

Deep Learning Szeminárium

I. előadás: bemutatkozás, kedvcsináló

Varga Dániel (Rényi Intézet)

Előadók:

- Csiszárík Adrián (Rényi Intézet)
- Lukács András (ELTE TTK Számítógéptudományi Tanszék)
- Varga Dániel (Rényi Intézet)
- Zombori Zsolt (Rényi Intézet)

<https://bit.ly/elite-ttk-deeplearning>

Feltételezett előismeretek: lineáris algebrai alapismeretek, többváltozós analízis alapismeretek, python alapismeretek

A jegyszerzést megkönnyítő további előismeretek: gépi tanulás alapfogalmai, valószínűségszámítási alapismeretek, python tudományos programozás (numpy, scipy)

Jegyszerzés módja:

- Olvasószemináriumi előadás vagy önálló labormunka.
- Azoknak a hallgatóknak, akik birtokában vannak python/numpy alapismereteknek, a labormunkát ajánljuk a jegyszerzésre.
- Ezek hiányában is elvégezhető a tárgy, olvasószemináriumi előadás megtartásával, korlátozott létszámban.
- Cikkosztás két hét múlva.
- További részletek a feltételekről és elvárásokról a tárgy weblapján hamarosan.

A kurzus felépítése:

1. Bevezető előadások a deep learning-ről
2. Közös labormunka python/numpy/tensorflow/keras környezetben
3. Előadások modern deep learning modellekkről
 - a téma előadóitól,
 - meghívott szakértőktől,
 - olvasószemináriumi formában a kurzus hallgatóitól.

Ajánlott olvasmányok:

- François Chollet - Deep Learning with Python
- <http://course.fast.ai/>
- <http://cs231n.stanford.edu/>
- <http://rll.berkeley.edu/deeprlcourse/>
- <http://www.deeplearningbook.org/>
- Aki nagyon naprakész szeretne lenni, annak <https://www.reddit.com/r/MachineLearning/> vagy Twitteren deep learning hírességek követését ajánljuk.
- <https://github.com/> A legizgalmasabb dolgok elég nagy része open source, vagy van elfogadható minőségű open source reimplementationációja. A kód módosítgatásából lehet a legtöbbet tanulni.

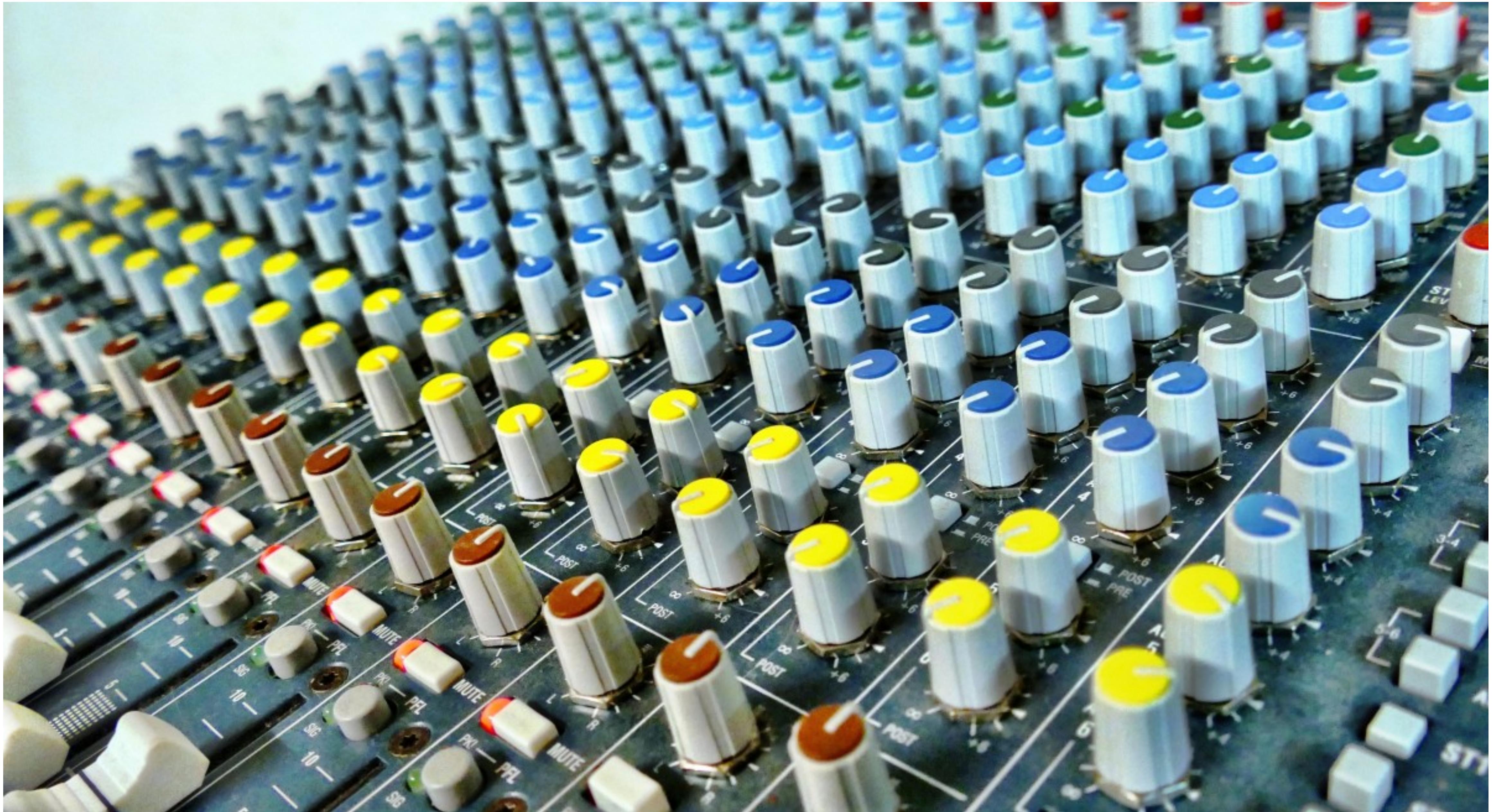
...ez mind közkincs:

- pip install tensorflow keras
- git clone <https://github.com/tensorflow/models> ; cd research/im2txt # image to text
- git clone <https://github.com/hanzhanggit/StackGAN.git> # text to image
- git clone https://github.com/matterport/Mask_RCNN # object detection & segmentation
- git clone <https://github.com/tensorflow/nmt> # machine translation (seq2seq)
- git clone <https://github.com/tomlepaine/fast-wavenet> # speech synthesis
- git clone <https://github.com/lengstrom/fast-style-transfer> # style transfer
- pip install pytorch
- git clone https://github.com/tkarras/progressive_gans # image generation

Megpróbálom végigcsinálni ezt az előadást úgy, hogy egyáltalán nem mondom el, hogy miből épül fel egy mesterséges neuronháló.

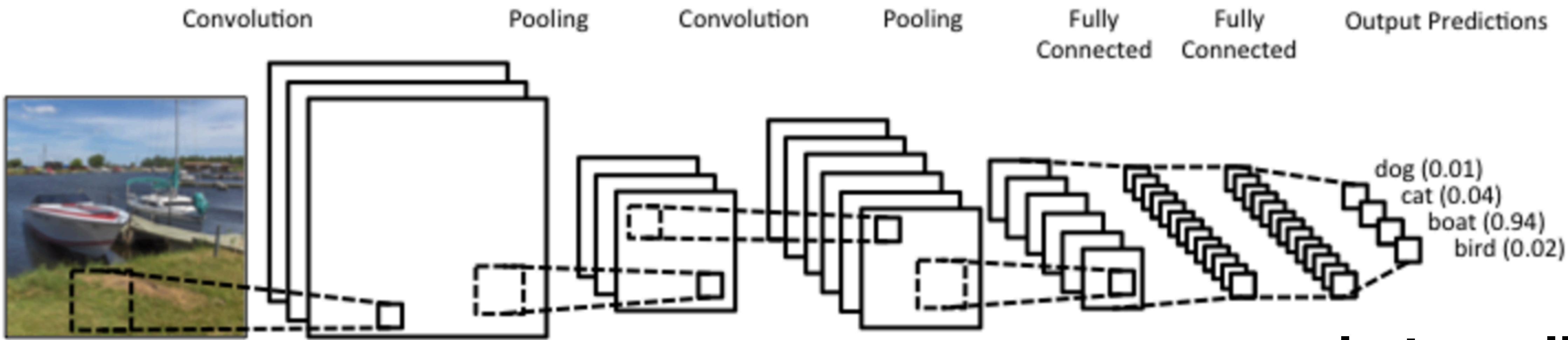
Jövő héten Zombori Zsolt teljesen pontos definíciót ad majd.

Egy fekete doboz nagyon sok potméterrel



Ennél azért egy kicsit részletesebben

- valós vektorokat valós vektorokba képező folytonos függvények paraméterezett családja
- bemenetek (kép, hang, videó, szöveg, térbeli pozíció, bármilyen idősor, ...)
- kimenetek (a fentiek közül bármelyik, klasszifikációnál valószínűségek vektora)
- veszteségfüggvény (Legegyszerűbb esetben $L(y, \hat{y})$, azaz hogy mennyire vagyok elégedett a kapott \hat{y} kimenettel az elvárt y kimenet ismeretében. Lehet pl. négyzetes hiba.)
- tanítás (paraméter-hangolás elvárt bemenet-kimenet párok halmazának felhasználásával, hogy a paraméterezett függvénycsaládból egy számunkra alkalmas konkrét elemet kiválasszunk.)

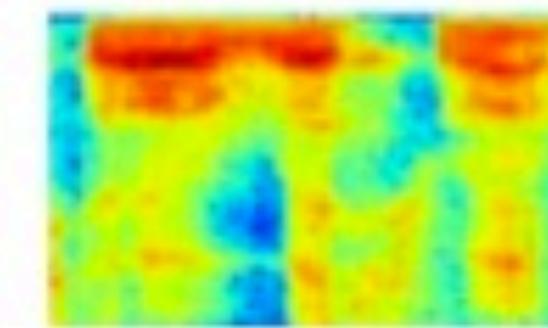
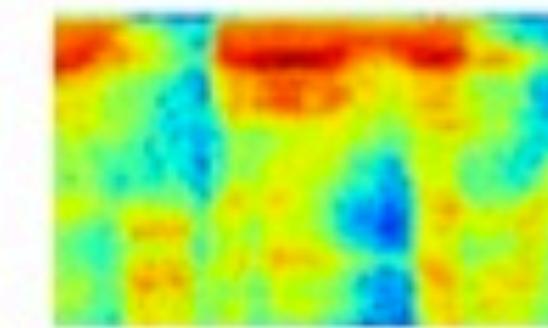
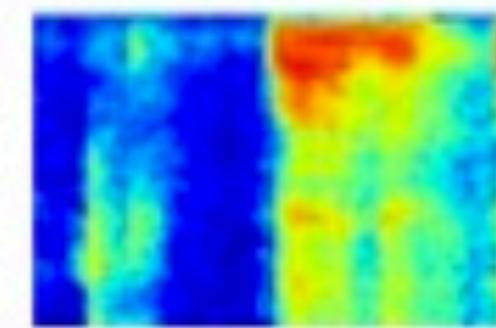
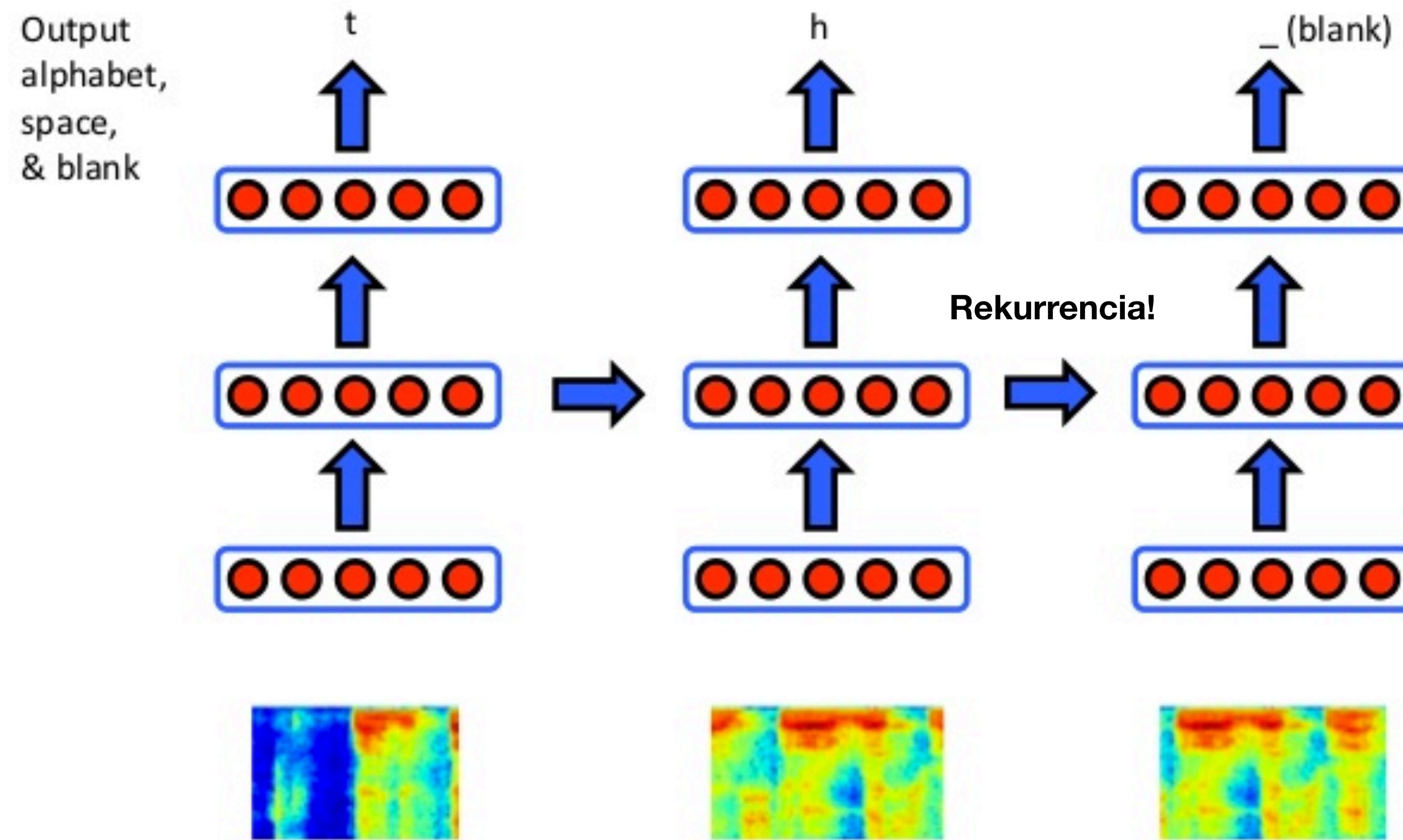


**RGB pixel bemenet
(224x224x3 dim)**

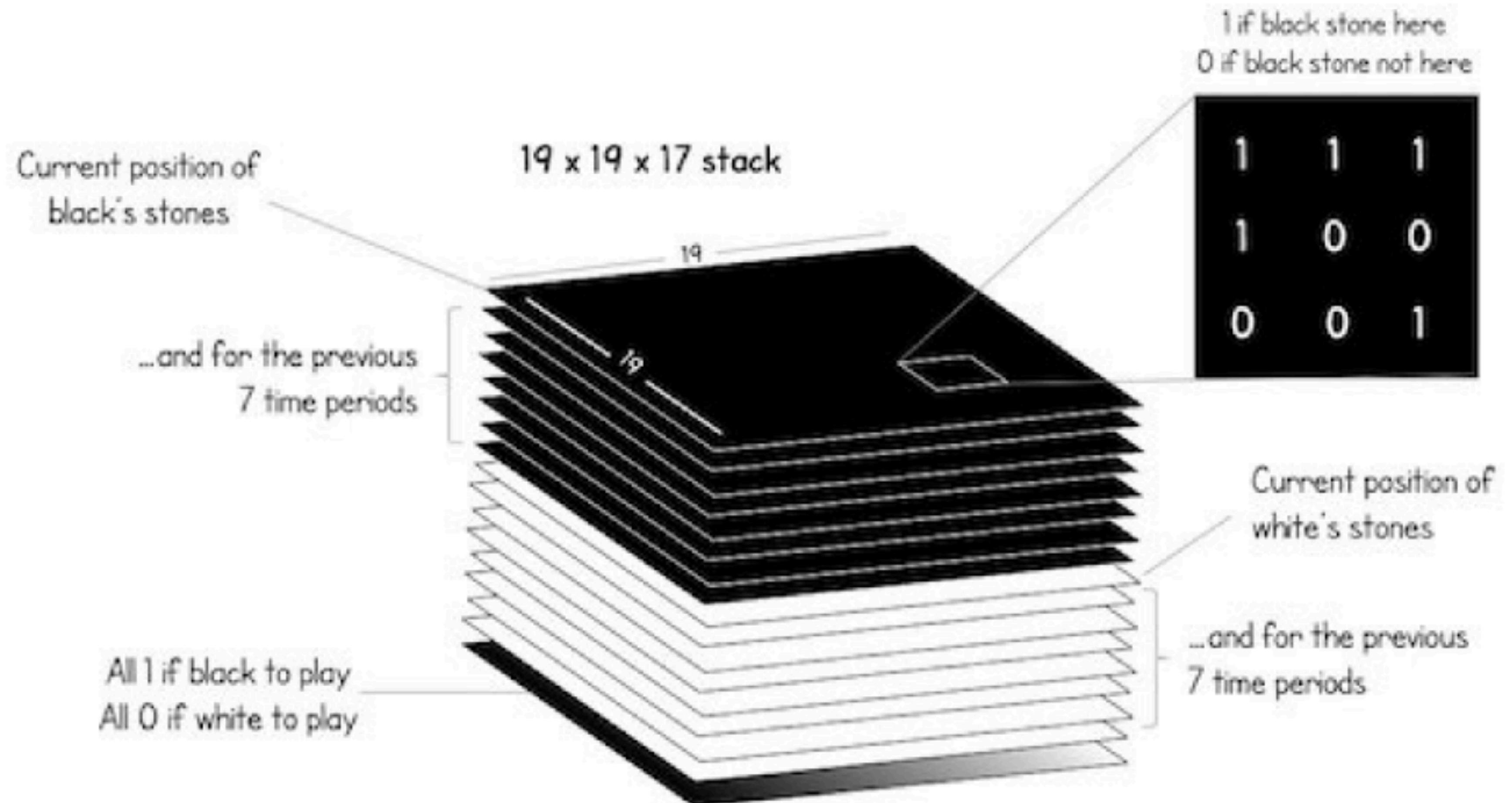
**kategorikus
kimenet
(4 dim)**

Kimenet: pixelenkénti klasszifikáció (haj vagy nem haj)





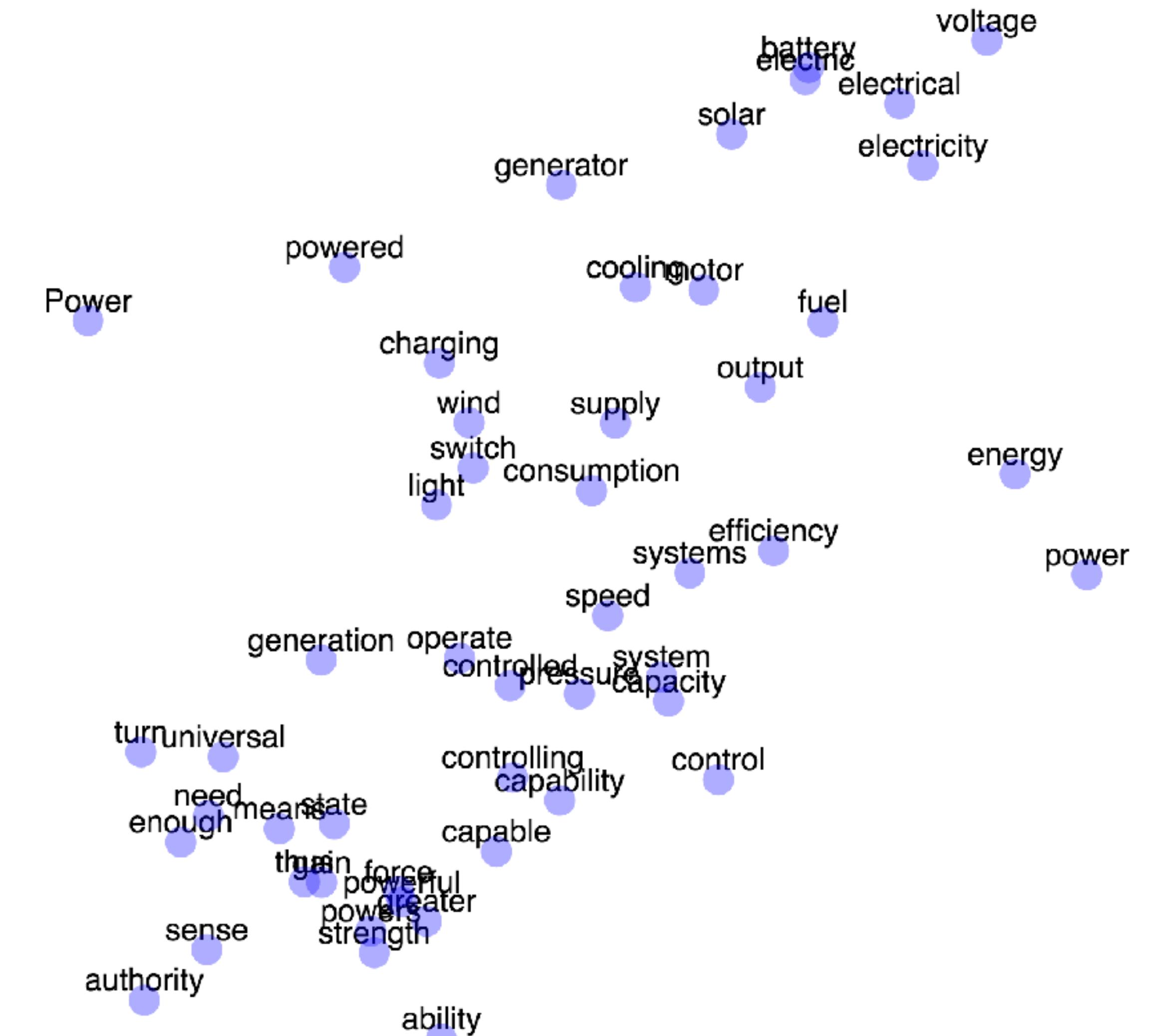
az AlphaGo Zero játékállás -reprezentációja



This stack is the input to the deep neural network

A word embedding módszereknek hála a szavak is reprezentálhatók valós vektorokként, és így használhatók neuronhálók bemeneteként és kimeneteként.

A beágyazás során megpróbálunk közelí vektorokba képezni olyan szavakat, amiknek a szövegkörnyezetük gyakran hasonló.

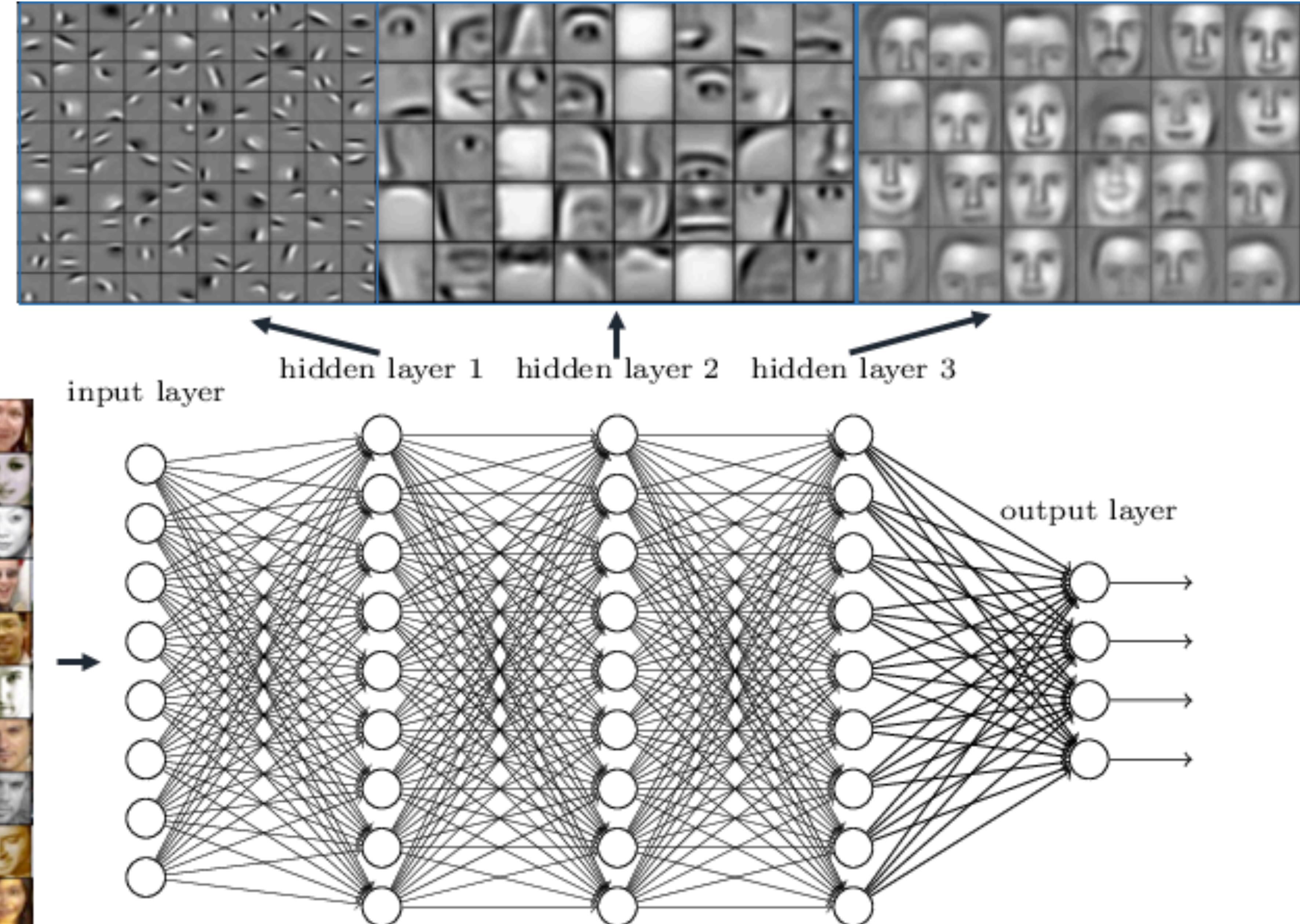


A neuronháló modellek építésének fő lépései (nem okvetlenül pontosan ebben a sorrendben)

- tanulóadatok (bemenet-kimenet párok) összegyűjtése
- veszteségfüggvény megtervezése
- neurális architektúra megtervezése (milyen paraméterezett függénycsaládok milyen kompozíciójaként álljon elő a neuronháló)
- while teljes kudarc:
 - hiperparaméter-hangolás, architektúra-hangolás
 - tanítás

Látens reprezentációk

Deep neural
networks learn
hierarchical feature
representations



<https://distill.pub/2017/feature-visualization/>

<https://distill.pub/2017/feature-visualization/appendix/>

POSITIVE CHANNEL



Channel Objective



Dataset examples

NEGATIVE CHANNEL



Negative Channel



Negative dataset examples

POSITIVE CHANNEL



Channel Objective

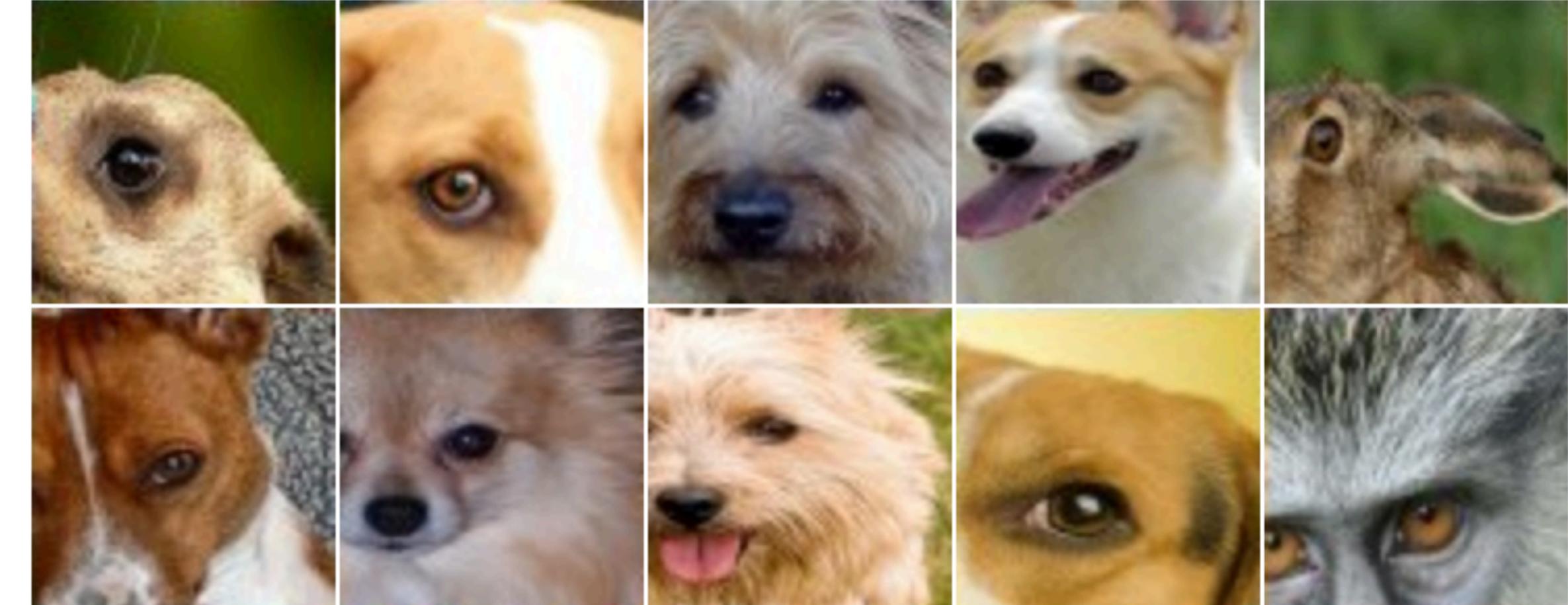


Dataset examples

NEGATIVE CHANNEL



Negative Channel



Negative dataset examples

POSITIVE CHANNEL



Channel Objective



Dataset examples

NEGATIVE CHANNEL



Negative Channel

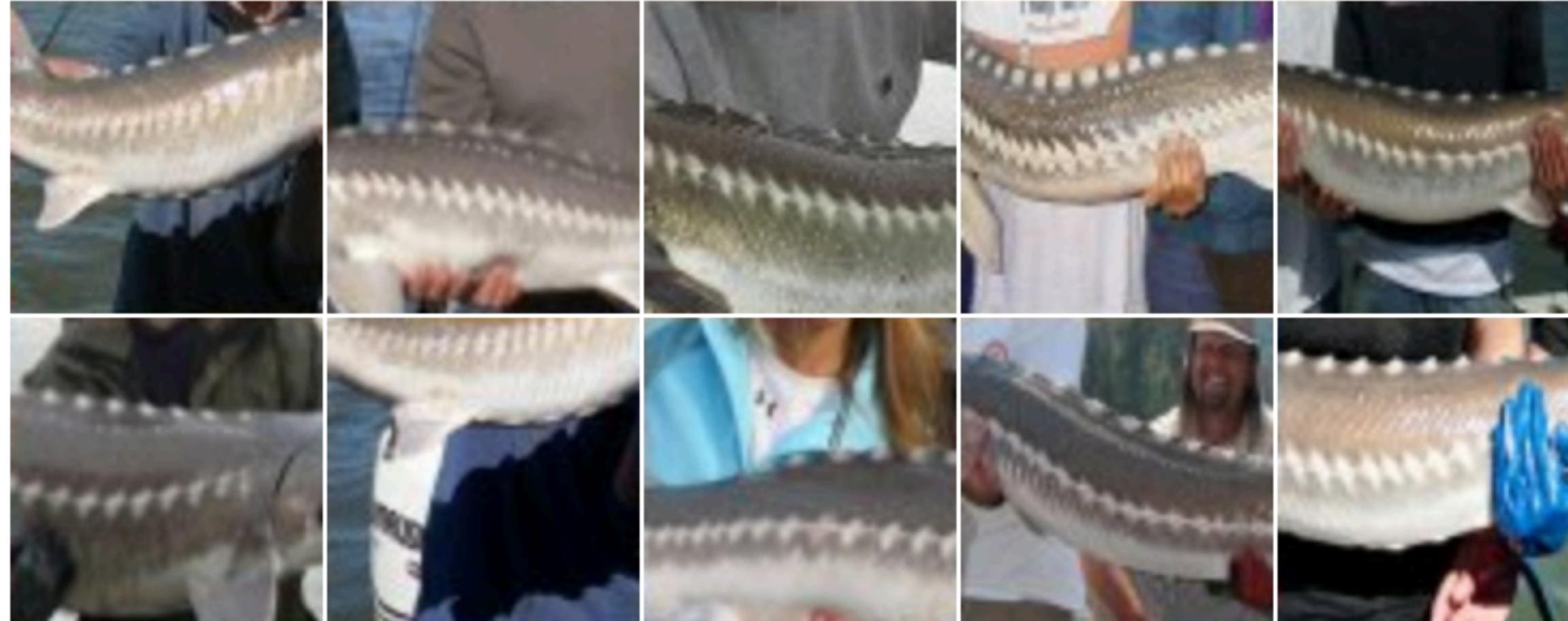


Negative dataset examples

POSITIVE CHANNEL



Channel Objective

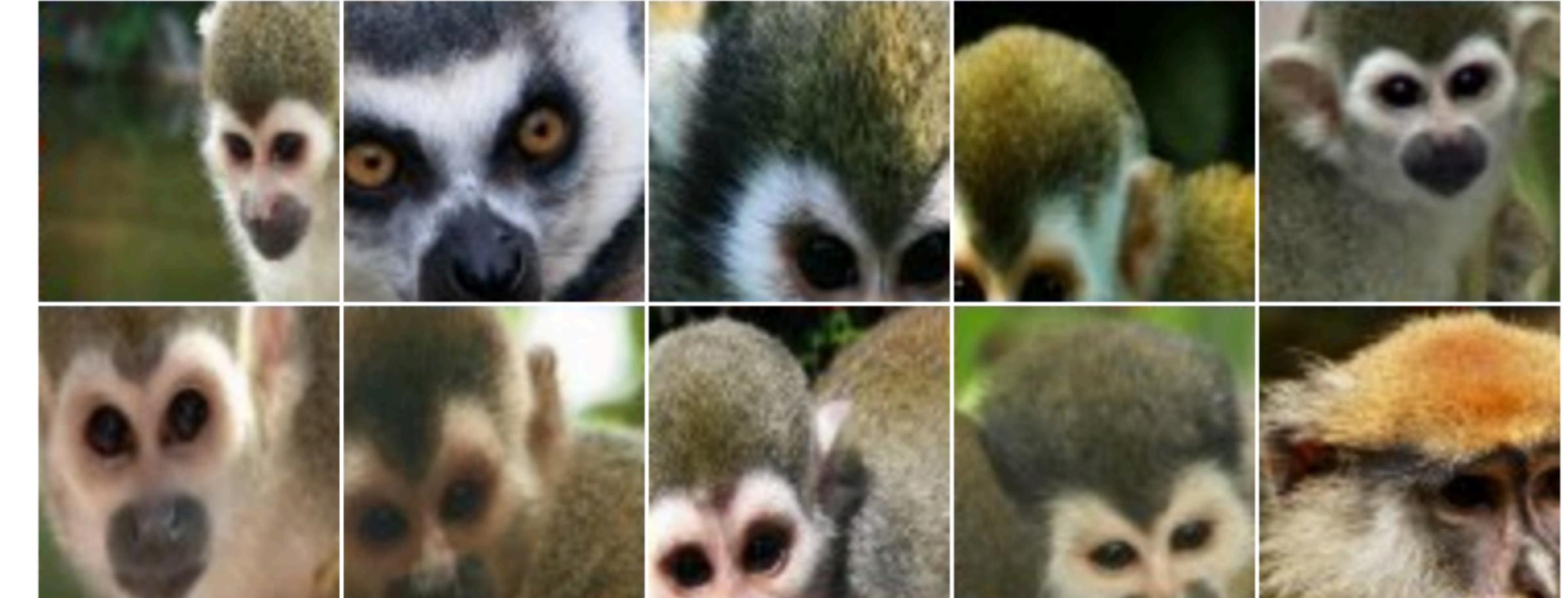


Dataset examples

NEGATIVE CHANNEL



Negative Channel



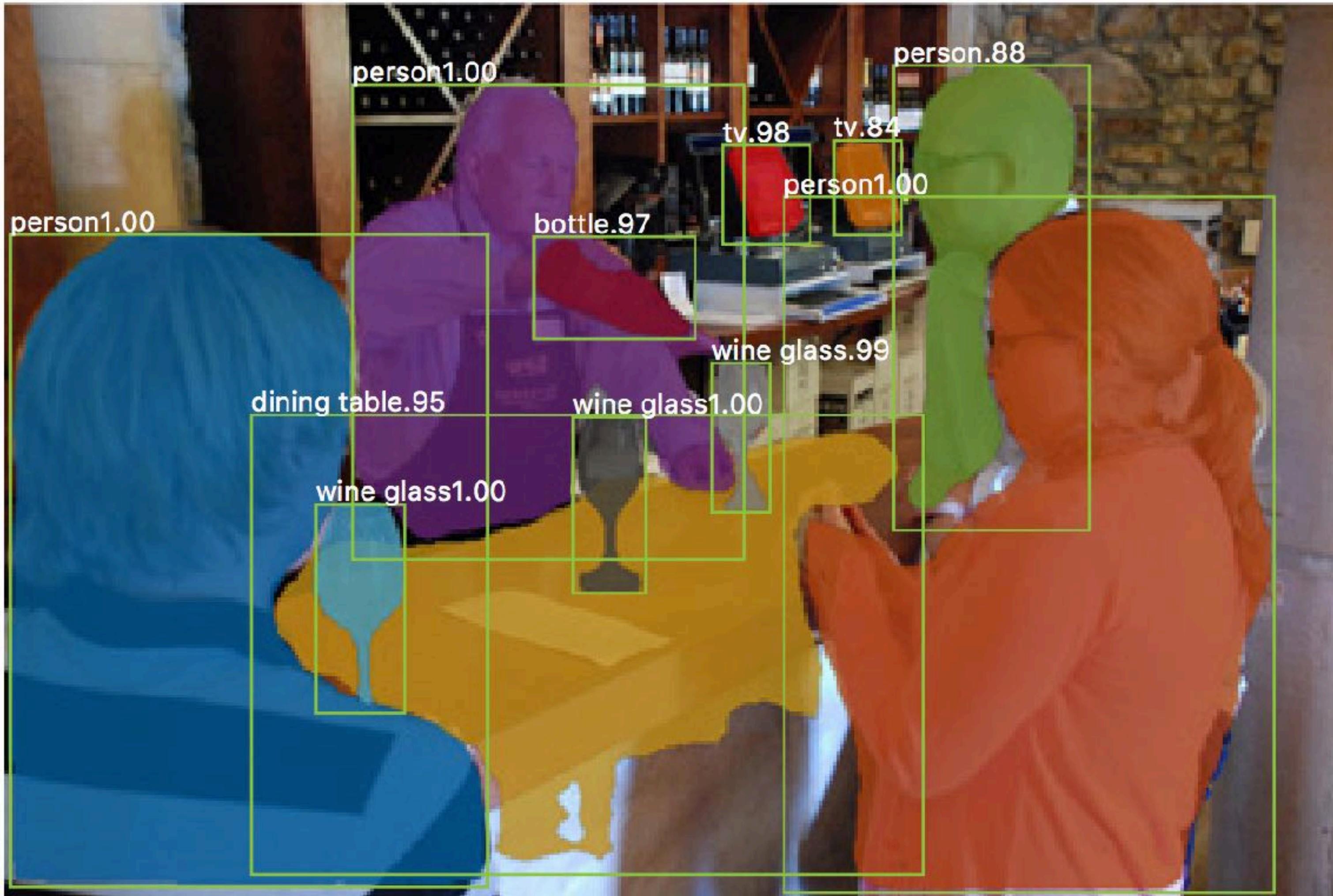
Negative dataset examples

Alkalmazások

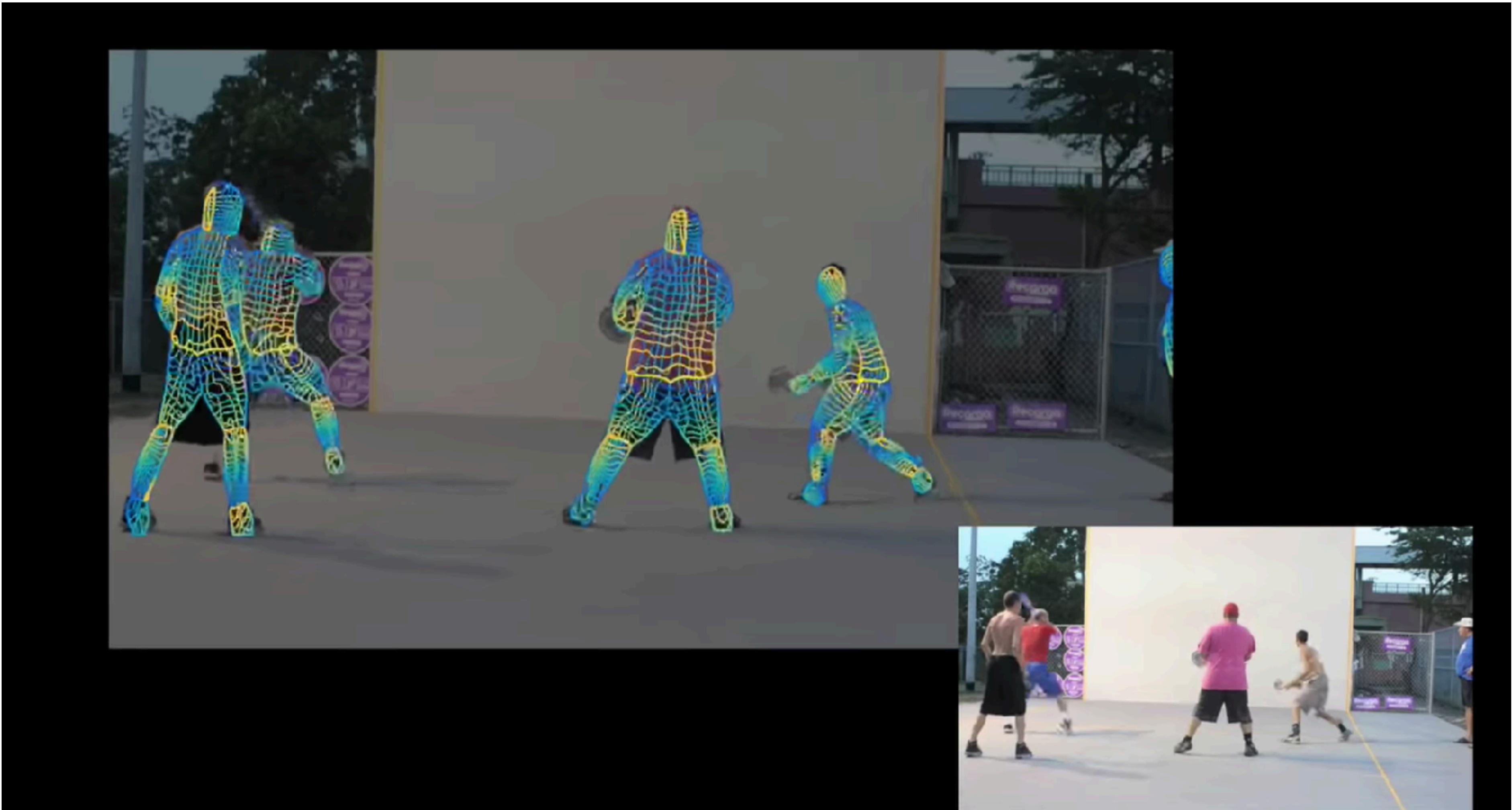
Alkalmazások

Nem reprezentatív lista, főleg az alapján válogatva, hogy szerintem mennyire érdekes vagy látványos az alkalmazás.

Mask R-CNN - képszegmentálás



DensePose - pose estimation



StackGAN - text to image

Text
description

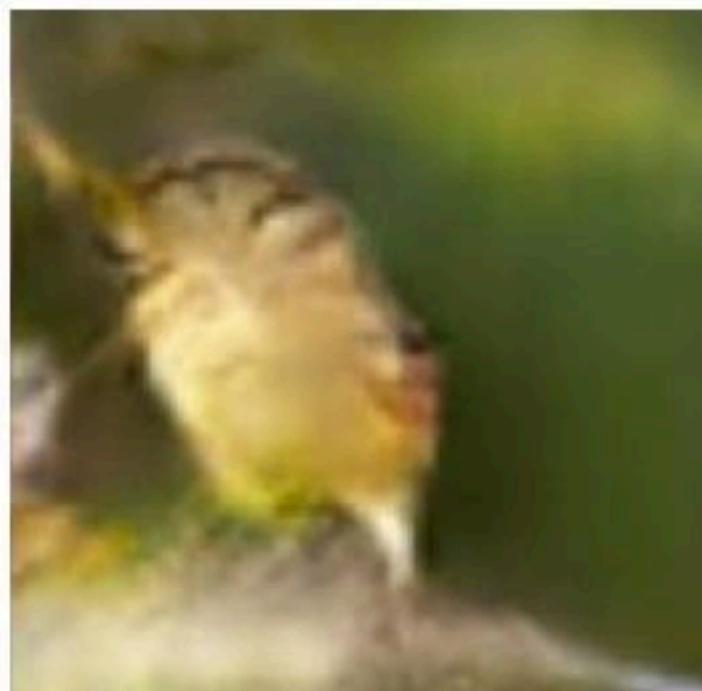
This bird is blue with white and has a very short beak



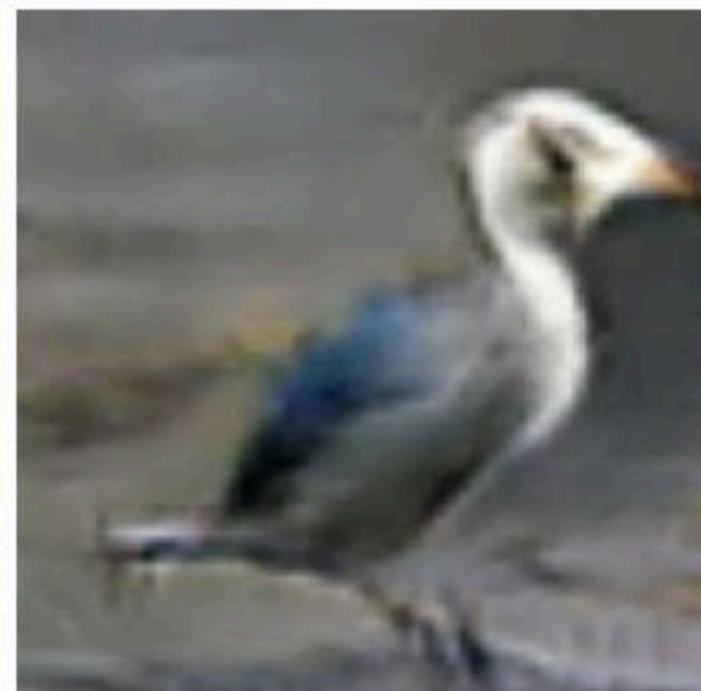
Stage-I
images

Stage-II
images

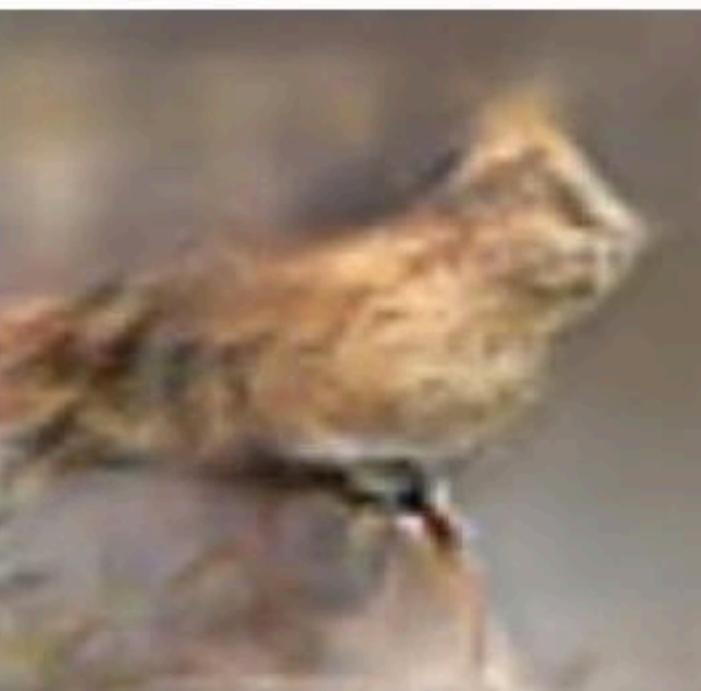
This bird has wings that are brown and has a yellow belly



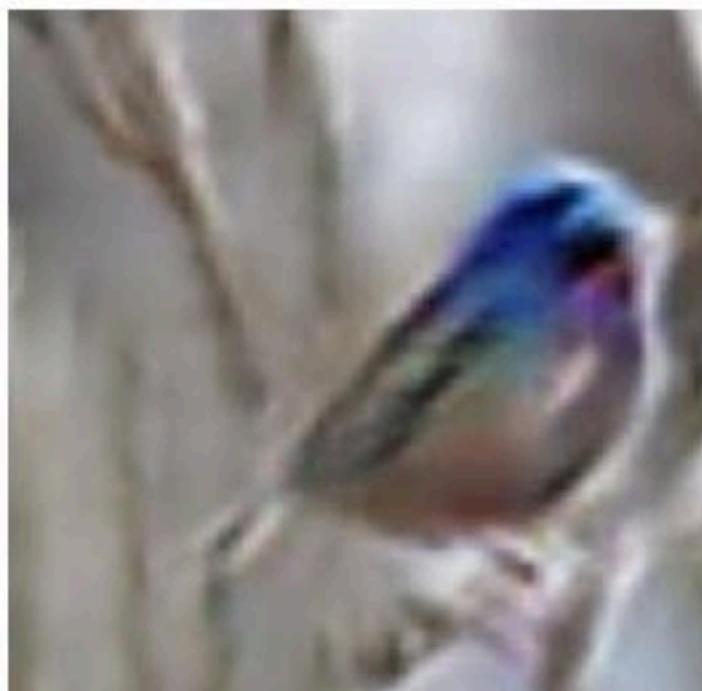
A white bird with a black crown and yellow beak



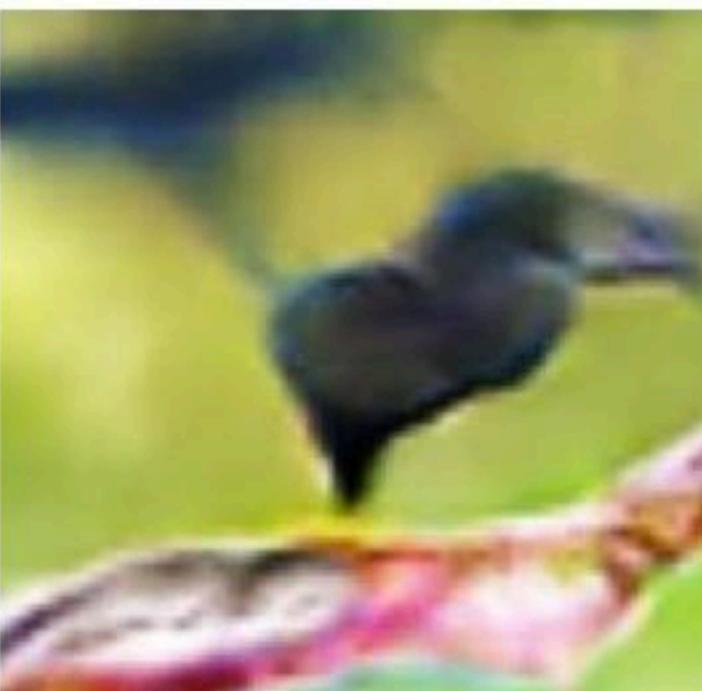
This bird is white, black, and brown in color, with a brown beak



The bird has small beak, with reddish brown crown and gray belly

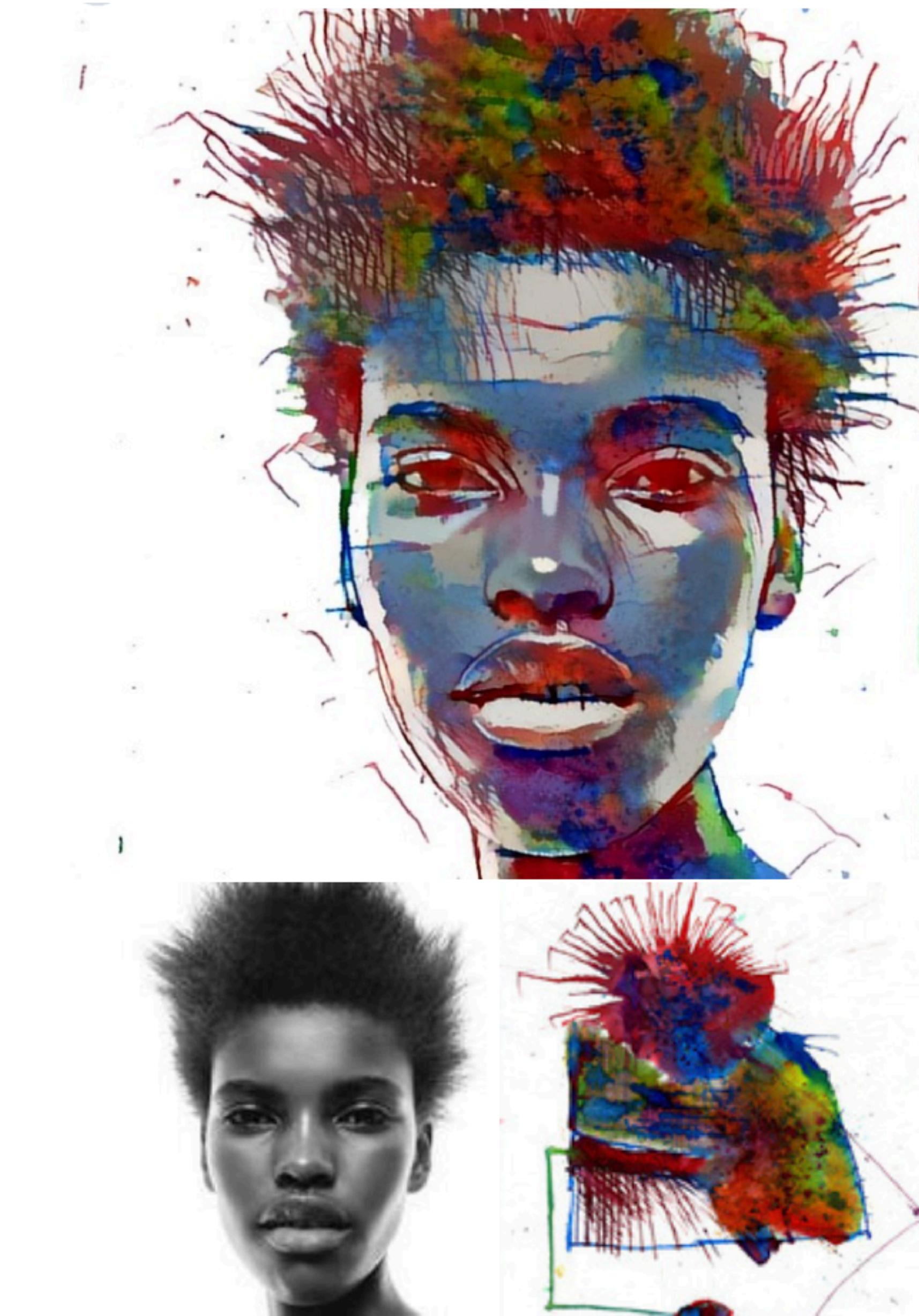


This is a small, black bird with a white breast and white on the wingbars.

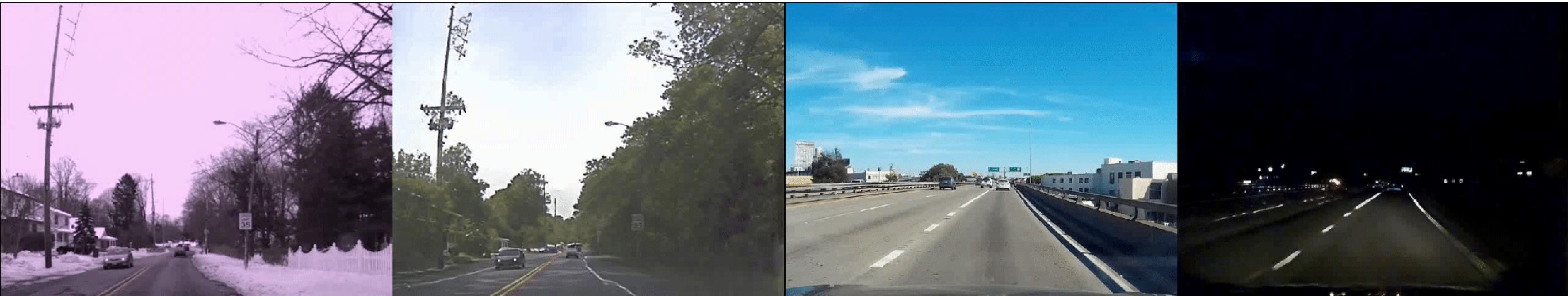


Style Transfer

link: Ostagram válogatás



Unsupervised Image-to-image Translation



DeepFakes

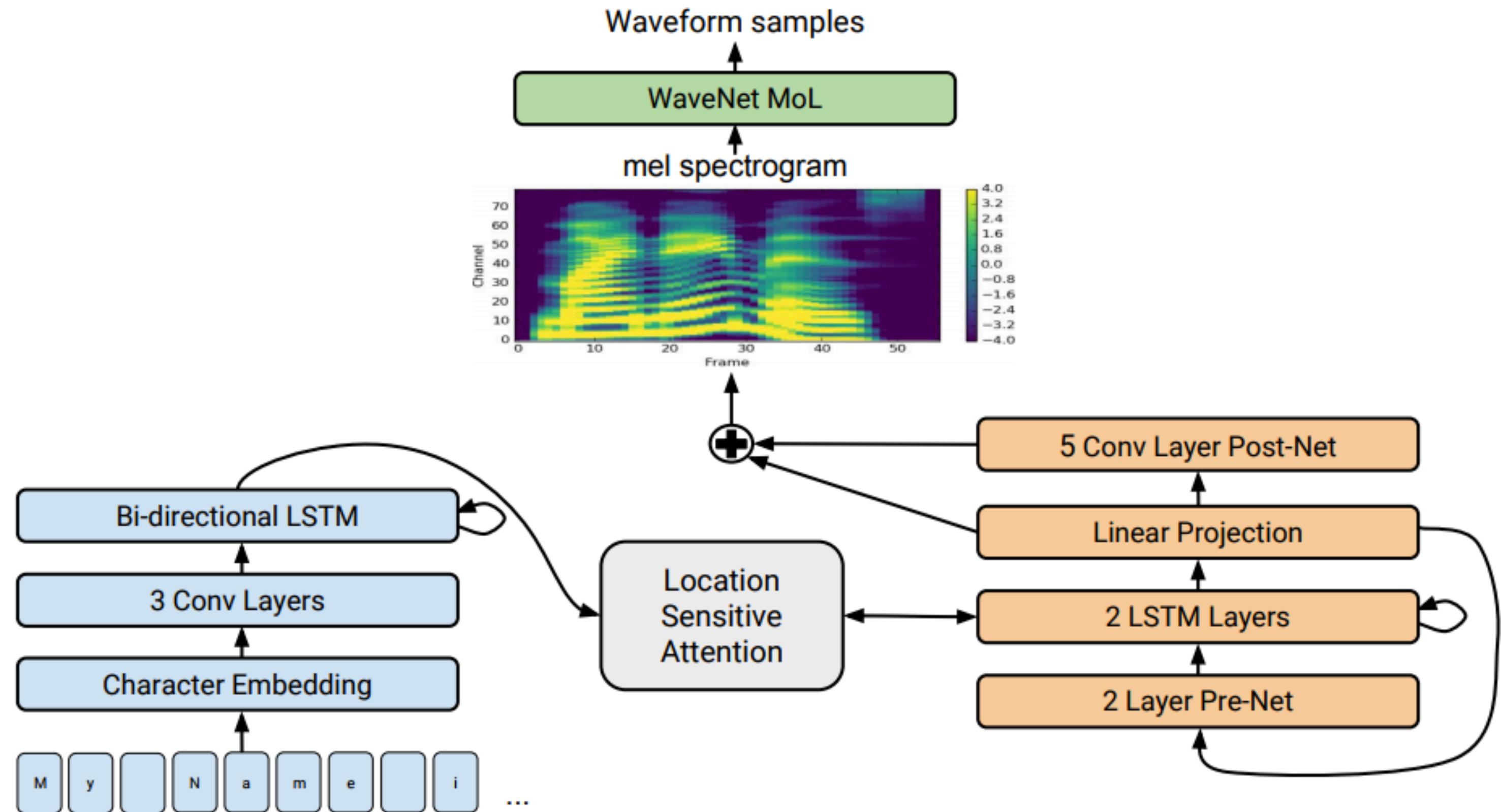


vídeo



Tacotron 2 - beszédszintézis

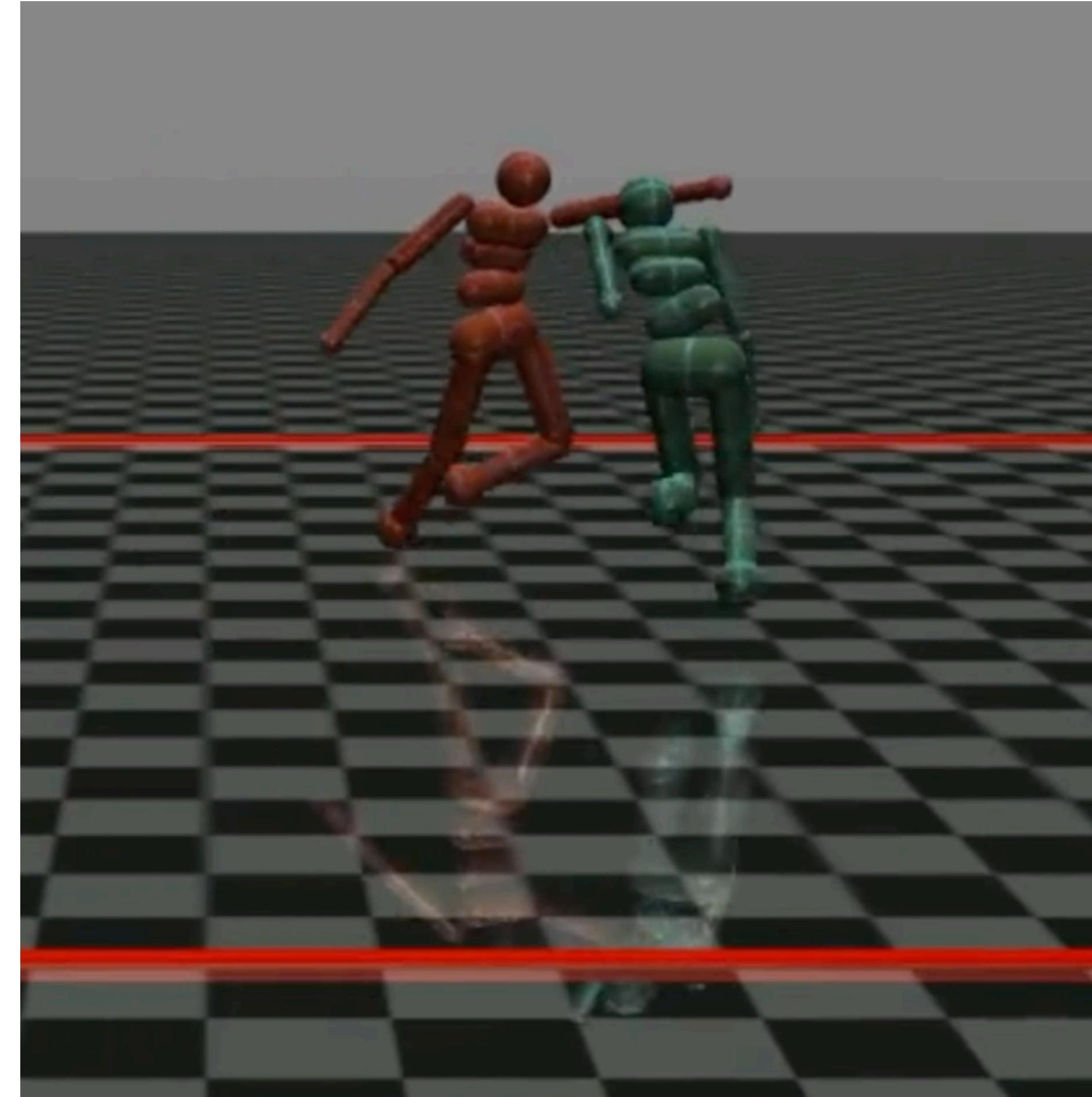
[link: Tacotron 2 szintézis példák](#)



Mély megerősítéses tanulás szimulált 3D környezetben

videó: tanult mozgás

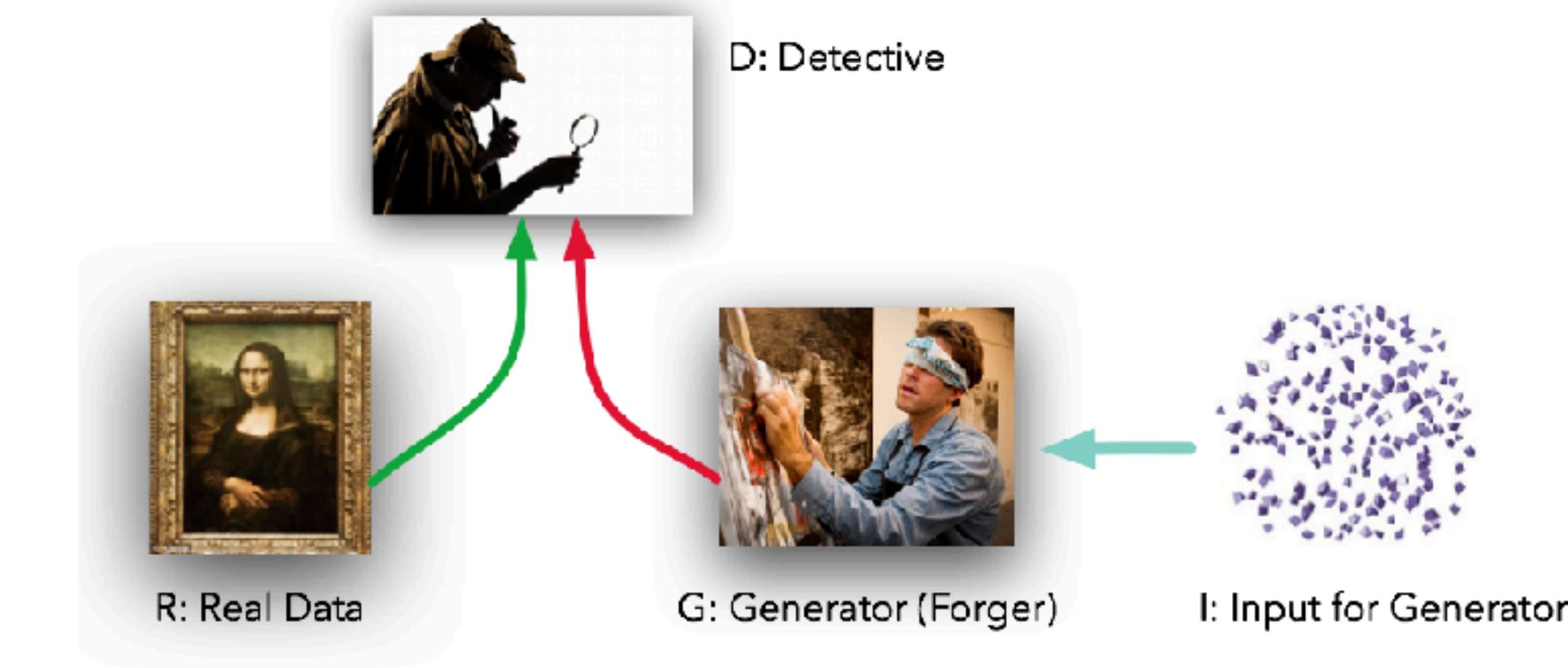
videó: competitive self-play



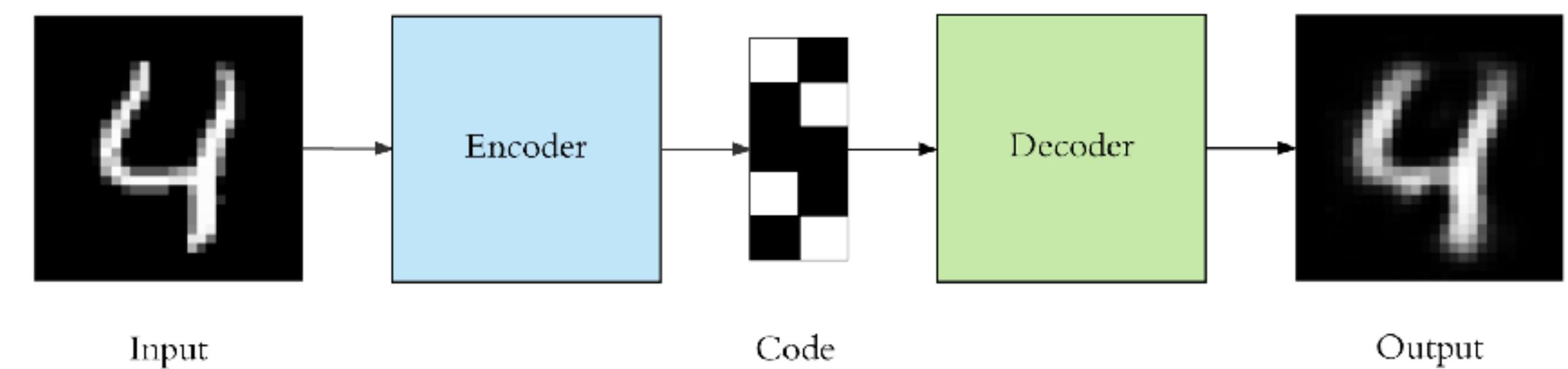
Generatív neuronhálók

Generatív neuronhálók

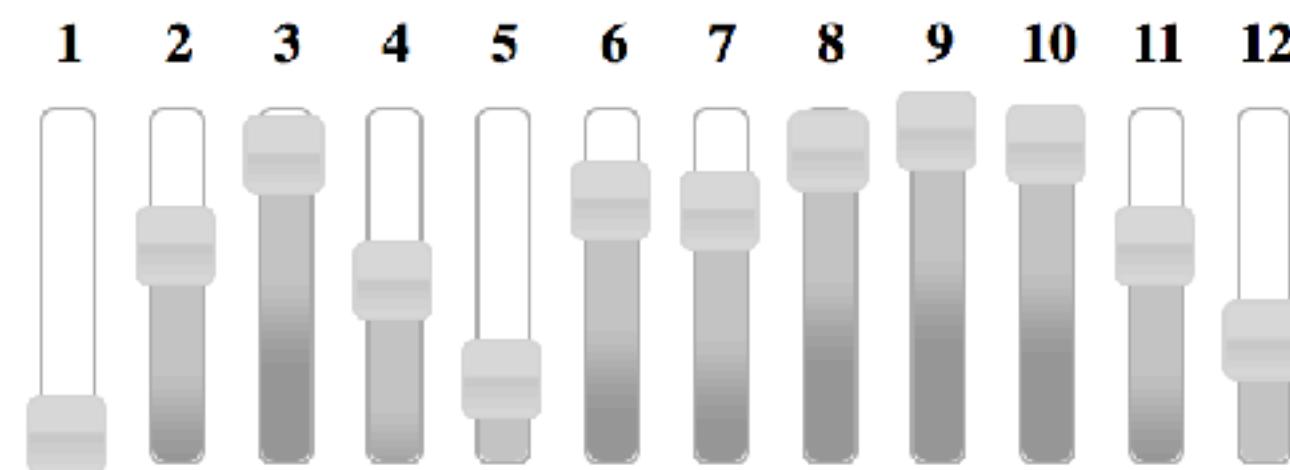
Generative Adversarial Network



Variational Autoencoder



Látens reprezentációk generatív modellekben



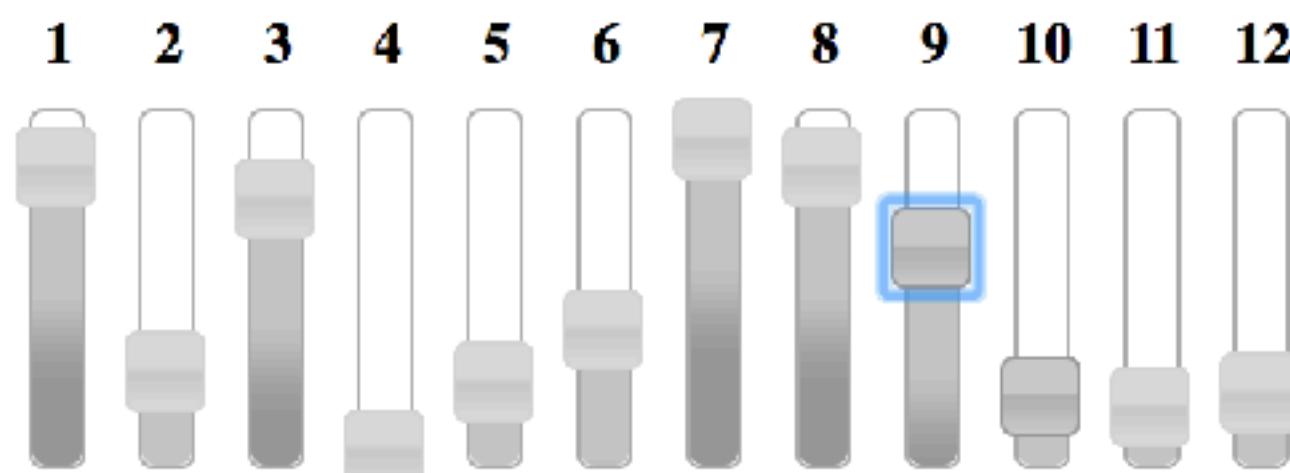
(Latent space) z



(Observed space) x



Digits autoencoder demo

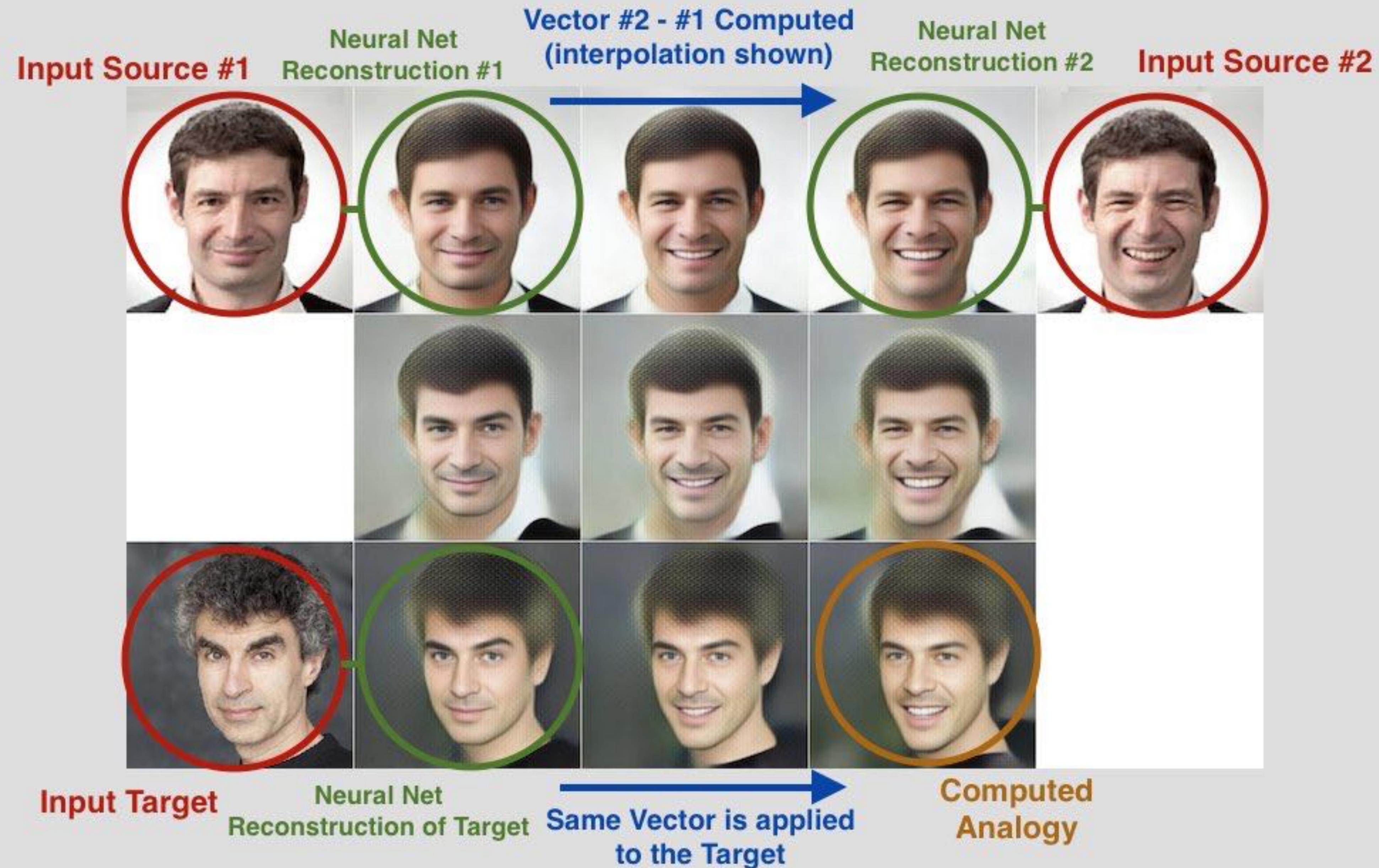


(Latent space) z



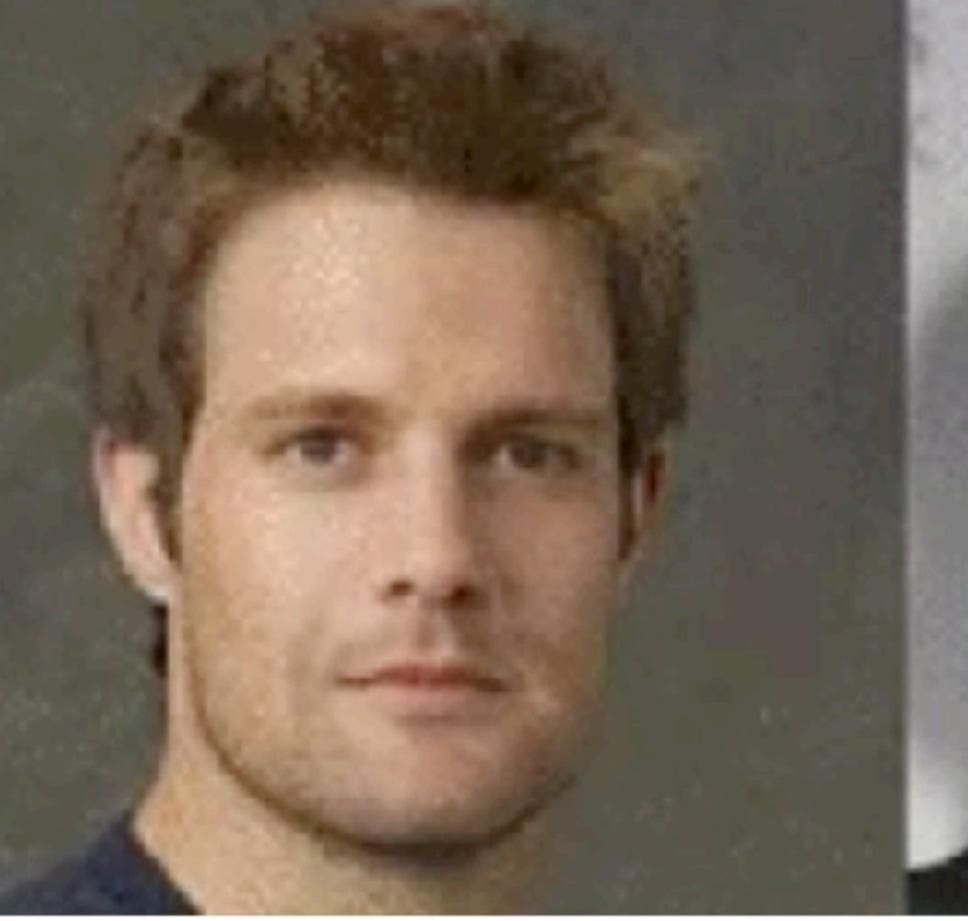
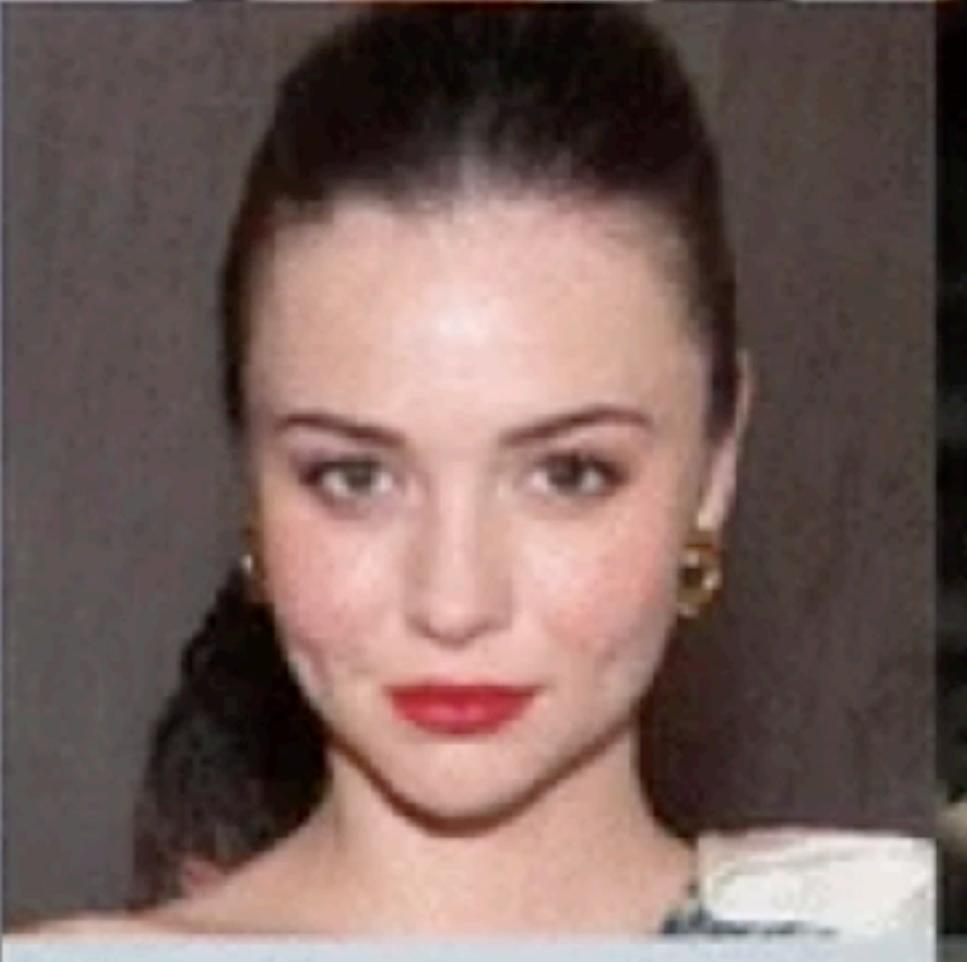
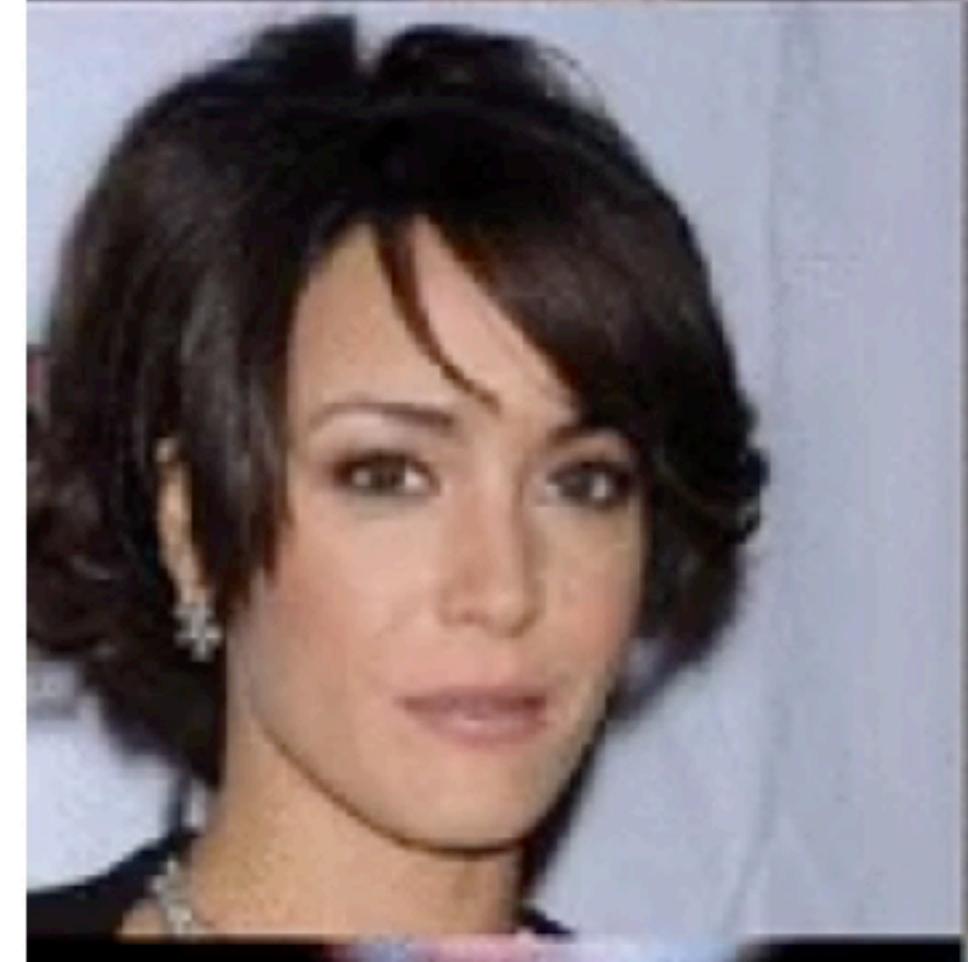
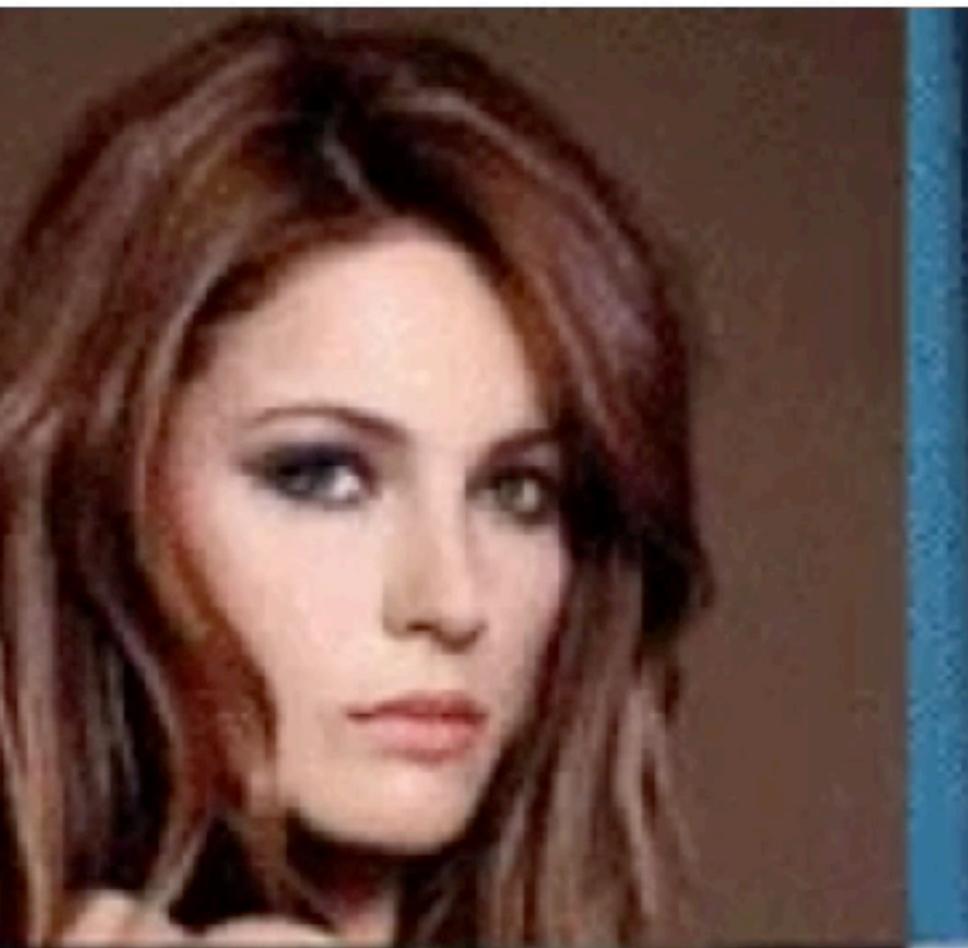
(Observed space) x





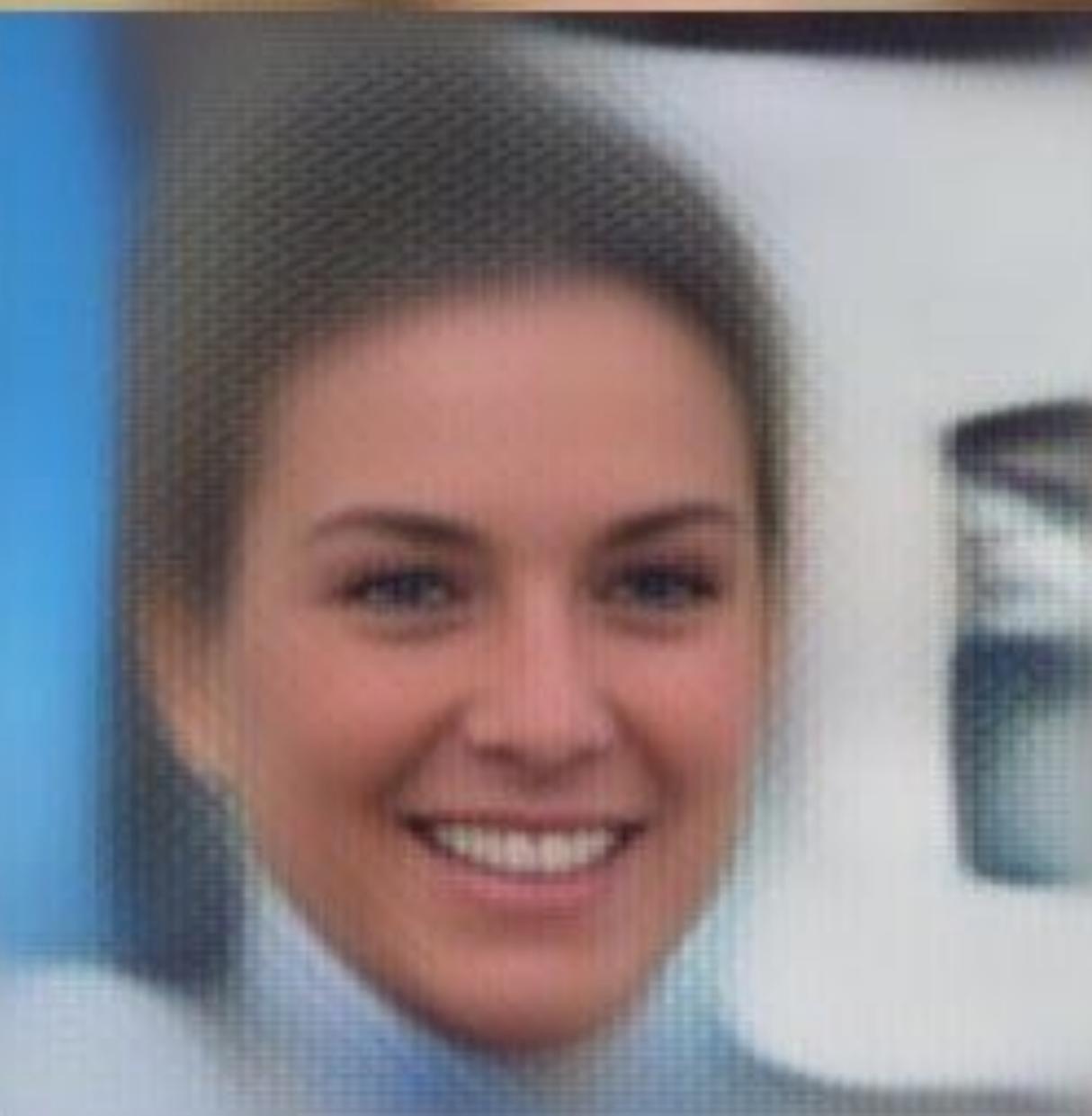
The relationship between Source #1 and Source #2 is applied to the target image.



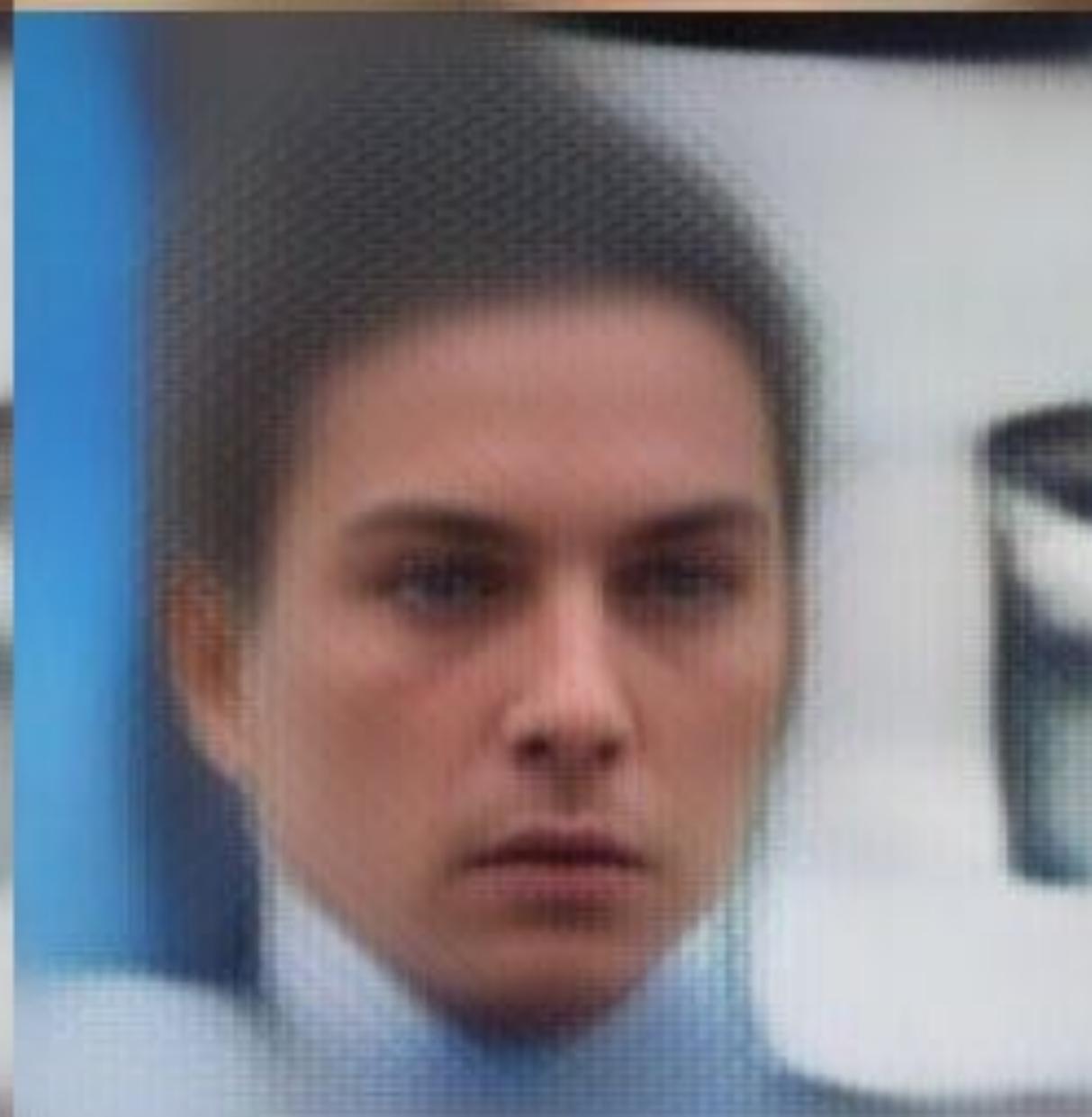


GIF

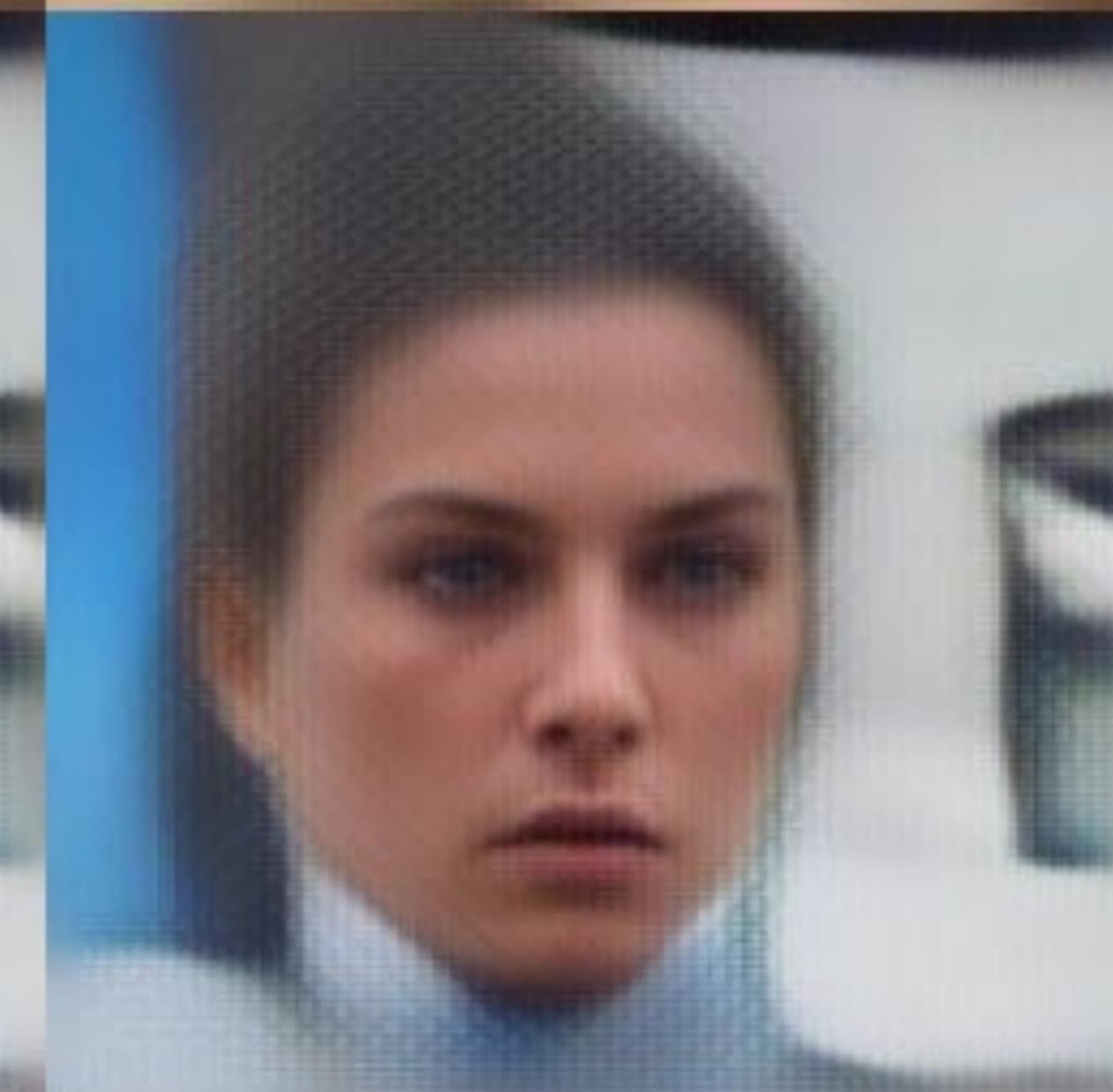
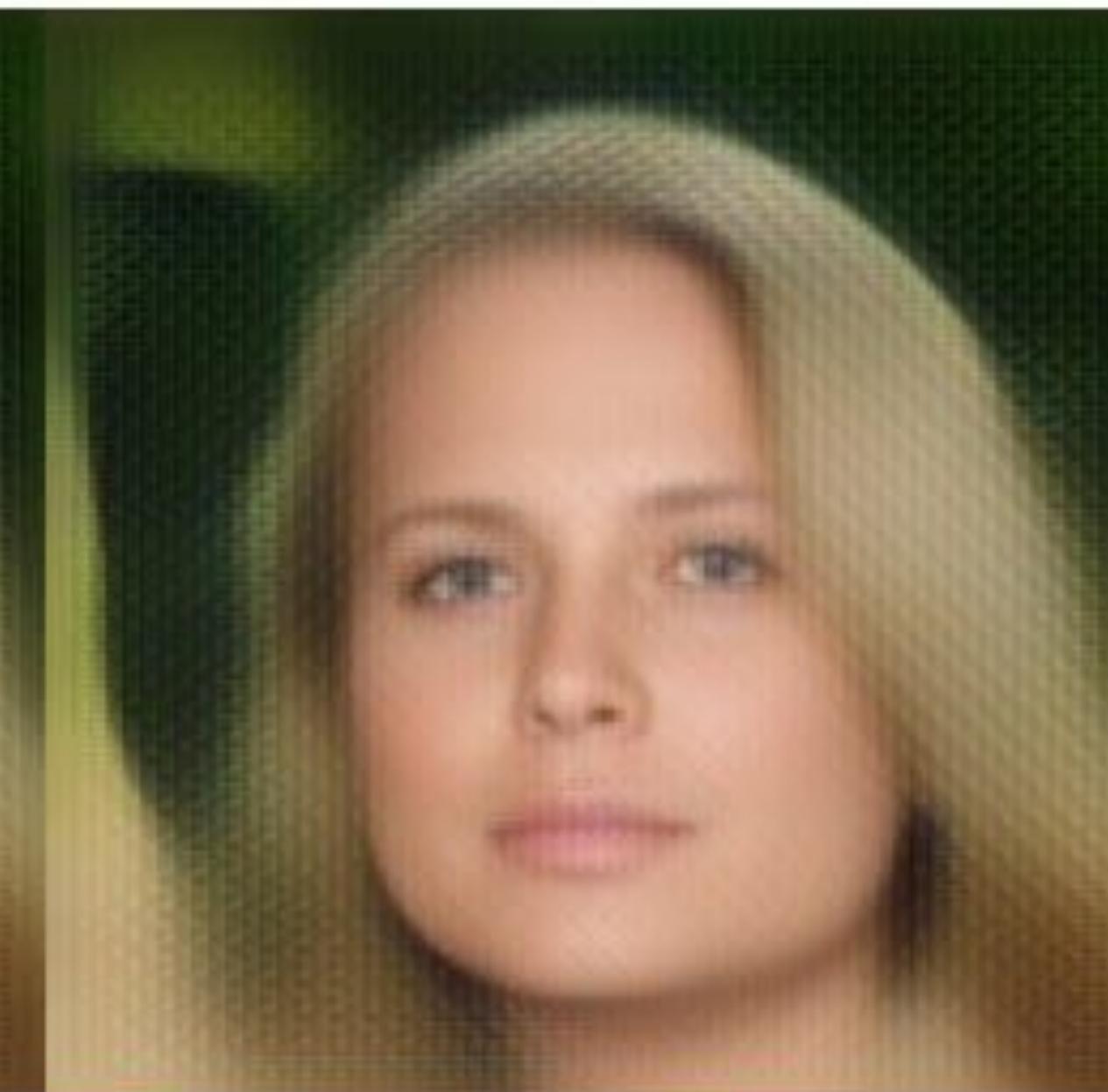
Reconstruction



Gender-biased
smilevector



Gender-balanced
smilevector







GAN state of the art

Progressive Growing of GANs

video: results

video: One hour of imaginary celebrities

...de máig nem könnyű GAN modelleket tanítani



Eddig csak az erényeit soroltam, de a deep learning nem varázsszer

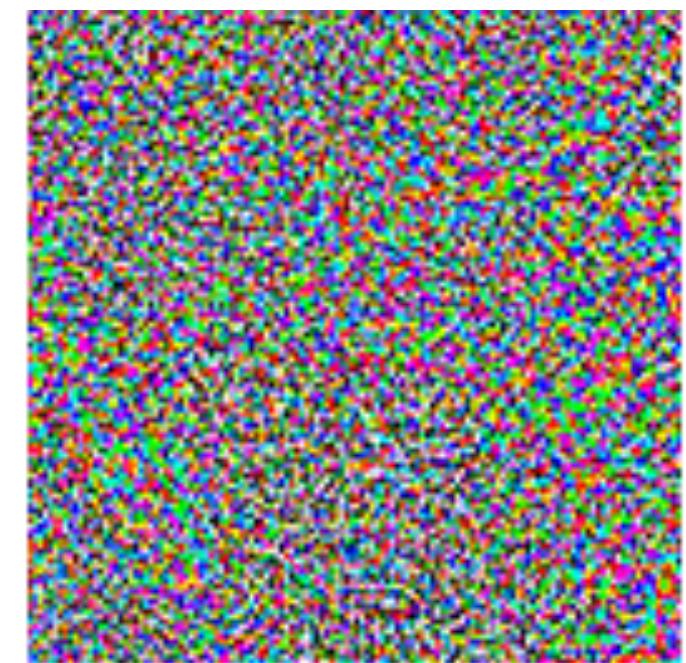
- Hatalmas az adat-igénye.
- Hatalmas a számítási kapacitás-igénye.
- Nincs szisztematizált matematikai vagy mérnöki megalapozása. Főleg próbálgtatás illetve szájhagyomány útján terjedő babonák alapján tervezik a hálózatokat.
- Kevés sikert mutathat fel azokon a területeken, ahol nem lehet olyan eltolás-invarienciákat kihasználni, mint a képeknél és a hangoknál.

...és még egy nagy gond

Adversarial perturbation



$+ \epsilon$



=



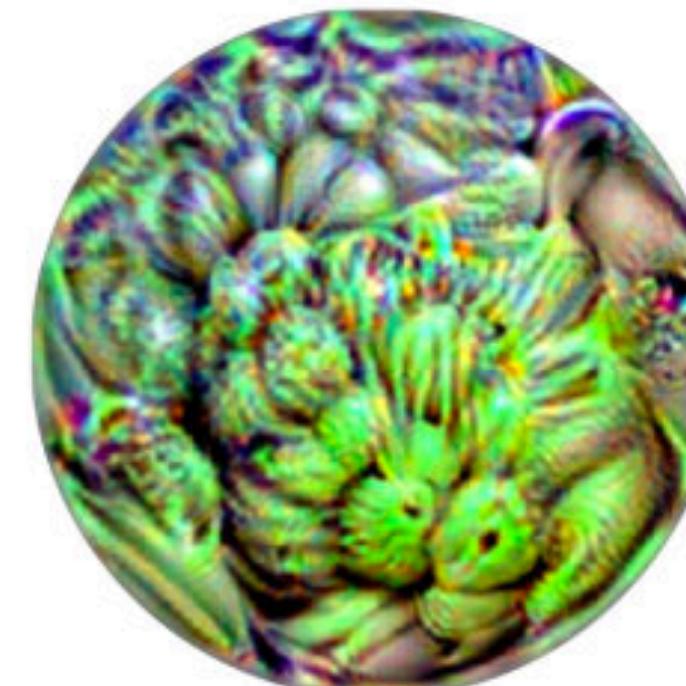
"panda"

57.7% confidence

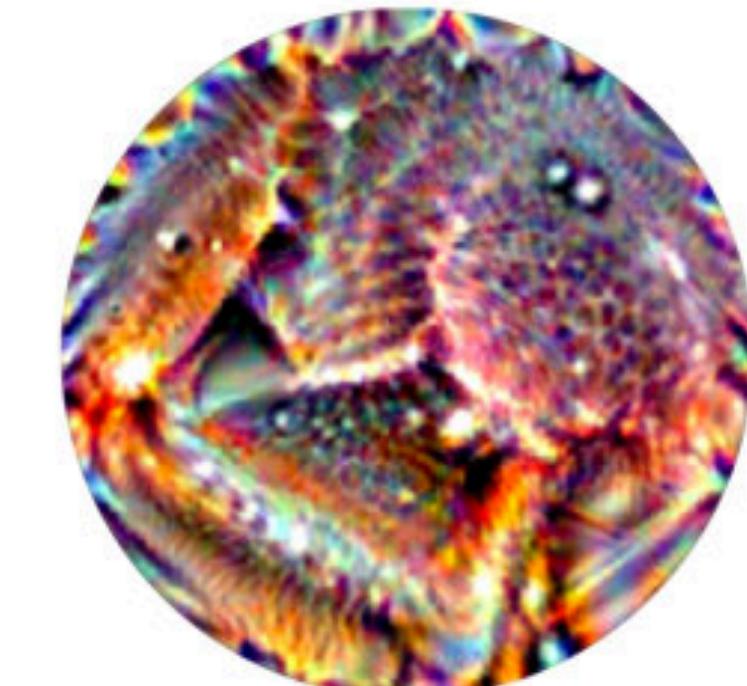
"gibbon"

99.3% confidence

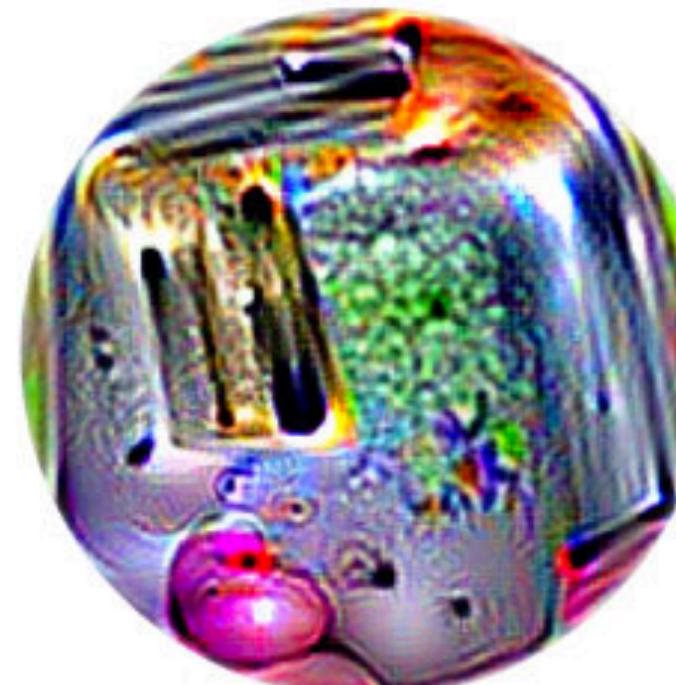
Adversarial patch



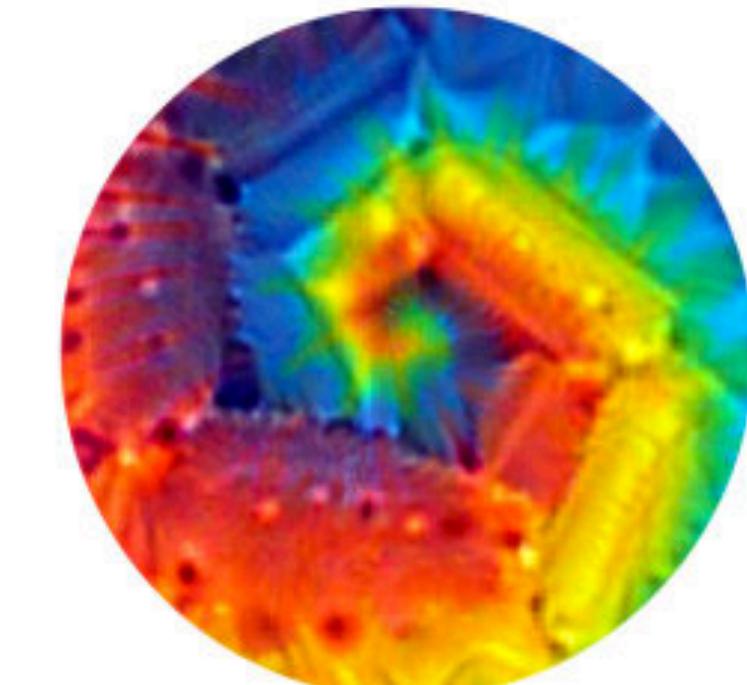
Banana



Crab



Toaster



Crab (disguised)

videó: fizikailag
megvalósított
adversarial példa

Köszönöm a figyelmet!

A következő héten a mesterséges
neuronhálók pontos bevezetése, tanításuk
alapelvei.