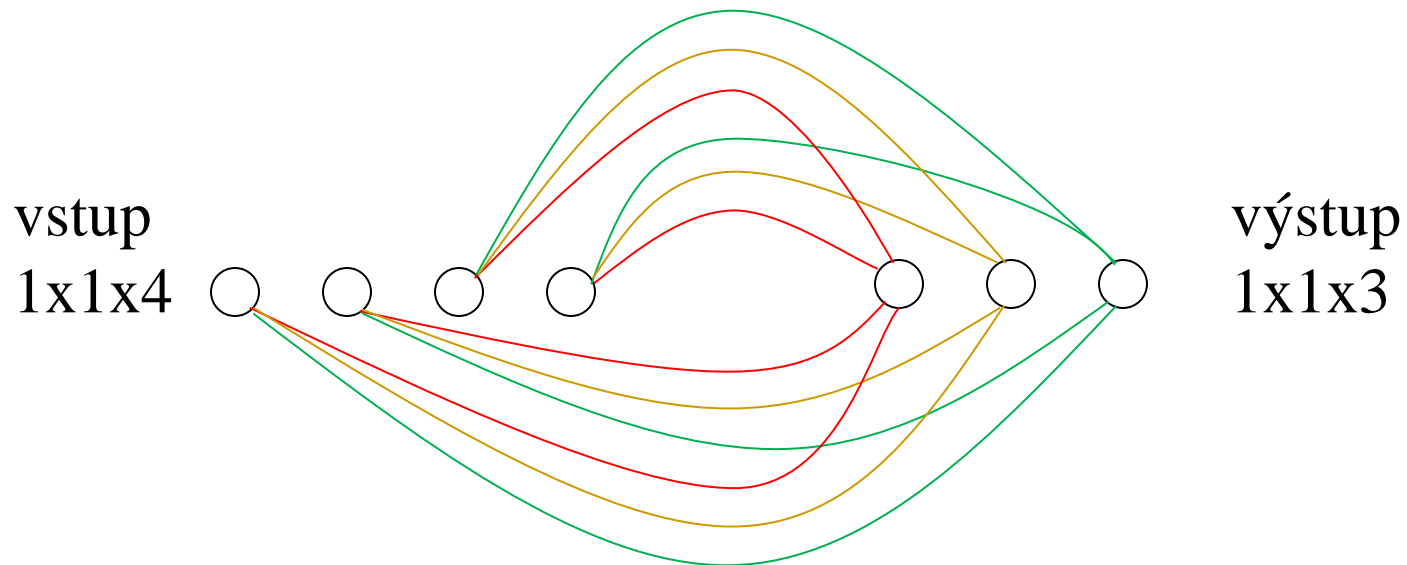
počet neurónov v plne prepojenej vrstve je v takom prípade identický s počtom konvolučných vrstiev v bloku



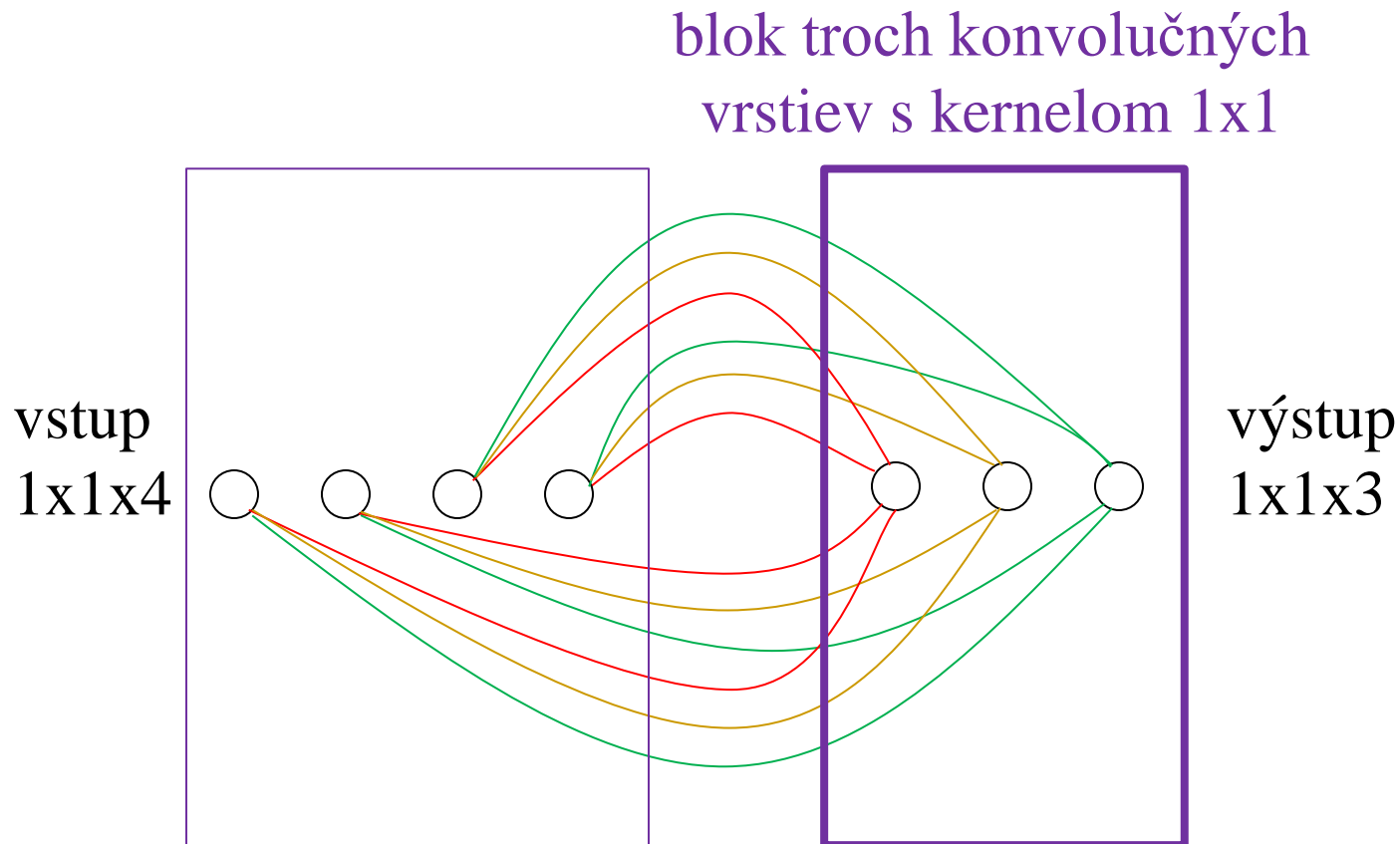
Plne prepojenú vrstvu môžeme „poľahnúť“



Miesto vstupné tenzora s dimenzou n máme potom vstup s dimenziou $1 \times 1 \times n$

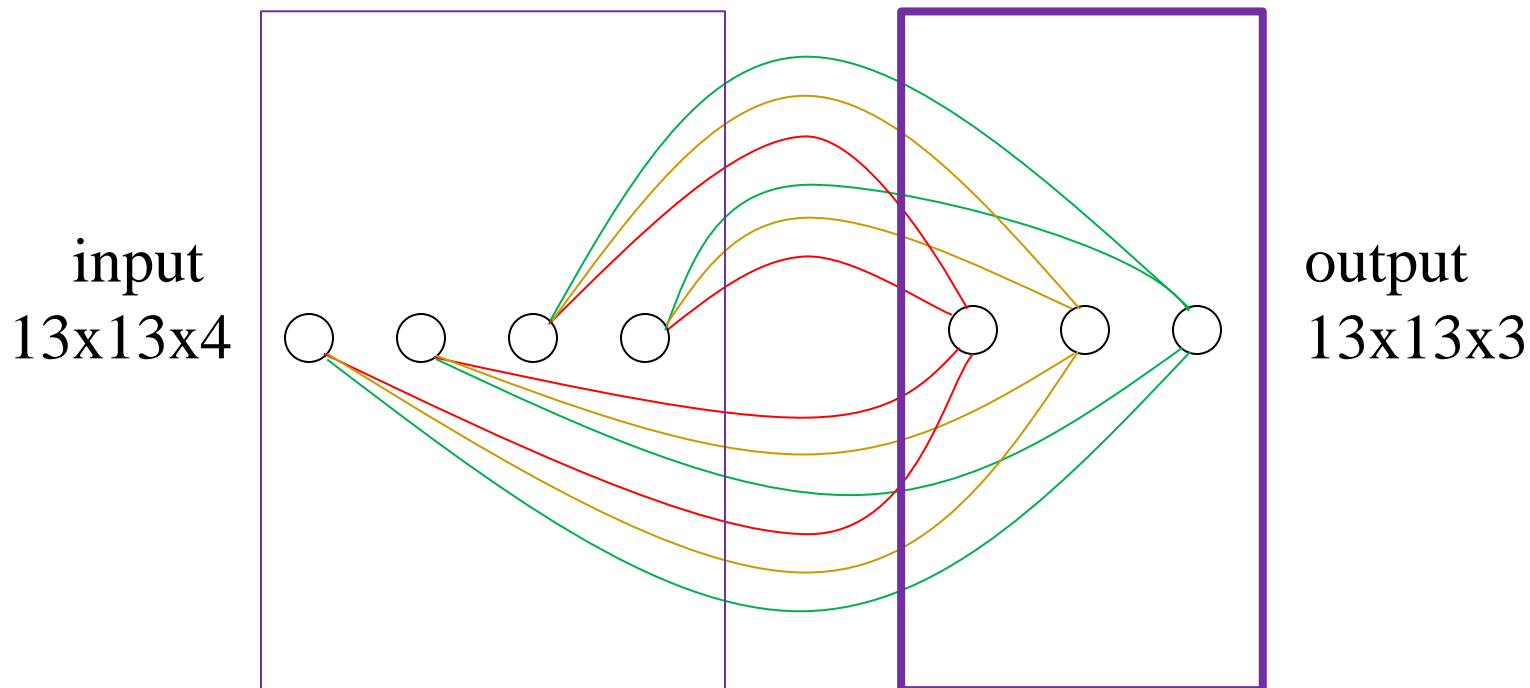


Keď do takého bloku konvolučných vrstiev pustíme ako vstup „obrázok“ s rozlíšením 1×1 a počtom kanálov rovným dimenzii pôvodného vstupu, vykoná činnosť jednej plne prepojenej vrstvy

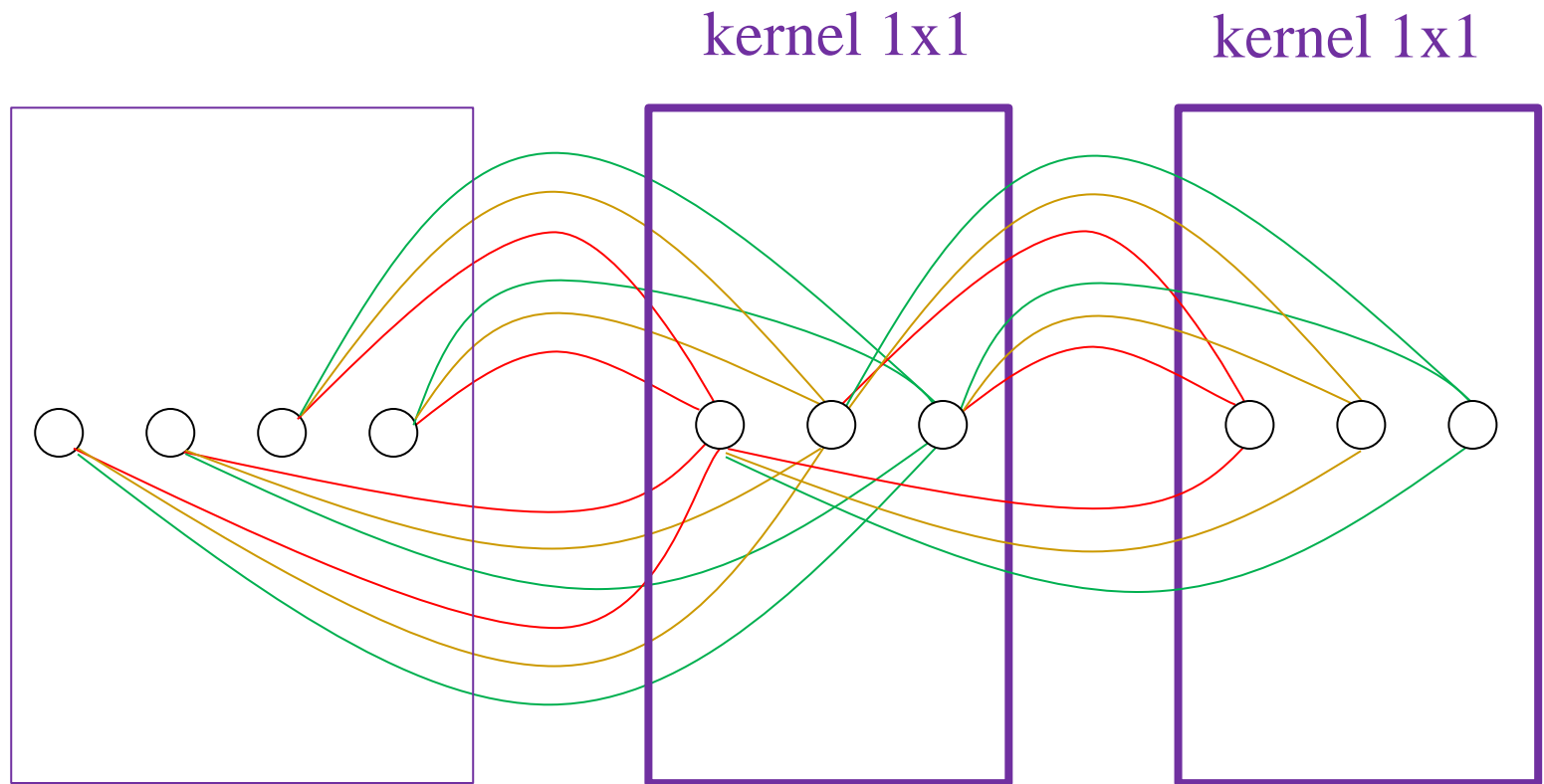


Do takého bloku konvolučných vrstiev nakrmíme vstup s rozlíšením napr. 13×13 jeho činnosť bude zodpovedať paralelnému spusteniu 169-tich plne prepojených vrstiev s rovnakými váhami a posunutiami

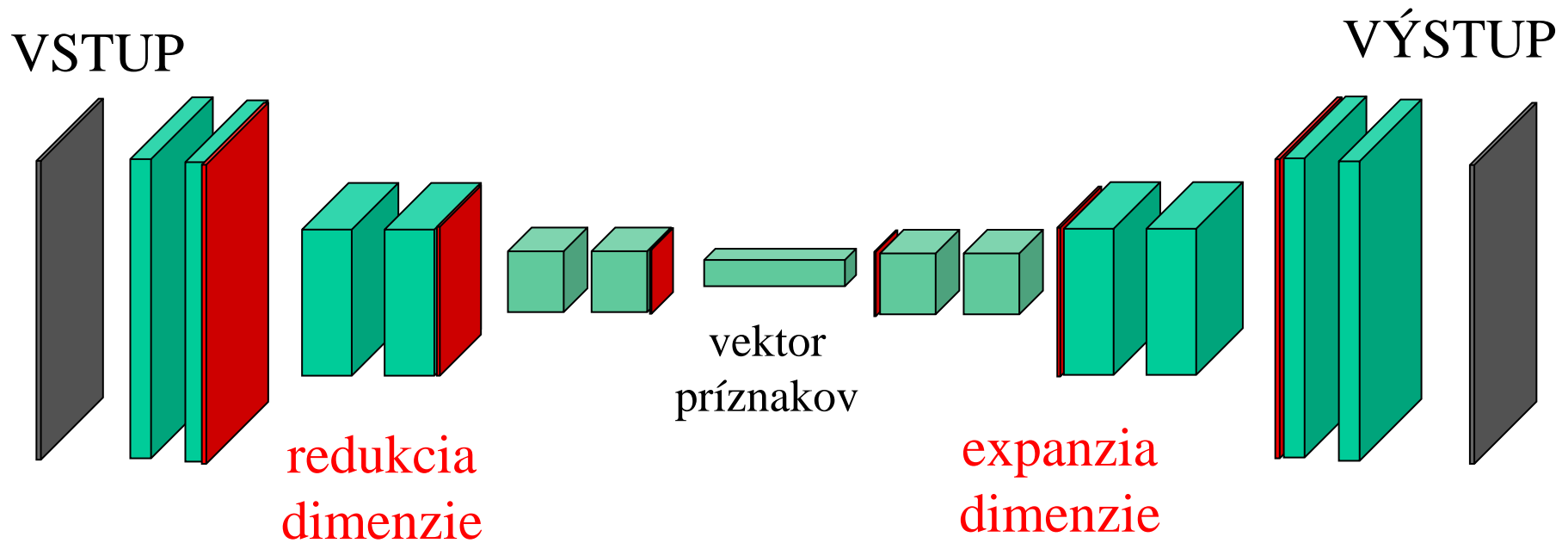
blok troch konvolučných
vrstiev s kernelom 1×1



A keď si takých blokov pustíme viac (dva, tri) implementujú nám napr. 169 paralelne bežiacich perceptrónov zdieľajúcich váhy a posunutia



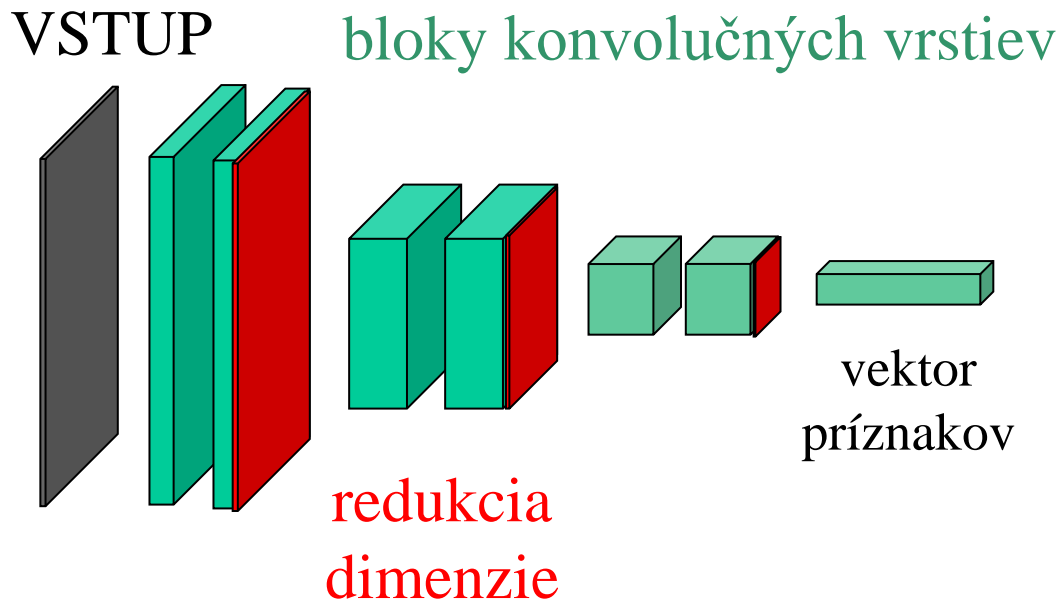
Konvolučný autokóder



bloky konvolučných vrstiev

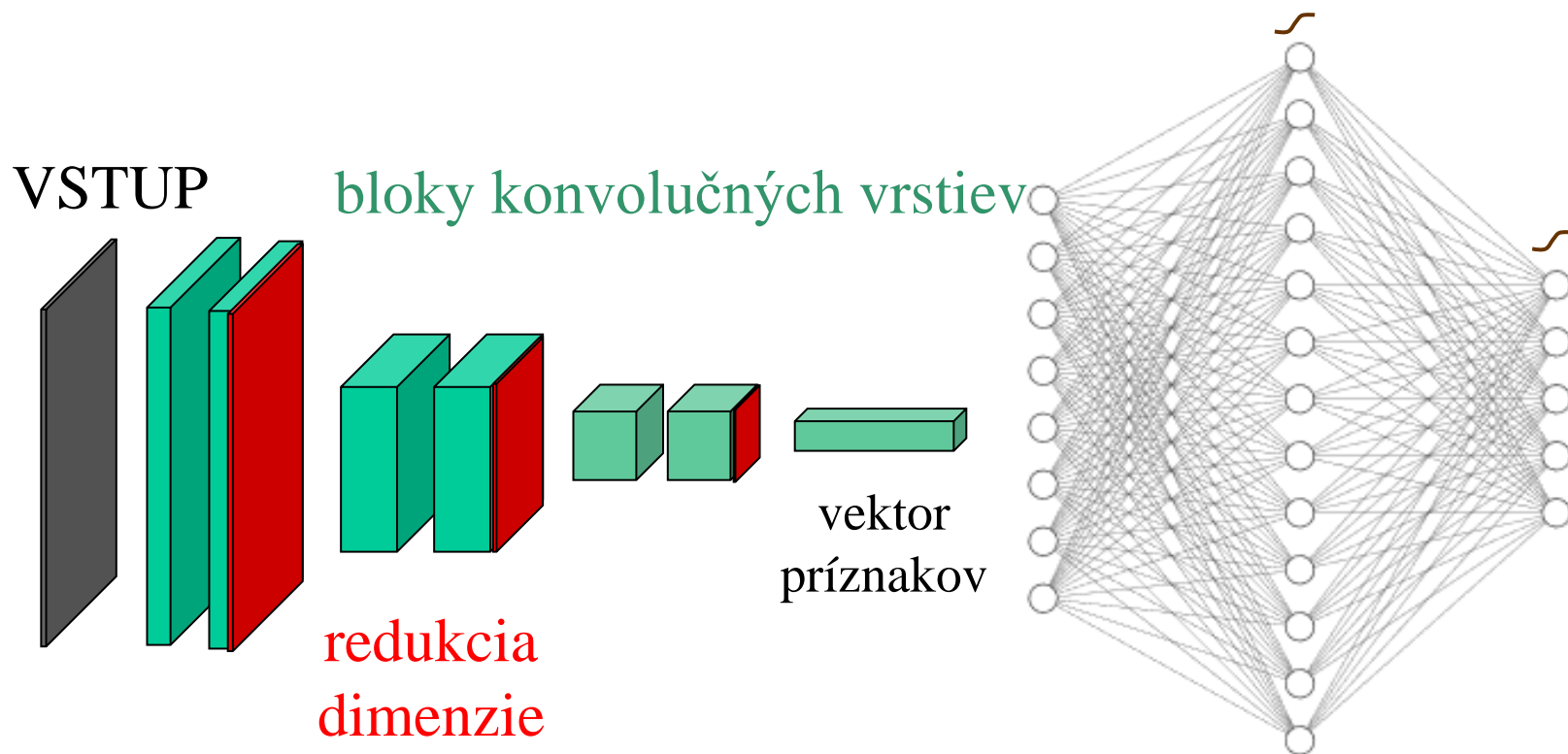
chceme, aby:
 $VSTUP = VÝSTUP$

Kóder



Kóder redukuje dimenziu dát,
premieňa obraz na príznaky

Hlboký klasifikátor / regresor



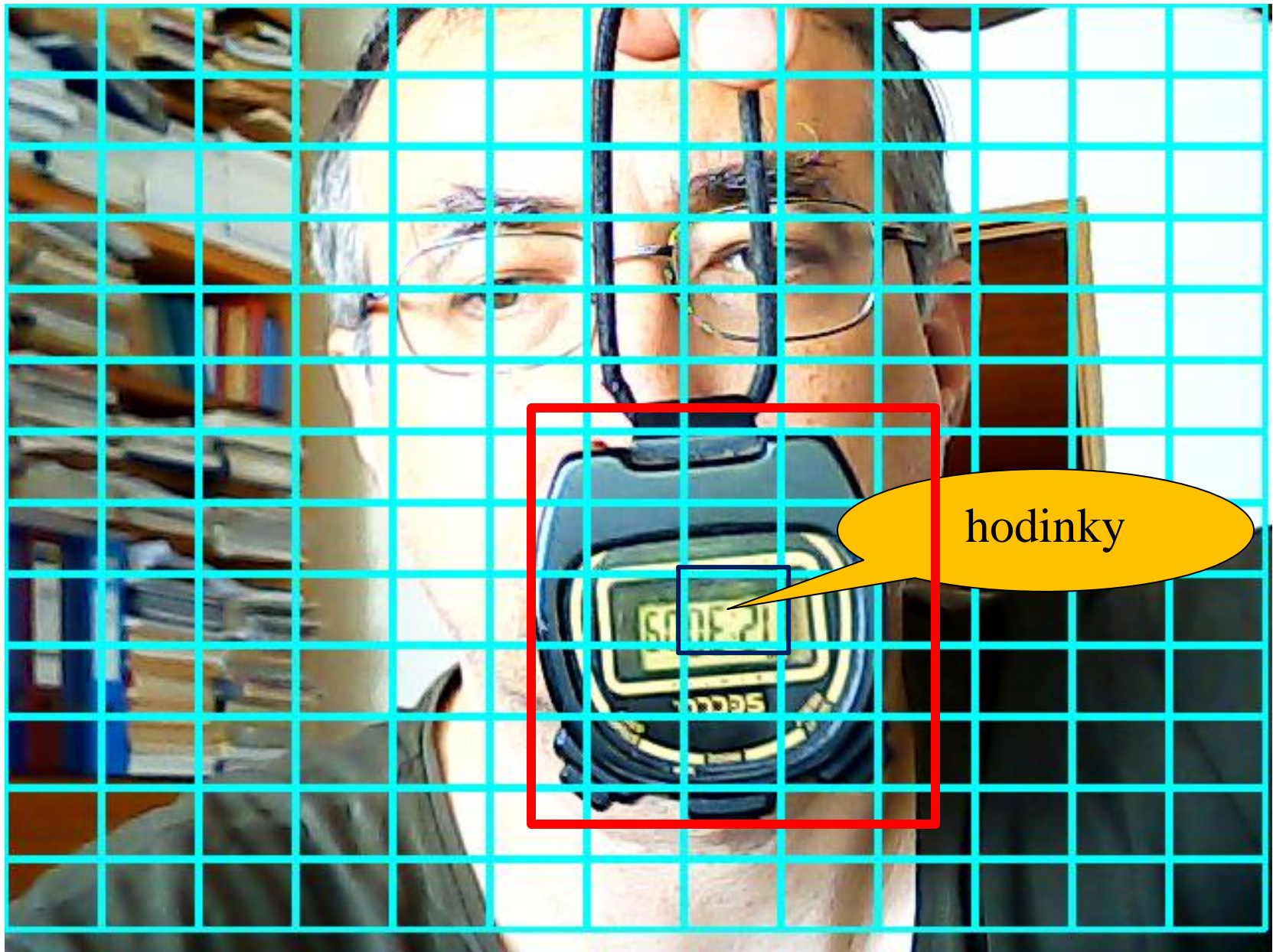
Následne môžeme príznaky spracovať perceptronom, implementujúc klasifikátor alebo regresor

YOLO Detektor (You Look Only Once)

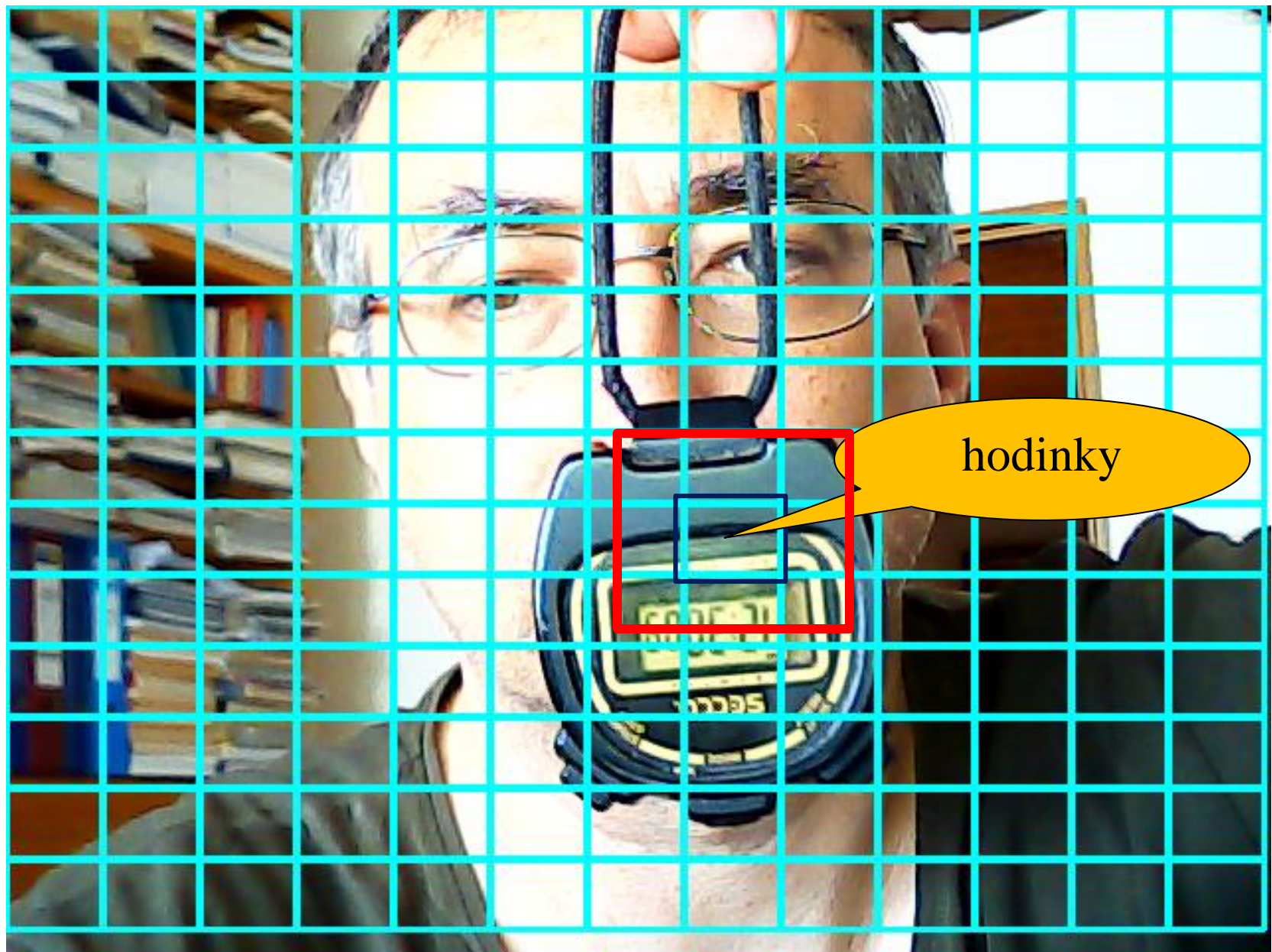
- obraz pokryjeme neprekrývajúcimi sa regiónmi
- nad každým regiónom vykonáme klasifikáciu a regresiu a zosumarizujeme výsledky



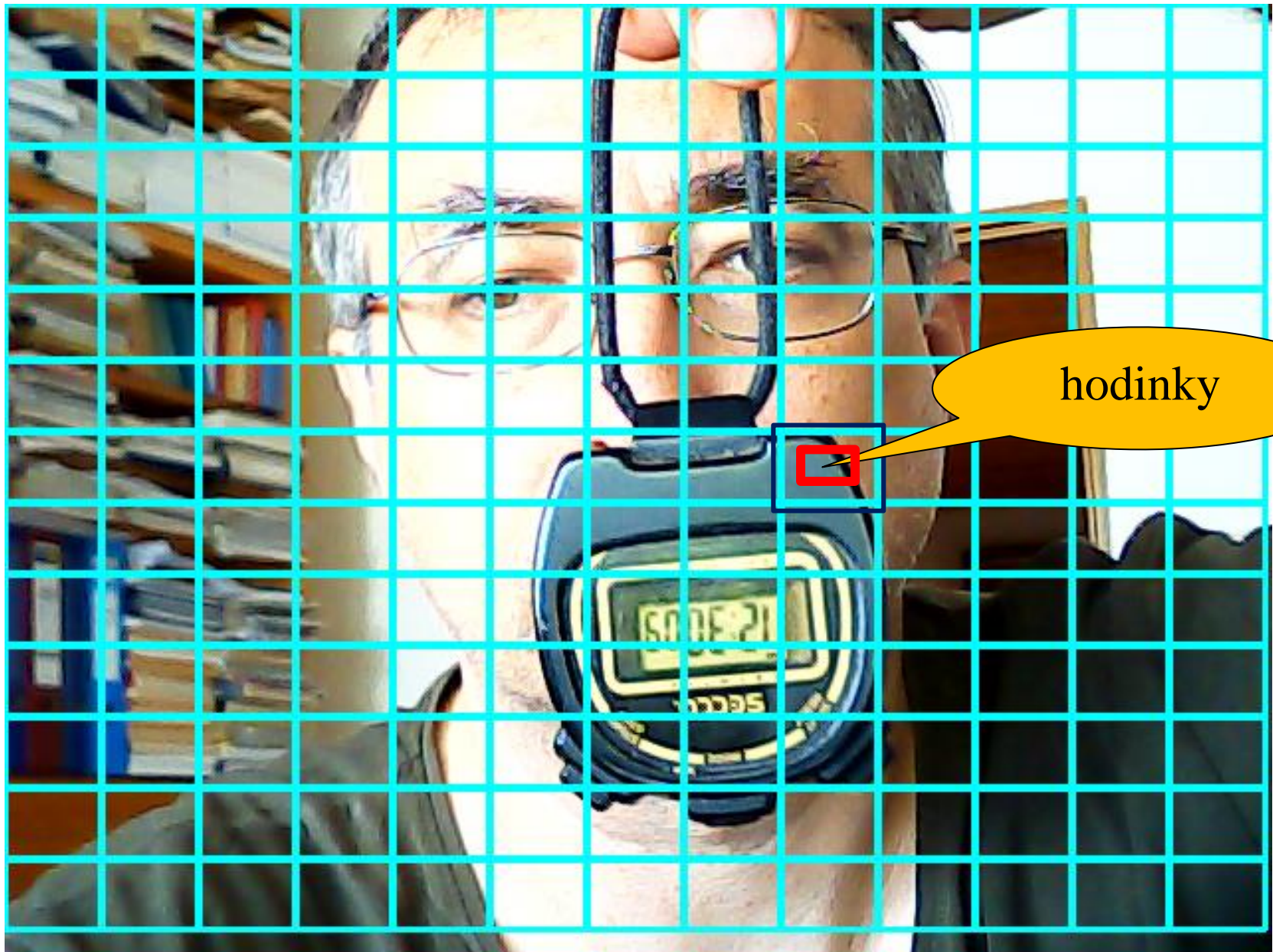
- **Klasifikátor** dáva pravdepodobnosť, že región patrí k detekovanému objektu
- **Regressor** dáva obdĺžnik obsahujúci objekt, ktorého je región súčasťou



hodinky



hodinky



hodinky

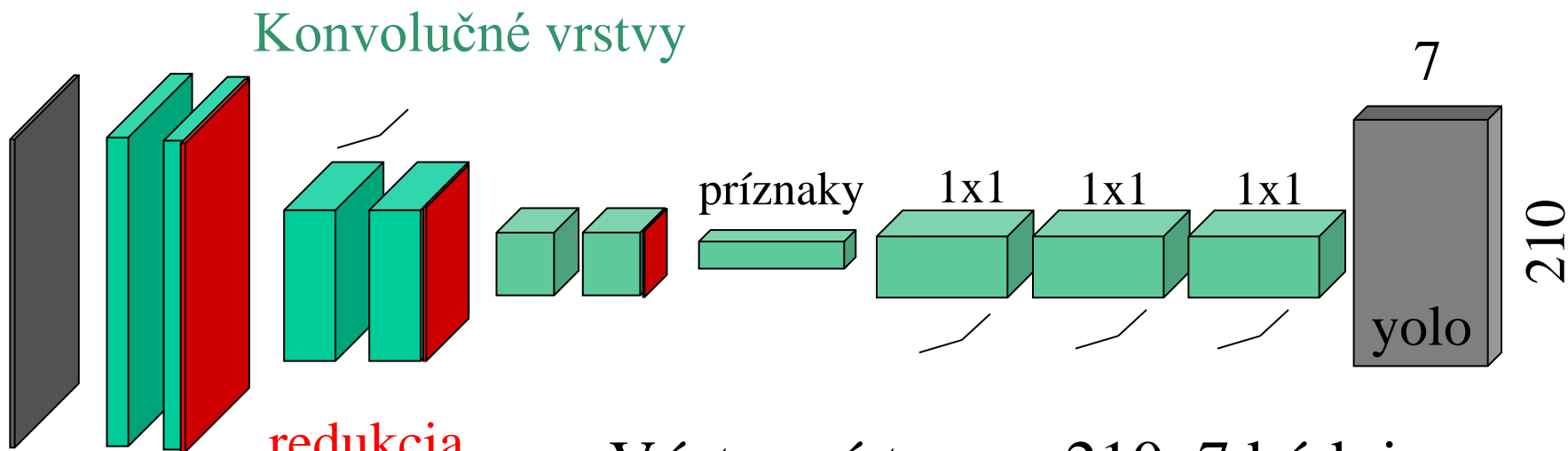


tvár



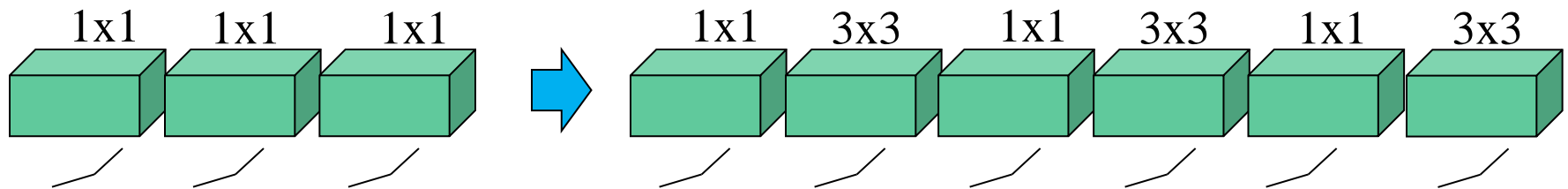
tvár

YOLO v1

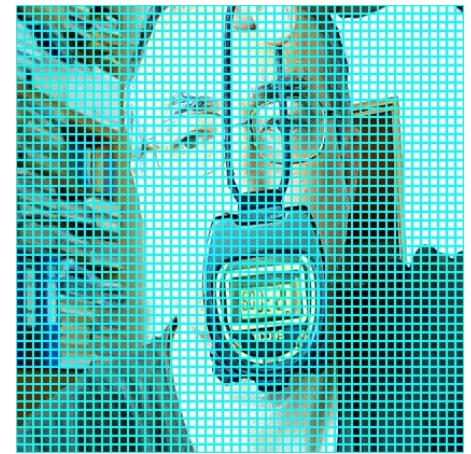
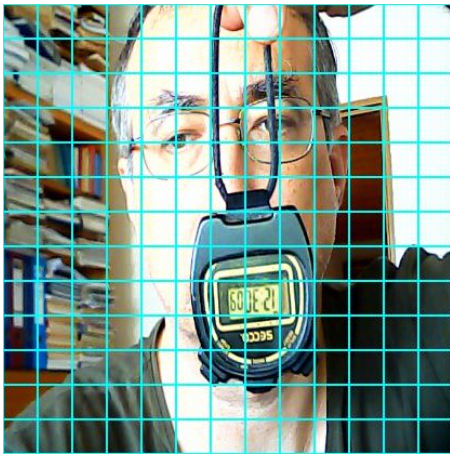


Výstupný tenzor 210×7 kóduje maximálne 210 detekcií, pričom každá obsahuje: relevantnosť, kategóriu, spoľahlivosť, x , y , w , a h (sumarizáciu vykonáva špeciálny stavebný prvok zvaný yolo)

- Ak máme kúsok hodínok v určitom regióne, zvyšuje ho pravdepodobnosť, že je aj v susedných regiónoch.
- Ako by sme mohli dosiahnuť, že paralelne pustené perceptrony budú kooperovať?
- Riešenie: Kernely 1×1 preložíme kernelmi 3×3

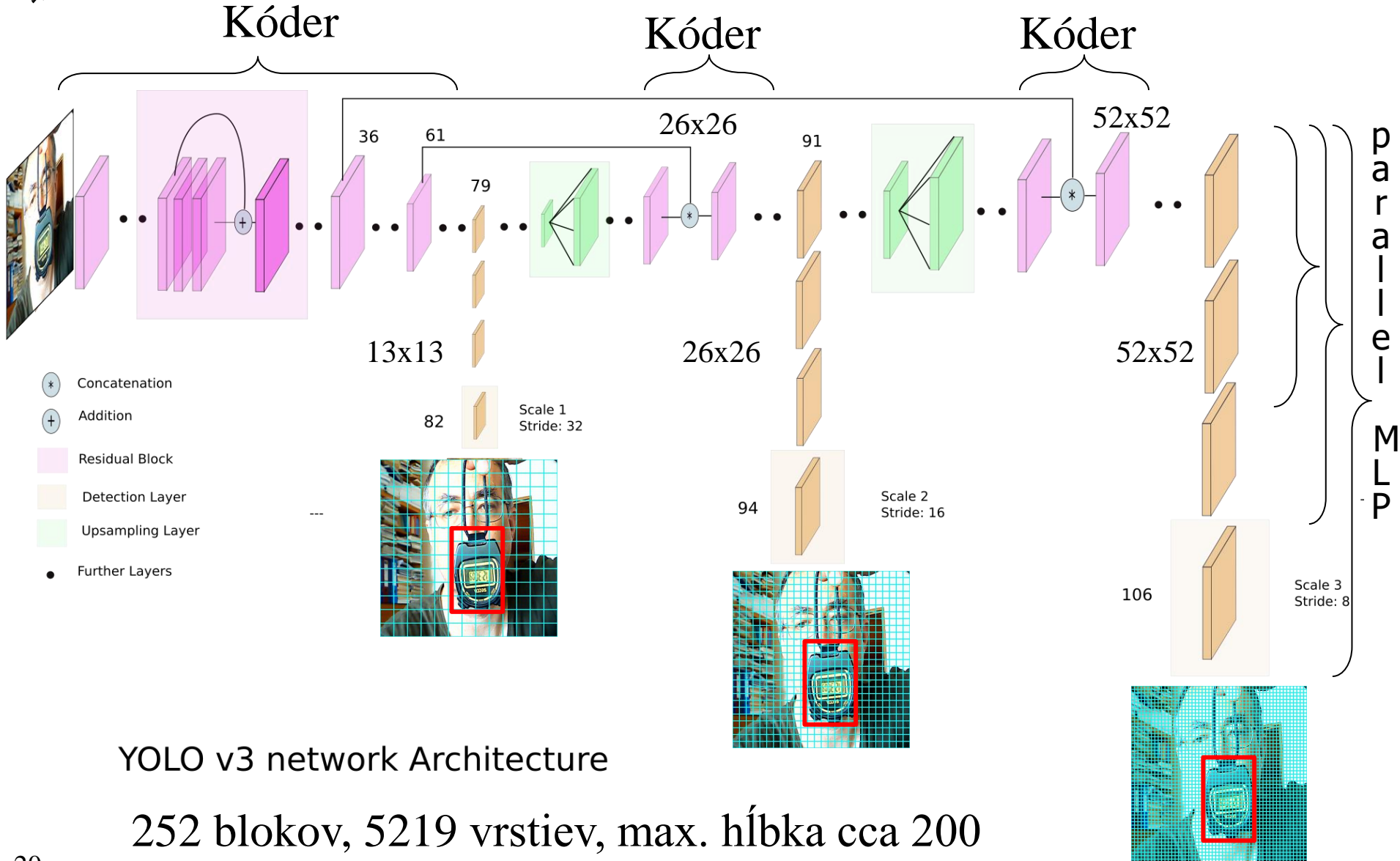


- Ako by sme mohli zohľadniť rôznu veľkosť objektov?
- Pustíme celé spracovanie paralelne viac krát pre 3×13 , 26×26 , a 52×52 regiónov



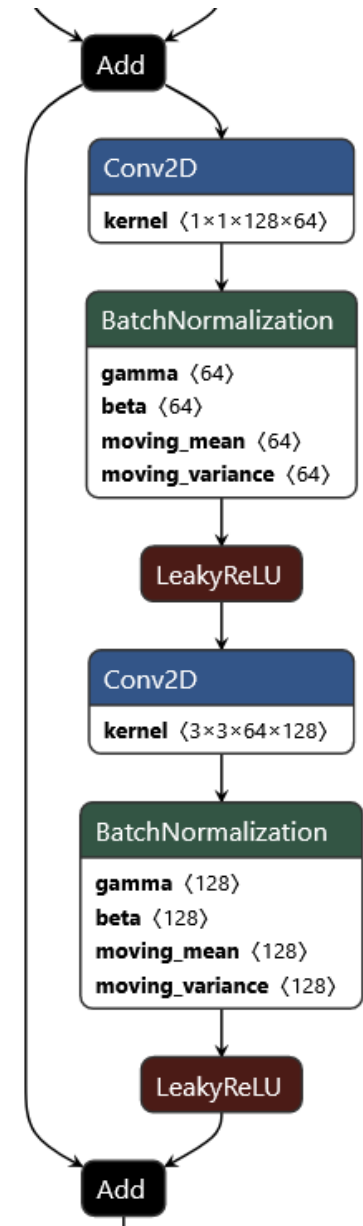
- V takom prípade aj menej detailné spracovanie môže radiť detailnejšiemu

YOLO v3 [Redmon, Farhadi 2018]

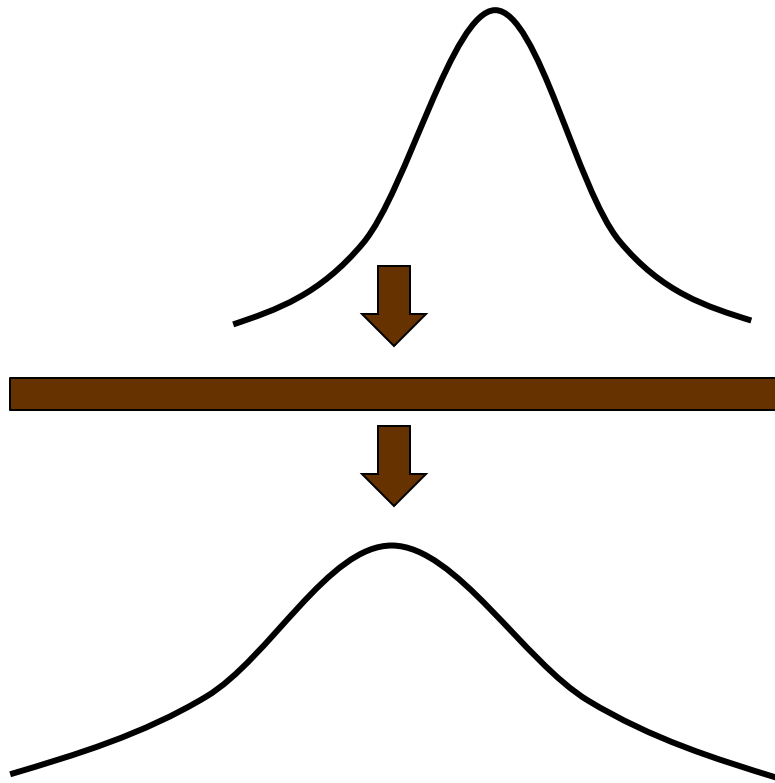


Trénovanie hlbokých sietí

- Hlboké siete dokážu mať sofistikovanú architektúru, ale trénujú sa oveľa ťažšie (problém miznúceho gradientu pri spätnom šírení)
- Preto do takýchto sietí vkladáme stavebné prvky, ktoré trénovanie podporujú. YOLO v3 využíva reziduálne spojenia



Normalizácia dávky



- Aby bolo trénovanie dostatočne rýchle, hlboké siete používajú sigmoidálne aktivácie len na výstupe a vnútri používajú ReLU
- ReLU ale nie je sigmoidálna a nedá sa s ňou implementovať aproximátor
- Preto pred (alebo za) ReLU vkladáme normalizáciu dávky

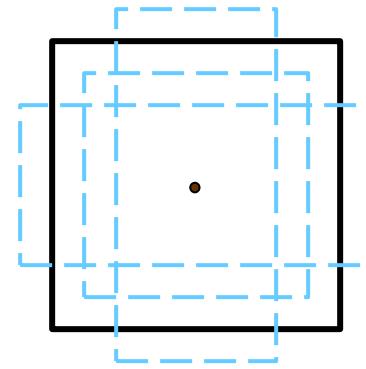
YOLO v3 schéma



YOLO v3 - výstup

- $13 \times 13 \times 255$, $26 \times 26 \times 255$, $52 \times 52 \times 255$
- Každý perceptron dáva tri detekcie,
 255 hodnôt = 3×85
- Každá detekcia dáva 85 hodnôt = $4+1+80$
obsahujúcich:
 - 4 ... x, y, w, h
 - 1 ... spoľahlivosť
 - 80 ... pravdepodobnosť príslušnosti
k jednej z 80 kategórii

Ukotvenia



- Každá detekcia vyjadruje x, y, w , a h relatívne k určitému ukotveniu. YOLO v3 využíva tri ukotvenia pre každú z troch veľkostí objektov (dohromady deväť), každé s rôznym pomerom strán
- Ukotvenia v YOLO v3:
[116x90, 156x198, 373x326] (výstup 52x52)
[30x61, 62x45, 59x119] (výstup 26x26)
[10x13, 16x30, 33x23] (výstup 13x13)

Transférové učenie

YOLO v3 poskytuje predtrénovaný model pre 80 kategórii z dátovej vzorky COCO:

person, bicycle, car, motorbike, aeroplane, bus, train, truck, boat, traffic light, fire hydrant, stop sign, parking meter, bench, bird, cat, dog, horse, sheep, cow, elephant, bear, zebra, giraffe, backpack, umbrella, handbag, tie, suitcase, frisbee, skis, snowboard, sports, ball, kite, baseball bat, baseball glove, skateboard, surfboard, tennis racket, bottle, wine glass, cup, fork, knife, spoon, bowl, banana, apple, sandwich, orange, broccoli, carrot, hot dog, pizza, donut, cake, chair, sofa, pottedplant, bed, diningtable, toilet, tvmonitor, laptop, mouse, remote, keyboard, cell phone, microwave, oven, toaster, sink, refrigerator, book, clock, vase, scissors, teddy bear, hair drier, toothbrush

Pri trénovaní pre vlastné dátové vzorky, nezačínáme s náhodnými parametrami, ale parametrami predtrénovaného modelu.

Trénovanie YOLO v3

<https://youtu.be/CPGR26Fhcro>



Spracovanie videa

