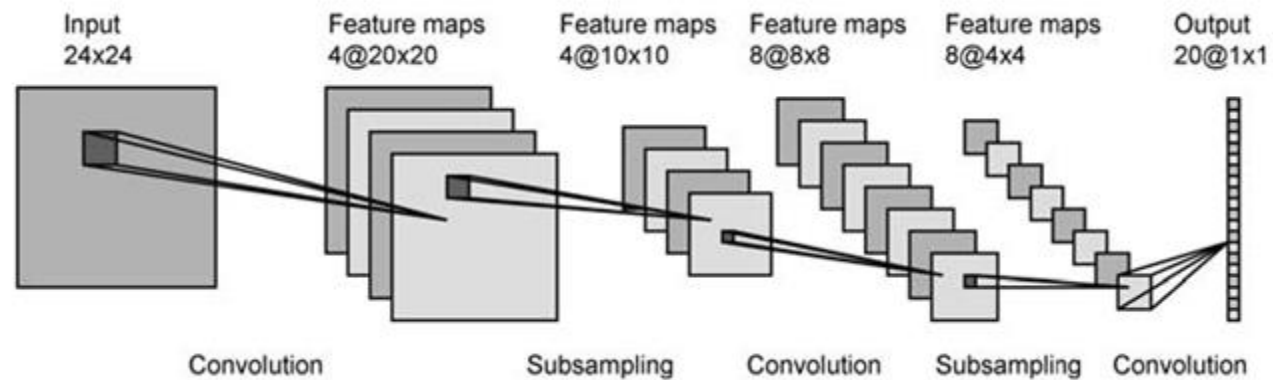


Kapitola 11

Hlboké konvolučné siete

Konvolučné (deep) siete

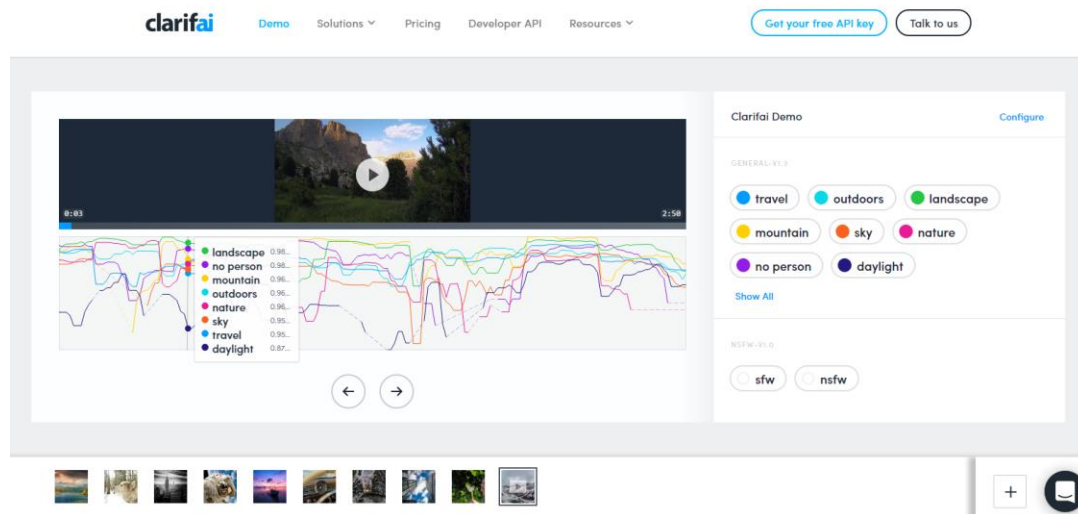


Obsah

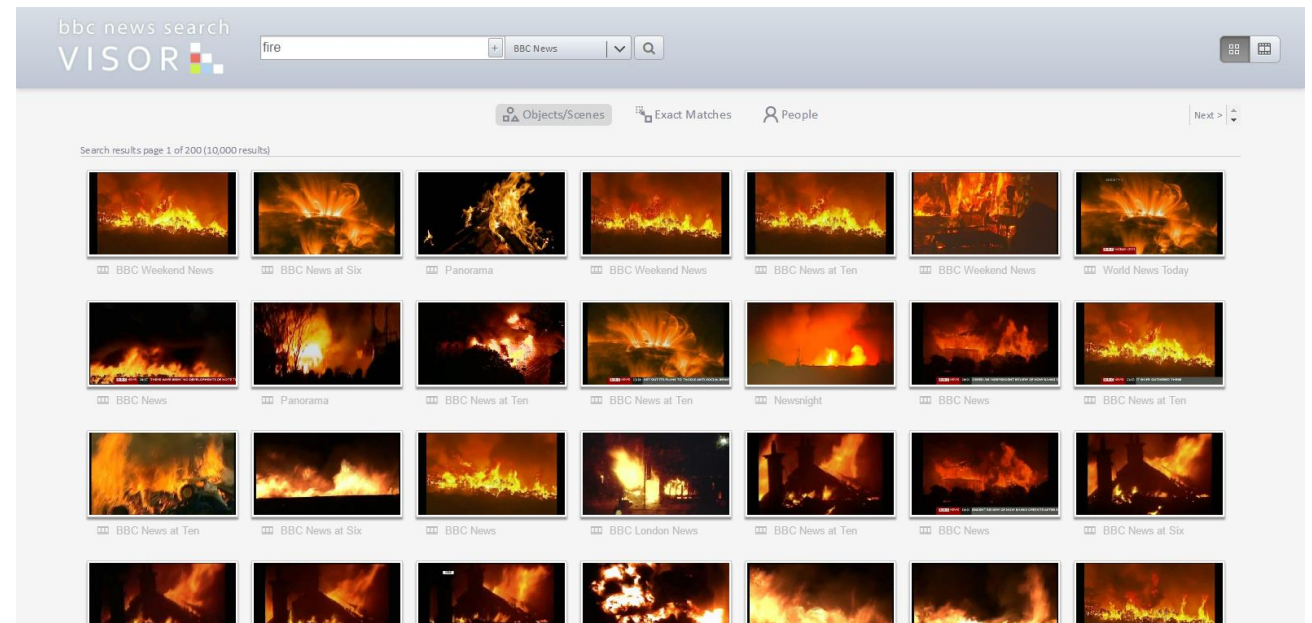
- Načrtnutie problému
- „Staré časy“
- Neurónové siete
- Konvolučné siete
- Ukážka obsahu jednotlivých vrstiev
- Zhrnutie

Demo

- <http://www.clarifai.com/>



- <http://www.robots.ox.ac.uk/~vgg/research/on-the-fly/>



Demo – výzva spracovať veľké dáta

Trvanie BBC nahrávok	# framov	# Snímok	Veľkosť dát (po úprave)	Počet detegovaných tvárí
3-40 K hodín	10-150 M	3-35 M	1-10 TB	5-20 M

Naučiť sa objekty, osoby za chodu aplikácie

- Vytvoriť model pre nové „queries“ za chodu

Rýchla odozva

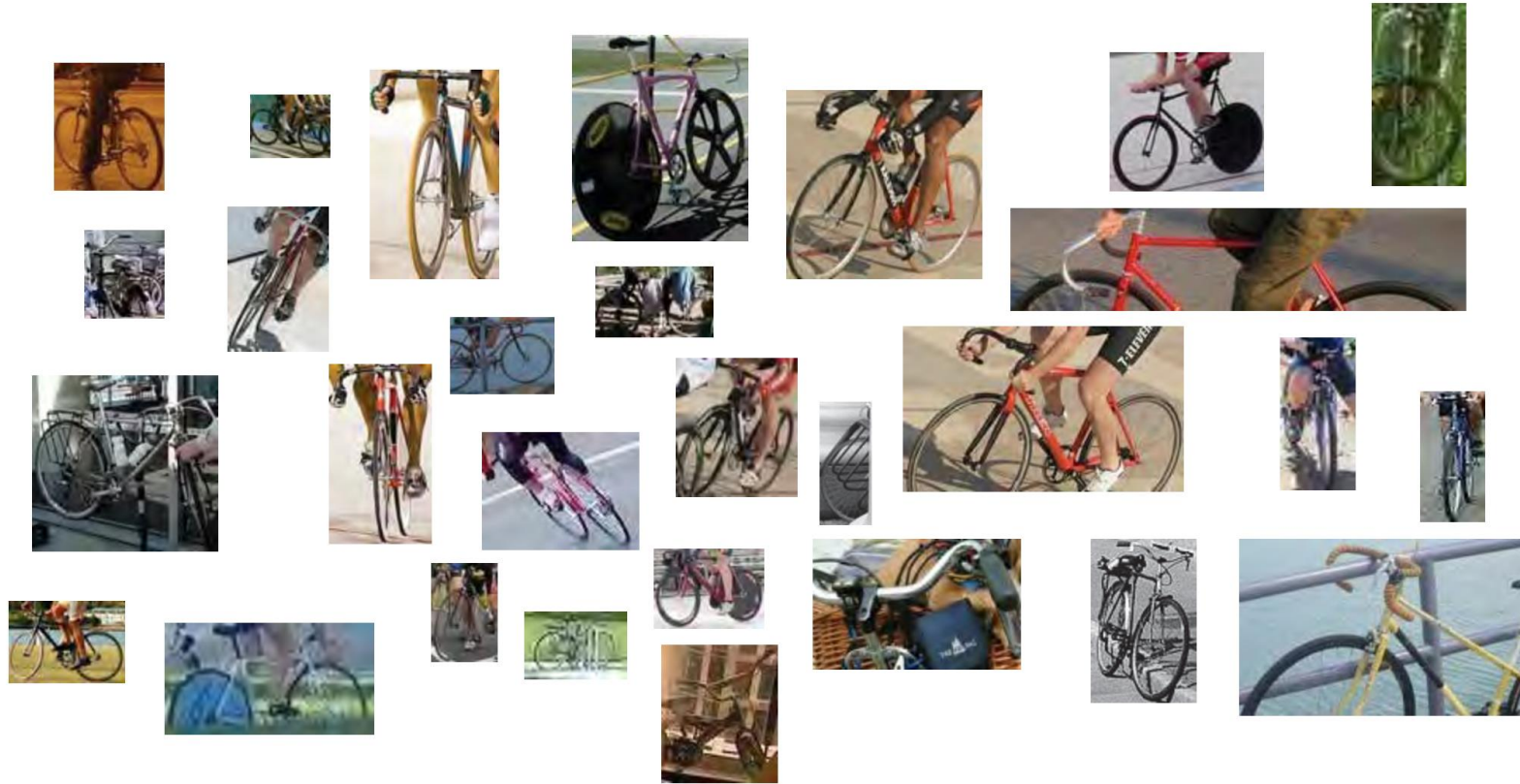
- Nájdenie objektu medzi miliónom snímok v priebehu sekúnd

Malé dáta

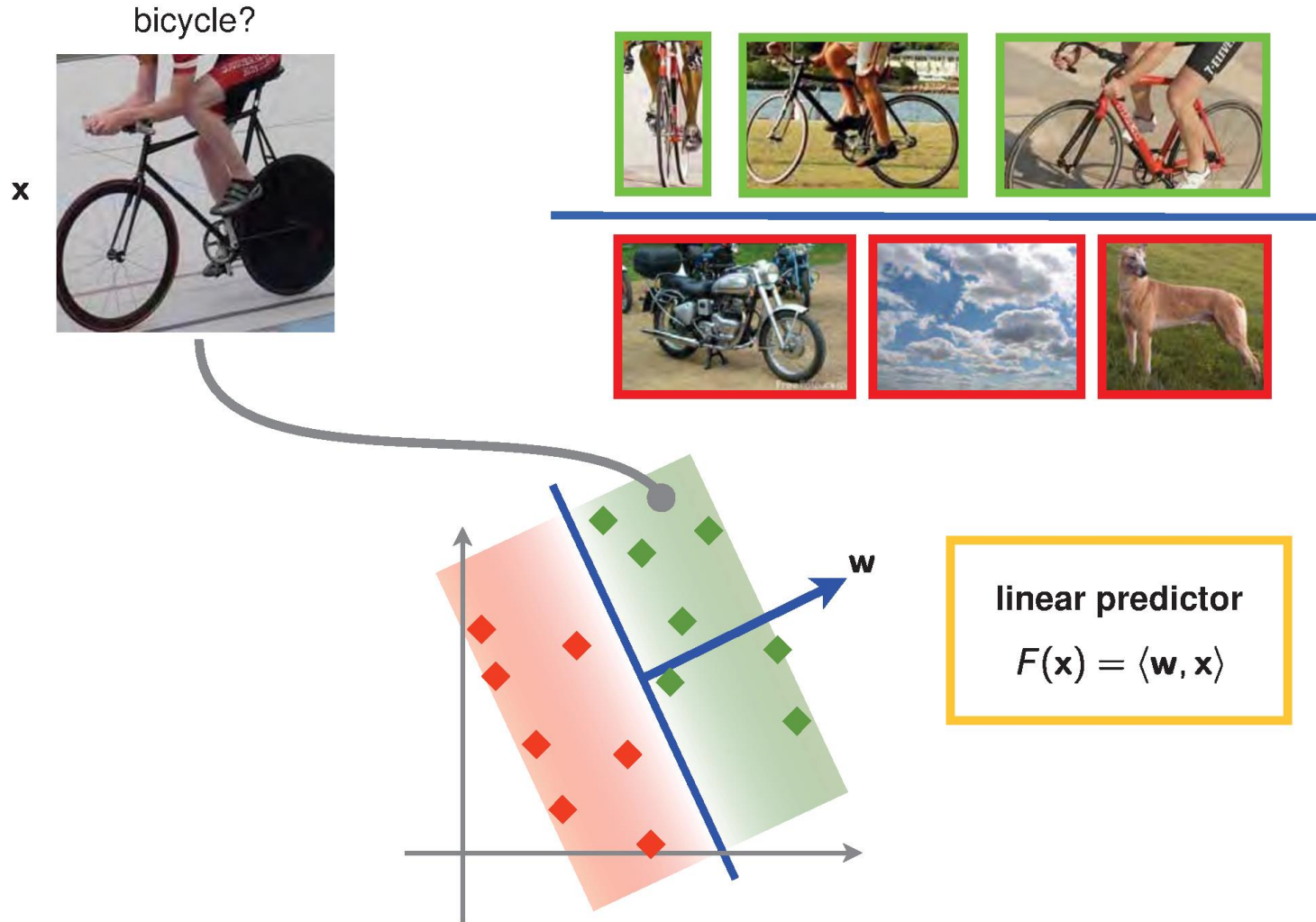
- Indexovať milióny snímok v RAM

Ako na to

Objekt = distribúcia 2D vzorov

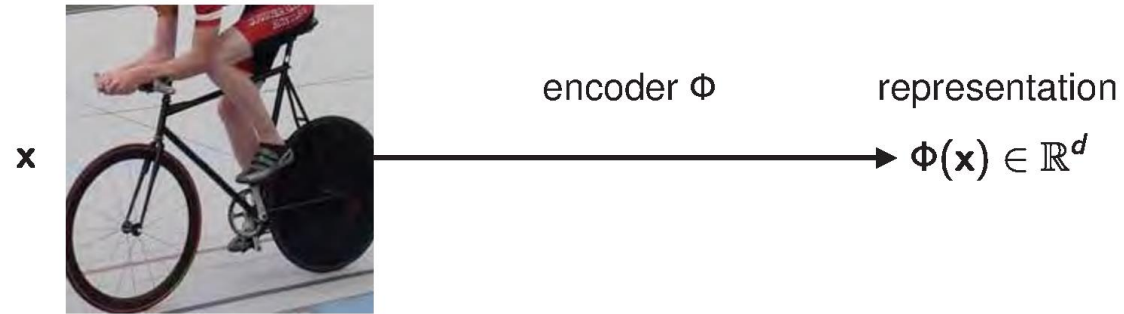


Lineárny klasifikátor



Reprezentácia obrazu

Použitie lineárneho klasifikátora na nevektorové dáta

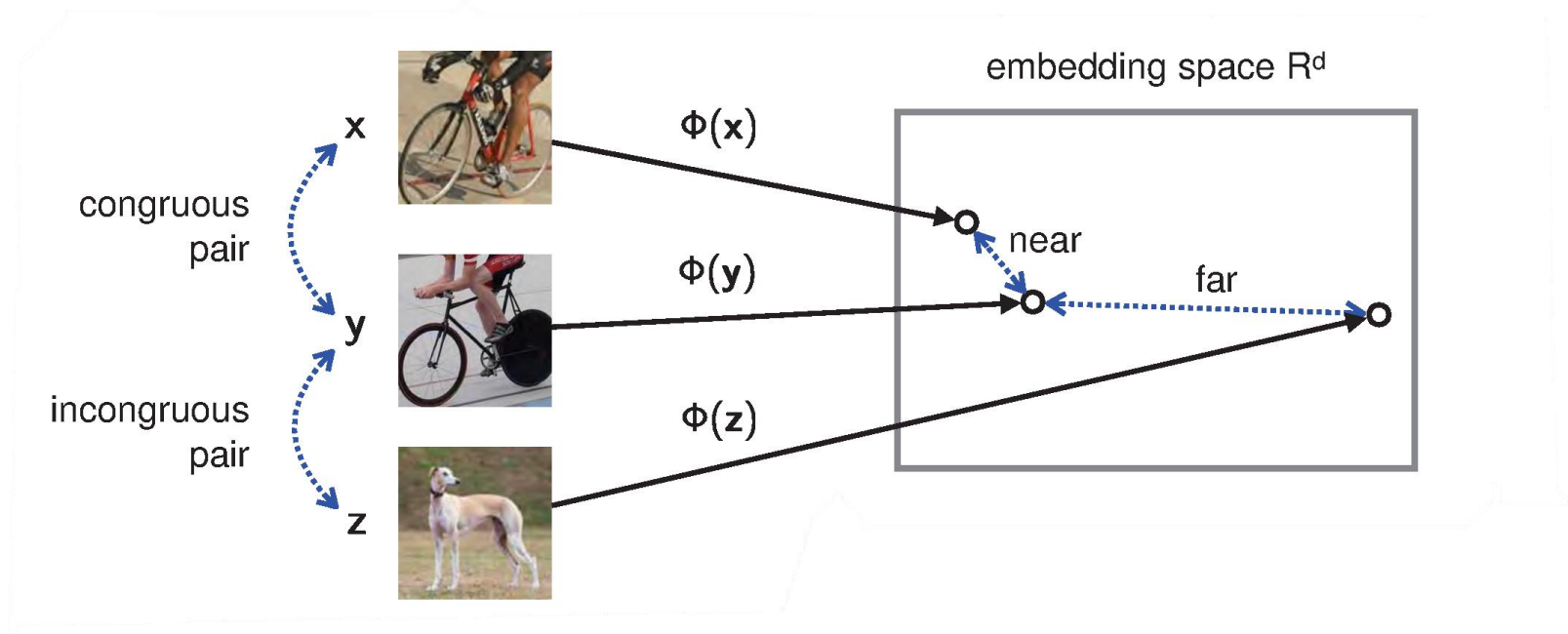
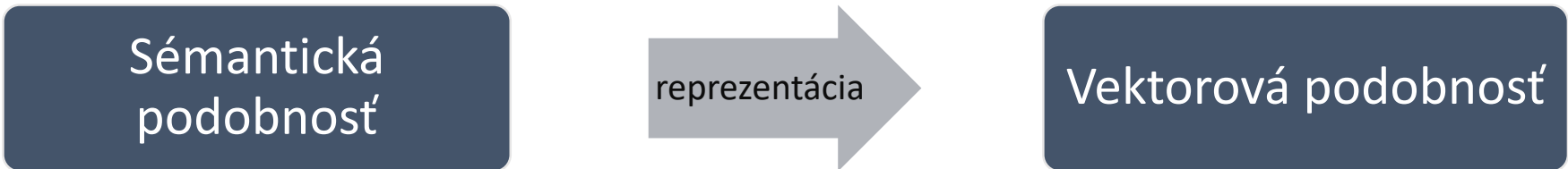


Enkodér mapuje dáta do **vektorovej podoby**.

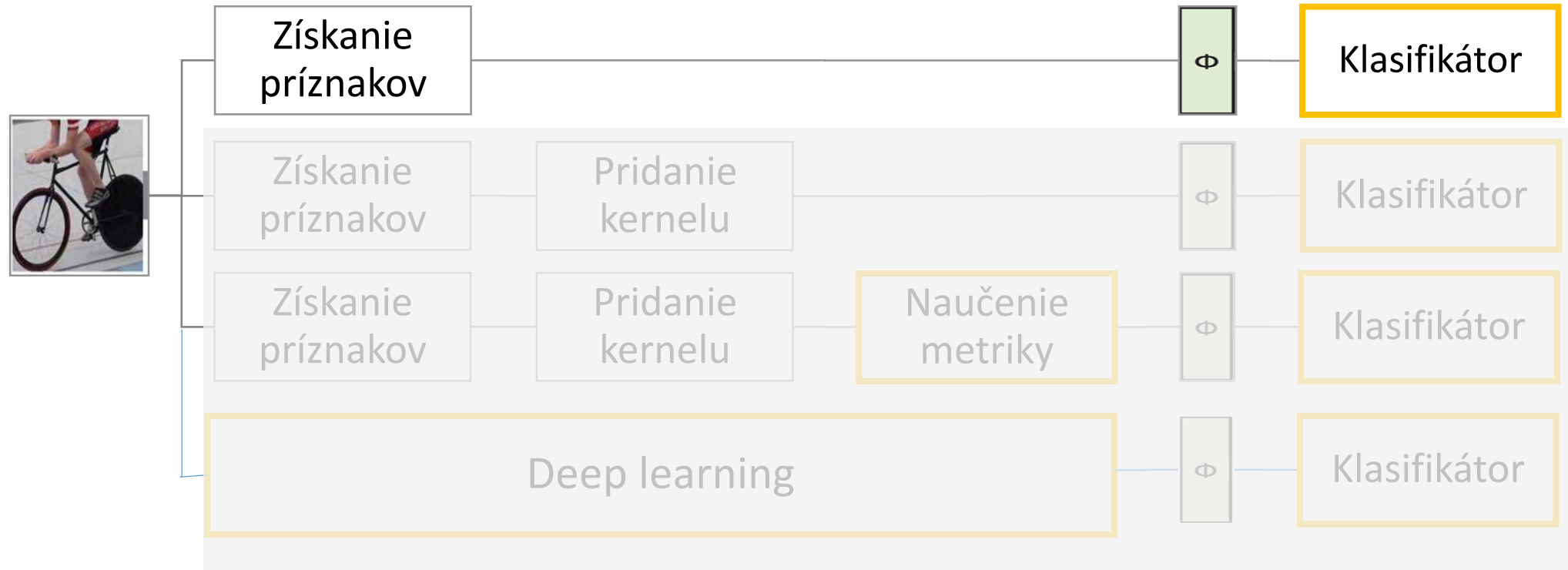
Umožňuje tak aplikáciu lineárneho klasifikátora na obraz, text, zvuk, video, ...

$$F(x) = \langle w, \Phi(x) \rangle$$

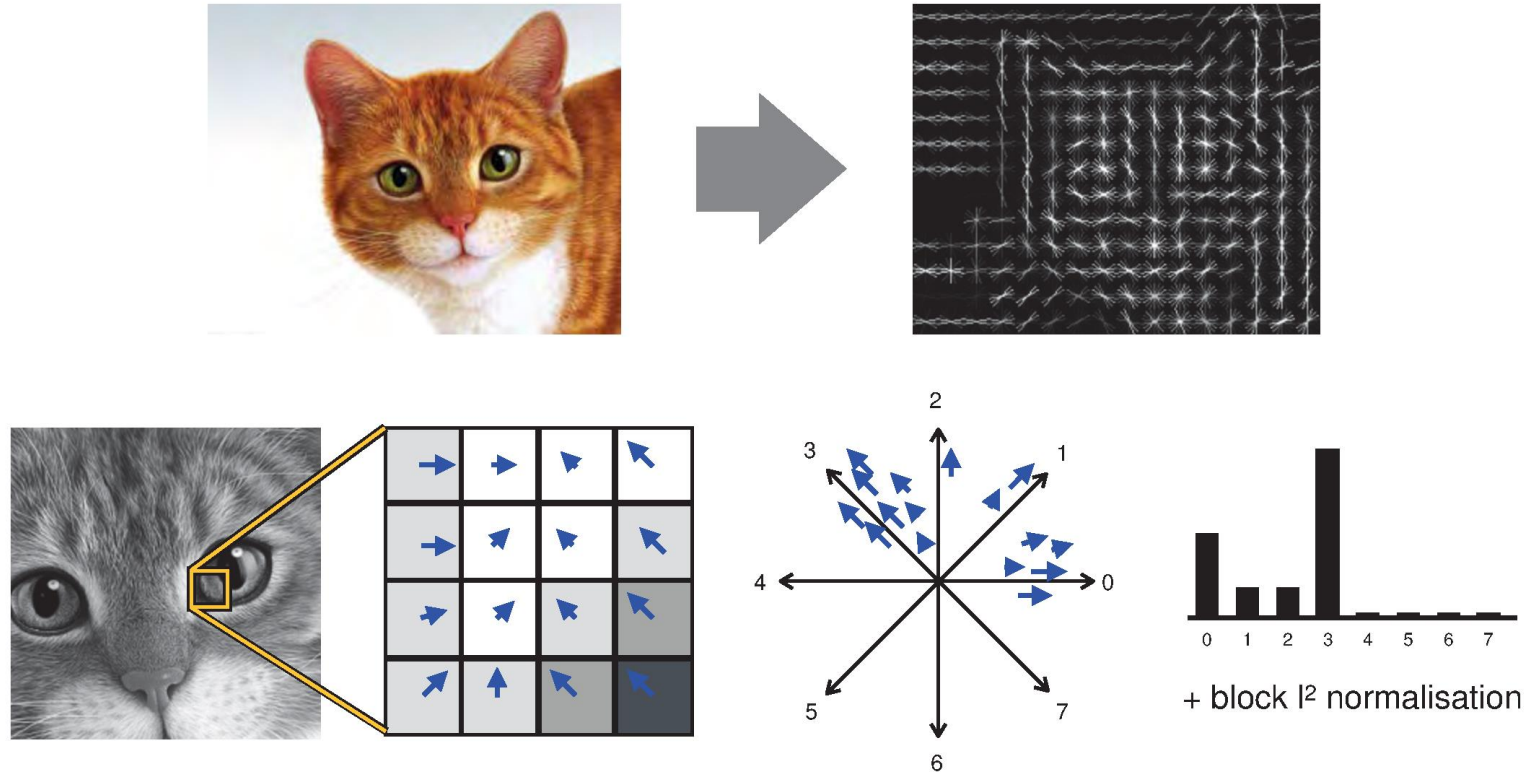
Zmysluplná reprezentácia



Lineárny klasifikátor



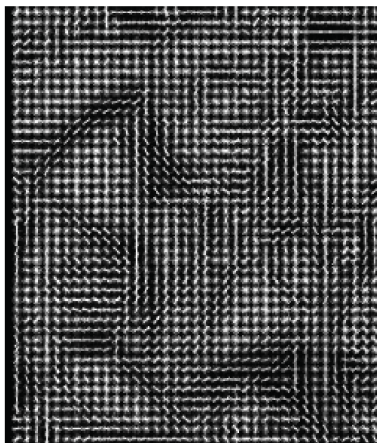
Histogram orientovaných gradientov



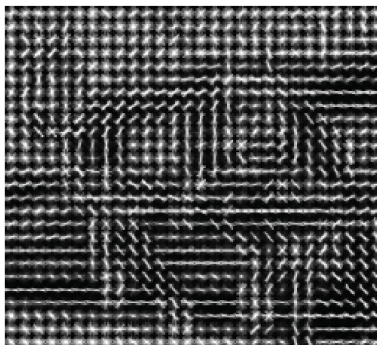
- Object recognition from local scale-invariant features, D.G. Lowe, Computer Vision, 1999.
- Histograms of Oriented Gradients for Human Detection Navneet Dalal and Bill Triggs 2005

Rekonštrukcia HOG

HOG(x)



$HOG^{-1}(x)$

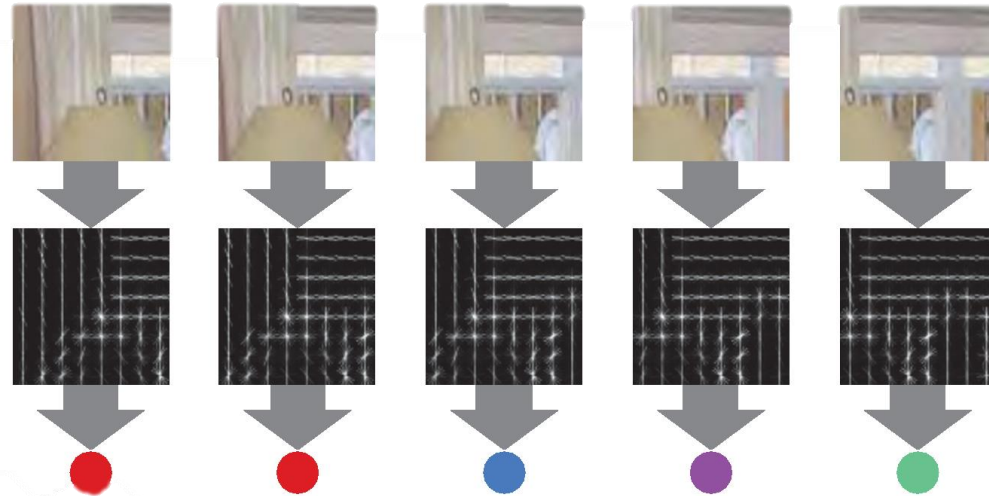


x



- [Visualizing Object Detection Features, Vondrick et al. 2013]

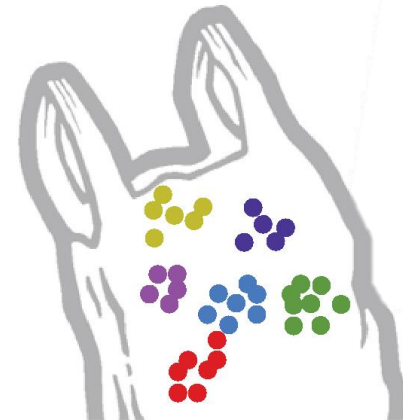
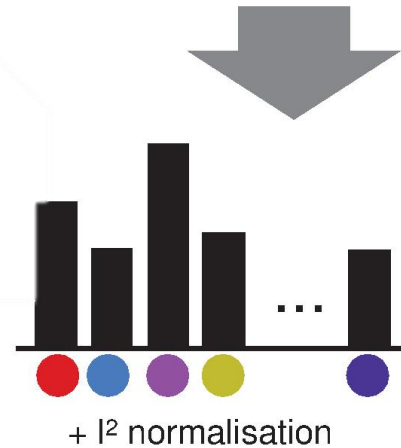
Vreće príznačov



Konštrukcia:

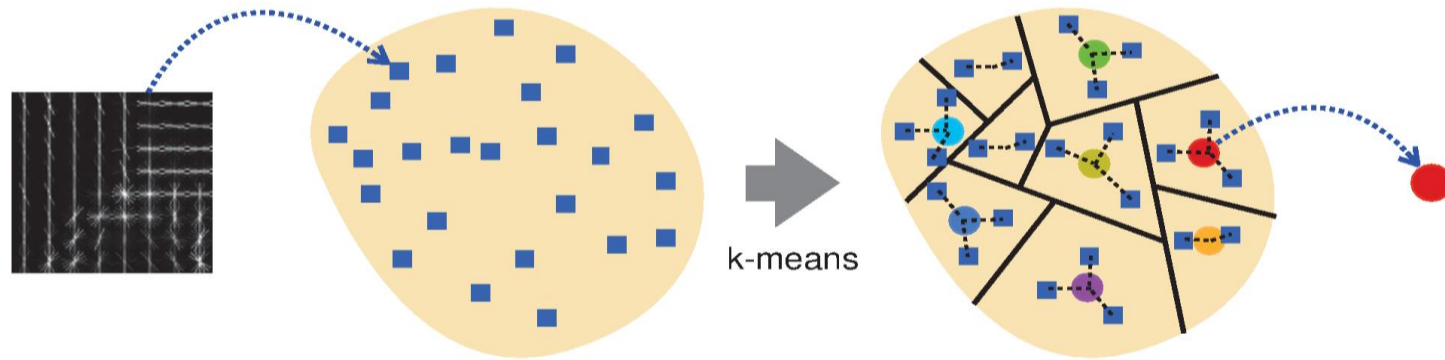
1. Extrakcia pomocou lokálneho deskriptora
2. Kvantizácia deskriptorov
3. Vytvorenie histogramu

Strata informácie o priestore



- Video Google: A Text Retrieval Approach to Object Matching in Videos, Sivic & Zissetman 2003
- Visual Categorization with Bags of Keypoints, Csurka et al. 2004
- Sampling Strategies for Bag-of-Features Image Classification, Novak et al. 2006

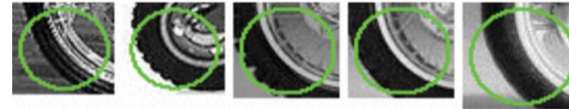
Kvantizácia



plane



torbike



face



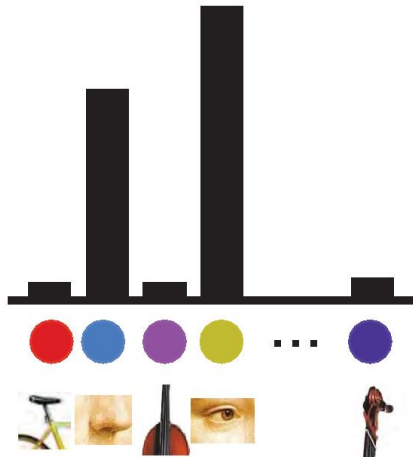
bike



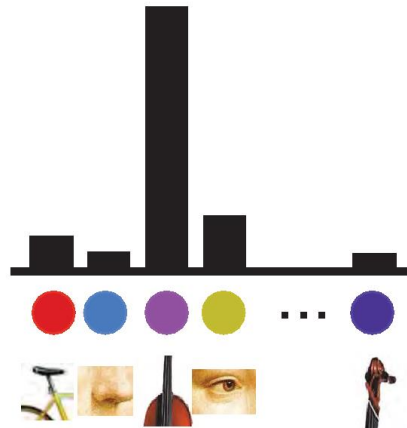
Vreće príznačov (zistenia)

Odstránením informácie o priestore získavame invarianciu (stálosť)

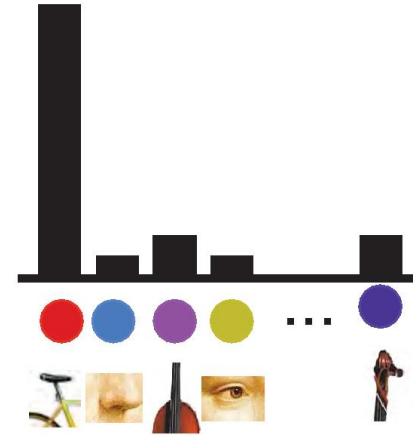
Pomocou viacerých kľúčových príznačov vieme vyskladať obraz predmetu.



person



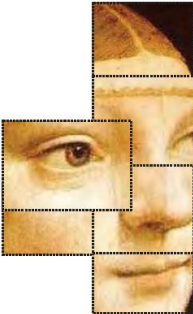
musical instrument



bike

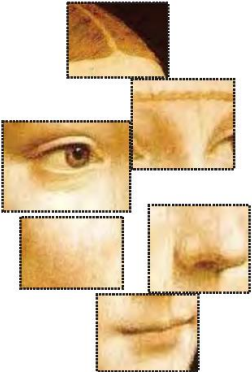
Dôsledok straty priestorovej informácie

image



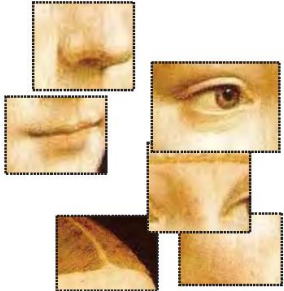
=

plausible deformation



=

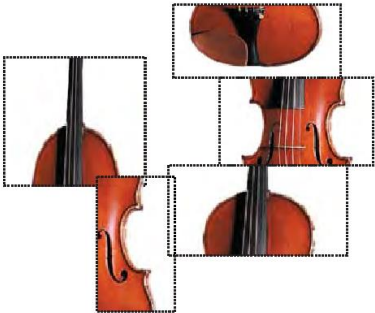
implausible deformation



=

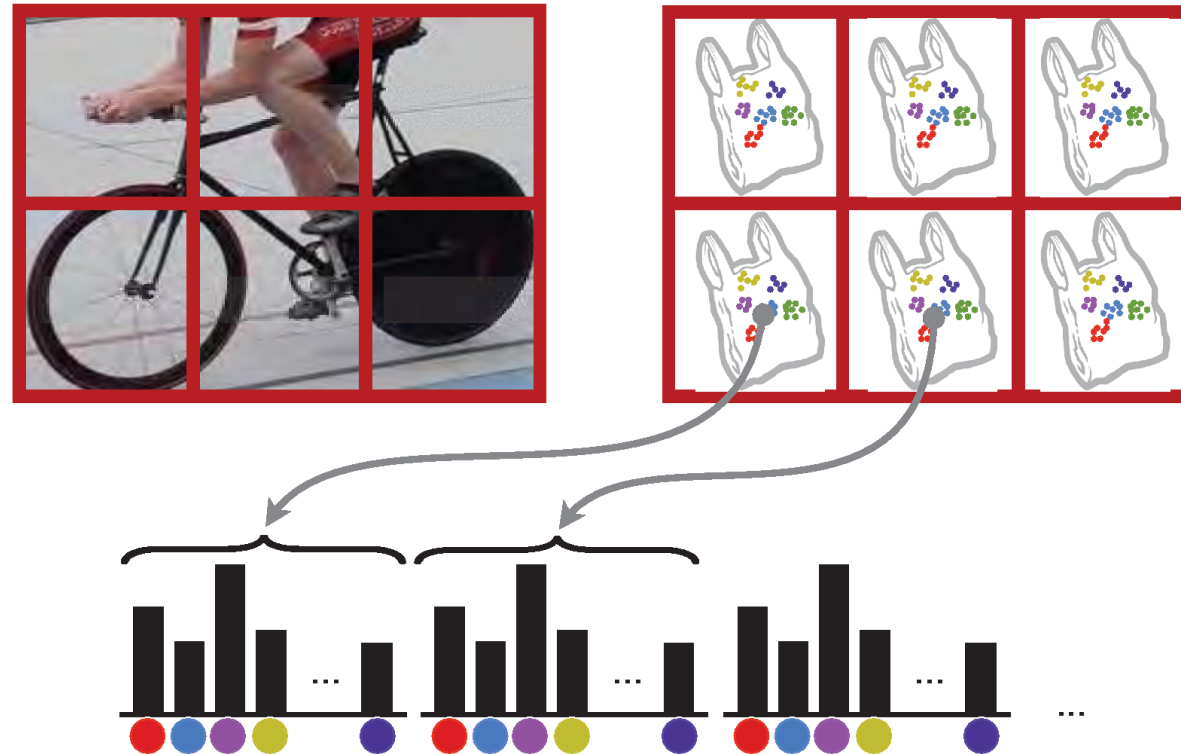


=



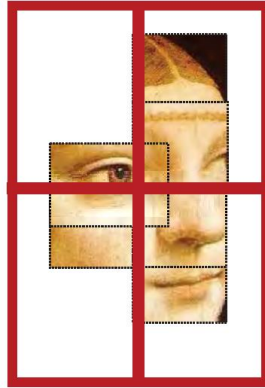
Priestorový histogram

Snaha uchovať priestorovú informáciu aspoň čiastočne. (pooling)

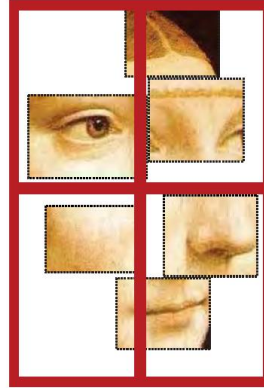


Uchovanie geometrie pomocou priestorového histogramu

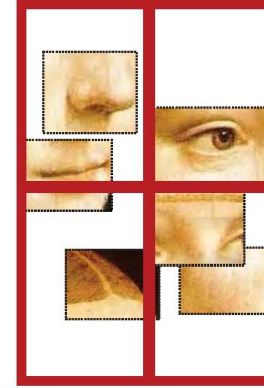
image



plausible deformation

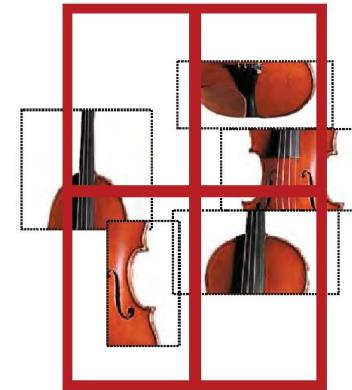
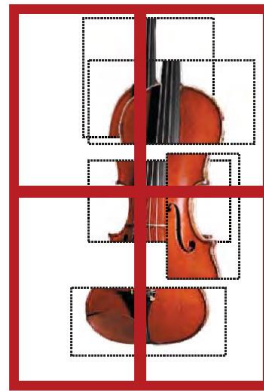
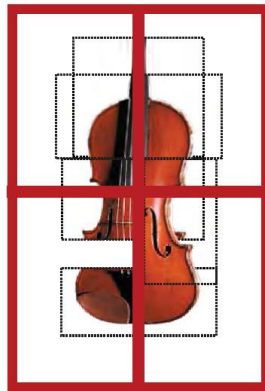


implausible deformation



=

≠



=

≠

Zložitejšie kódovanie obrazu

Soft and **sparse** assignments, e.g.

- ▶ [Philbin et al CVPR 08, Gemert et al ECCV 08]
- ▶ Locality-constrained linear coding (LLC) – [Wang et al CVPR 10]

Representing SIFT distribution **mean** in Voronoi cell, e.g.

- ▶ Super-Vector Coding [Zhou et al ECCV 10]
- ▶ VLAD [Jegou et al CVPR 10]

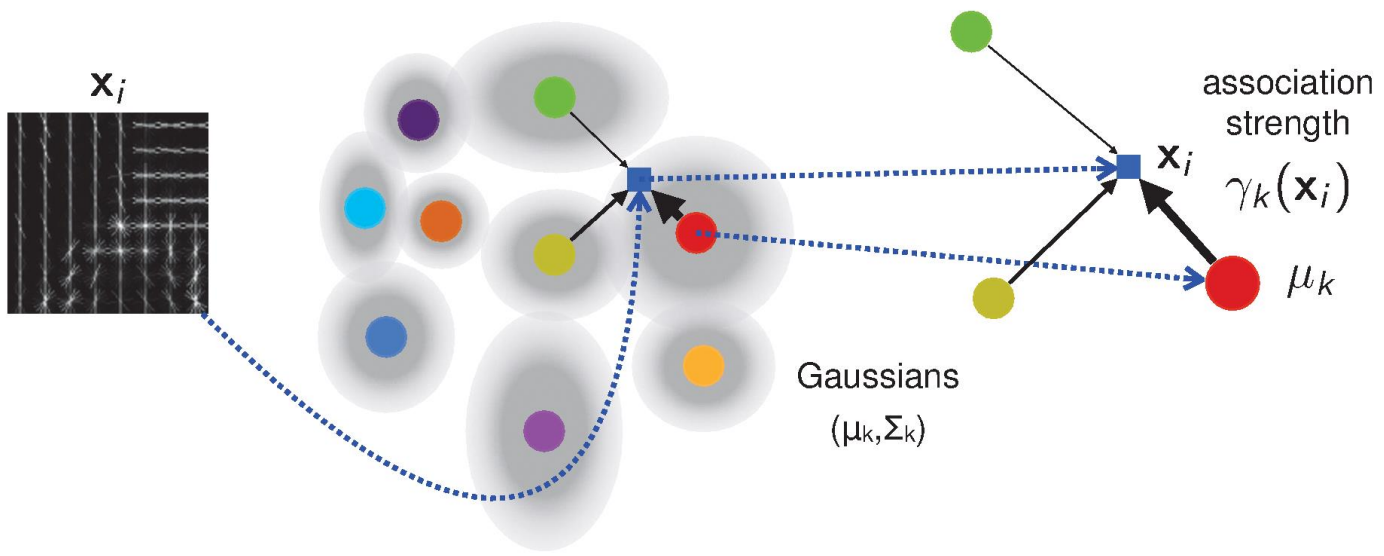
Representing SIFT distribution **mean** and **covariance** in Voronoi cell, e.g.

- ▶ Fisher vector [Perronnin et al CVPR 07 & 10, ECCV 10]

Improvements to **normalization**, PCA, **whitening** for VLAD/FV

- ▶ Chen et al 2011 [Jegou & Chum ECCV 12]
- ▶ All about VLAD [Arandjelovic & Zisserman CVPR 13]

Fisherov vektor



FV encoding $\Phi =$

$$\begin{bmatrix} \mathbf{v}_1 \\ \mathbf{u}_1 \\ \mathbf{v}_2 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{v}_K \\ \mathbf{u}_K \end{bmatrix}$$

+ sqrt-l² normalisation

first and second order statistics

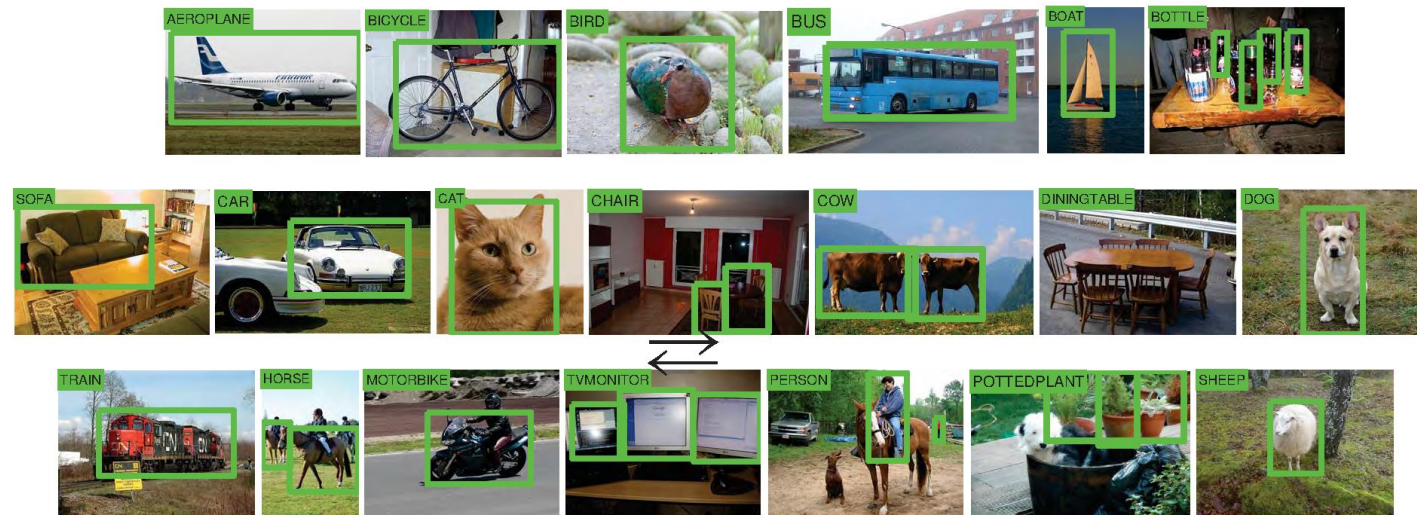
$$\mathbf{v}_k = \frac{1}{M\sqrt{\pi_k}} \sum_{i=1}^M \gamma_k(\mathbf{x}_i) \frac{\mathbf{x}_i - \mu_k}{\sigma_i}$$

$$\mathbf{u}_k = \frac{1}{M\sqrt{2\pi_k}} \sum_{i=1}^M \gamma_k(\mathbf{x}_i) \left(\frac{\mathbf{x}_i - \mu_k}{\sigma_i} - 1 \right)^2$$

- Improving the Fisher Kernel for Large-Scale Image Classification, Perronnin et al. ECCV 2010
- Local higher-order statistics (LHS) for texture categorization and facial analysis, Sharma, Hussain, Jurie ECCV 2012
- Image Classification with the Fisher Vector: Theory and Practice , Sanchez et al. 2013

Benchmark: PASCAL VOC

Úloha: rozhodnúť, či obraz obsahuje aspoň jeden z 20 hľadaných objektov

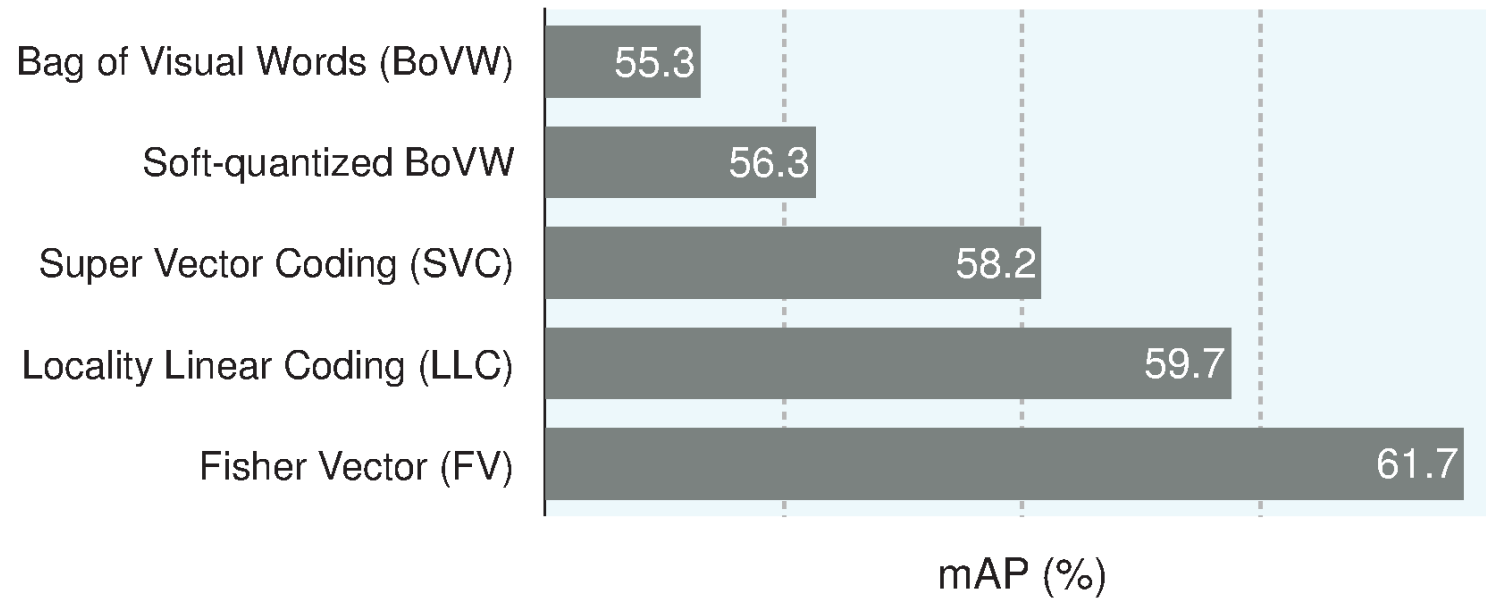


Výkon:
Mean Average Precision

mAP = 50% → 50% objektov bolo rozoznaných dôveryhodne

- [Everingham et al, 2006-2012, <http://host.robots.ox.ac.uk/pascal/VOC/>]

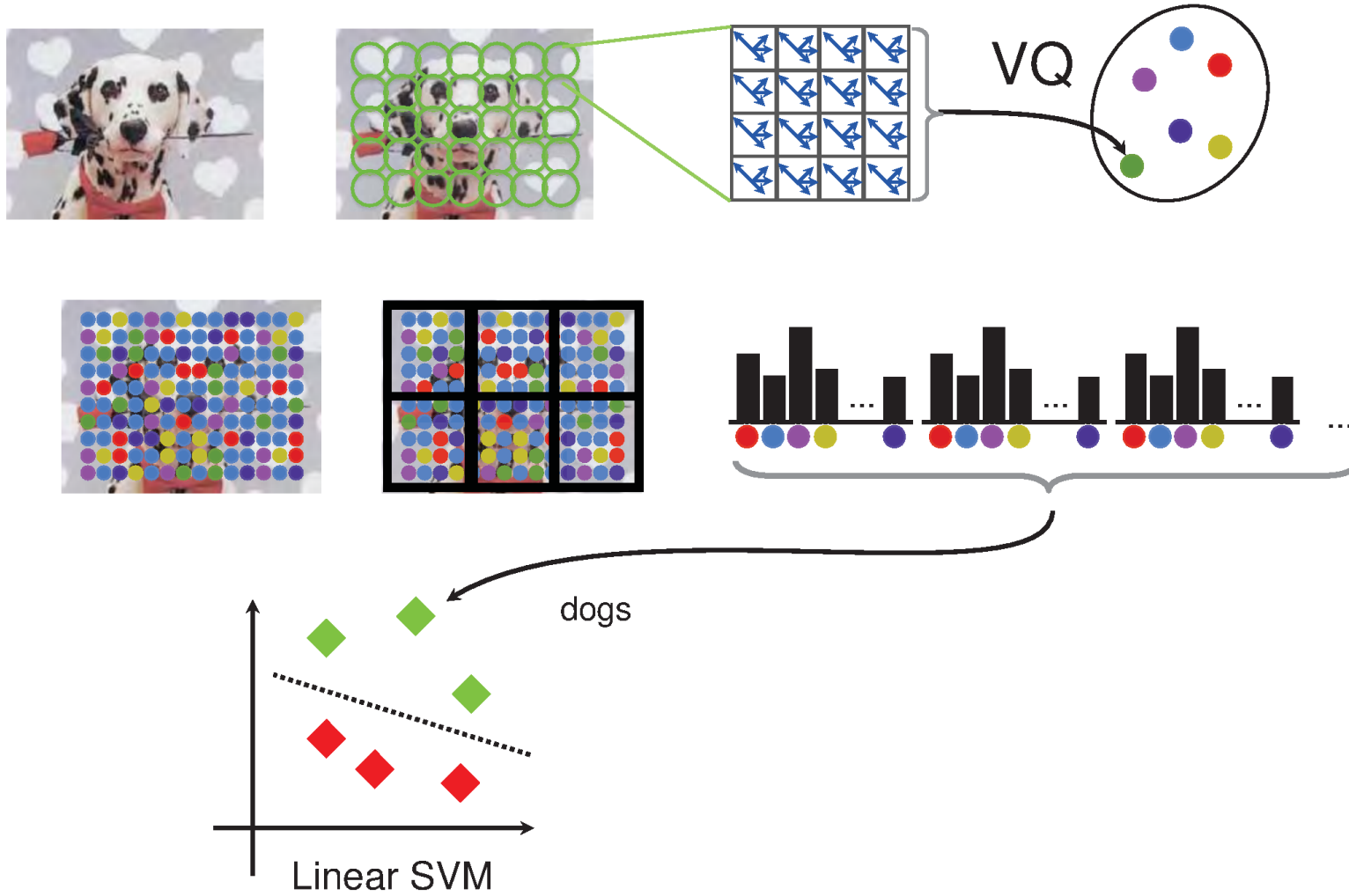
Porovnanie výsledkov



- Video Google: A Text Retrieval Approach to Object Matching in Videos, Sivic & Zissetman 2003
- Visual Categorization with Bags of Keypoints, Csurka et al. 2004
- A Simple High Performance Approach to Semantic Segmentation, G. Csurka and F. Perronnin, BMVC08,....

1999 - 2012

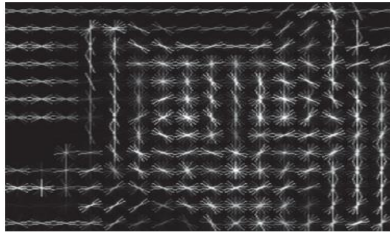
28



Podstatné zistenia

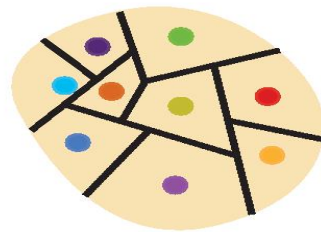
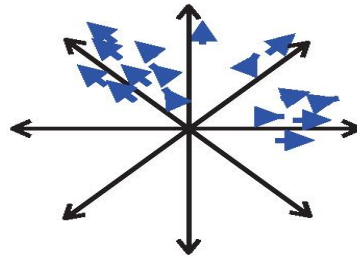
Lokálne operátory a
operátory invariantné voči
natočeniu

Gradienty, filtre, vizuálne
slová



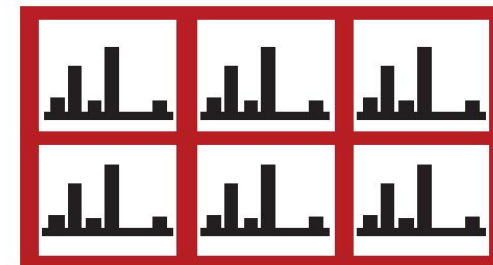
„Rozmotávanie“

hustota, kvantizácia

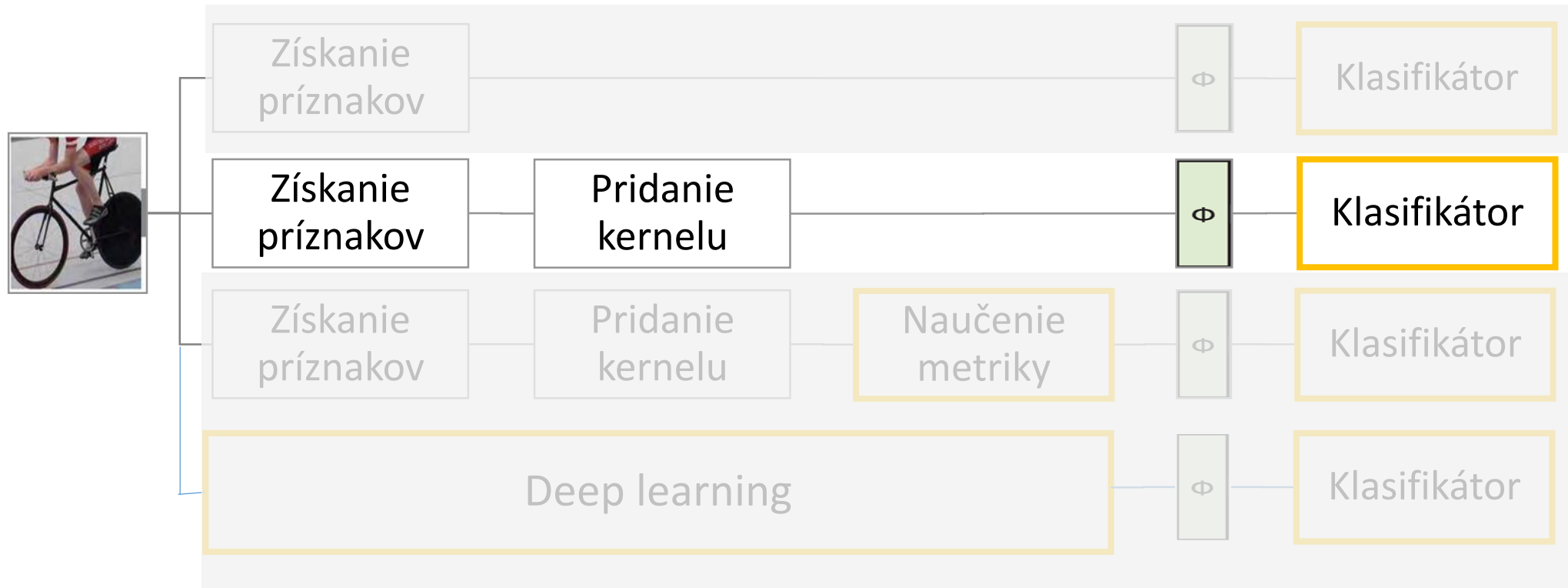


Pooling

Max, suma, priestorový
pooling



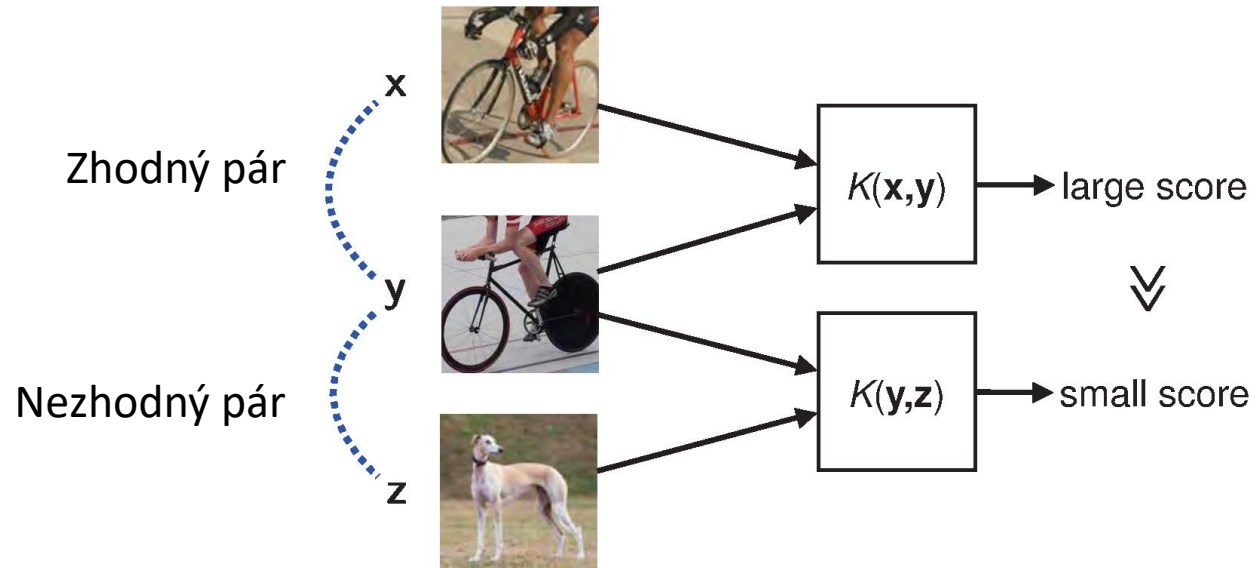
Lineárny klasifikátor



Kernely (jadrá)

Kernely (jadrá) priamo kódujú mieru podobnosti dát.

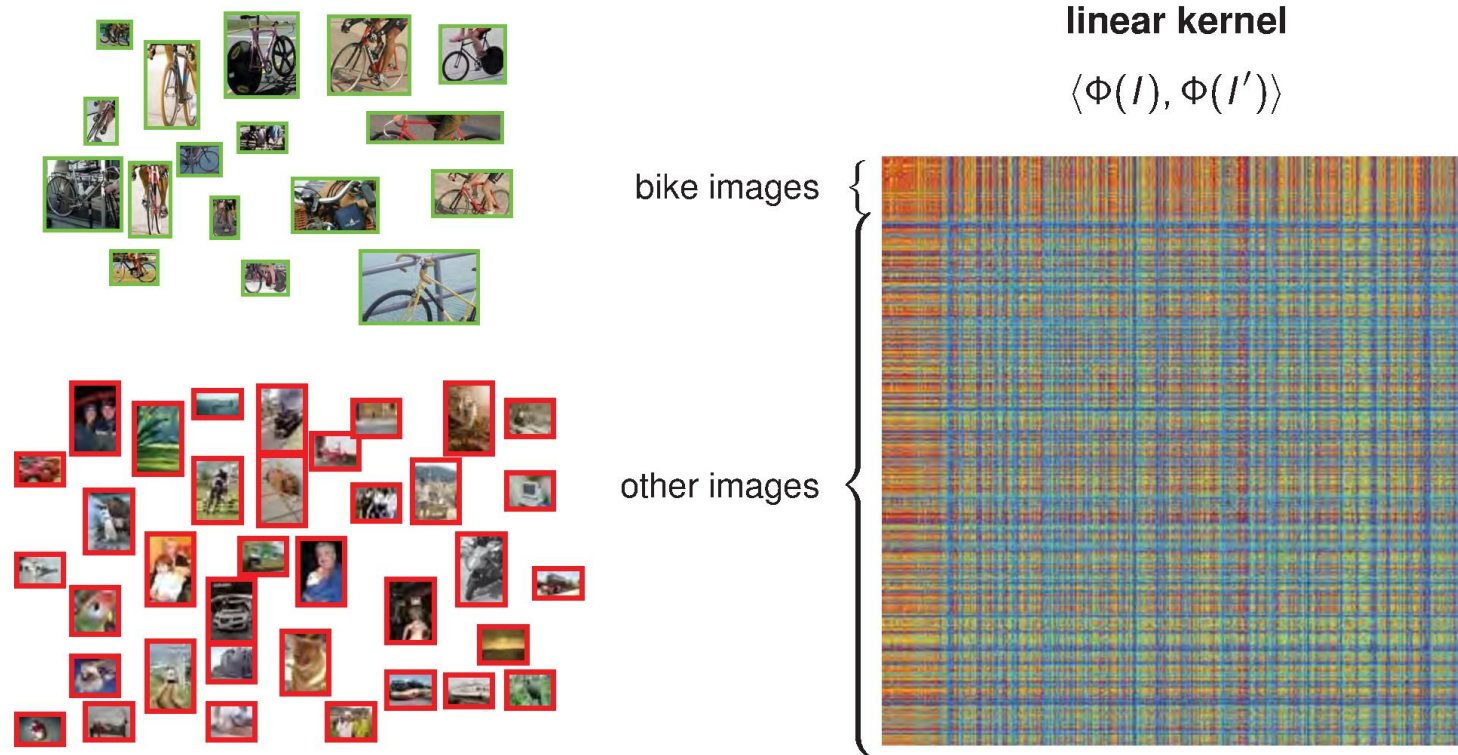
$$K : (\mathbf{x}, \mathbf{y}) \mapsto \mathbb{R}$$



Podobnosť a jadrá

Pre oživenie: Enkodér $\Phi(I)$ predstavuje užitočnú mieru podobnosti.

$$\langle \Phi(I), \Phi(I') \rangle = \text{lineárny kernel}$$



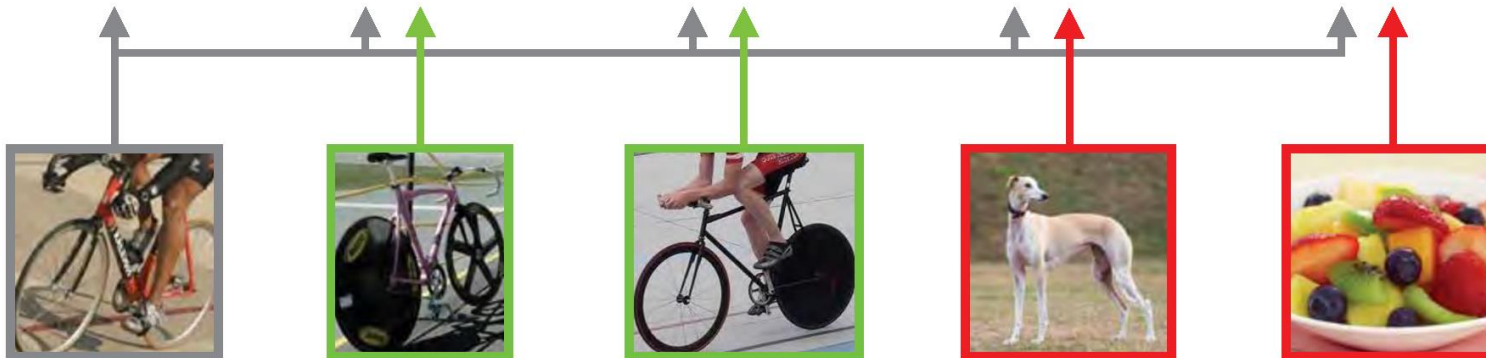
Prediktor s jadrom

Úloha: predyhovať triedu údajov x

Riešenie : použiť K na porovnanie vzorky x so všetkými $x_1, x_2 \dots$

$$F(\mathbf{x}) = \sum_{i=1}^N \alpha_i K(\mathbf{x}, \mathbf{x}_i)$$

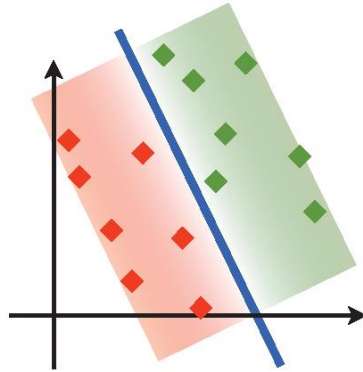
$$F(\mathbf{x}) = \alpha_1 K(\mathbf{x}, \mathbf{x}_1) + \alpha_2 K(\mathbf{x}, \mathbf{x}_2) + \alpha_3 K(\mathbf{x}, \mathbf{x}_3) + \alpha_4 K(\mathbf{x}, \mathbf{x}_4) + \dots$$



Nelineárne kernely

Lineárny SVM

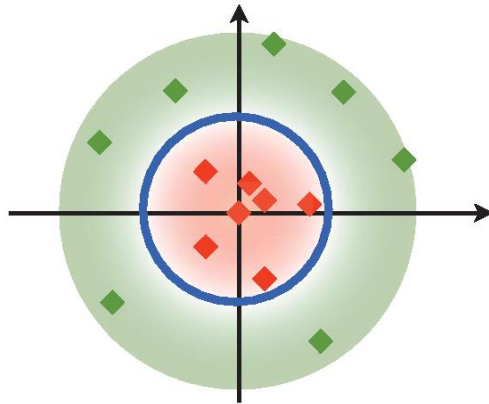
- ✓ Rýchly
- ✗ Reštriktívny



$$F(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle$$

Nelineárny SVM

- ✓ Pomalší
- ✗ Silnejší



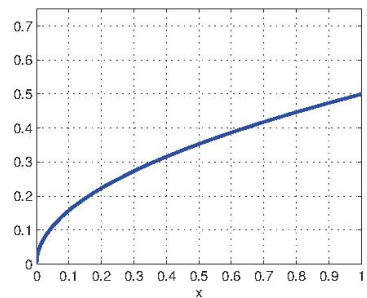
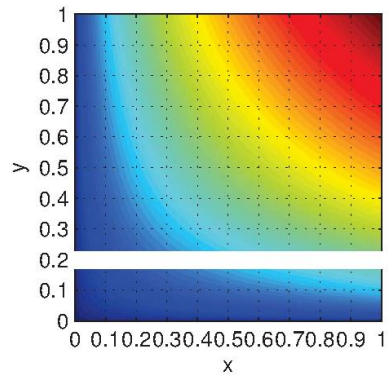
$$F(\mathbf{x}) = \sum_{i=1}^N \alpha_i K(\mathbf{x}, \mathbf{x}_i)$$

Aditívne homogénne kernely

$$K(\mathbf{x}, \mathbf{y}) = \sum_{l=1}^d k(x_l, y_l)$$

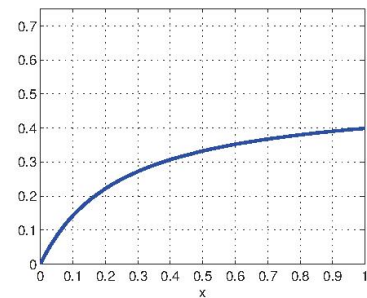
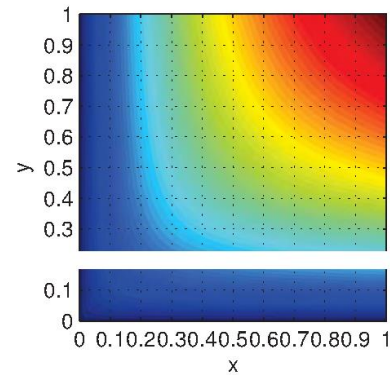
Hellinger

$$\sqrt{xy}$$



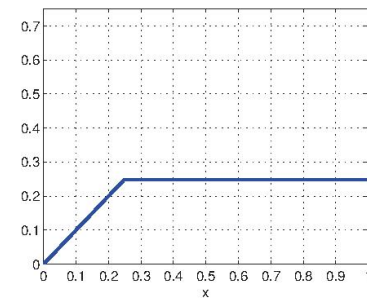
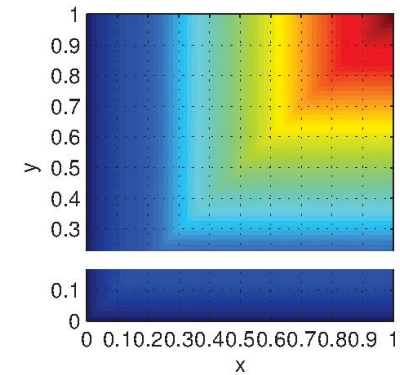
x²

$$\frac{2xy}{x+y}$$



intersection

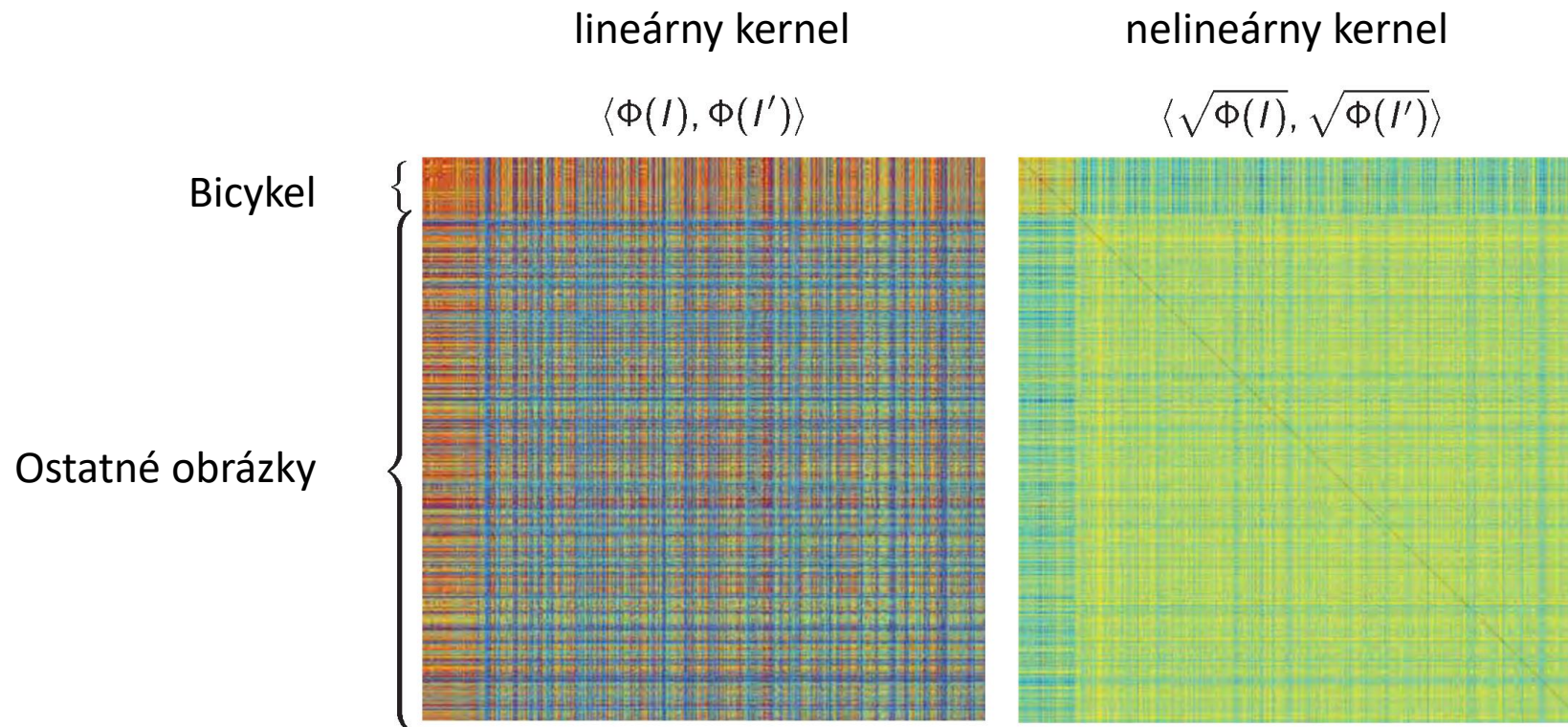
$$\min\{x, y\}$$



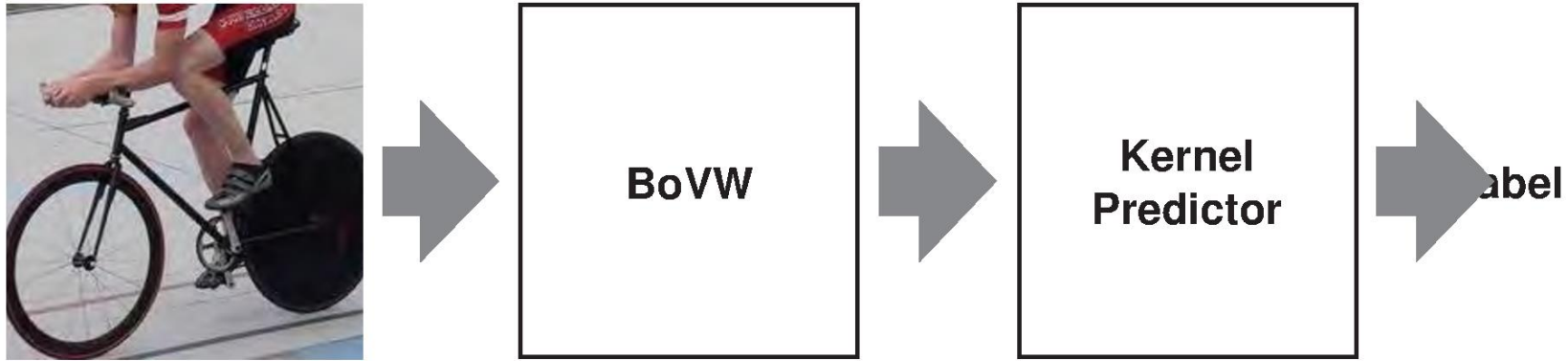
Efektivita pridaním kernelov

Nahradením lineárnych kernelov nelineárnymi zvykneme dostať lepšie výsledky pri klasifikácii tried.

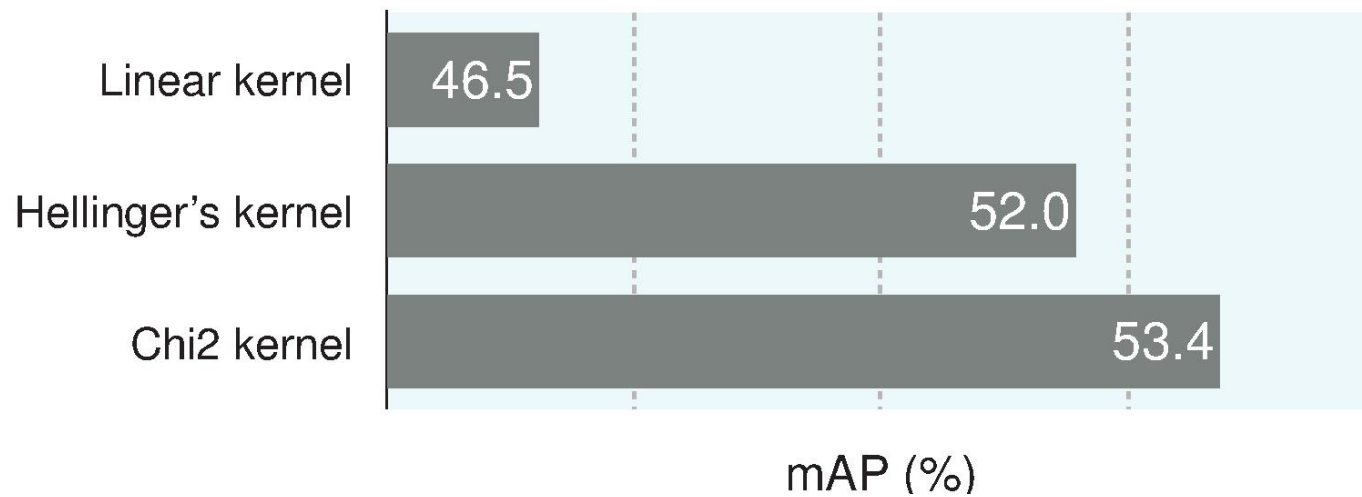
Dokonca aj jednoduché jadrá vedia pomôcť.



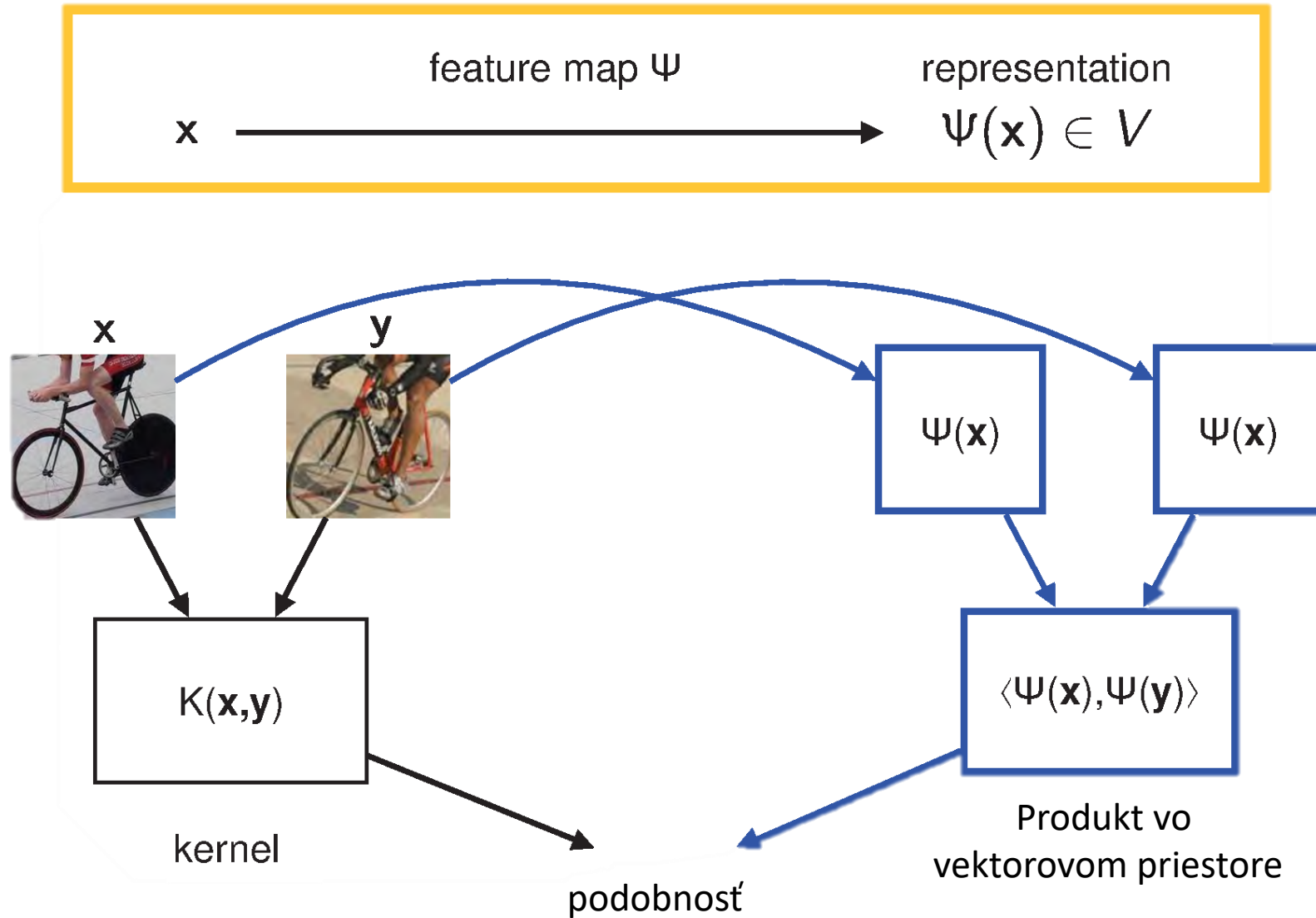
Efektivita pridaním kernelov



Bag of Visual Word on PASCAL VOC 07



Kernelova mapa



Explicitne mapy sú efektívnejšie

Rýchlejšie ohodnocovanie

$$F(\mathbf{x}) = \sum_{i=1}^N \alpha_i K(\mathbf{x}, \mathbf{x}_i)$$

O(N)

explicit map



$$F(\mathbf{x}) = \langle \mathbf{w}, \Phi(\mathbf{x}) \rangle$$

O(1)

Rýchlejšie učenie

Non-linear SVM

LibSVM

O(N²)

explicit map

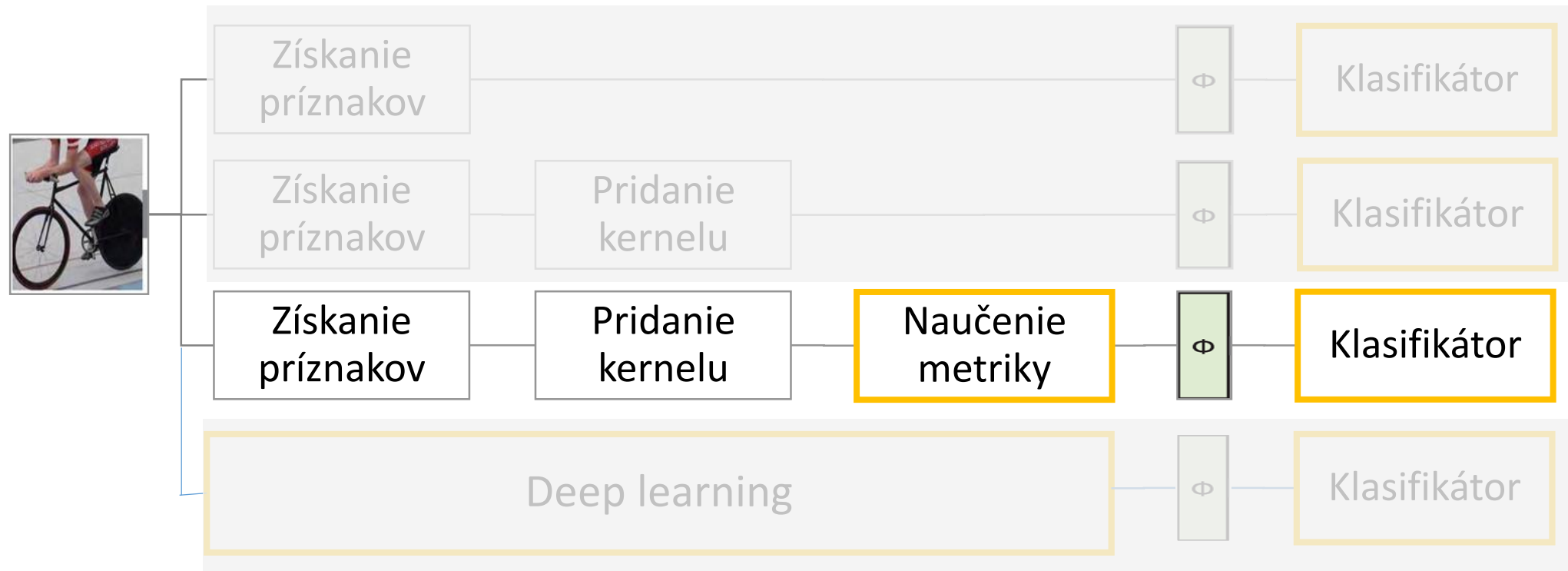


Linear SVM solver

LibLinear

O(N)

Lineárny klasifikátor



Učíme sa porovnávať

Cieľ

- Porovnávať objekty x, y
- Naučiť sa metriku $d^2(x, y)$

Desideratum

- Ak x a y sú zhodné objekty \rightarrow malá vzdialenosť
- Ak x a y nie sú zhodné \rightarrow veľká vzdialenosť

Parametrizácia vzdialenosti

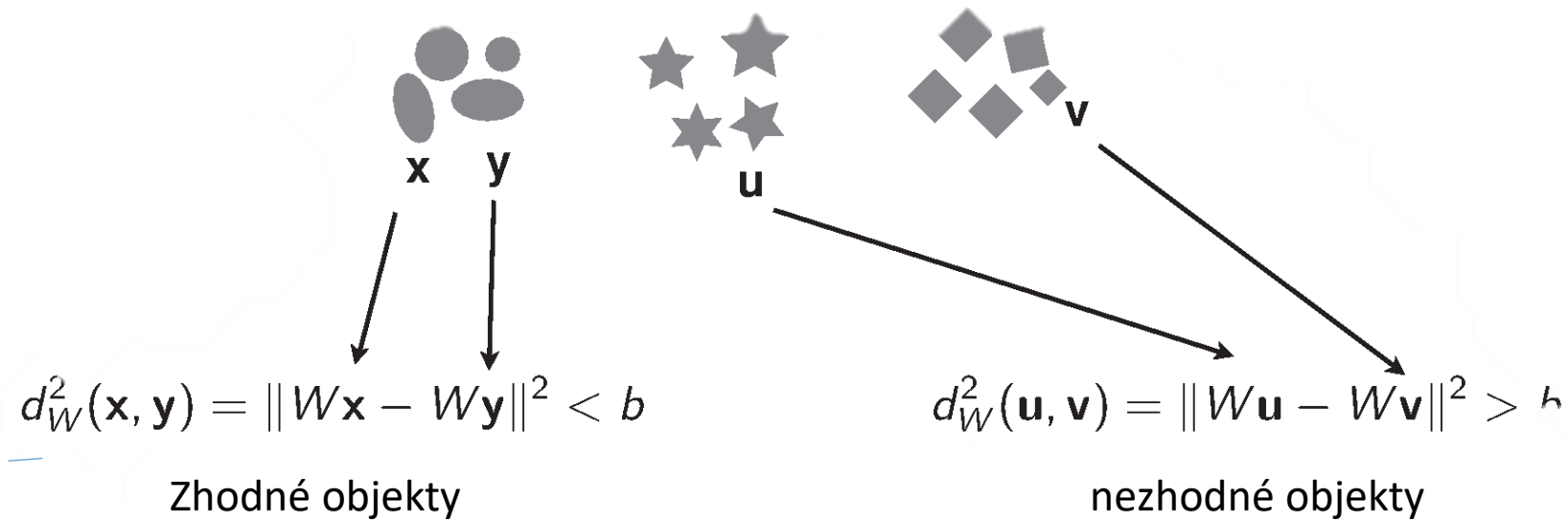
Euklidovská vzdialenosť + lineárna projekcia W

$$d_W^2(\mathbf{x}, \mathbf{y}) = \|W\mathbf{x} - W\mathbf{y}\|^2$$

Obmedzenia podobné klasifikácii

Pre všetky páry objektov x, y

- Zhodné \rightarrow vzdialenosť je menšia než threshold - margin
- Rozdielne \rightarrow vzdialenosť je väčšia než threshold + margin



$$d_W^2(x, y) < b - 1, \quad d_W^2(u, v) > b + 1$$

Porovnať a zmenšiť

Euklidovská vzdialenosť

$$d_W^2(\mathbf{x}, \mathbf{y}) = \|\mathbf{W}\mathbf{x} - \mathbf{W}\mathbf{y}\|^2$$

+

lineárna projekcia

$$\mathbf{x} \in \mathbf{R}^n \xrightarrow{\mathbf{W} \in \mathbf{R}^{m \times n}} \bar{\mathbf{x}} = \mathbf{W}\mathbf{x} \in \mathbf{R}^m$$

\mathbf{W} dokáže zlepšiť separáciu dát = (učí sa zmysluplné podobnosti)

\mathbf{W} taktiež redukuje dimenziu dát

$$\begin{array}{|c|} \hline \bar{\mathbf{x}} \\ \hline \end{array} = \begin{array}{|c|} \hline \mathbf{W} \\ \hline \end{array} \begin{array}{|c|} \hline \mathbf{x} \\ \hline \end{array}$$

Podíme porovnávať ľudí

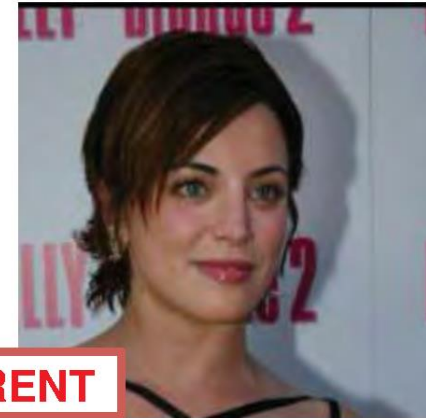
Klasická aplikácia vytvorenia metriky



SAME



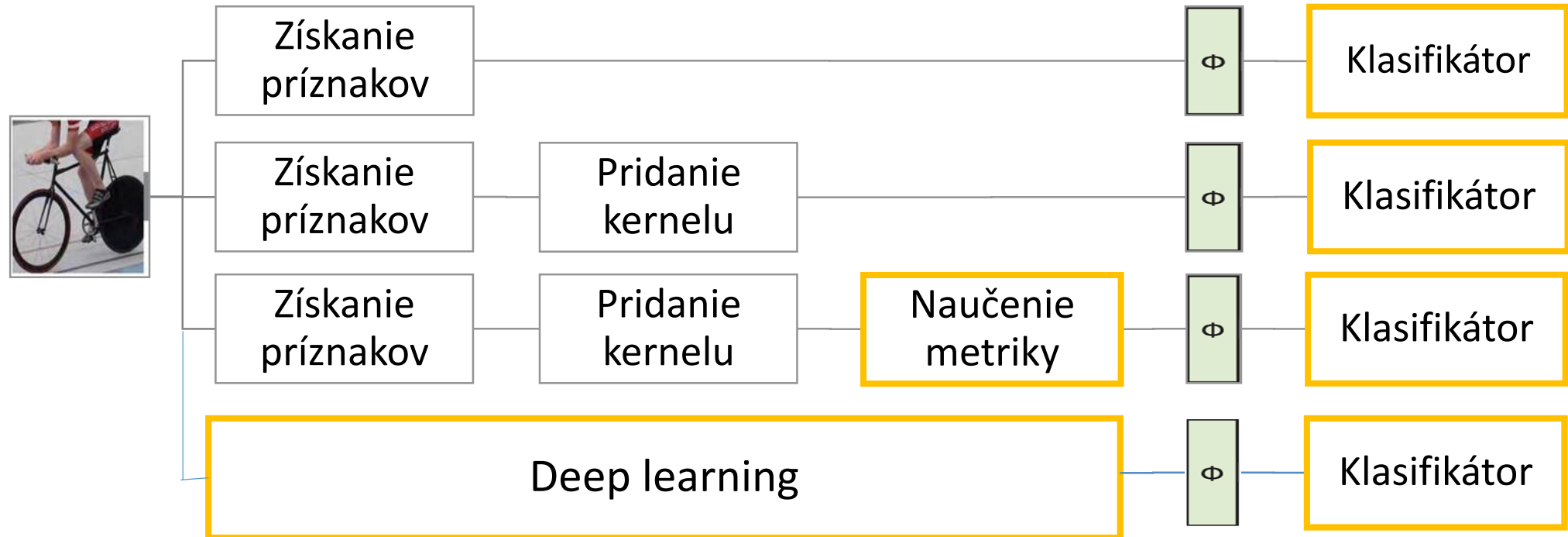
DIFFERENT



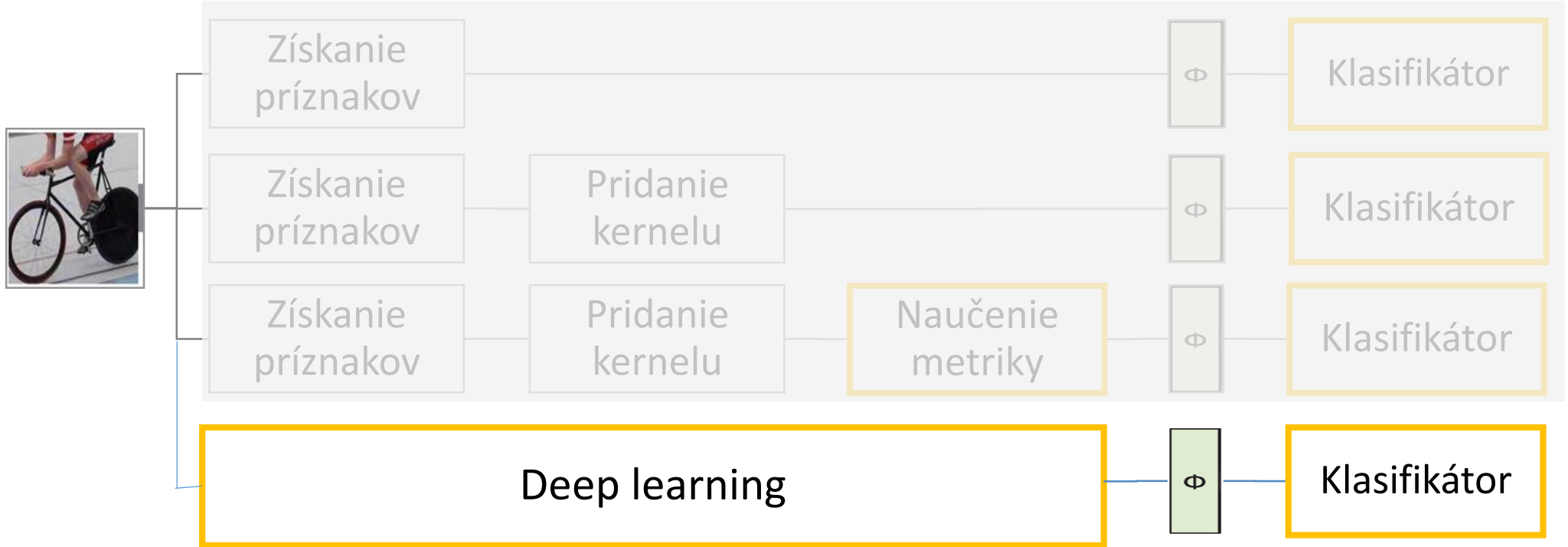
Úloha

- Rozhodnime či obrázky zobrazujú rovnakú osobu
- Vyberme presný a kompaktný deskriptor

Lineárny klasifikátor



Lineárny klasifikátor



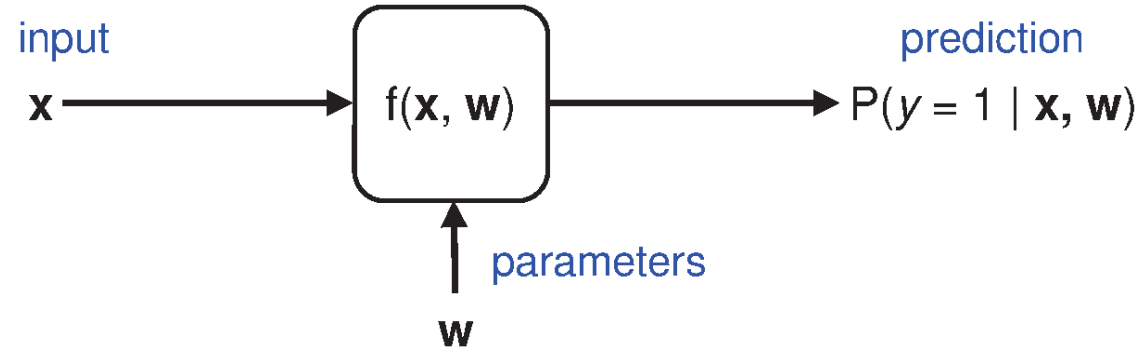
Konvolučné neurónové siete

Úvod do problematiky

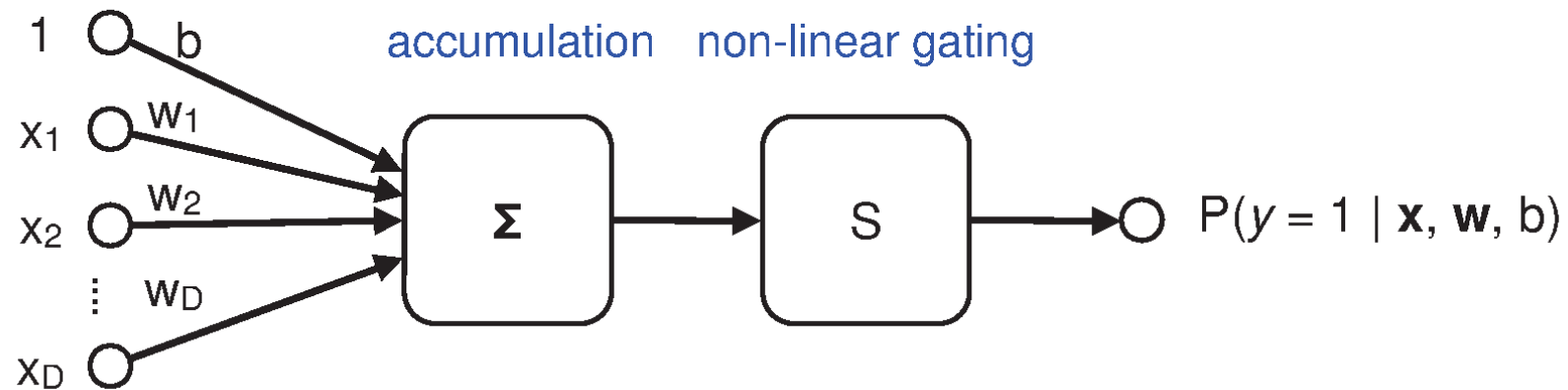
Vizualizácia

Perceptrón

Úloha: určiť pravdepodobnosť priradenia vstupu x do triedy y



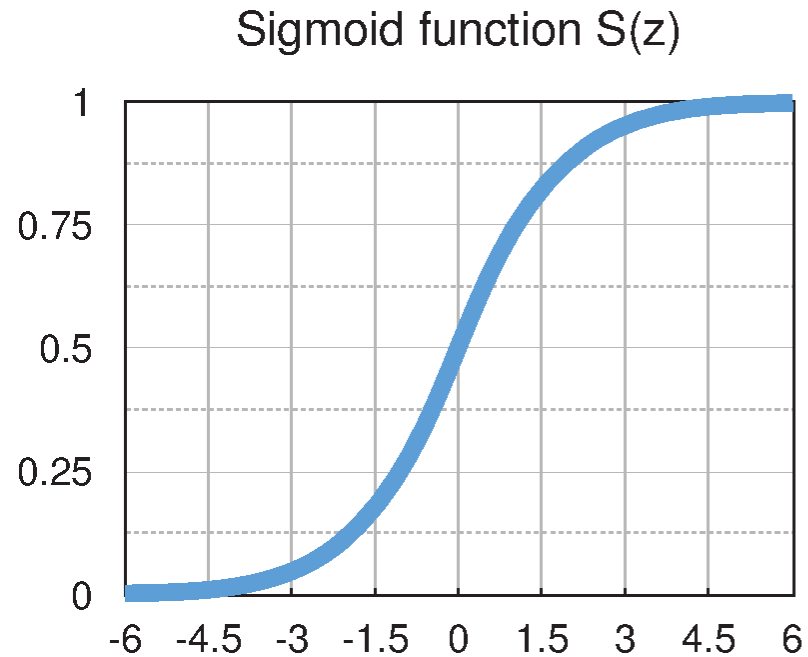
linear weighting



Perceptrón

Aktivačná funkcia (sigmoid)

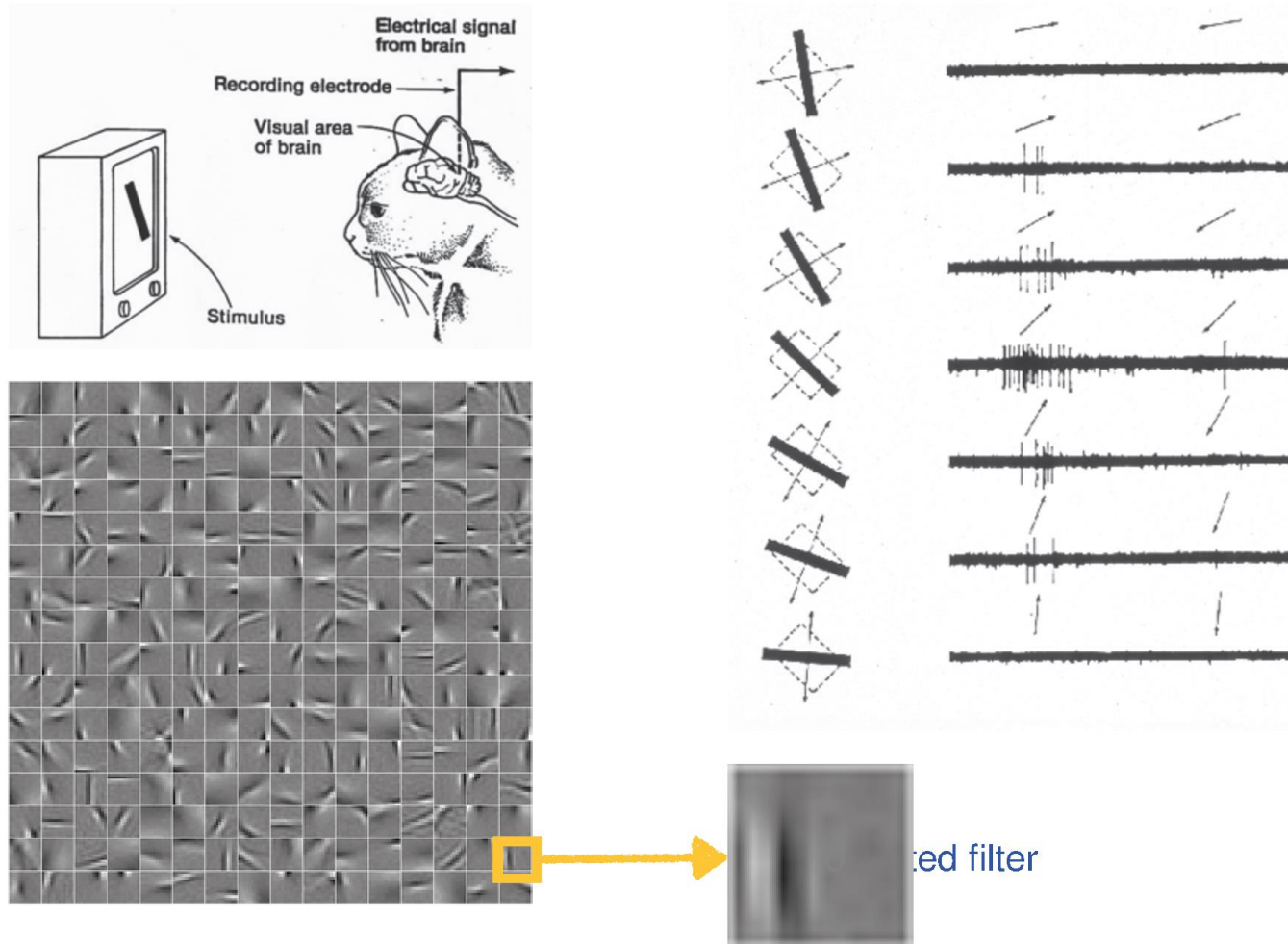
Pretože dokáže konvertovať $(-\infty, +\infty)$ na pravdepodobnostnú hodnotu $(0,1)$



$$S(z) = \frac{1}{1 + e^{-z}}$$

Objav orientovaných buniek vo vizuálnom kortexe

[Hubel and Wiesel 59]

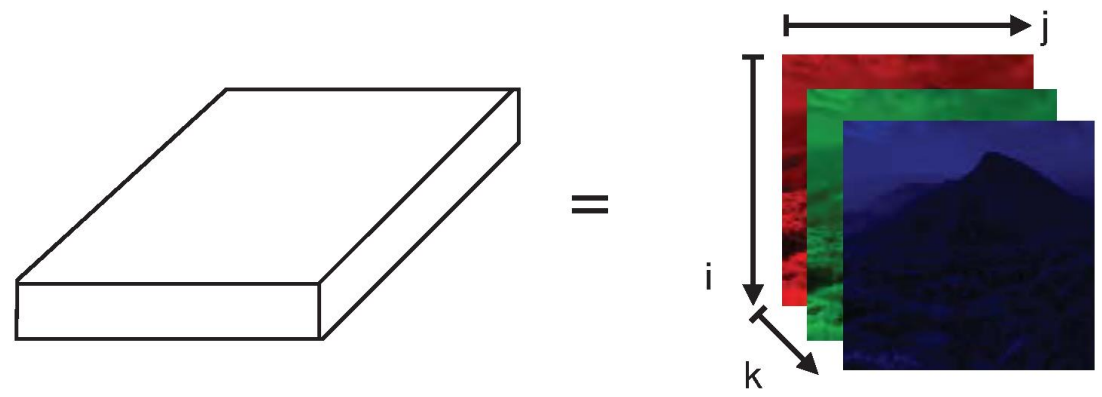
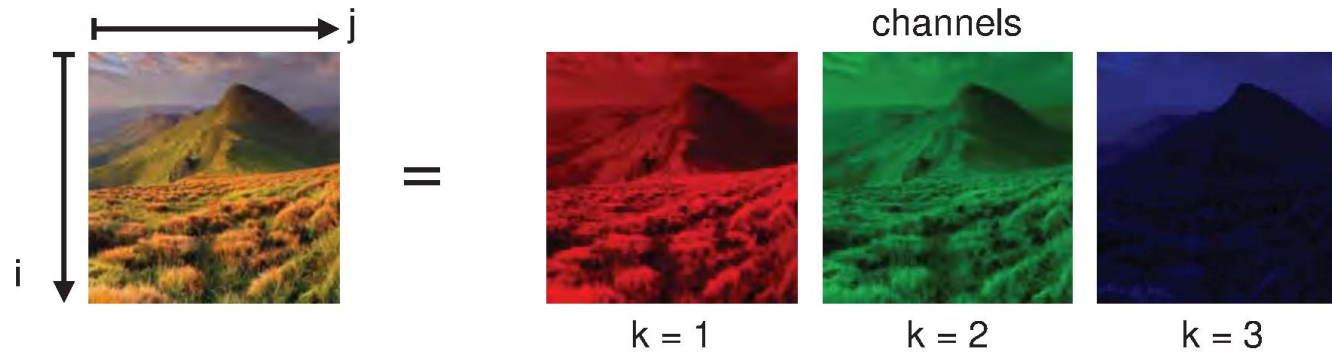






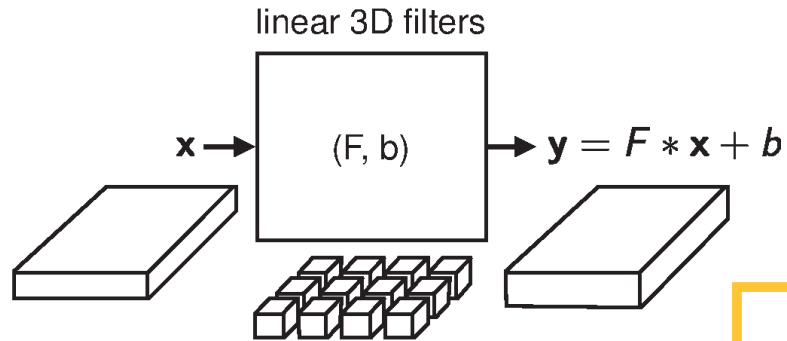
Zápis pri spracovaní signálu

Dáta sú reprezentované ako vektorové polia.



Lineárna konvolúcia

Banka „3D“ lineárnych filtrov.

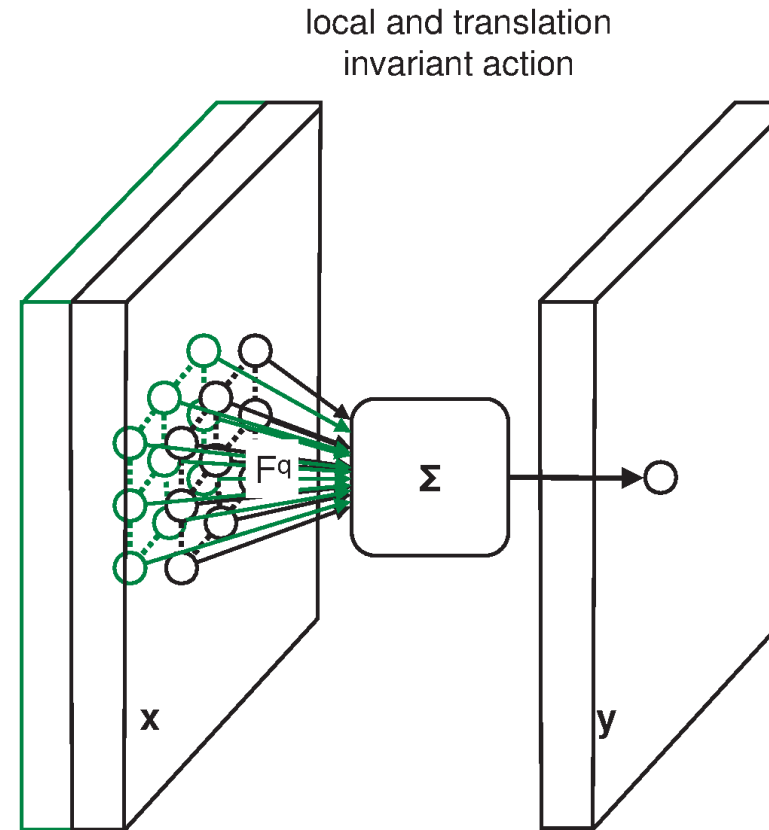
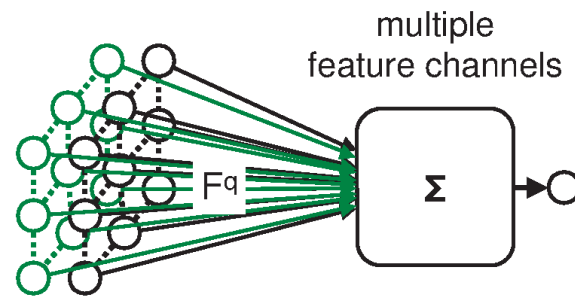
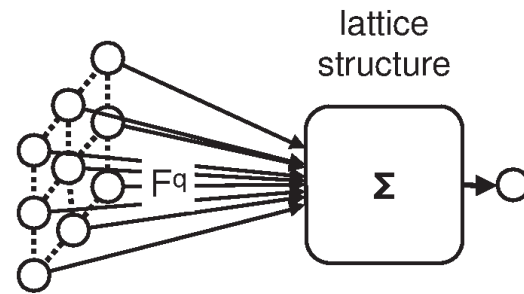
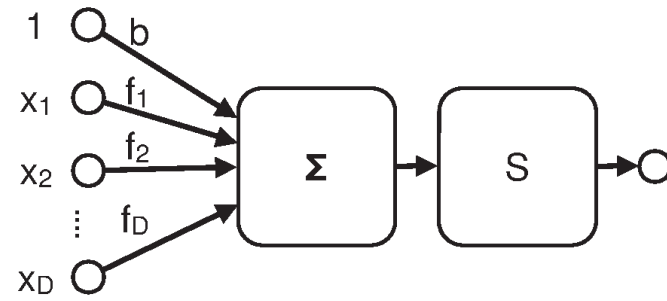


$$y_{ijq} = b_q + \sum_{u=0}^{H-1} \sum_{v=0}^{W-1} \sum_{k=1}^K x_{u+i, v+j, k} f_{u, v, k, q}$$

- Vstup $x = H \times W \times K$ pole
- Banka filtrov $F = H' \times W' \times K \times Q$ pole
- Výstup $y = (H - H' + 1) \times (W - W' + 1) \times Q$ pole

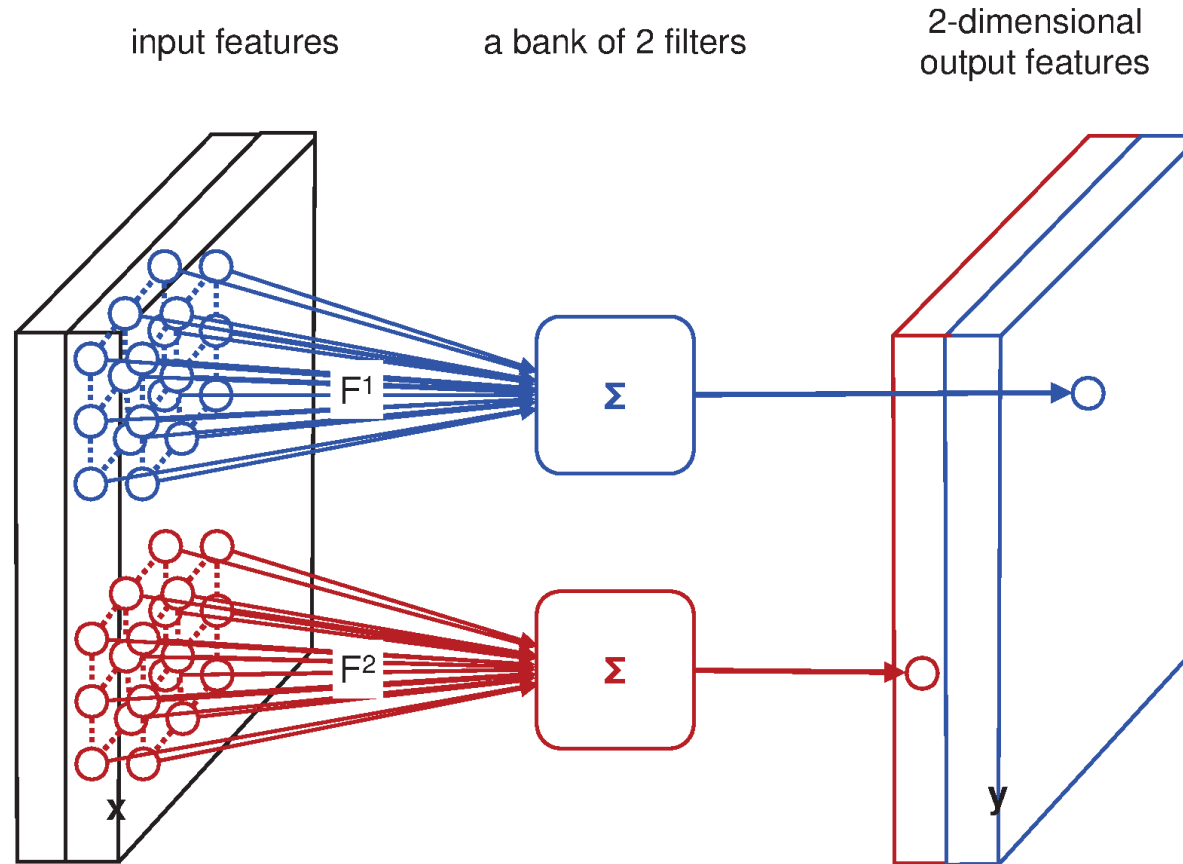
Lineárna konvolúcia

Ako neurónová sieť

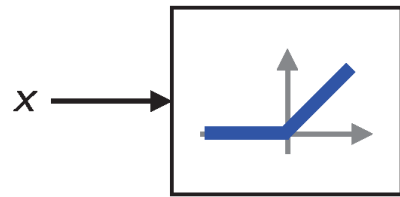


Lineárna konvolúcia

Ako neurónová sieť



Aktivačné funkcie



$$y = \frac{1}{1 + e^{-x}}$$

sigmoid

$$y = \tanh(x)$$

hyperb. tan

$$y = \max\{0, x\}$$

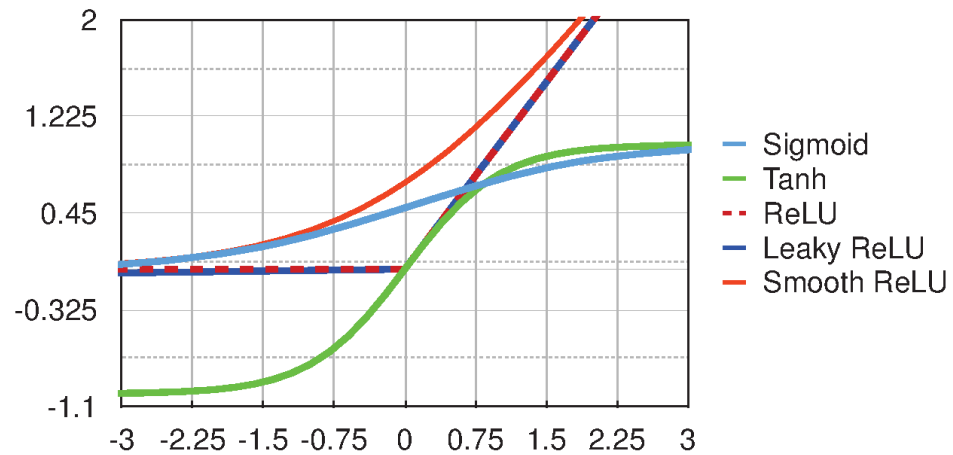
ReLU

$$y = \log(1 + e^x)$$

Soft ReLU

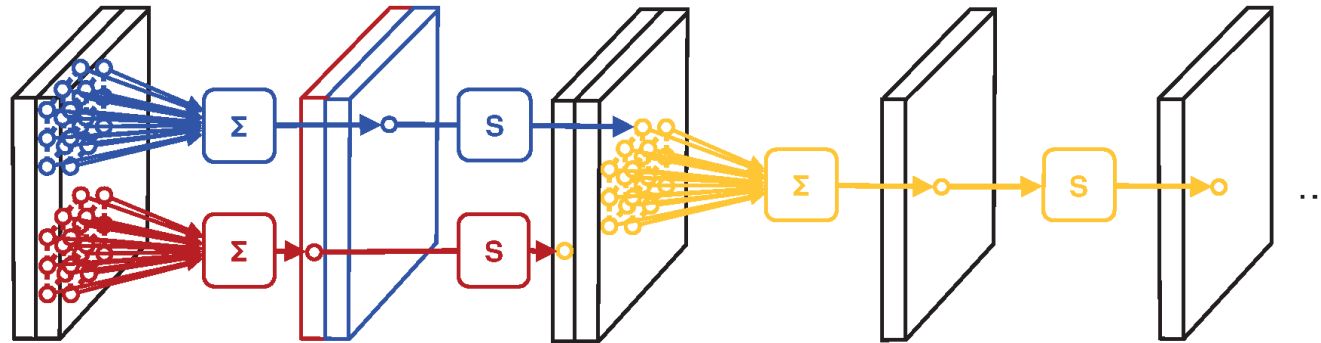
$$y = \epsilon x + (1 - \epsilon) \max\{0, x\}$$

Leaky ReLU



Viaceré vrstvy

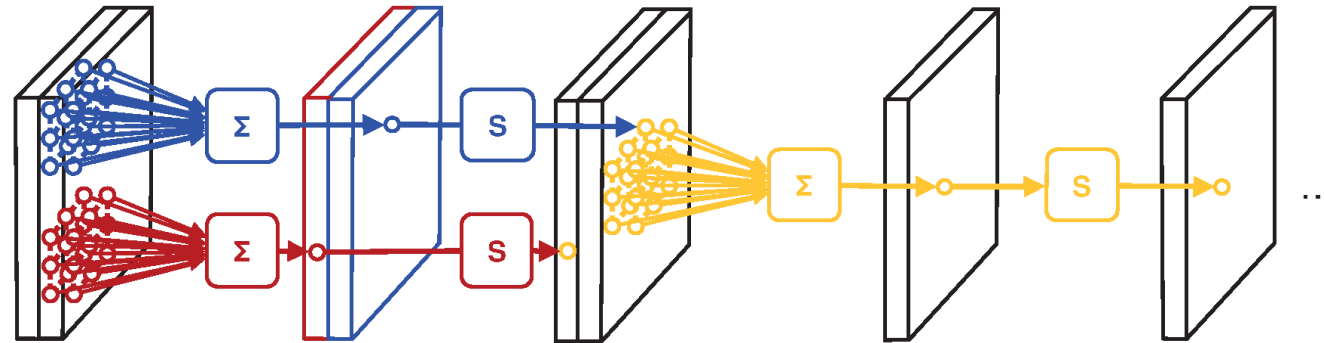
Konvolúcia, priepustnosť, konvolúcia,...



Filtre sú dopĺňované nelineárnymi operátormi (aktivačné funkcie)

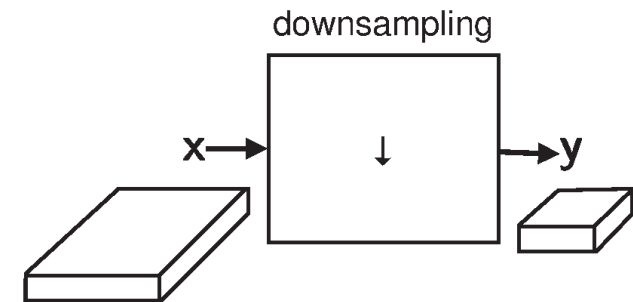
Viaceré vrstvy

Downsampling

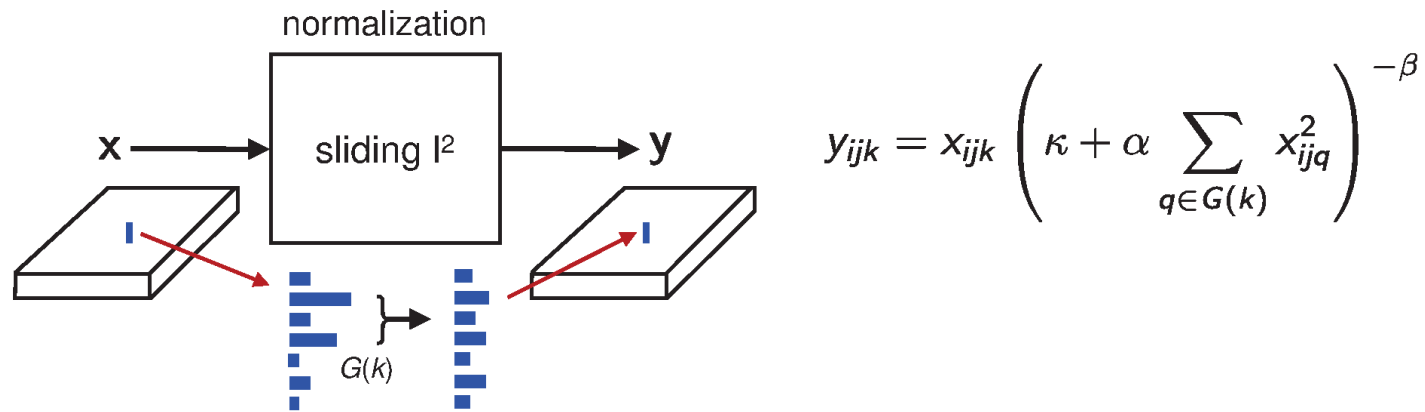


Za filtrami nasleduje downsampling.

To je zvyčajne kompenzované zvýšeným množstvom kanálov príznakov.



Normalizácia lokálnych príznačov



Normalizácia sa vykonáva po častiach.

Normalizuje sa grupa $G(k)$ príznačov v rámci jedného kanálu.

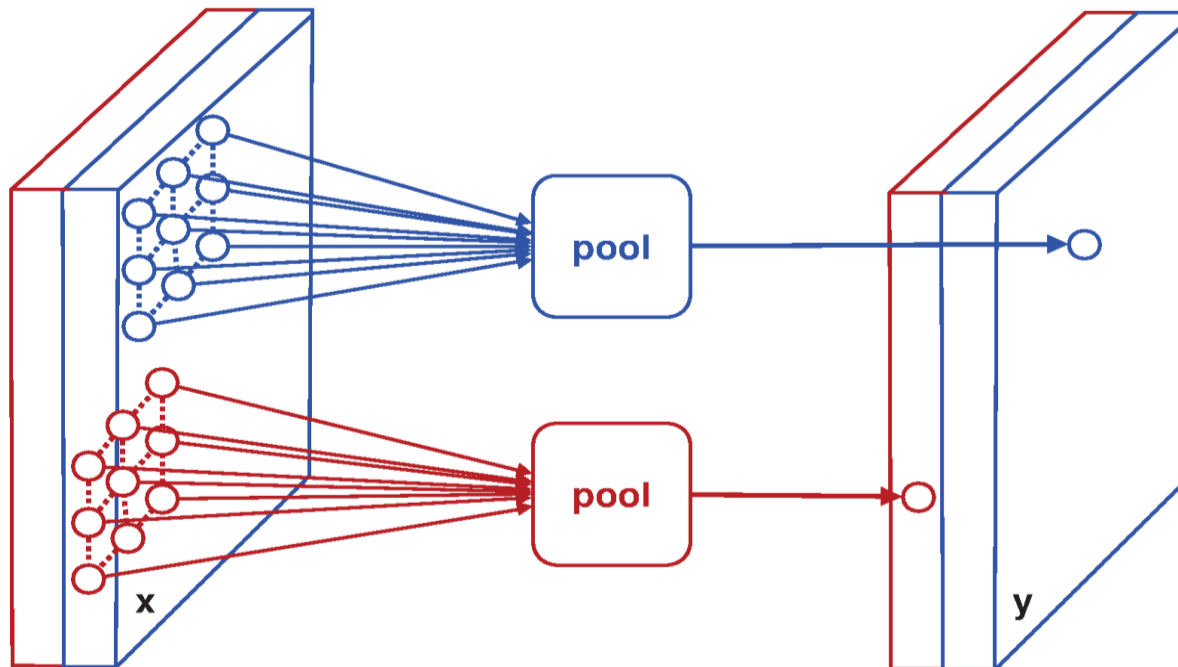
Grupy sú väčšinou definované ako klzavé okno.

Priestorový pooling

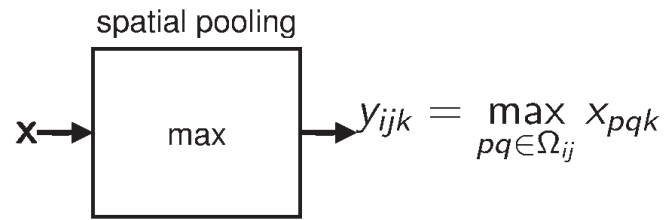
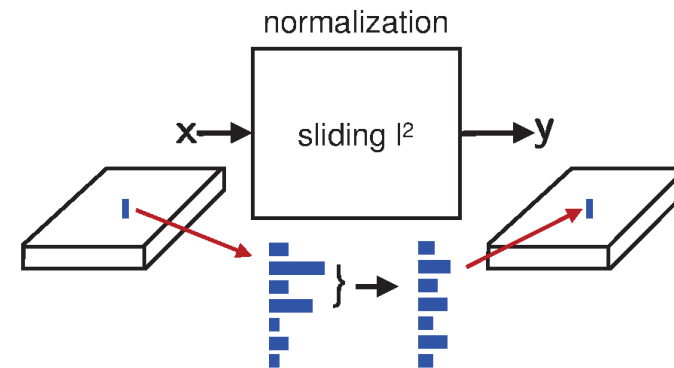
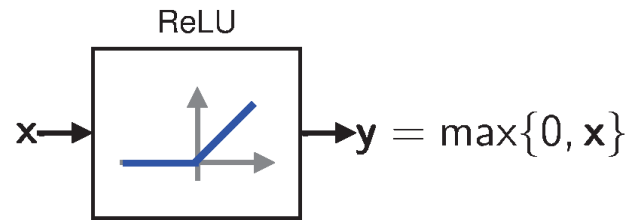
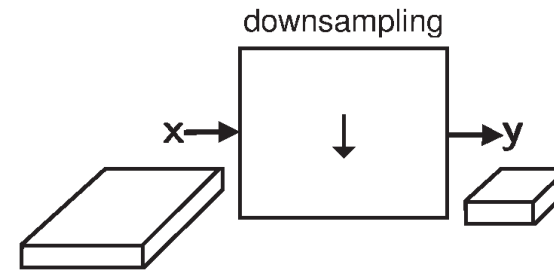
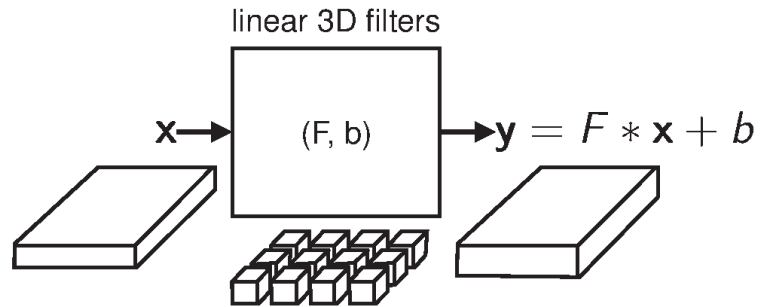
Znižuje závislosť na presných lokáciách

Pooling vyrába priemernú/maximálnu hodnotu príznaku v okolí.

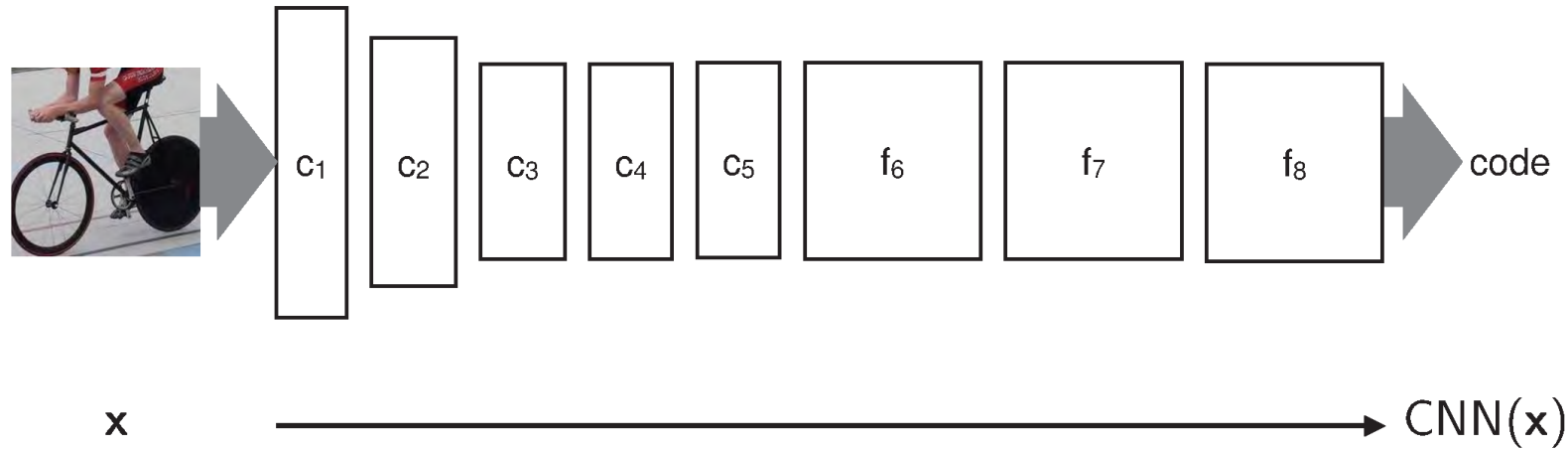
Táto hodnota sa zisťuje pre každý z kanálov.



Sumár komponentov



Konvolučná neurónová sieť



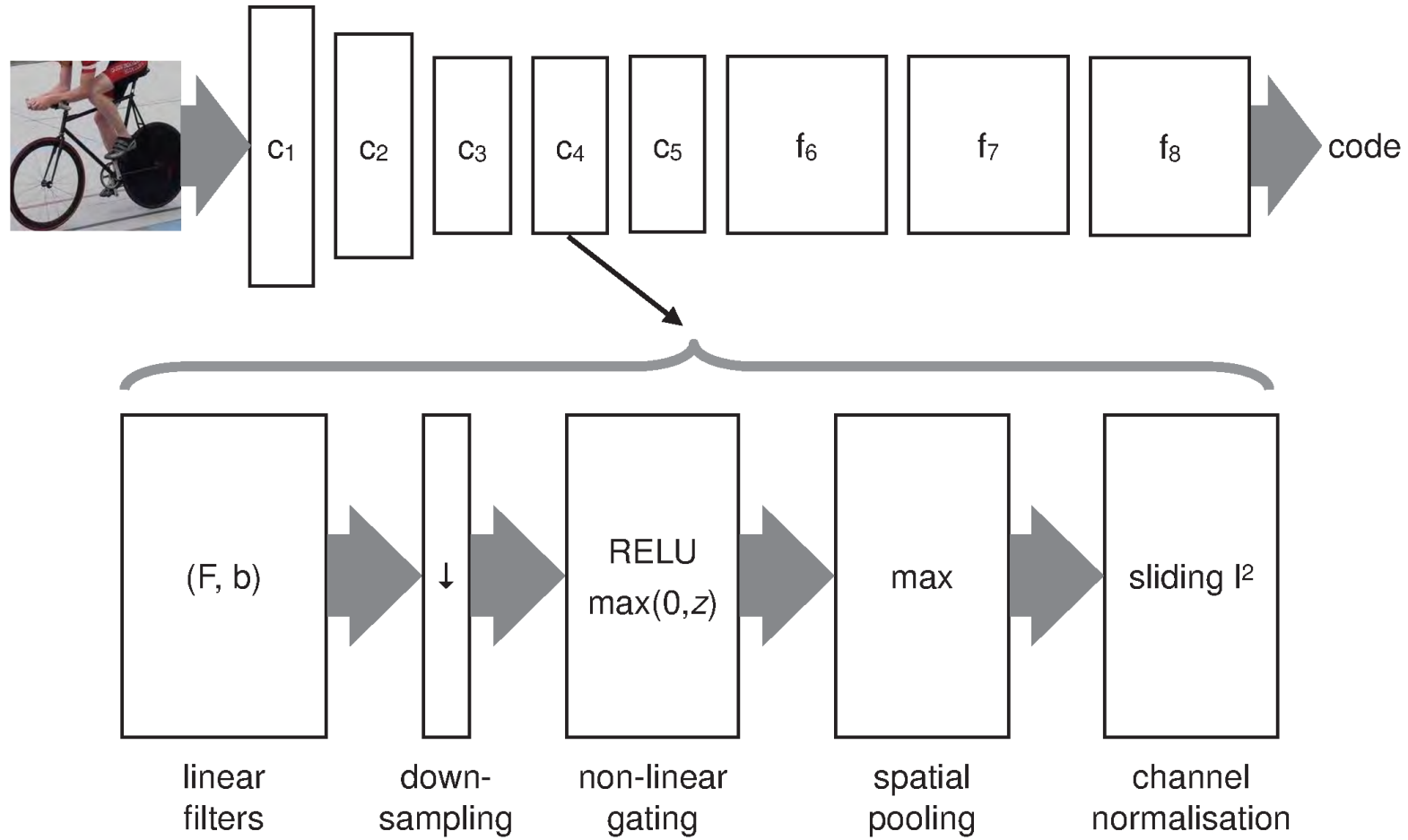
Zľava do prava:

- Znižuje sa rozlíšenie (veľkosť) obrazu
- Zvyšuje sa dimenzionalita príznakov

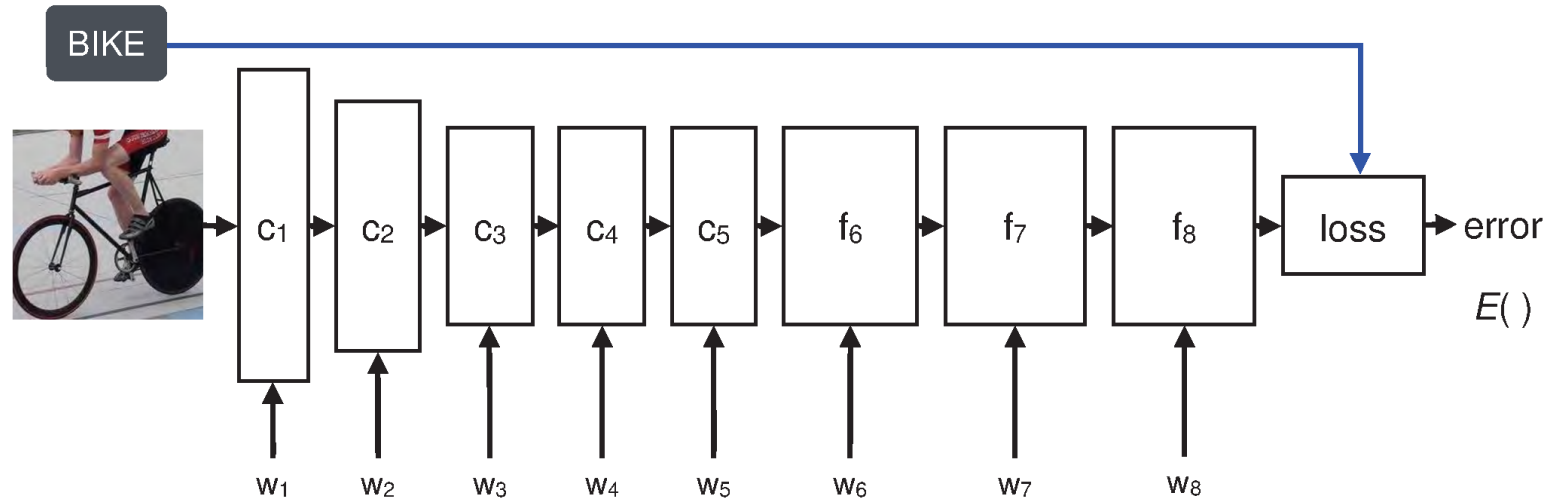
Plne prepojené vrstvy

- Rovnaké ako konvolučné, ale priestorové rozloženie je 1×1
- Obsahuje väčšinou parametre

Konvolučné vrstvy



Učenie sa CNN



$$\operatorname{argmin} E(\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_8)$$

Stochastický gradient klesá

Učenie sa CNN klasifikátora

Výzva:

- Veľa parametrov, tendencia pretrénovať

Kľúčové prísady

- Veľké množstvo (označených) dát
- Silná regularizácia (dropout)
- Stochastické klesanie gradientu
- GPU



Trénovací čas

- Cca. 90 epoch
- Dni – týždne tréovania
- Rýchlosť spracovania cca. 150-300 snímok za sekundu

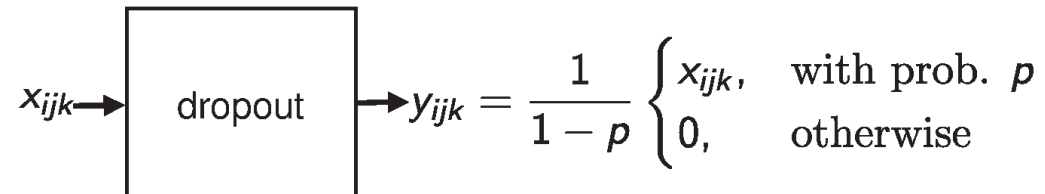
Dropout

Bežné problémy:

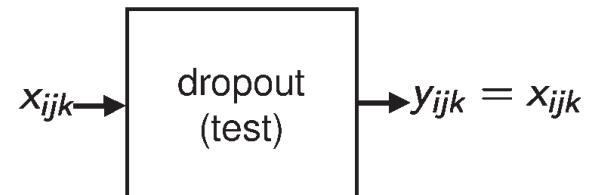
- Underfitting (lokálne maximum)
- Overfitting (kvôli veľkému počtu parametrov)

Dropout regularizácia:

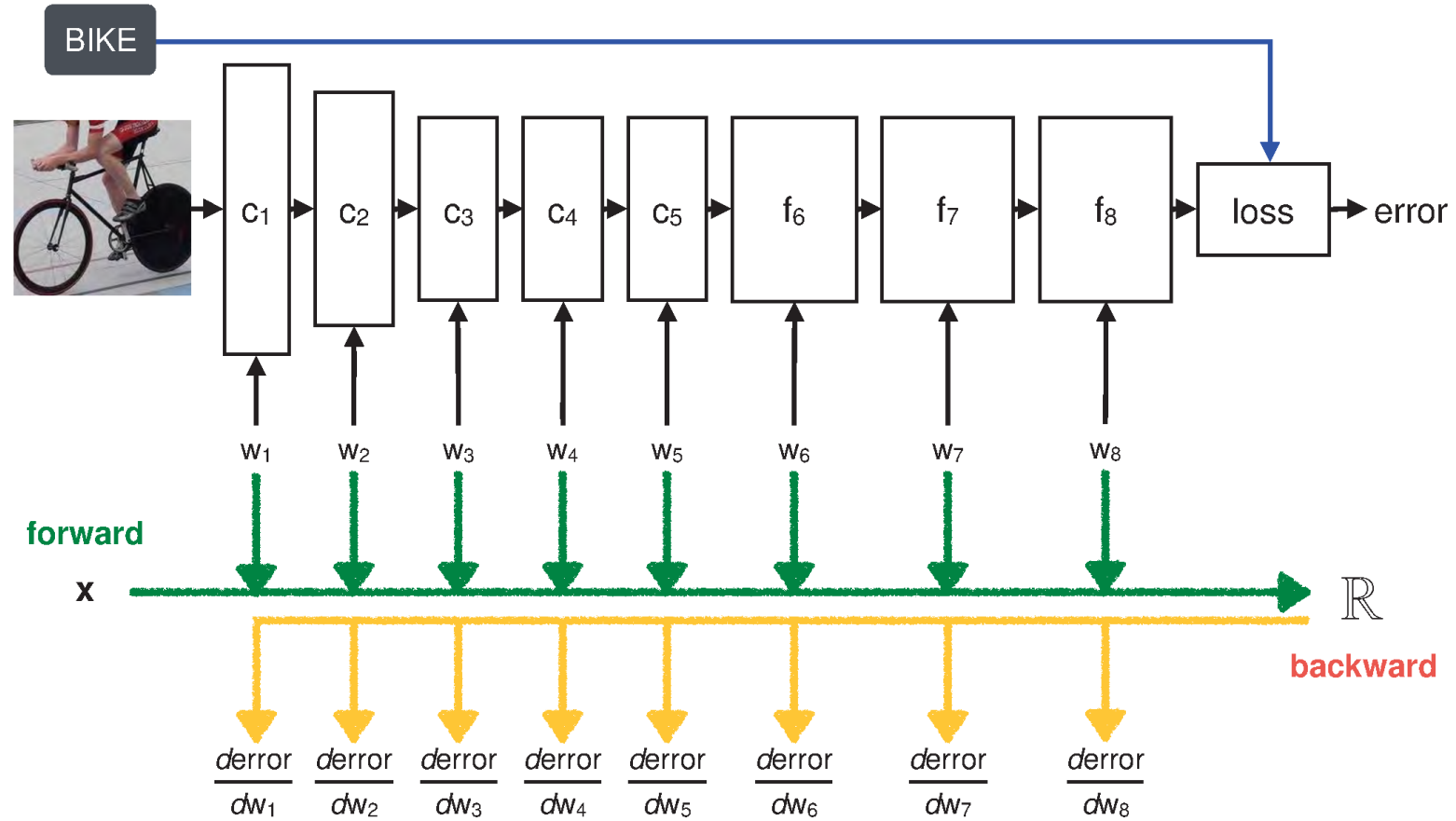
- Pridávame ju medzi vrstvy (zvyčajne plne-prepojené)
- Náhodne vypína neuróny



Pri testovaní je regularizácia pozmenená.



Spätne šírenie chýb

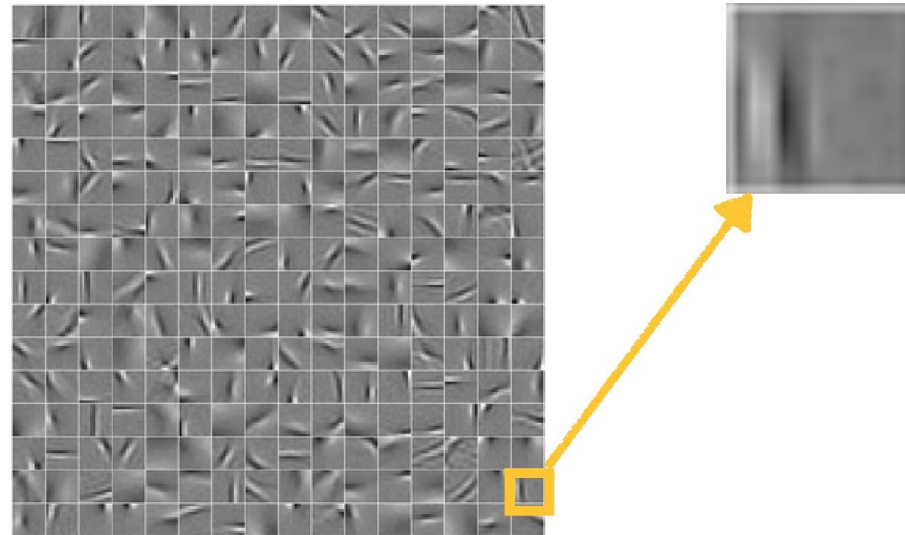


Lineárna konvolúcia

Banka z 256 filtrov (naučená na dátach)

Každý z filtrov je 1D (šedotónový obraz)

Veľkosť 16x16



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

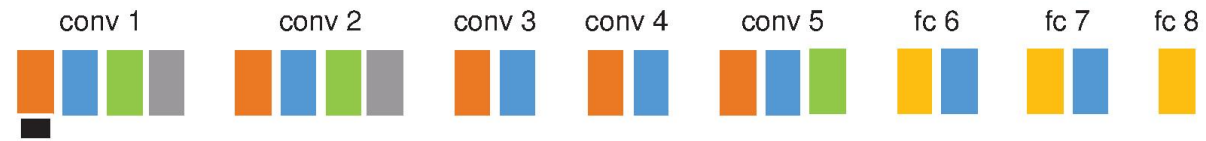


Konvolučné neurónové siete

Úvod do problematiky

Vizualizácia

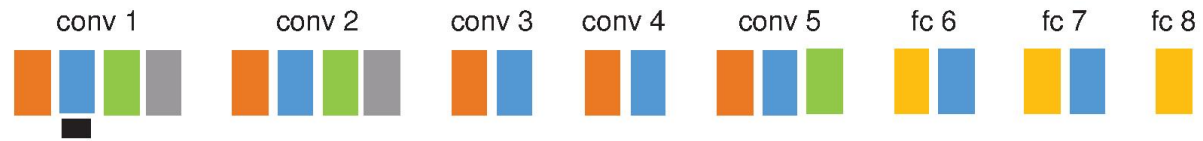
Invertné zobrazenie hlbokéj konvolučnej neurónové siete



Original
Image



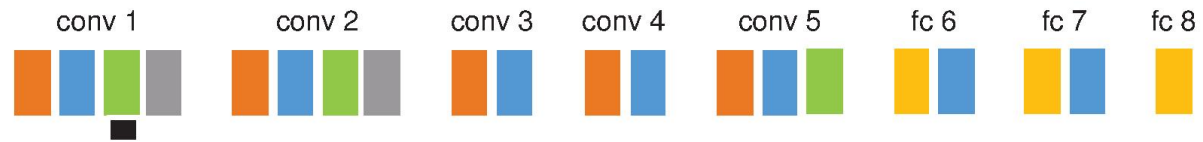
Invertné zobrazenie hlbokoj konvolučnej neurónové siete



Original
Image



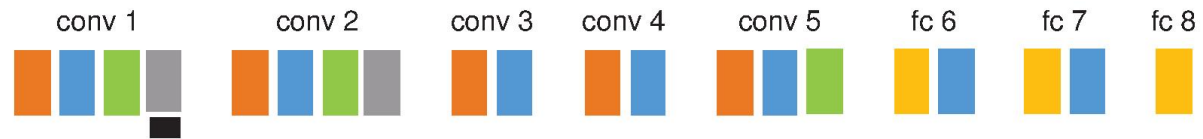
Invertné zobrazenie hlbokoj konvolučnej neurónové siete



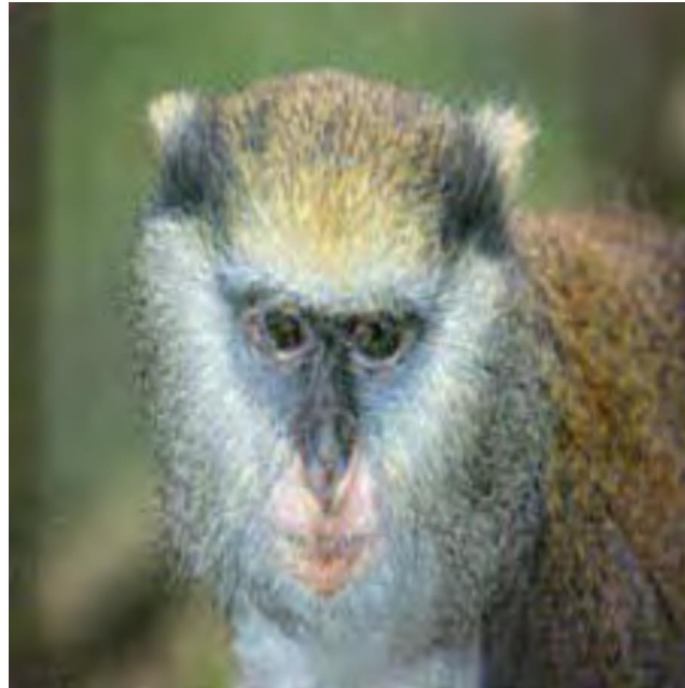
Original
Image



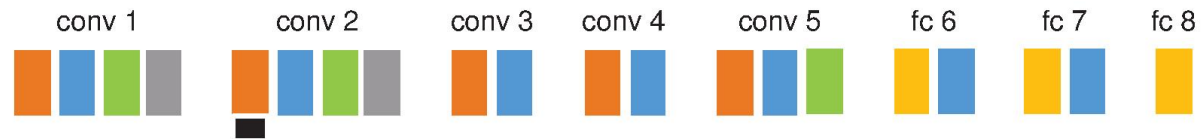
Invertné zobrazenie hlbokoj konvolučnej neurónové siete



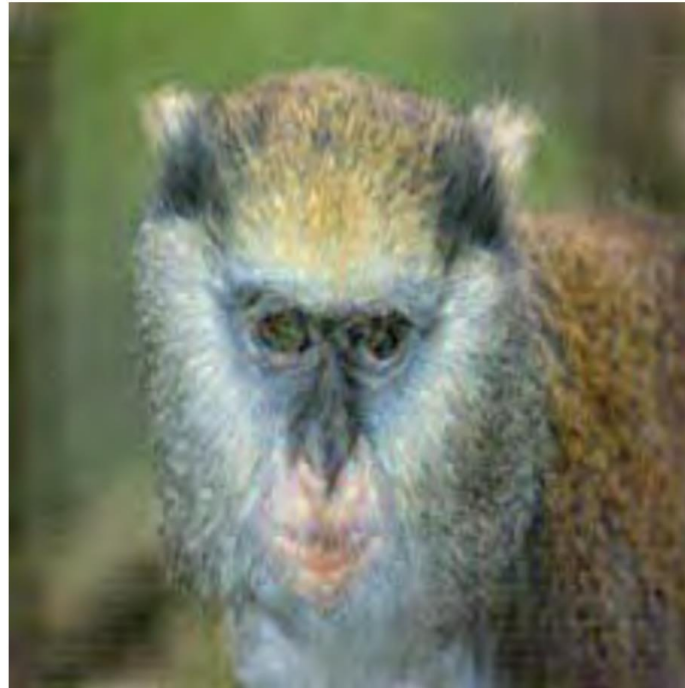
Original
Image



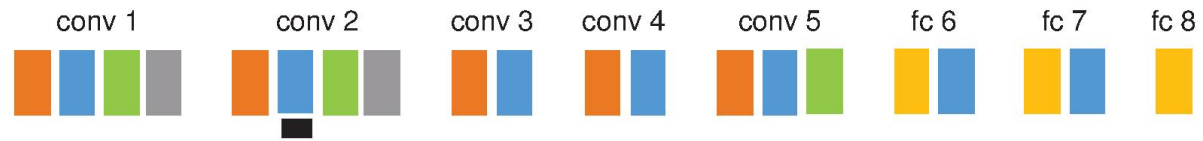
Invertné zobrazenie hlbokoj konvolučnej neurónové siete



Original
Image



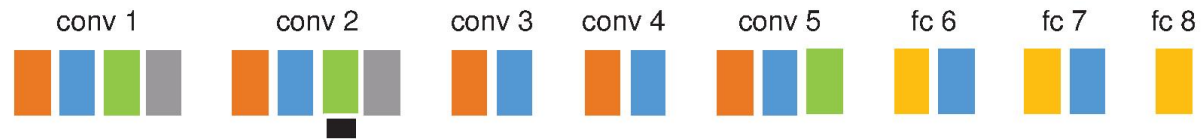
Invertné zobrazenie hlbokoj konvolučnej neurónové siete



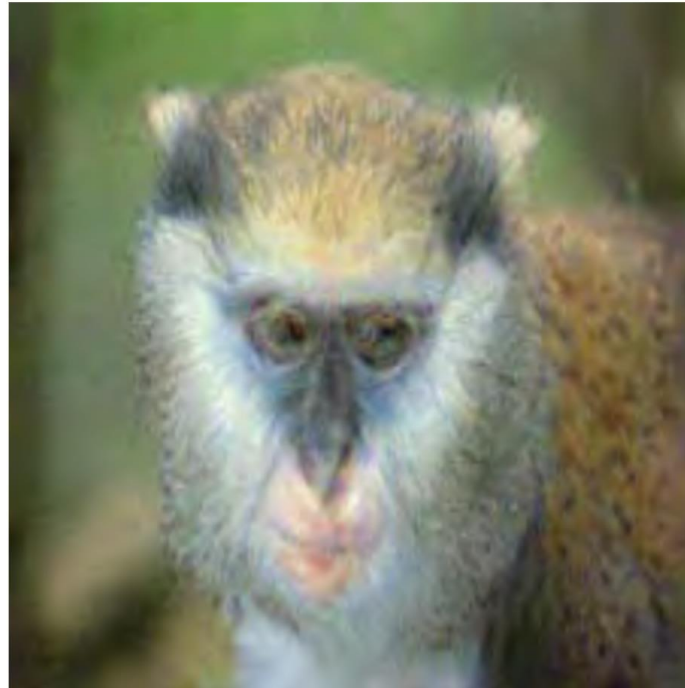
Original
Image



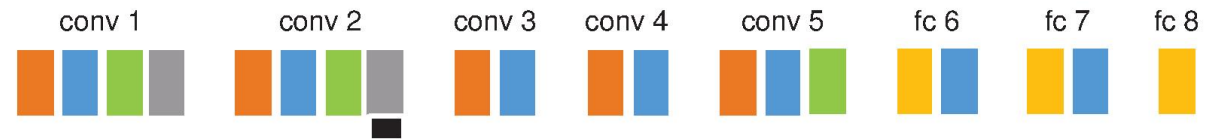
Invertné zobrazenie hlbokoj konvolučnej neurónové siete



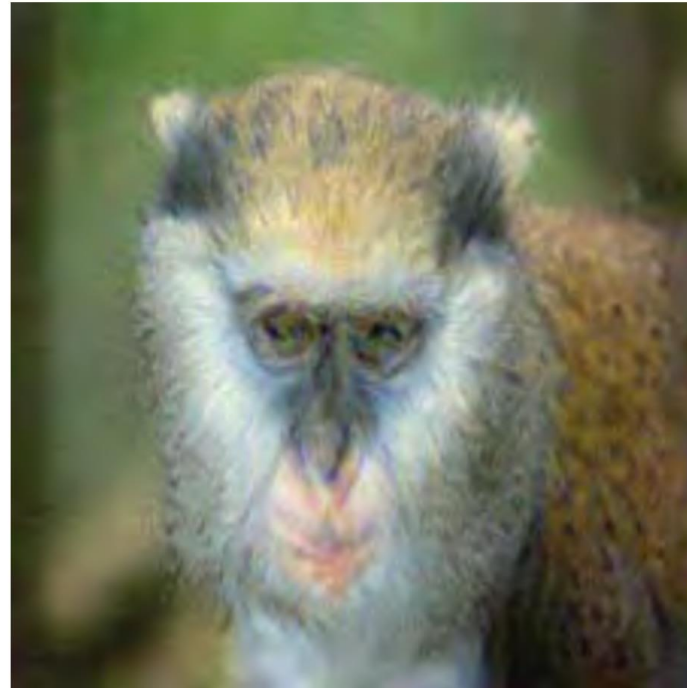
Original
Image



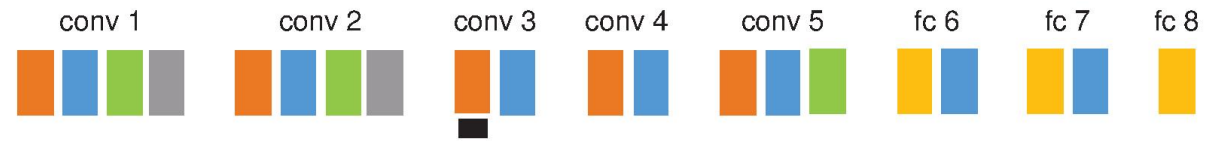
Invertné zobrazenie hlbokéj konvolučnej neurónové siete



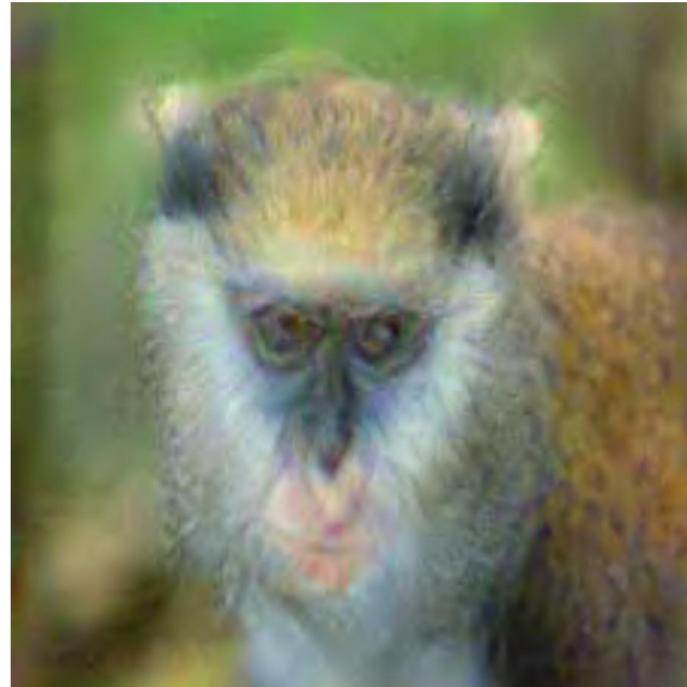
Original
Image



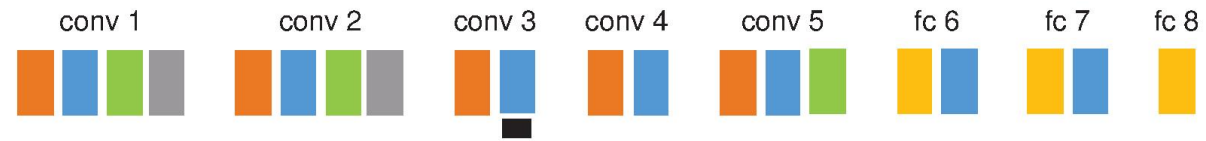
Invertné zobrazenie hlbokoj konvolučnej neurónové siete



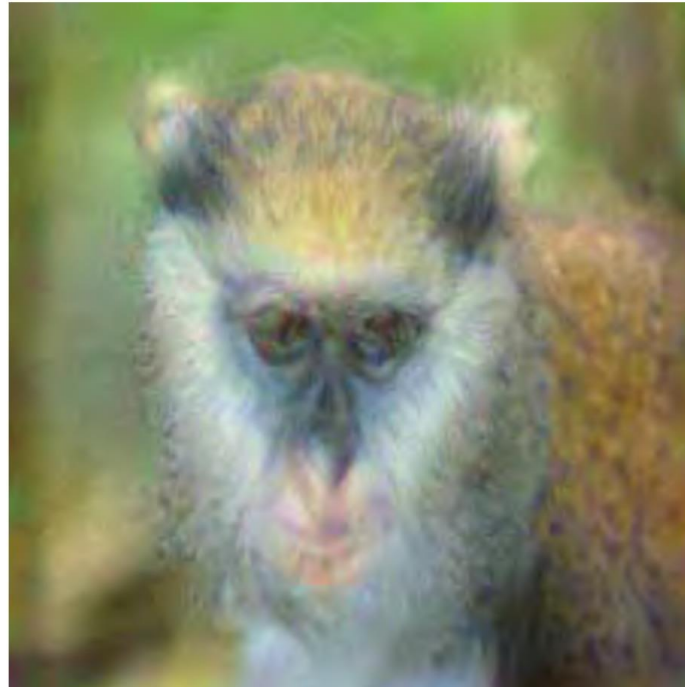
Original
Image



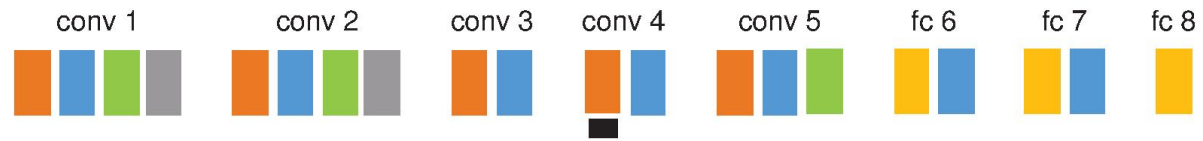
Invertné zobrazenie hlbokéj konvolučnej neurónové siete



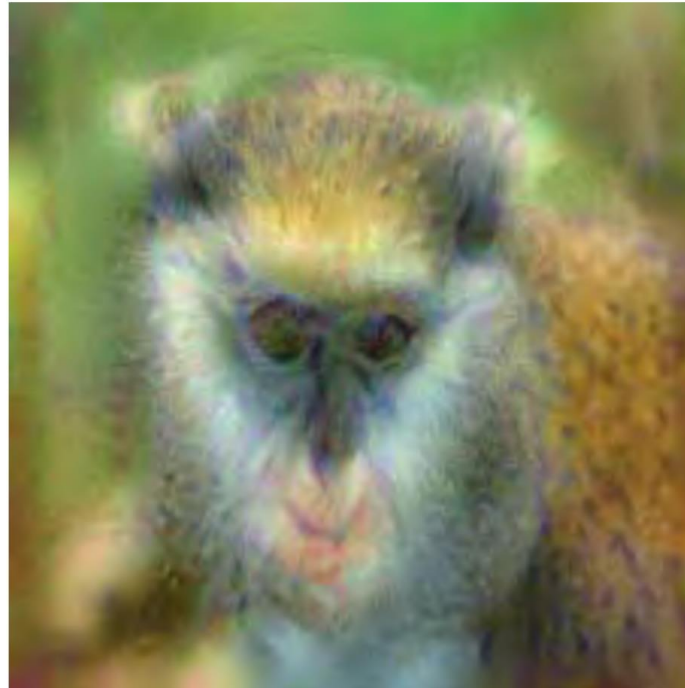
Original
Image



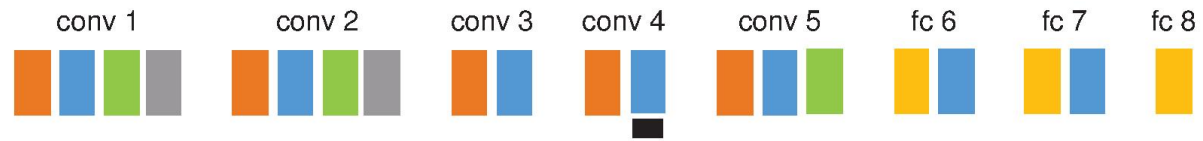
Invertné zobrazenie hlbokoj konvolučnej neurónové siete



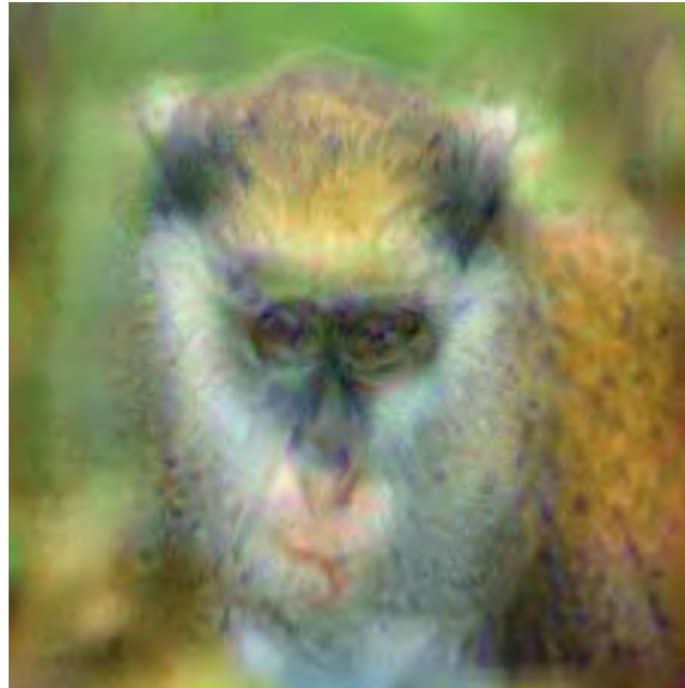
Original
Image



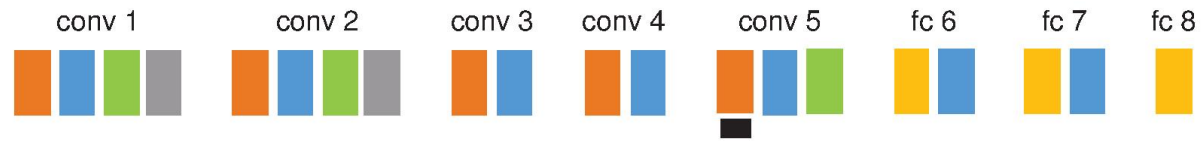
Invertné zobrazenie hlbokoj konvolučnej neurónové siete



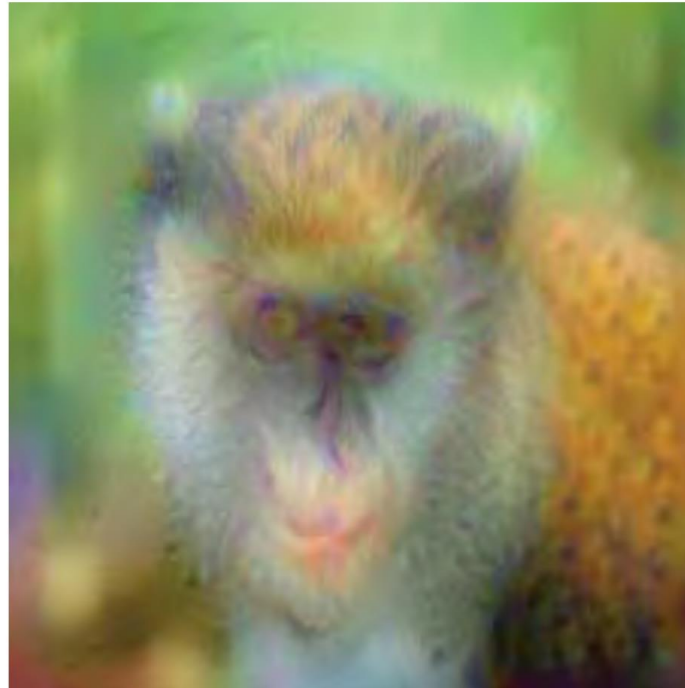
Original
Image



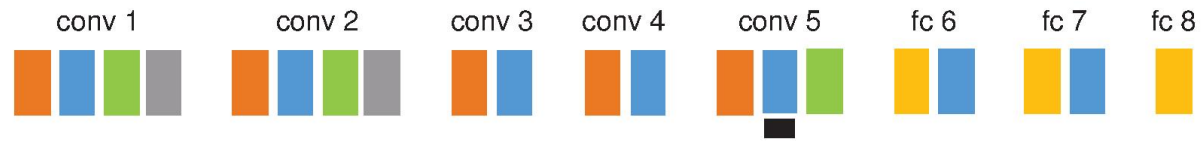
Invertné zobrazenie hlbokoj konvolučnej neurónové siete



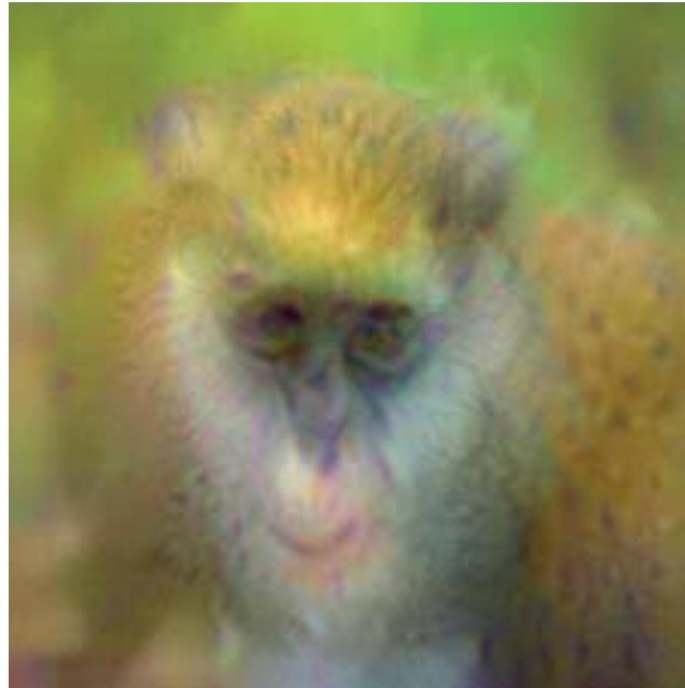
Original
Image



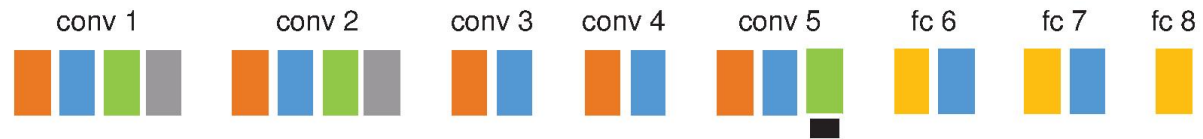
Invertné zobrazenie hlbokéj konvolučnej neurónové siete



Original
Image



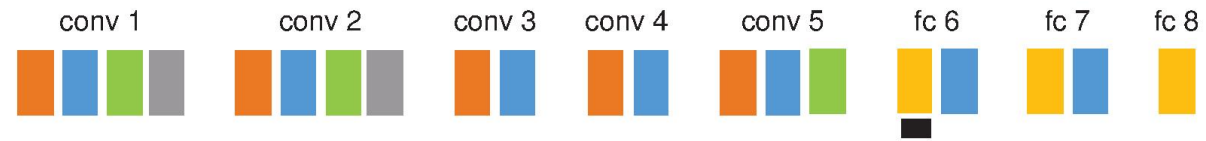
Invertné zobrazenie hlbokoj konvolučnej neurónové siete



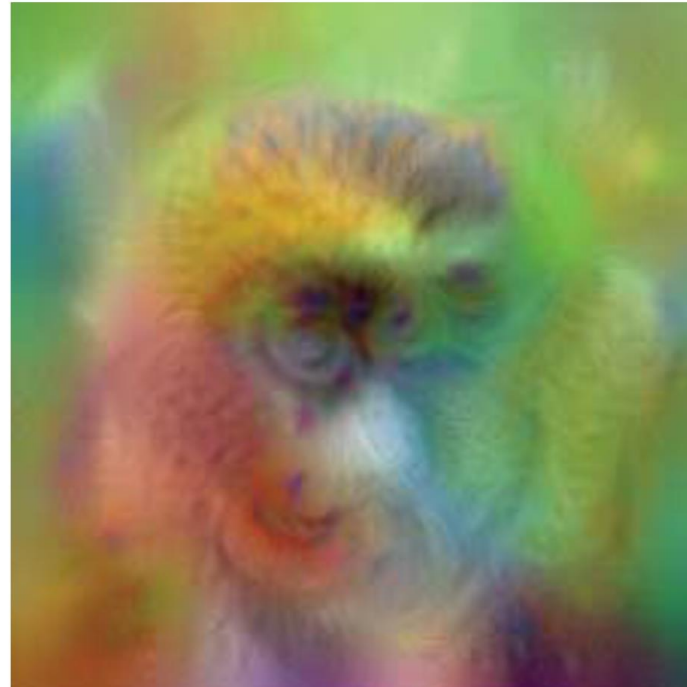
Original
Image



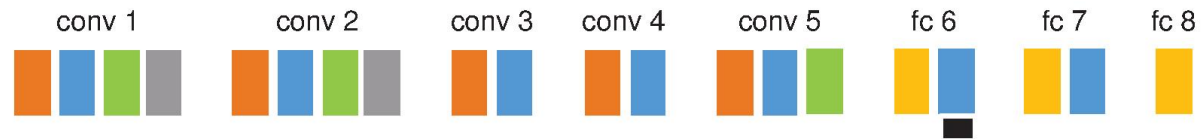
Invertné zobrazenie hlbokéj konvolučnej neurónové siete



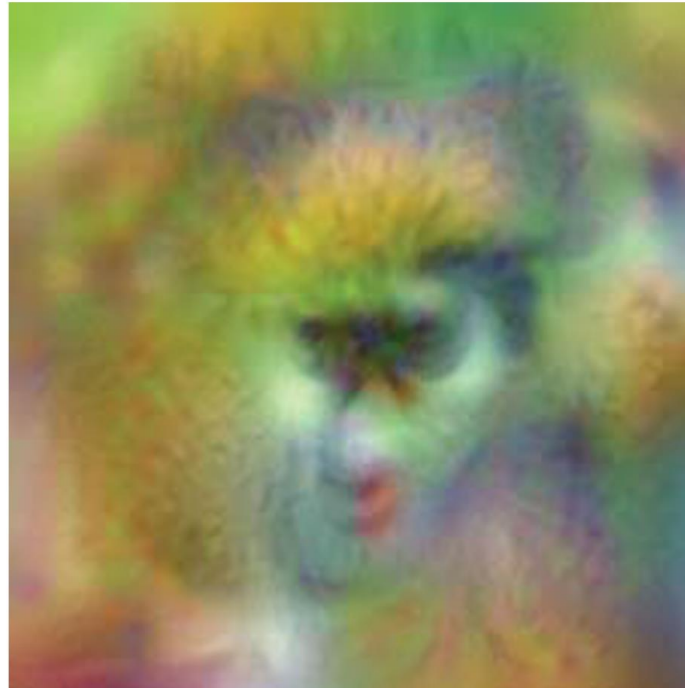
Original
Image



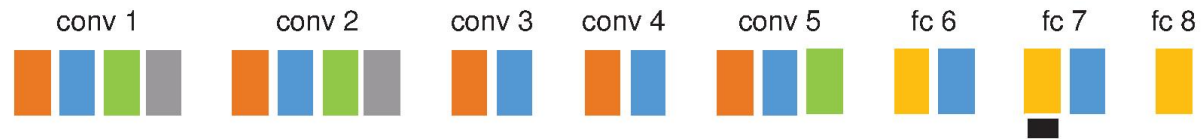
Invertné zobrazenie hlbokoj konvolučnej neurónové siete



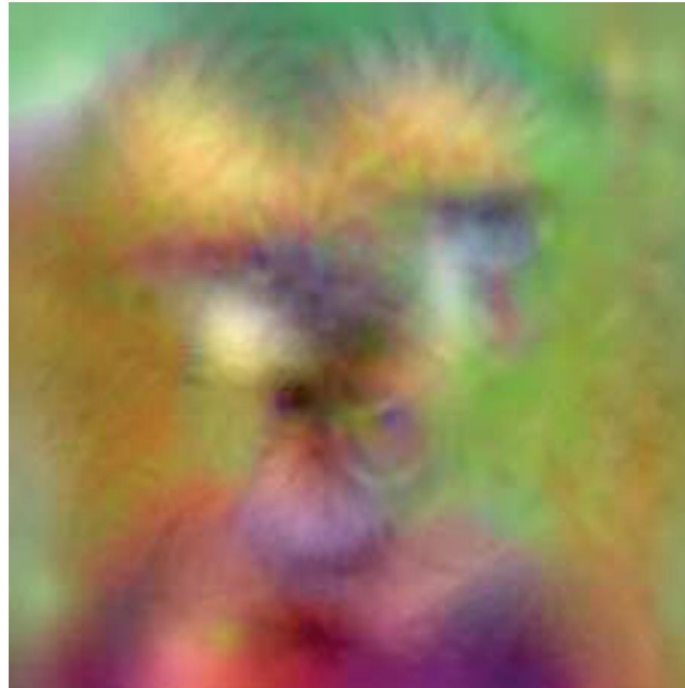
Original
Image



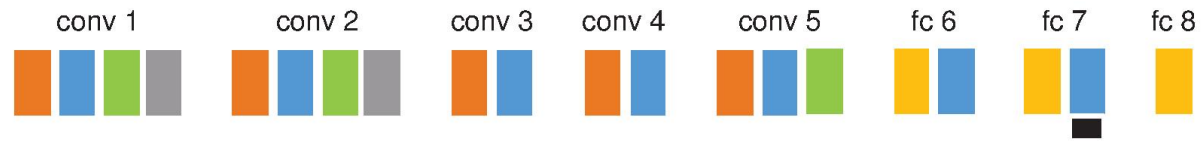
Invertné zobrazenie hlbokéj konvolučnej neurónovej siete



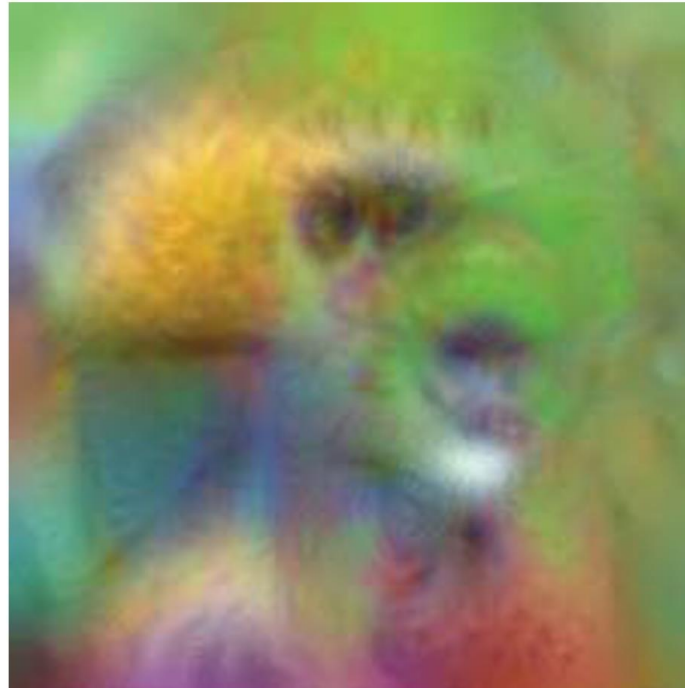
Original
Image



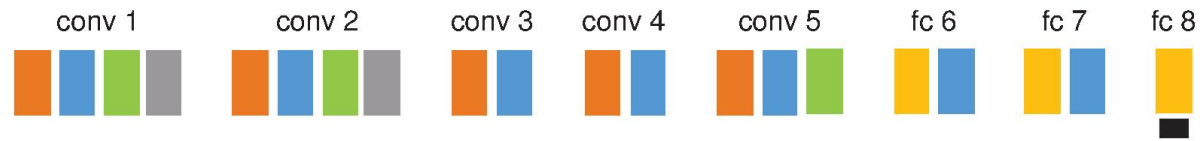
Invertné zobrazenie hlbokoj konvolučnej neurónové siete



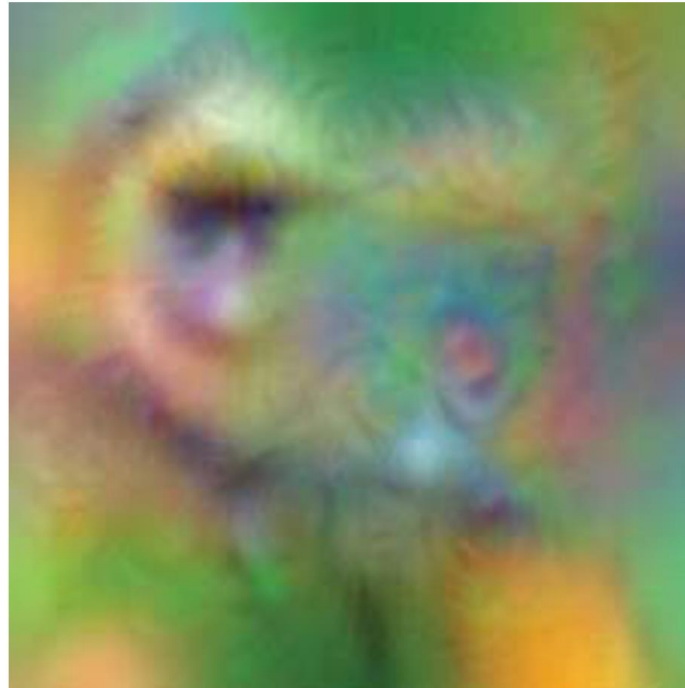
Original
Image



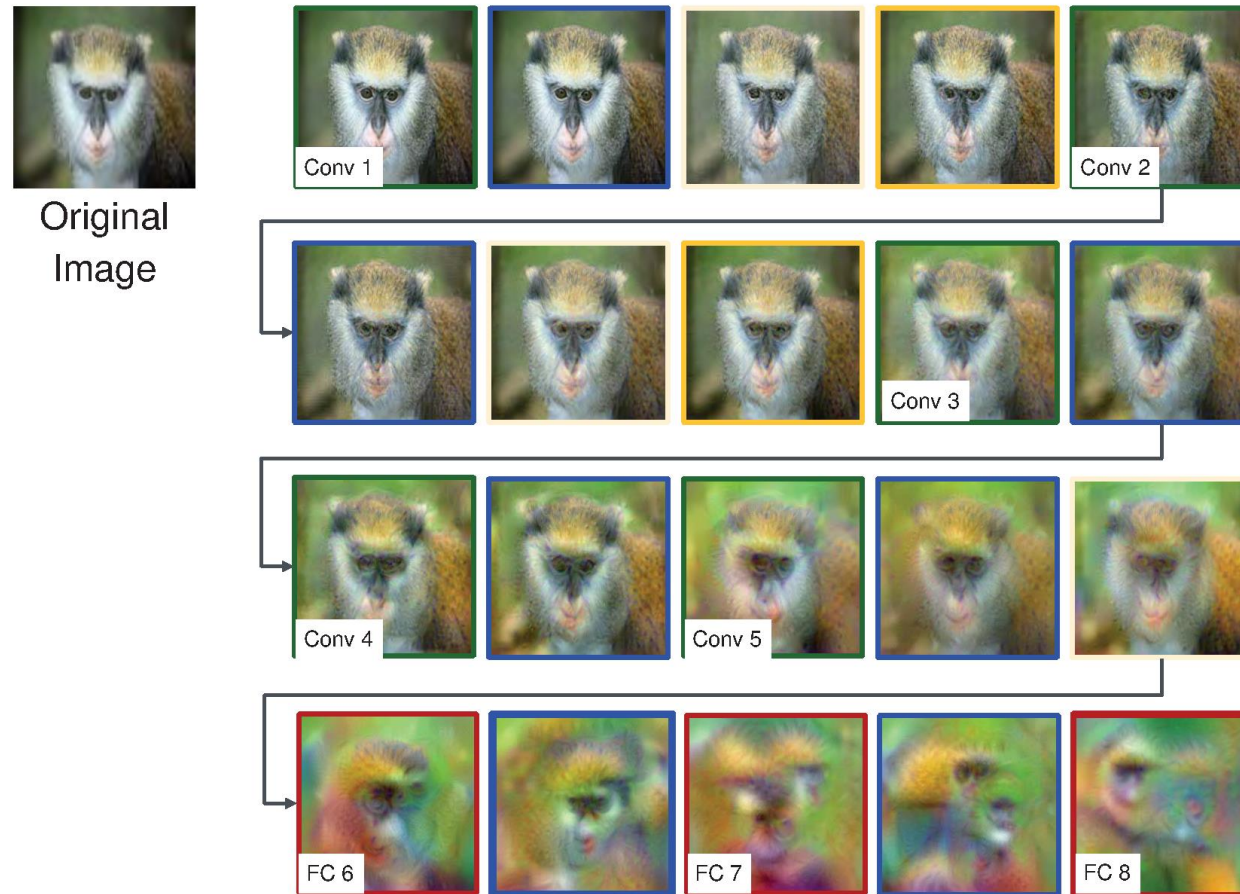
Invertné zobrazenie hlbokoj konvolučnej neurónové siete



Original
Image



Invertné zobrazenie hlbokéj konvolučnej neurónovej siete



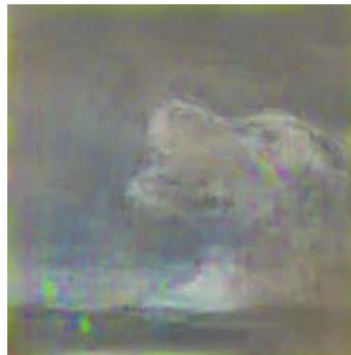
Looking Across Frames in a Video

Original Video



By James Drake
<https://vimeo.com/29132743>

Temporally-smooth
inverse



Temporally-indep.
inverse



conv 1

conv 3

Invertné zobrazenie hlbokéj konvolučnej neurónové siete

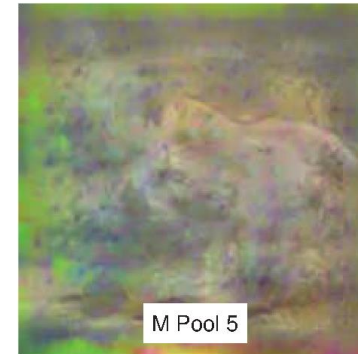
Original Video



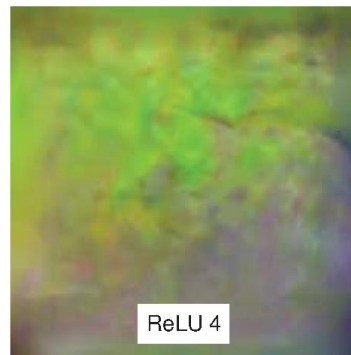
Temporally-independent
inverse



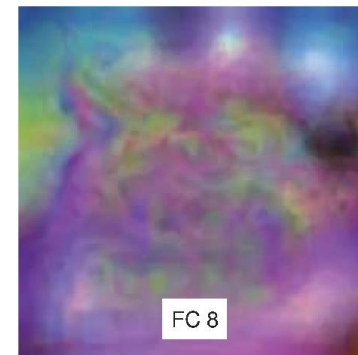
Conv 2



M Pool 5

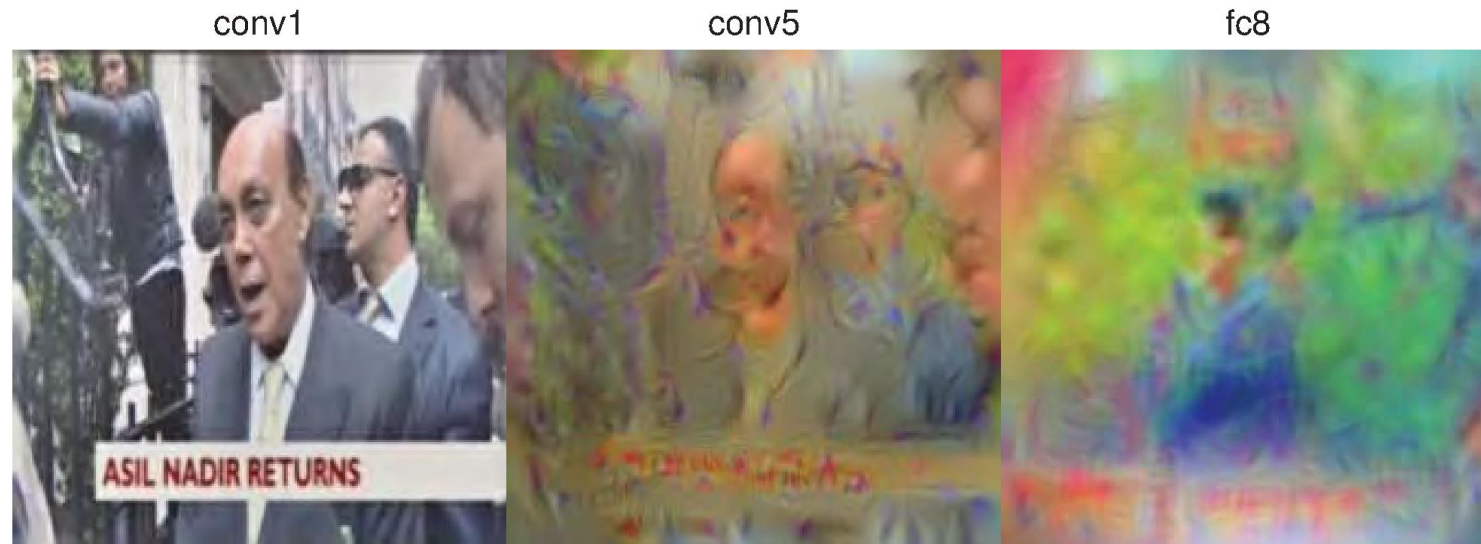


ReLU 4



FC 8

Invertné zobrazenie hlbokej konvolučnej neurónové siete



Invertné zobrazenie hlbokoj konvolučnej neurónové siete



Deep inside CNNs
(aka deep dreams)
[Simonyan et al. 2014]

Invertné zobrazenie hlbokej konvolučnej neurónové siete

Deep inside CNNs (aka deep dreams)
[Simonyan et al. 2014]



Plus TV regularis + jitter
[Mahendran et al. 2015, Google Inceptionism]



Invertné zobrazenie hlbokej konvolučnej neurónové siete



Invertné zobrazenie hlbokéj konvolučnej neurónové siete



Invertné zobrazenie hlbokkej konvolučnej neurónové siete



Invertné zobrazenie hlbokoj konvolučnej neurónové siete



Invertné zobrazenie hlbokkej konvolučnej neurónové siete



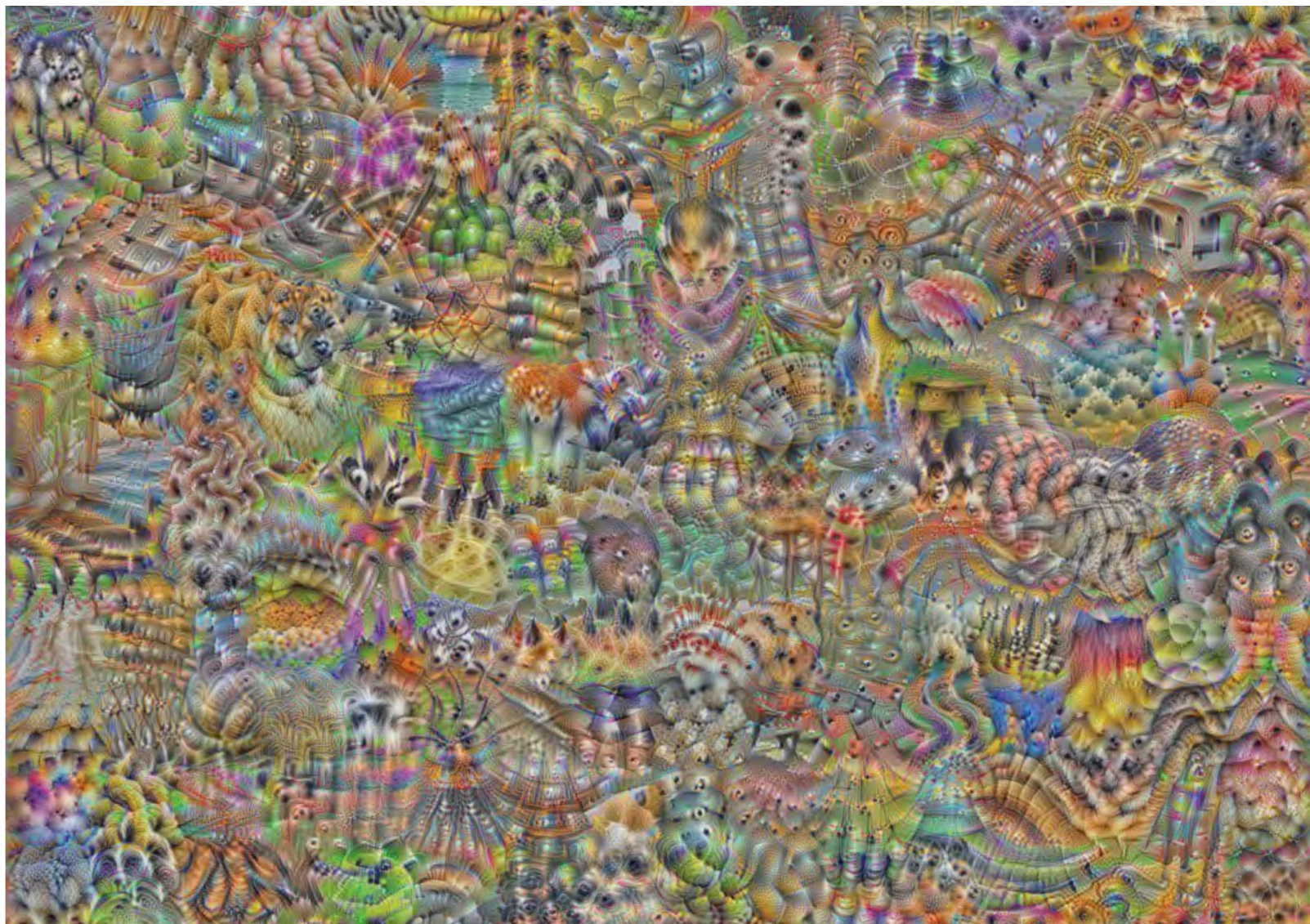
Invertné zobrazenie hlbokoj konvolučnej neurónové siete



Invertné zobrazenie hlbokej konvolučnej neurónové siete



Invertné zobrazenie hlbokéj konvolučnej neurónové siete



Zhrnutie

„Staré časy“:

- Veľa zaujímavých nástrojov
- Veľa myšlienok, nápadov a trikov

Sila hlbokých sietí

- Lepšia úspešnosť
- Všeobecná architektúra (stačí ju nakrmiť údajmi)

Top výhody

- Učenie sa všeobecnej reprezentácie
- Nie je potreba kontroly vizuálnych štruktúr

Limitácie

- Stále je tu veľa miesta na zlepšenie
- Chýbajúca teória ako tieto siete fungujú (Viete vôbec ako funguje SVM?)
- 3D geometria
-

Zdroje

- Tutorial k neurónovým sieťam: <http://ufldl.stanford.edu/tutorial/supervised/LinearRegression/>
- Andrea Vedaldi (prezentácia deep siete): <https://www.youtube.com/watch?v=pb9oNDS7Cj4>
- Matthew Zeiler: Hierarchical Convolutional Deep Learning in Computer Vision , ICVSS2015, Sicily

- zaujímavé nástroje:
 - TensorFlow (<https://www.tensorflow.org/>)
 - Theano
 - Caffe (<http://caffe.berkeleyvision.org/>)