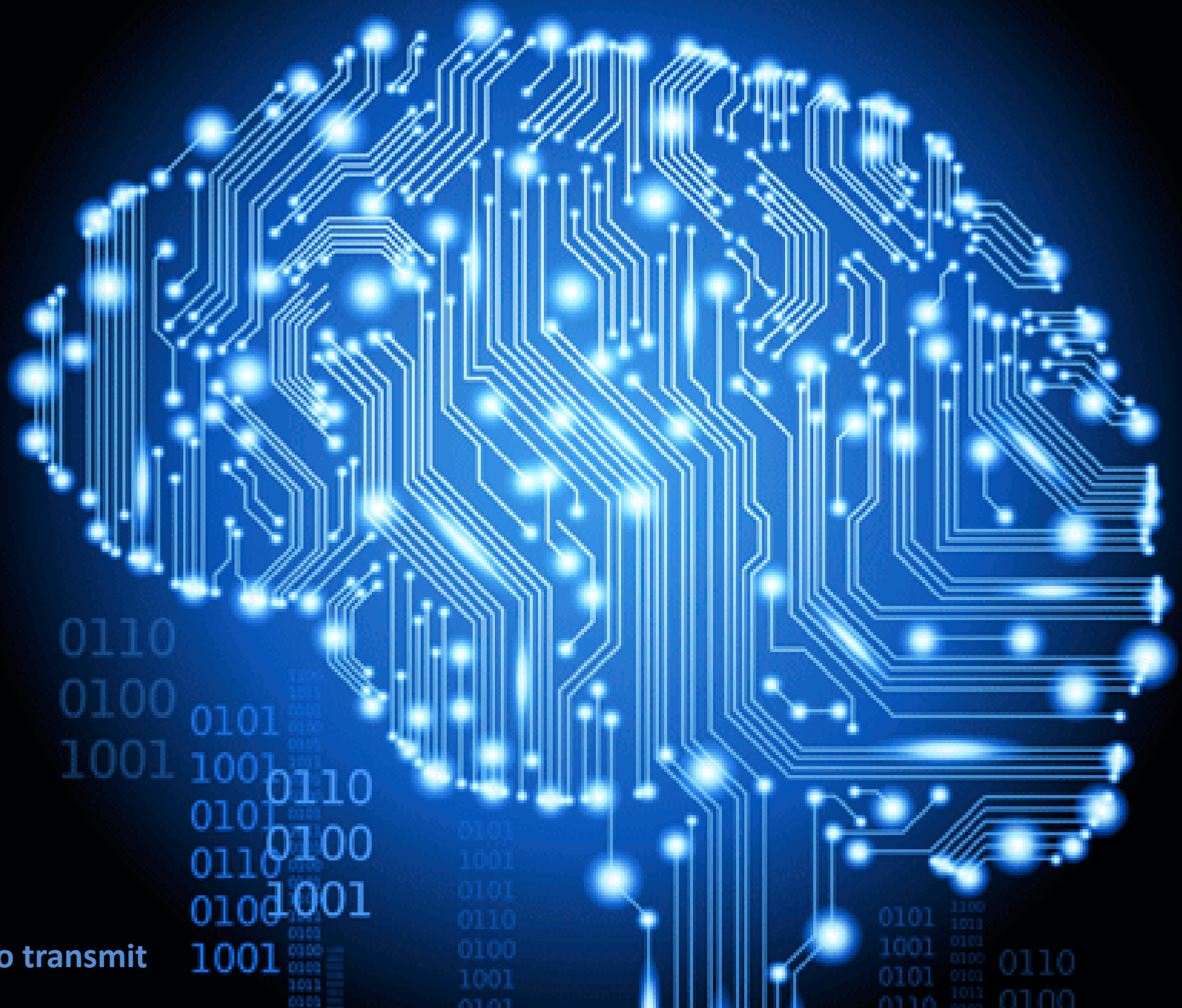




0110

0100

1001

0101

1001

0101

0110

0100

1001

0110

0100

1001

0101

1001

0101

0110

1001

0101

0101

1001

0110

1100

1011

0100

0110

0100

>ready to transmit

Konvolúciós neurális hálózatok, multimodális tanulás elméleti hátttere

Szoldán Péter

Facebook csoport

- <https://www.facebook.com/groups/496484064185906>
- Feltettem linket a konvolúciós demó weboldalra

A kép pixelekből áll

206	205	247	245
244	151	137	244
192	154	75	200
90	109	98	143



A konvolúciós mag („kernel”)

- Az alábbi példa egy 3 x 3 –as kernel.
- Ha a kernel összege 1, akkor „normál”, vagy „normalizált” a kernel

1	0	2
-3	-5	5
2	0	-1

A konvolúció művelete

- A kernel és a pixel környezetének szorzatösszege lesz az új érték

206	205	247	245
244	151	137	244
192	154	75	200
90	109	98	143

1	0	2
-3	-5	5
2	0	-1

	207		

Legegyszerűbb az identitás

- Csak a középső pixel értéke számít, a többi nem: visszakapjuk az eredeti képet
- Persze ez nem nagyon jó semmire

206	205	247	245
244	151	137	244
192	154	75	200
90	109	98	143

0	0	0
0	1	0
0	0	0

	151		

Konvolúció online demo weboldal

<http://setosa.io/ev/image-kernels/>

Szélek detektálása

- Azonos színű területnél a konvolúció eredménye 0, de ha változás van, akkor nem

0	0	255	255
0	0	255	255
0	0	255	255
0	0	255	255

0	0	0
-1	1	0
0	0	0

0	255	255	0
0	255	255	0
0	255	255	0
0	255	255	0

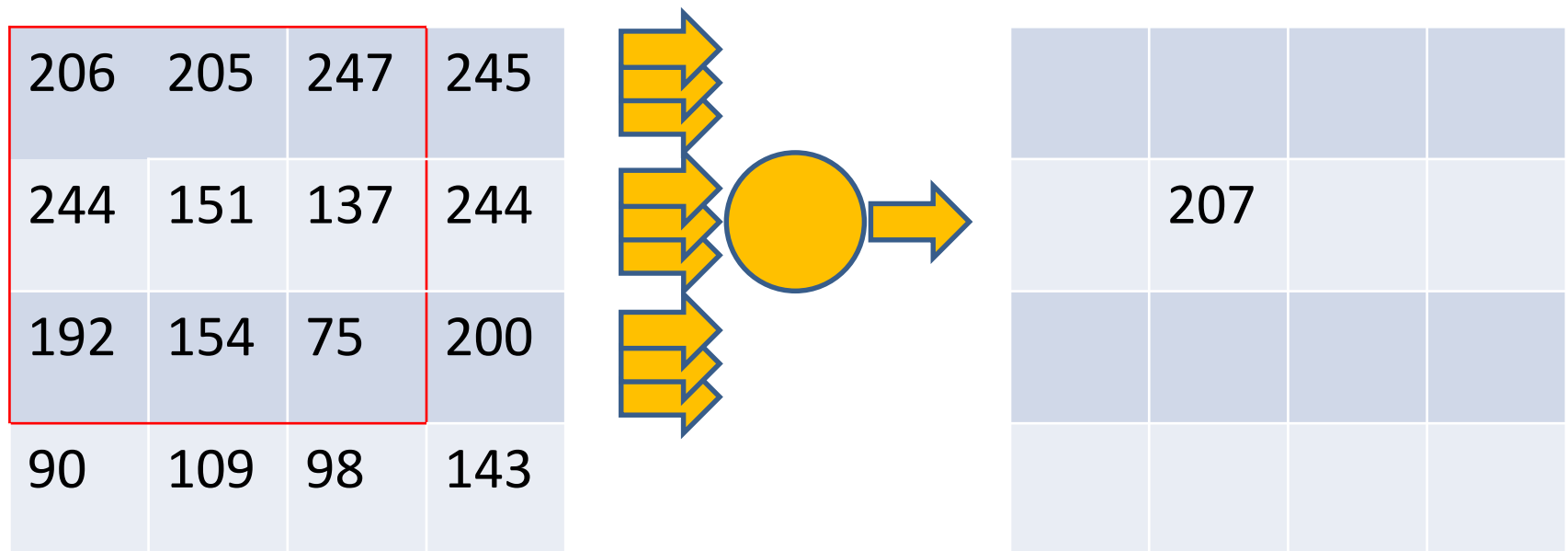
Szél detektálás példa: kameraman

- Mindkét irányba is lehet egyszerre



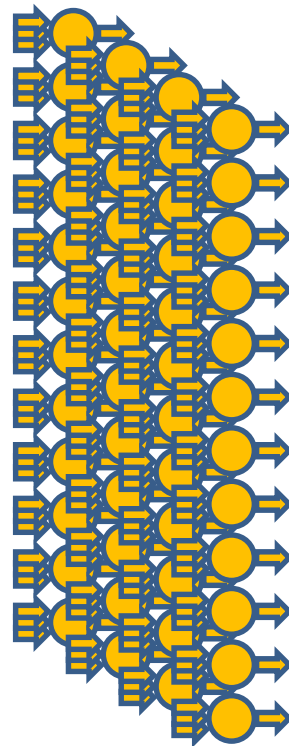
Konvolúciós hálózat

- A konvolúciós mag felfogható egy neuron bemeneti súlyainak is



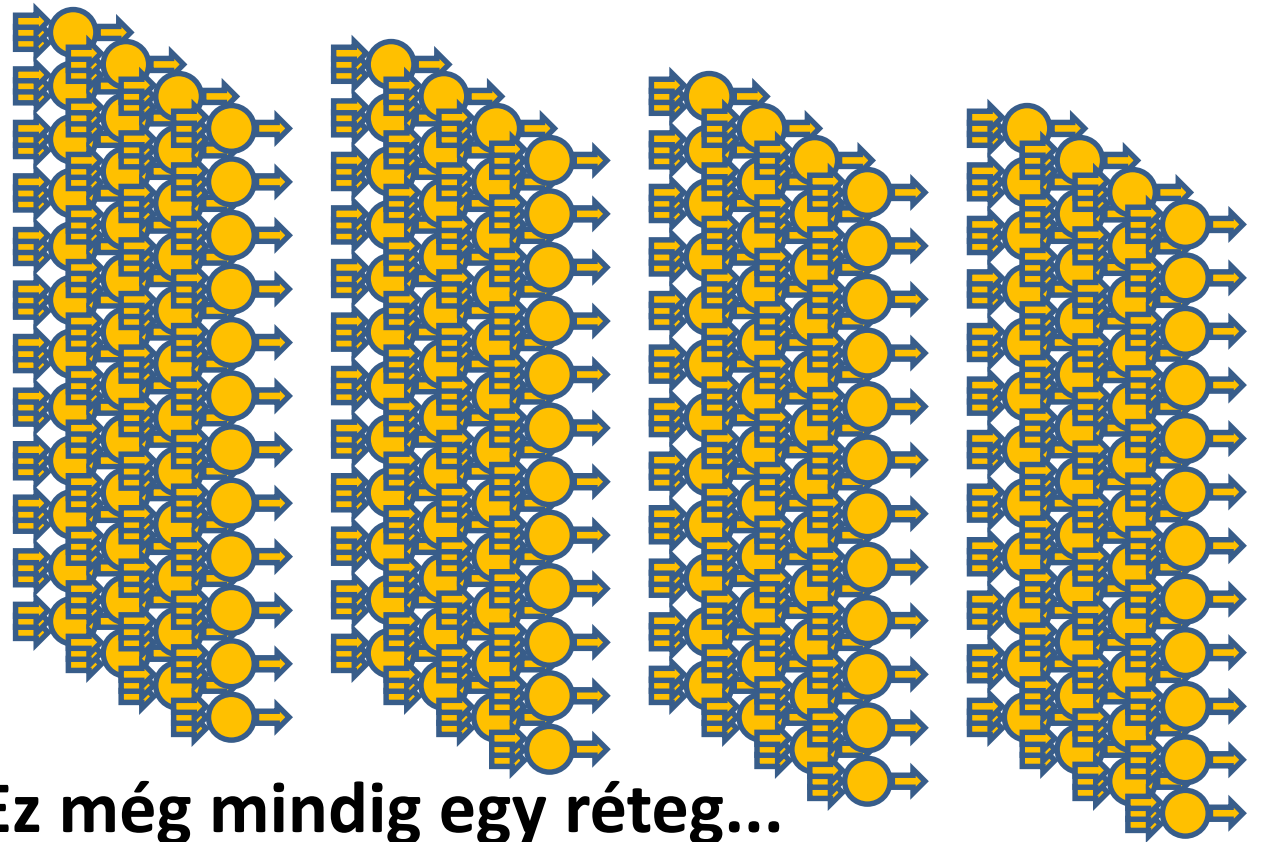
Ahány pixel, annyi neuron

- A réteg négyzetbe rendezett neuronokból áll
- A neuronok *ugyanazokat a súlyokat* használják („weight sharing”)



Feature depth

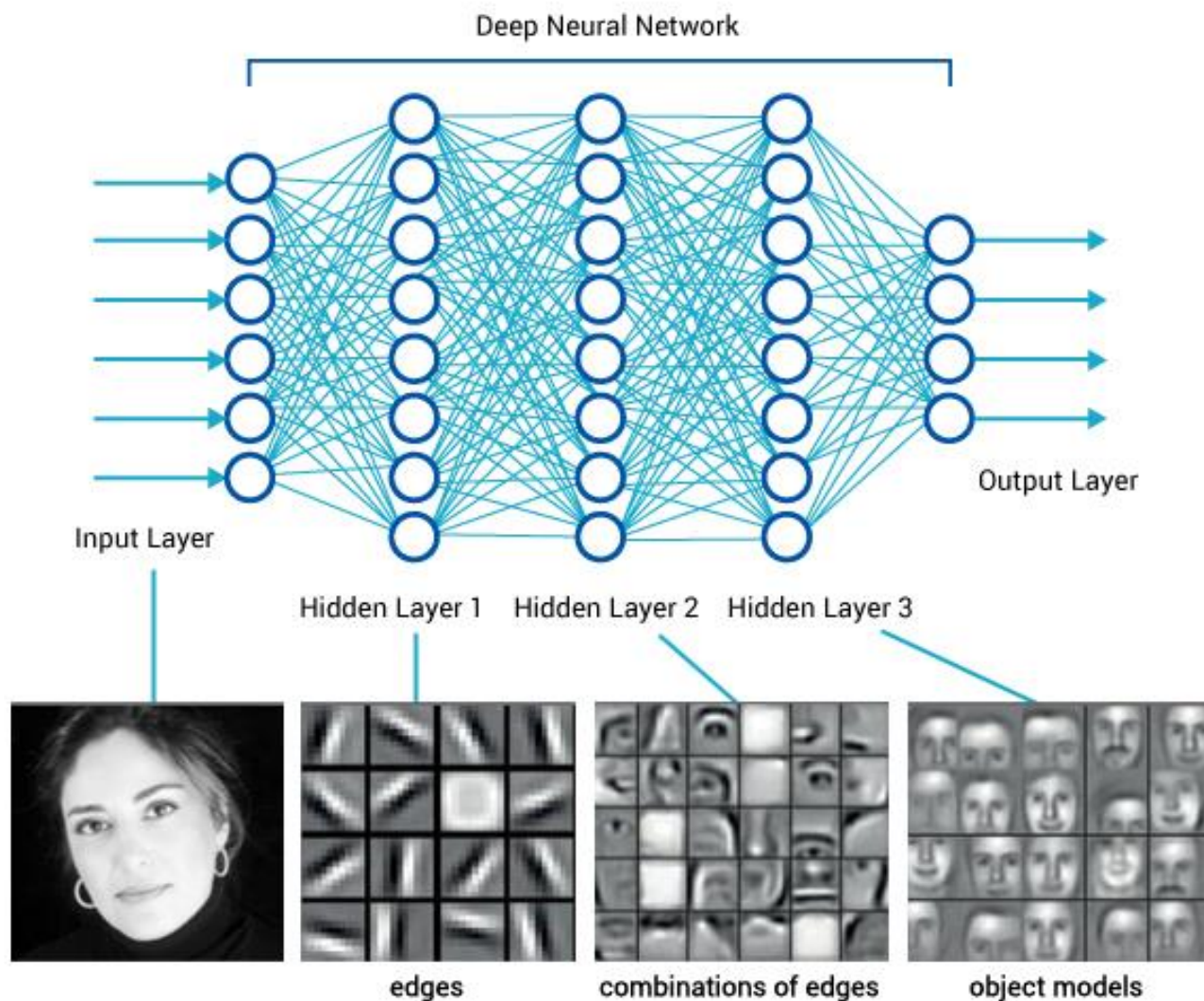
- Többféle konvolúcióval engedjük a hálózatot kísérletezni, Inception pl. 64-192 félével



Konvolúciós rétegek egymás után

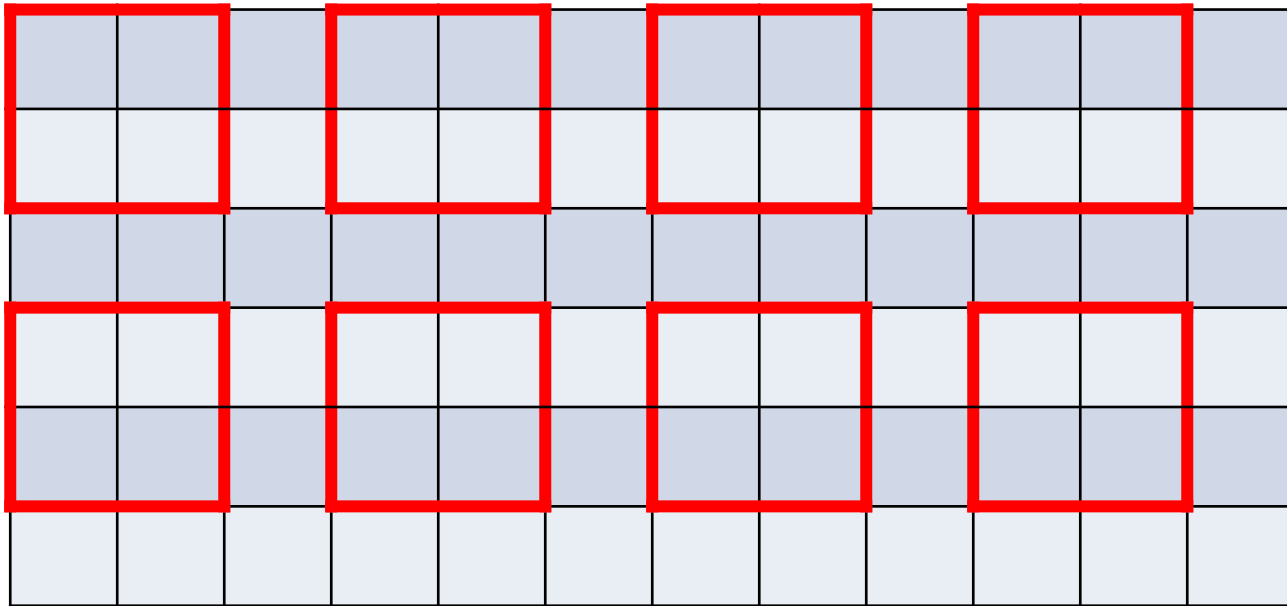
- A következő réteg megkapja az összes féle módon konvolvált verzióját a bemenetnek
- Ha például három, megfelelő szögben álló szélrt lát egymás mellett, akkor fel tud ismerni egy háromszöget
- Az egymás után következő rétegek egyre komplikáltabb dolgokat tudnak összerakni: ez volt Szegedy Krisztián újítása 2014-ben

Egyre magasabb rendű struktúrák



Lépésköz („stride”)

- Lehet úgy is 2x2 konvolválni, hogy csak például minden harmadik pixelen állunk meg



(Általában a lépésköz kisebb, mint a kernel, itt csak az ábrázolás érthetősége miatt van másképp)

Lépésköz előnyei és hátrányai

- Hátránya a potenciális információvesztés
- Előnye, hogy a kimenet már kevesebb adatot tartalmaz: ha a lépésköz kettő, akkor mindkét irányba feleződik a méret, tehát a bemenet egynegyede lesz a kimeneti adat

Max pooling

- A vizsgált területről a maximális értéket választjuk

206	205	247	245
244	151	137	244
192	154	75	200
90	109	98	143



244	247	247	247
244	247	247	247
244	244	244	244
192	192	200	200

Max pooling előnyei

- Növeli a transzláció invarianciát (az eredmény nem változik kis elmozdulásoktól)
- Mivel elég sok redundáns információt tartalmaz, ezért a lépésközt lehet nagyobbra választani, például kettőnek, komolyabb információvesztés nélkül, és akkor a következő rétegeknek már csak negyedannyi adattal kell dolgozniuk

Probléma: konvolúció lokális

- A nagy összefüggések felismerésére csak a legutolsó rétegekben kerülhet sor, mert a konvolúció csak az egymás melletti pixeleket, vagy egymás melletti „alkatrészeket” tudja együtt figyelni
- A kép értelmezéséhez viszont gyakran nagy segítség, ha már az elején van valami elképzelés a nagy összefüggésekről

Lyukas konvolúció („atrous convolution”)

- A kernel kihagyásokkal kerül alkalmazásra

206	205	247	245	244
244	151	137	244	254
192	154	75	200	249
90	109	98	143	223
87	89	107	196	236

1	0	2
-3	-5	5
2	0	-1

		926		

Lyukas konvolúció előnyei és hátrányai

- Lehetőséget ad a nagy összefüggések figyelembe vételére már a hálózat elején
- Hátránya, hogy nem lehet olyan jól optimalizálni a számításokat, relatív sokáig tart a kiszámítás
- És ha nagy a kihagyás, akkor meg a helyi összefüggéseket nem látja

Képfelismerés már mindennapos

- Az elmúlt években rengeteget fejlődött ez a fajta hálózat is
- A képfelismerés már API szinten elérhető, programozók beépíthetik az appjukba, nem is kell megírni: <https://cloud.google.com/vision/>

Valós idejű objektum felismerés

- Az alábbi videón valós idejű objektum felismerést láthatunk
- <https://youtu.be/MPU2HistivI>
- Ez pedig az önvezető autó által „látott” kép:
<https://youtu.be/tiwVMrTLUWg?t=545>
- Túl „téglalapos”? Akkor:
<https://youtu.be/bI7atFO0glg?t=62>

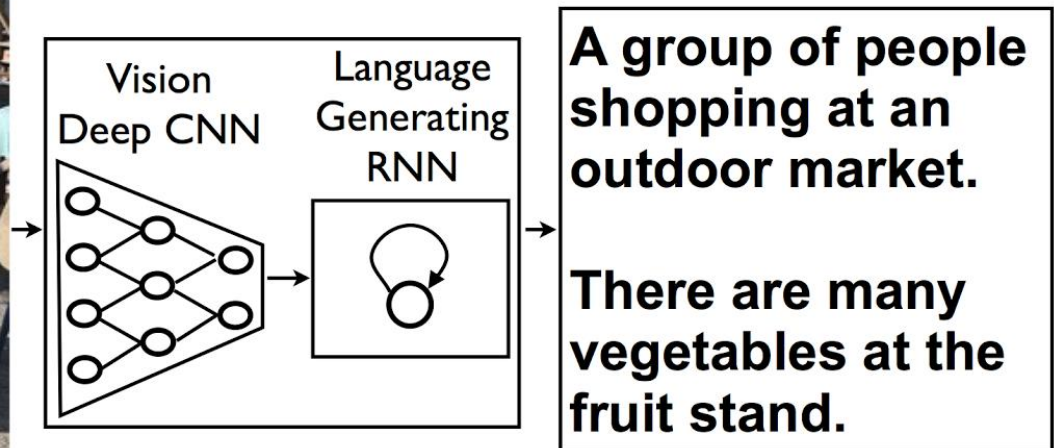
Multimodális tanulás

- Szövegértést segítheti az ajakmozgás követése („bar-far”, vagy HAL a *2001 Űrodisszeiá*-ban): a Deep Mind hálózata már 50%-os pontosságot produkál¹
- Az esemény-felismerés videó és audió csatornából például az objektum őrzés egyik fontos igénye

¹Joon Son Chung et al.: Lip Reading Sentences in the Wild, 2016.11.16. <https://arxiv.org/abs/1611.05358>
<https://youtu.be/0BbYeMyxMy0>

Képfeliratozás I.

- Nem sokkal az Inception után már megjelent ez az applikáció¹



¹Oriol Vinyals et al.: A picture is worth a thousand (coherent) words: building a natural description of images, 2014.11.17.
<https://ai.googleblog.com/2014/11/a-picture-is-worth-thousand-coherent.html>

Képfeliratozás II.

Describes without errors



A person riding a motorcycle on a dirt road.



A group of young people playing a game of frisbee.



A herd of elephants walking across a dry grass field.

Describes with minor errors



Two dogs play in the grass.



Two hockey players are fighting over the puck.



A close up of a cat laying on a couch.

Somewhat related to the image



A skateboarder does a trick on a ramp.



A little girl in a pink hat is blowing bubbles.



A red motorcycle parked on the side of the road.

Unrelated to the image



A dog is jumping to catch a frisbee.



A refrigerator filled with lots of food and drinks.



A yellow school bus parked in a parking lot.

Feliratból kép¹

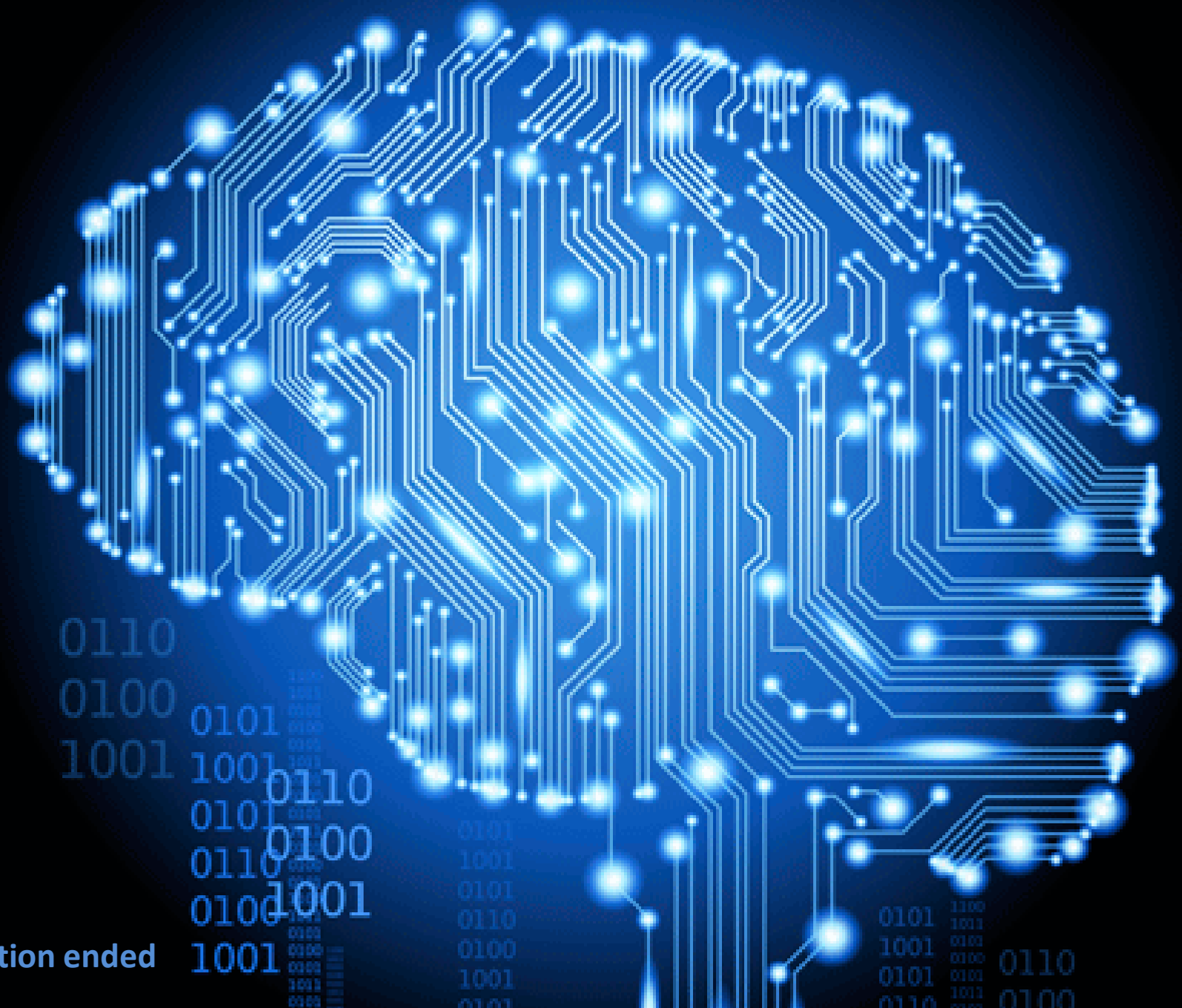
This flower has white petals with a yellow tip and a yellow pistil



This bird is black with green and has a very short beak



¹Han Zhang et al.: StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks , 2016.12.10. <https://arxiv.org/pdf/1612.03242v1.pdf>



>connection ended