

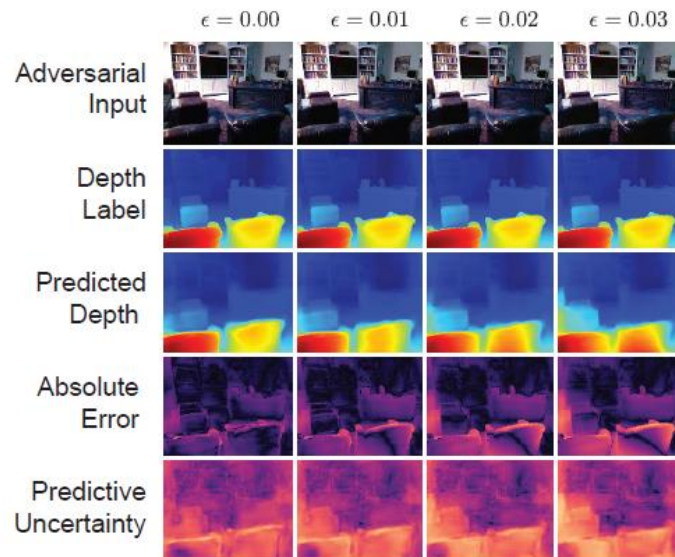
Csapó Tamás Gábor

Deep Learning a gyakorlatban Python és LUA alapon

Hiperparaméter optimalizálás & praktikus tanácsok

Deep learning híradó (1)

- Neural network - when it should not be trusted?
 - Current methods: often 99% accuracy
 - But what about the remaining 1% - e.g. healthcare?
 - How can we detect those situations reliably and efficiently?
 - Uncertainty analysis
 - *evidential distributions* directly capture the model's confidence in its prediction
 - e.g. analyze a monocular color image and estimate a depth value (i.e. distance from the camera lens) for each pixel.
 - Result: high uncertainty for pixels where it predicted the wrong depth
- ... will be presented in December at NeurIPS



Forrás: <https://news.mit.edu/2020/neural-network-uncertainty-1120>

Deep learning híradó (2)

- NeurIPS 2020: December 6-12, virtual only
 - Registration
 - <https://nips.cc/Conferences/2020/Pricing2>
 - Full Time Student, **Conference + Workshops = \$25**
 - Keynotes
 - <https://neuripsconf.medium.com/announcing-the-neurips-2020-keynote-speakers-ca6fbbb64206>
 - Proceedings
 - <https://proceedings.neurips.cc/paper/2020>
 - Glow-TTS: A Generative Flow for Text-to-Speech via Monotonic Alignment Search
 - The Hateful Memes Challenge: Detecting Hate Speech in Multimodal Memes
 - Teaching a GAN What Not to Learn
 - Gergely Neu, Ciara Pike-Burke: A Unifying View of Optimism in Episodic Reinforcement Learning



5. kis házi feladat:

Hiperparaméter optimalizálás

Készíts kísérletsorozatot, melyben automatikus hiperparaméter optimalizálást végzel a CIFAR-10 adatbázison képosztályozási feladatra. Legyen legalább 100 tanítás, melynek eredményeit elemzed is (pl. hálózat mérete és az osztályozás pontossága hogyan függ össze?).

Tetszőleges deep learning keretrendszert és github forráskódokat (megfelelően hivatkozva) használhatsz. A megoldást részletesen kommentezd és a tanítás kimenetét (log fájlját) is mellékelj. A leírás és kommentek angolul legyenek!

<https://github.com/BME-SmartLab-Education/vitmav45/blob/master/kishazi/5-kis-hazi-feladat.md>

Beadási határidő: 2020. december 8. 23:59

Jogi nyilatkozat

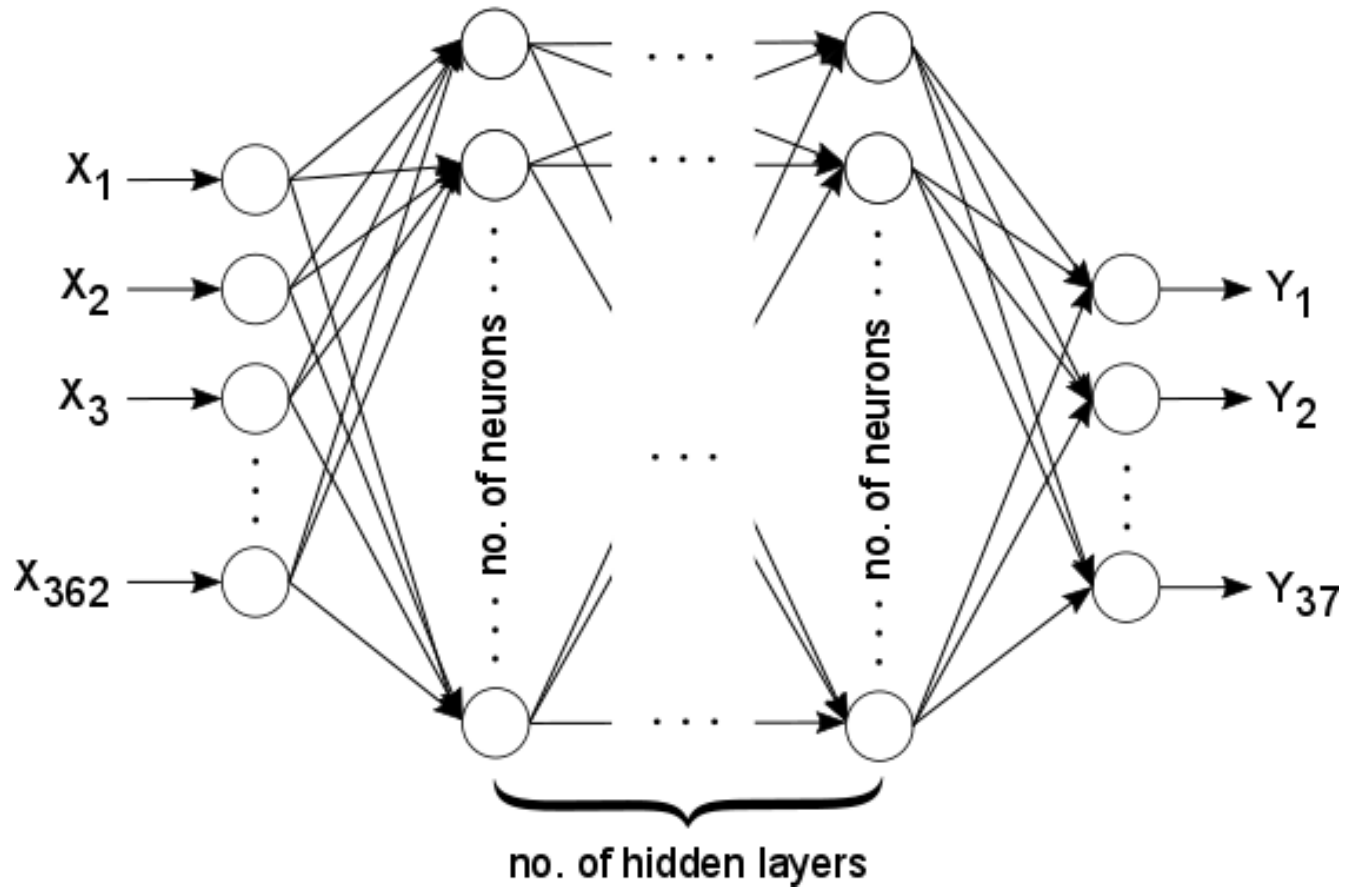
Jelen előadás diái a „*Deep Learning a gyakorlatban Python és LUA alapon*” című tantárgyhoz készültek és letölthetők a <http://smartlab.tmit.bme.hu> honlapról.

A diák nem helyettesítik az előadáson való részvételt, csupán emlékeztetőül szolgálnak.

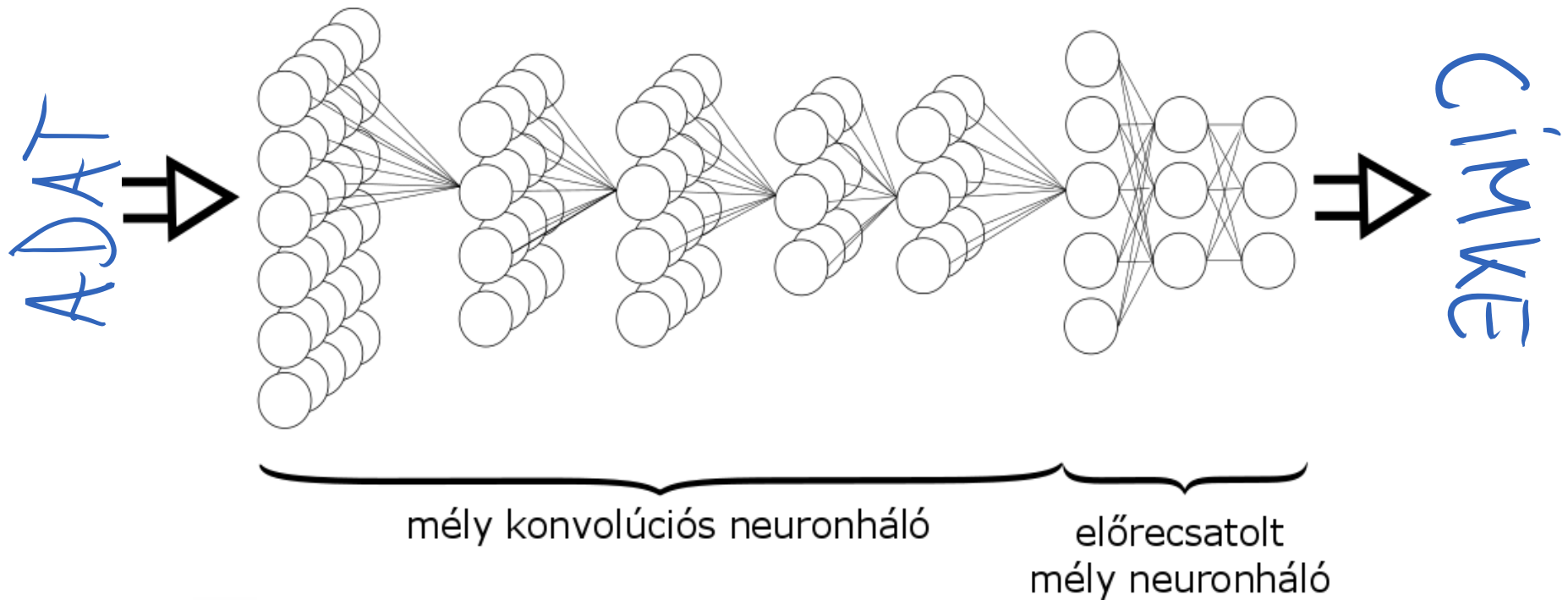
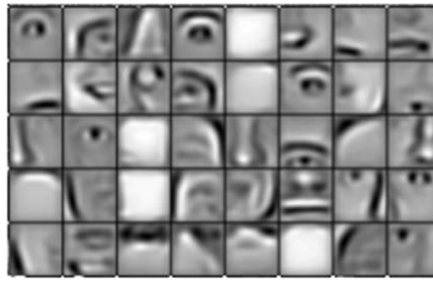
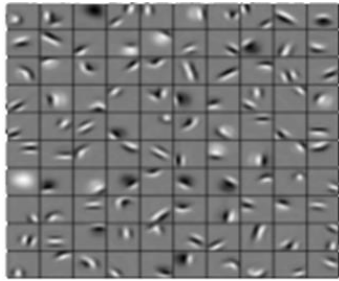
Az előadás diái a szerzői jog védelme alatt állnak. Az előadás diáinak vagy bármilyen részének újra felhasználása, terjesztése, megjelenítése csak a szerző írásbeli beleegyezése esetén megengedett. Ez alól kivétel, mely diákon külső forrás külön fel van tüntetve.

Hálózat architektúrák (ism.)

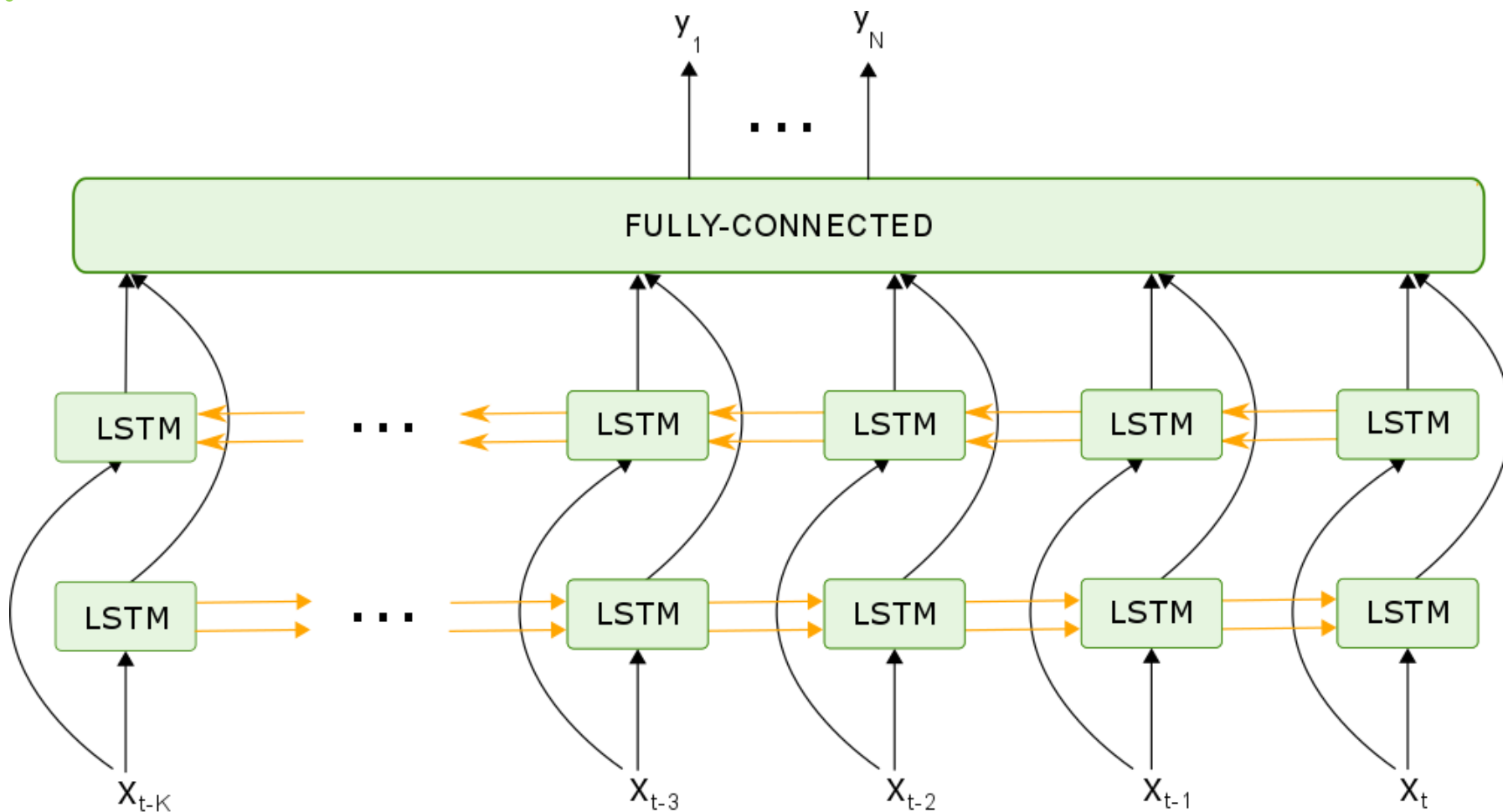
FC



CNN (1D / 2D / 3D / ...)



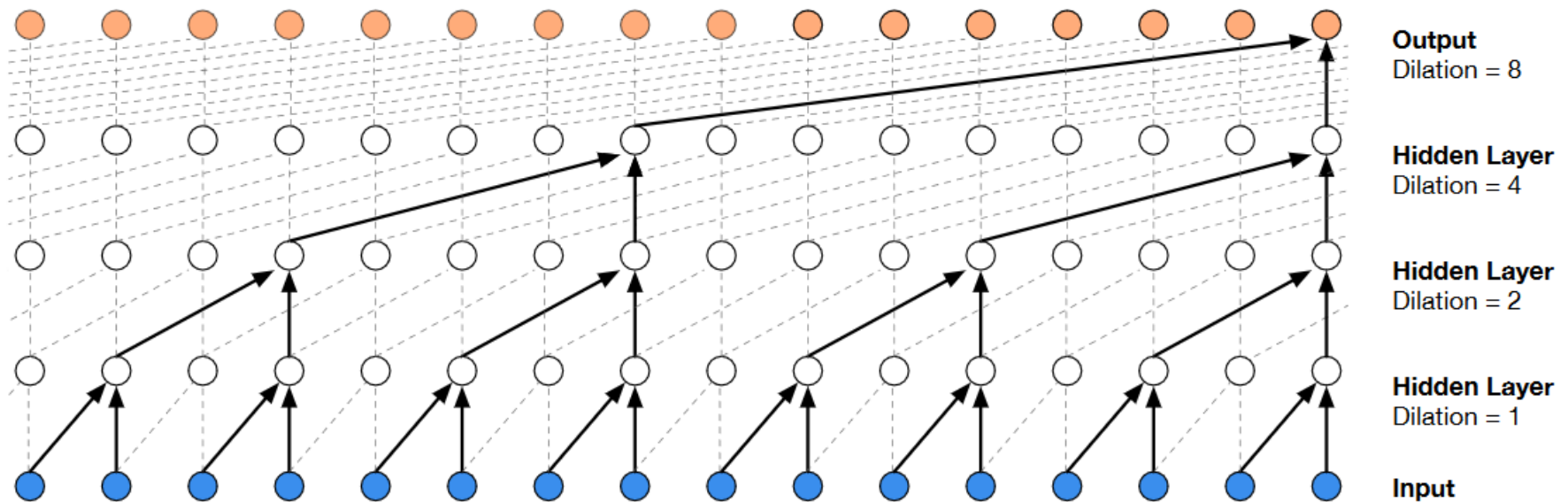
Kétirányú LSTM (Bidirectional LSTM, BLSTM)



Minden LSTM réteghez külön osztott súlyok.

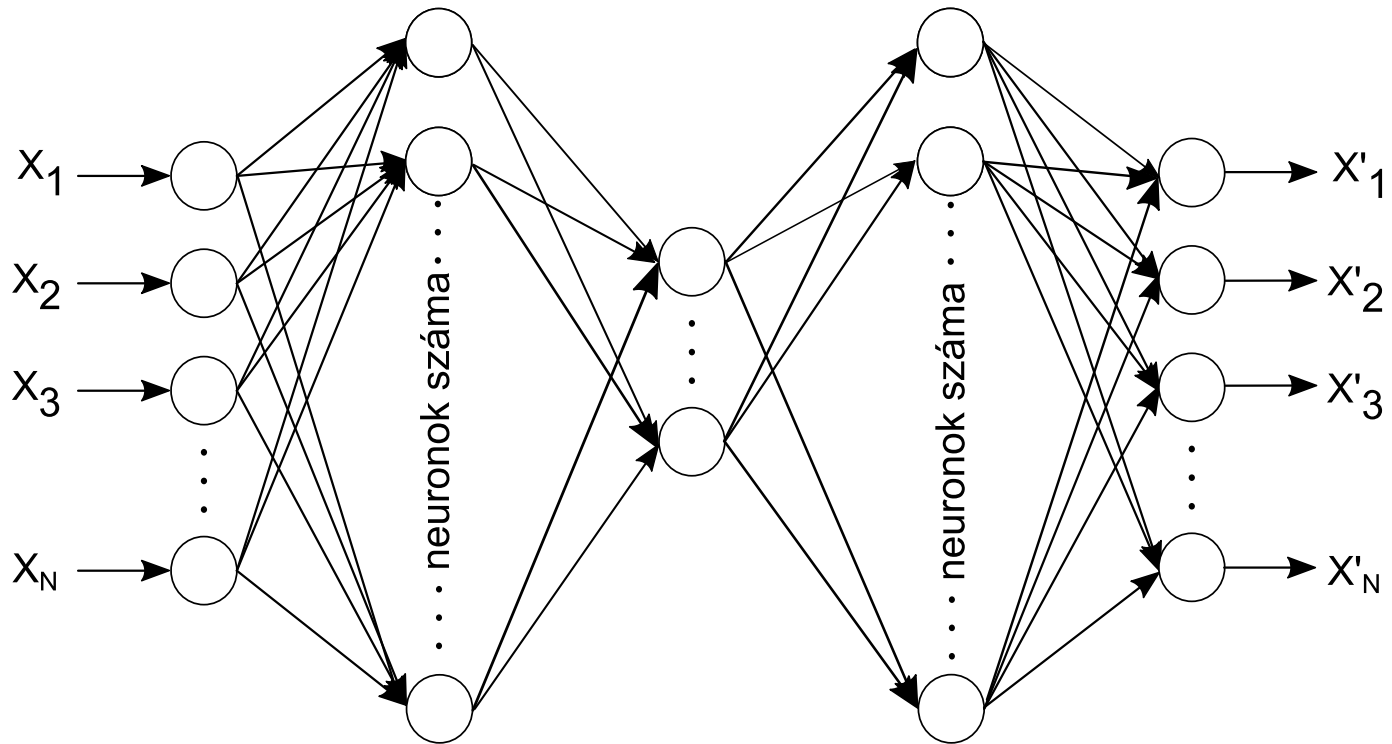
Komplex architektúrák, pl. WaveNet

- Dilated convolution

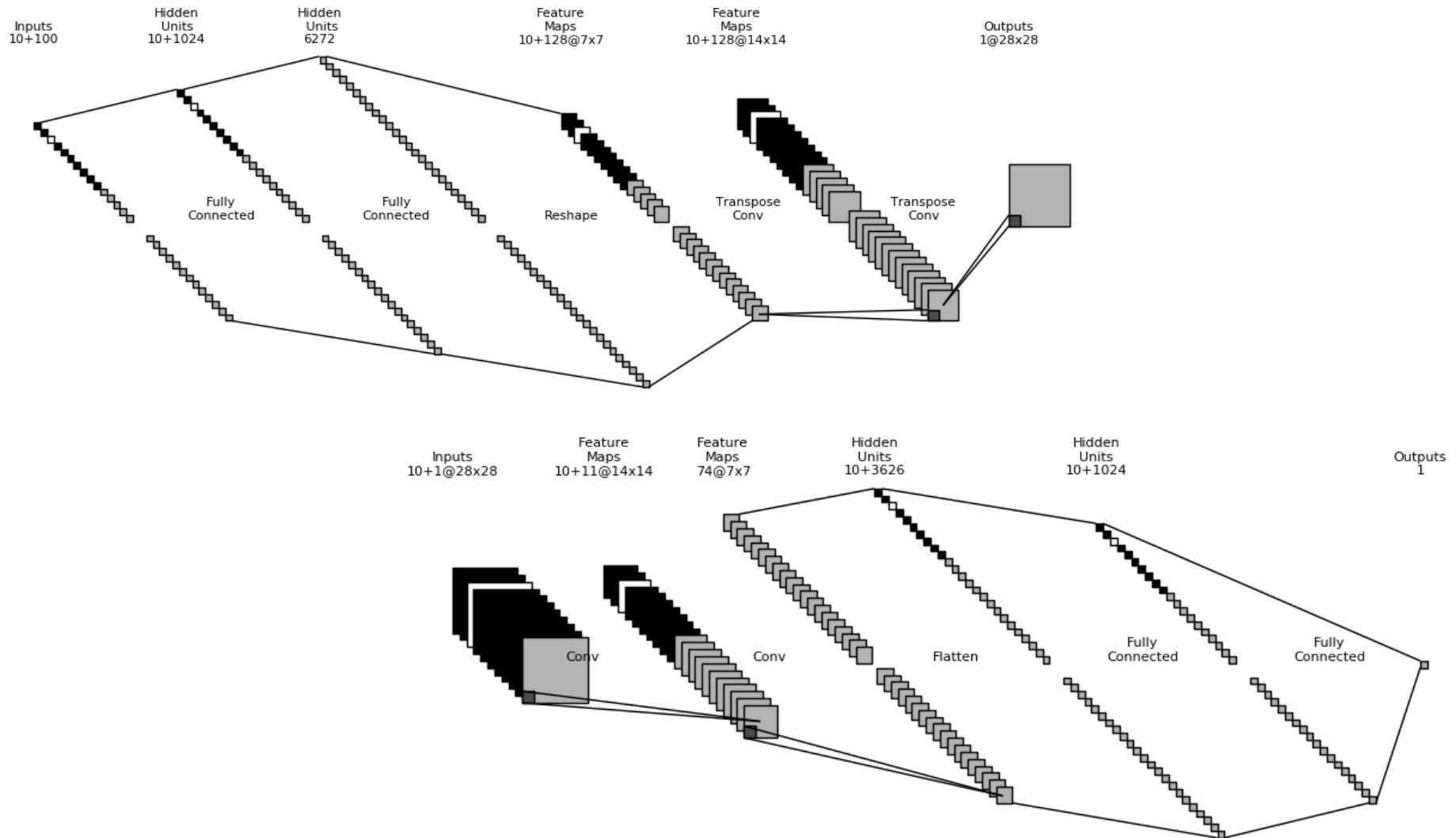


- 3 rejtett réteg -> 16 minta

AE

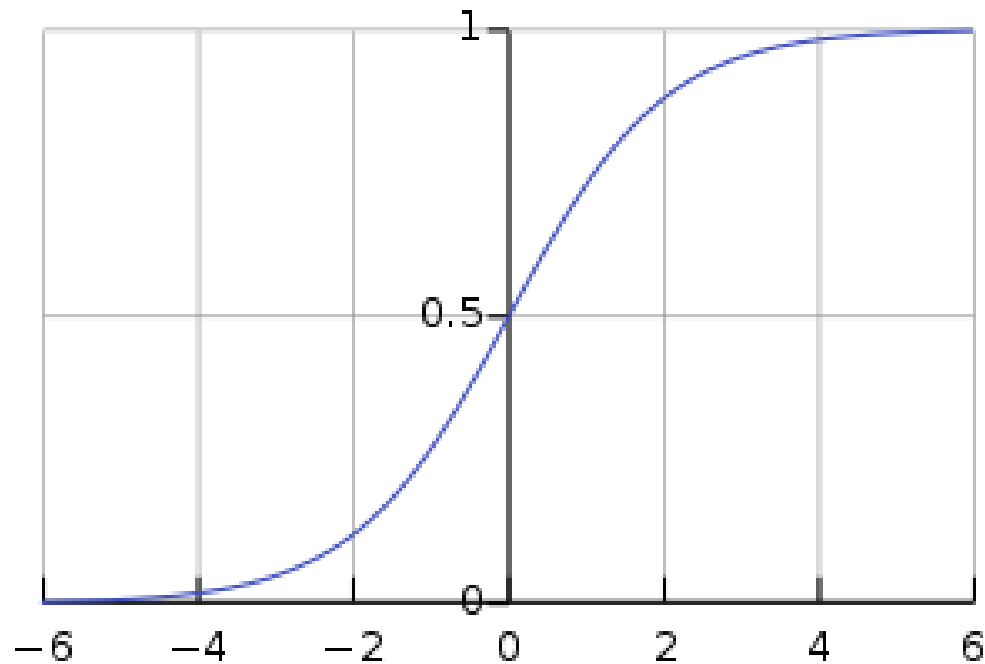


GAN



Aktivációs függvények (ism.)

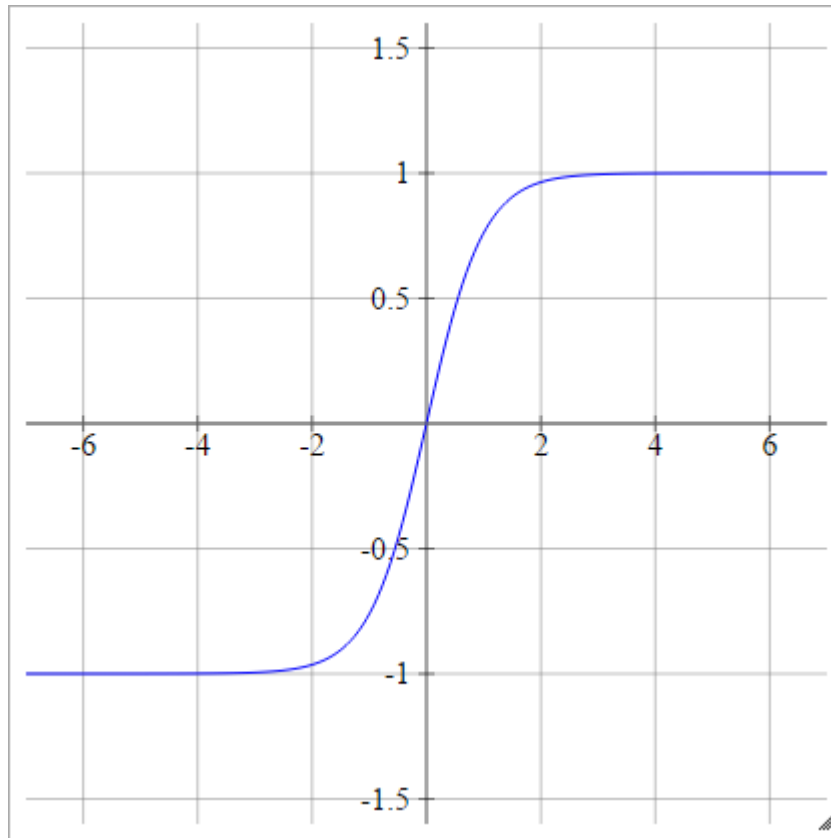
Sigmoid



$$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}}$$

Forrás: https://en.wikipedia.org/wiki/Sigmoid_function

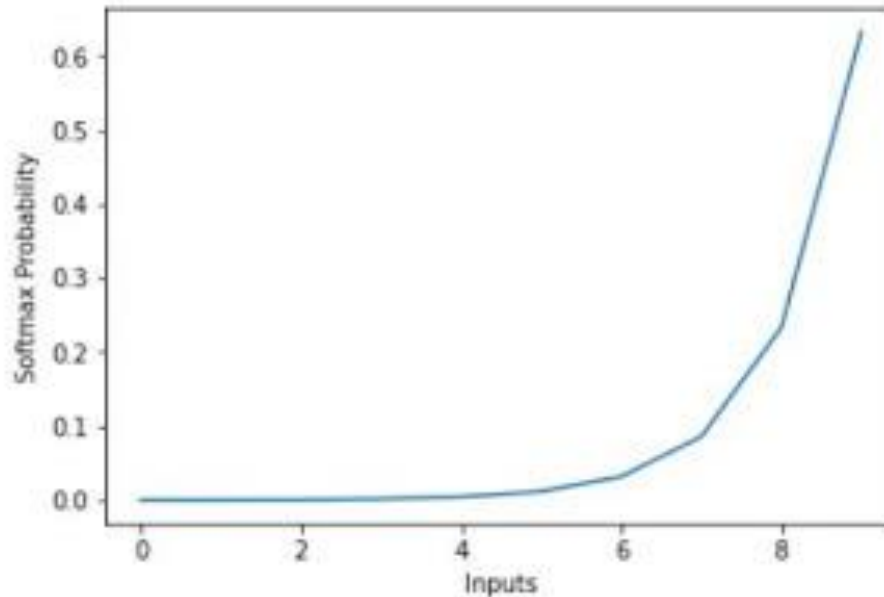
Tanh



$$f(x) = \tanh(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})}$$

Forrás: <https://stats.stackexchange.com/questions/115258/comprehensive-list-of-activation-functions-in-neural-networks-with-pros-cons>

Softmax

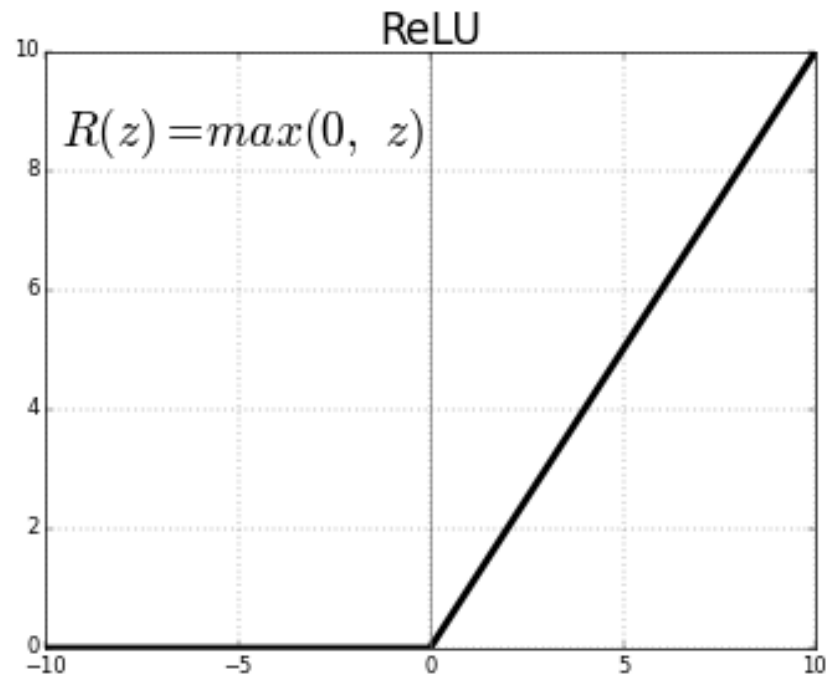


z is a vector of the inputs to the output layer (if you have 10 output units, then there are 10 elements in z)

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}}$$

Forrás: <https://github.com/Kulbear/deep-learning-nano-foundation/wiki/ReLU-and-Softmax-Activation-Functions>

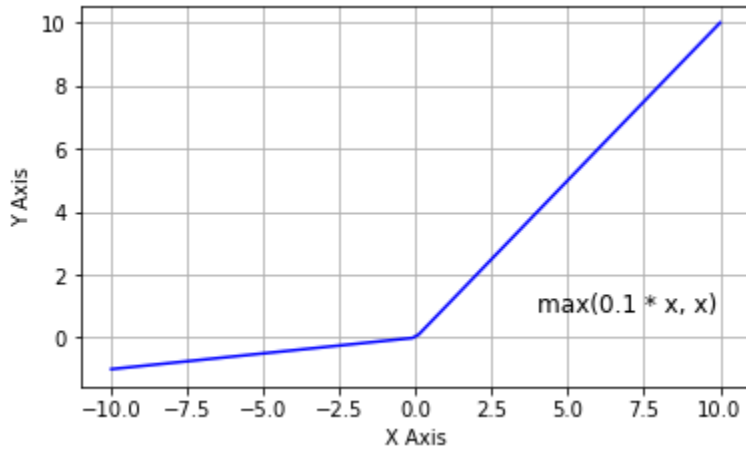
ReLU



$$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$$

Forrás: <https://medium.com/@kanchansarkar/relu-not-a-differentiable-function-why-used-in-gradient-based-optimization-7fef3a4cecec>

Leaky ReLU, PReLU

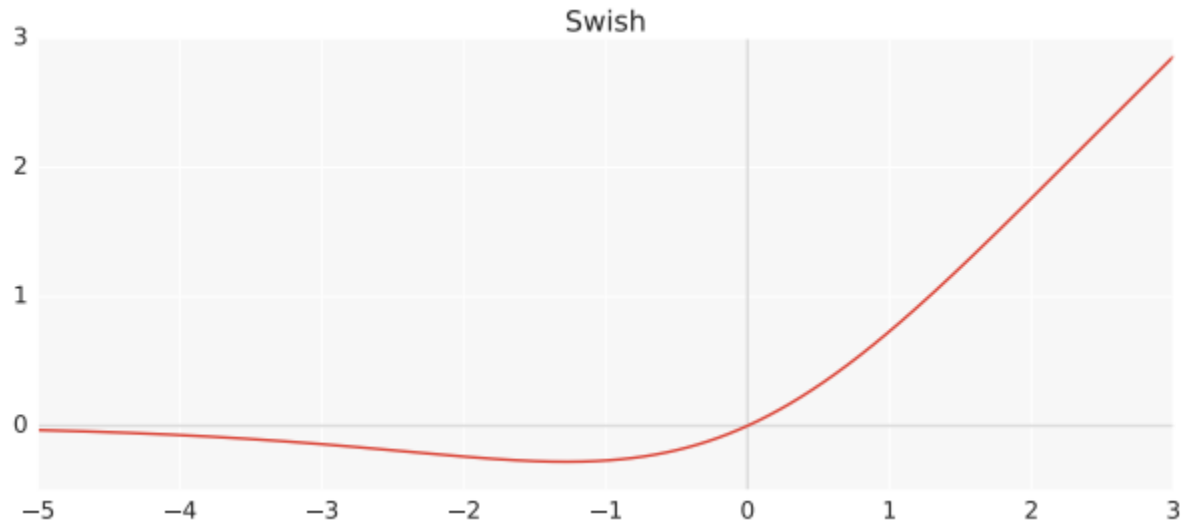


$$f(x) = \begin{cases} 0.01x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$$

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ ax & \text{otherwise} \end{cases}$$

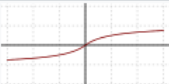
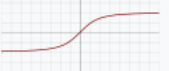
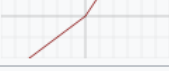
Forrás: <https://medium.com/@kanchansarkar/relu-not-a-differentiable-function-why-used-in-gradient-based-optimization-7fef3a4cecec>

SWISH



$$\begin{aligned} f(x) &= x * \text{sigmoid}(x) \\ &= x * (1 + e^{-x})^{-1} \end{aligned}$$

Forrás: <https://medium.com/@neuralnets/swish-activation-function-by-google-53e1ea86f820>

Name ↕	Plot ↕	Equation ↕	Derivative (with respect to x) ↕
Identity		$f(x) = x$	$f'(x) = 1$
Binary step		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x \neq 0 \\ ? & \text{for } x = 0 \end{cases}$
Logistic (a.k.a. Sigmoid or Soft step)		$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}}$ [1]	$f'(x) = f(x)(1 - f(x))$
Tanh		$f(x) = \tanh(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})}$	$f'(x) = 1 - f(x)^2$
ArcTan		$f(x) = \tan^{-1}(x)$	$f'(x) = \frac{1}{x^2 + 1}$
ElliotSig ^{[9][10][11]} Softsign ^{[12][13]}		$f(x) = \frac{x}{1 + x }$	$f'(x) = \frac{1}{(1 + x)^2}$
Inverse square root unit (ISRU) ^[14]		$f(x) = \frac{x}{\sqrt{1 + \alpha x^2}}$	$f'(x) = \left(\frac{1}{\sqrt{1 + \alpha x^2}} \right)^3$
Inverse square root linear unit (ISRLU) ^[14]		$f(x) = \begin{cases} \frac{x}{\sqrt{1 + \alpha x^2}} & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} \left(\frac{1}{\sqrt{1 + \alpha x^2}} \right)^3 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Square Nonlinearity (SQNL) ^[11]		$f(x) = \begin{cases} 1 & : x > 2.0 \\ x - \frac{x^2}{4} & : 0 \leq x \leq 2.0 \\ x + \frac{x^2}{4} & : -2.0 \leq x < 0 \\ -1 & : x < -2.0 \end{cases}$	$f'(x) = 1 \mp \frac{x}{2}$
Rectified linear unit (ReLU) ^[15]		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Leaky rectified linear unit (Leaky ReLU) ^[16]		$f(x) = \begin{cases} 0.01x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0.01 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Parameteric			

Forrás: https://en.wikipedia.org/wiki/Activation_function

SWISH vs ReLU

- Ramachandran et al. 2017
write that their
“extensive experiments show that Swish consistently matches or outperforms ReLU on deep networks applied to a variety of challenging domains such as image classification and machine translation”.

Forrás: <https://arxiv.org/pdf/1710.05941v1.pdf>

Optimizáció algoritmusok (ism.)

THE EVOLUTION OF GRADIENT DESCENT



Forrás: Siraj Raval, <https://www.youtube.com/watch?v=nhqo0u1a6fw> (1:15-től)

Praktikus tanácsok (ism.)

Train-validation-test set



70%

60

15%

20

15%

20

! 10% edelpont

pl. 1M edelpont

98%

93,5

1%

0,4

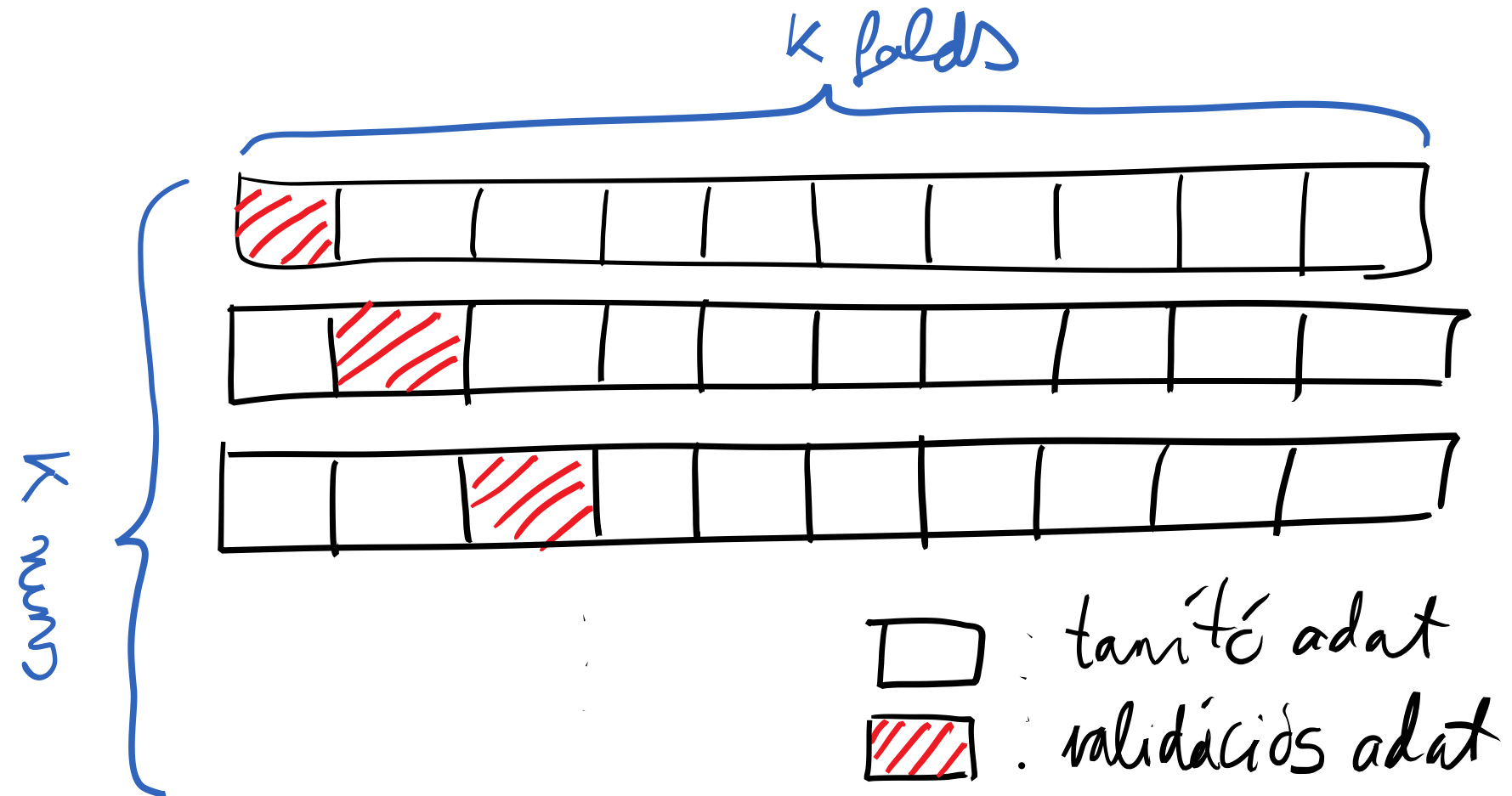
1%

0,1

Train-validation-test eloszlása

- Valódi alkalmazásban pl:
 - Tanítóadat: macska fényképek internetről
 - Dev/teszt: macska fényképek okostelefon kamerával
- Ökölszabály: legalább a dev & teszt legyen ugyanabból az eloszlásból!

Tanító-validációs-teszt adatok III.

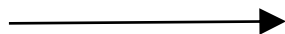


Adat dúsítás

- Data augmentation



4



4

4

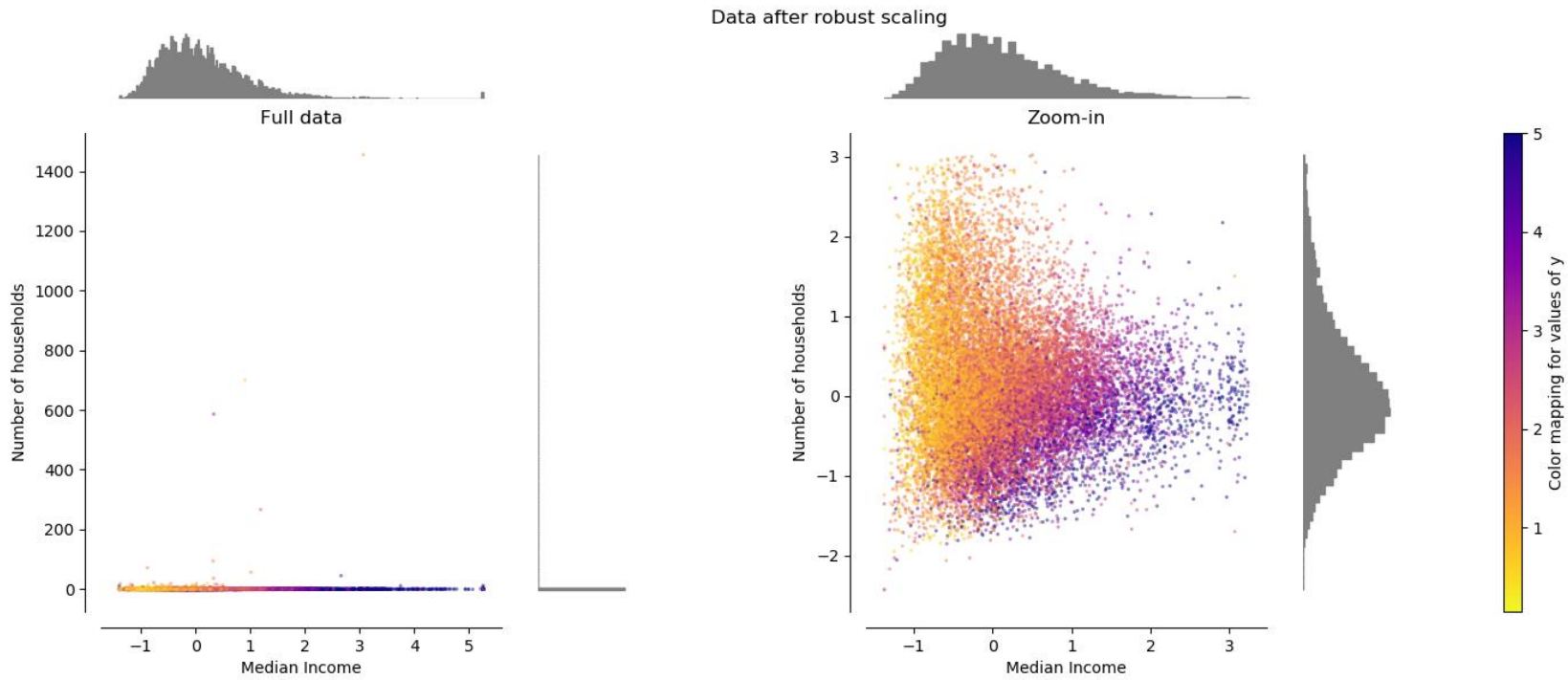
4

4

Forrás: <https://www.coursera.org/learn/deep-neural-network/home/welcome>

Adatok normalizálása / standardizálása

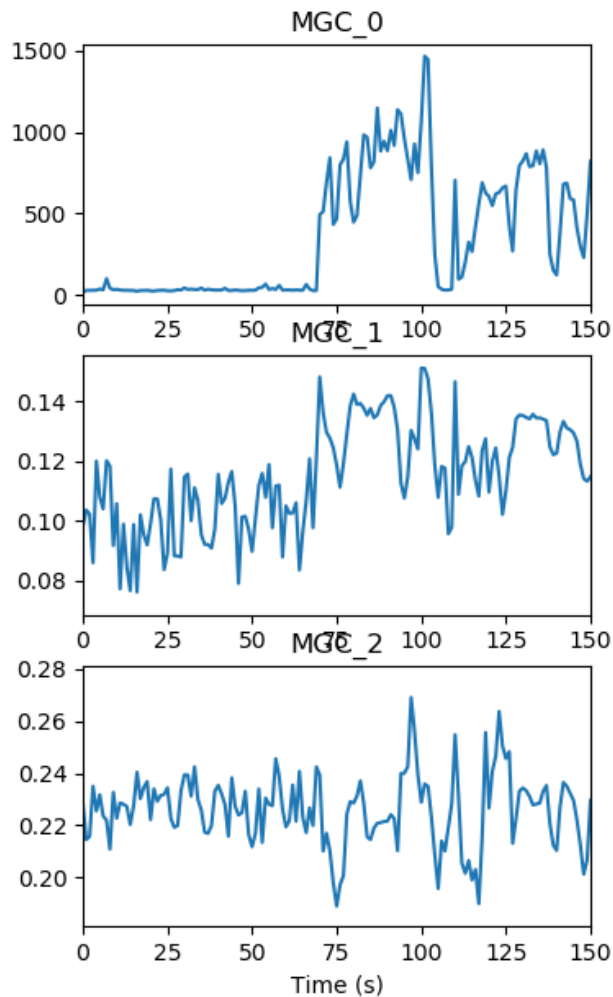
- MinMax, pl. [0-1]
- Normál eloszlás, pl. „zero mean, unit variance”
- RobustScaler



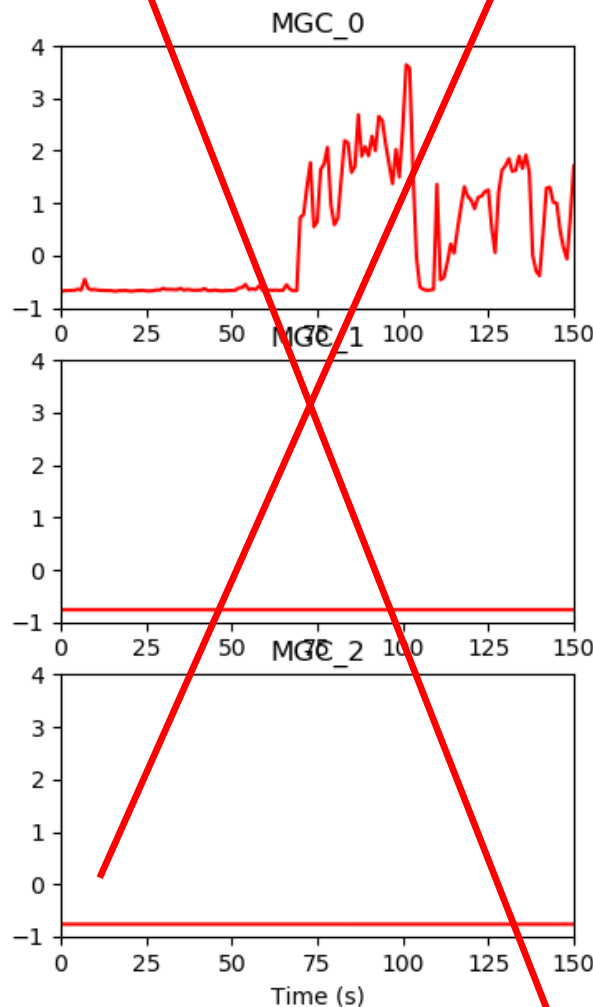
Forrás: https://scikit-learn.org/stable/auto_examples/preprocessing/plot_all_scaling.html

Jellemzőket érdemes megnézni!

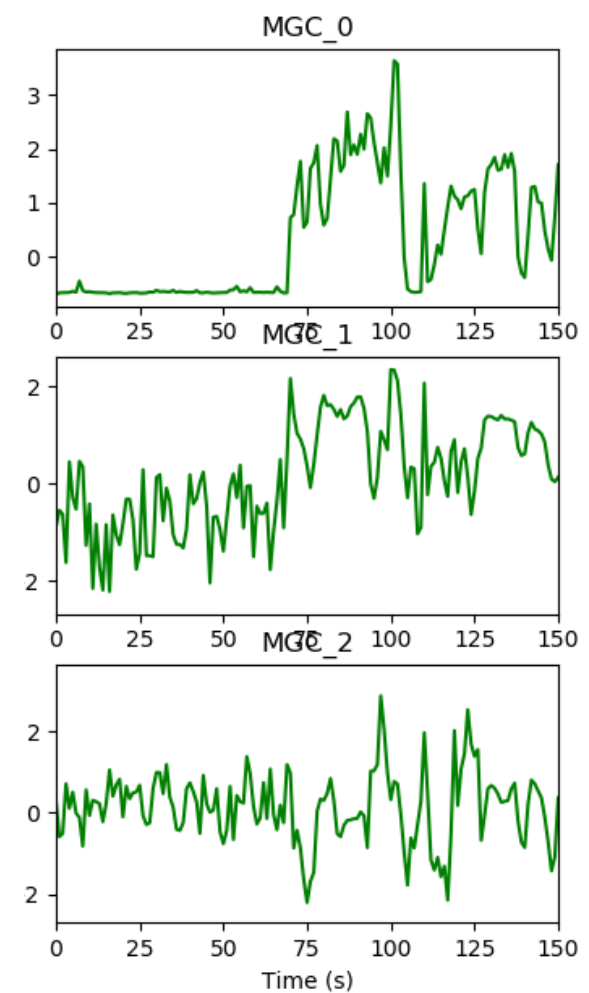
Skálázás előtt



Skálázás után / v1

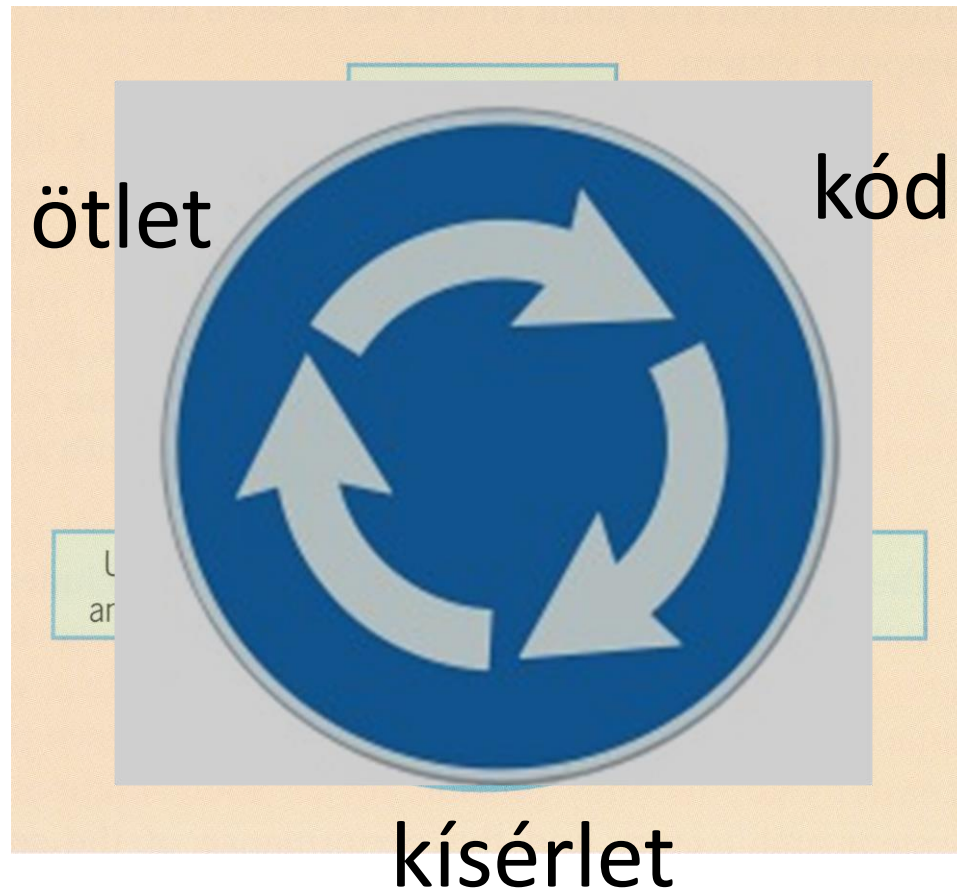


Skálázás után / v2



Hiperparaméterek

ML / DL: iteratív folyamat...

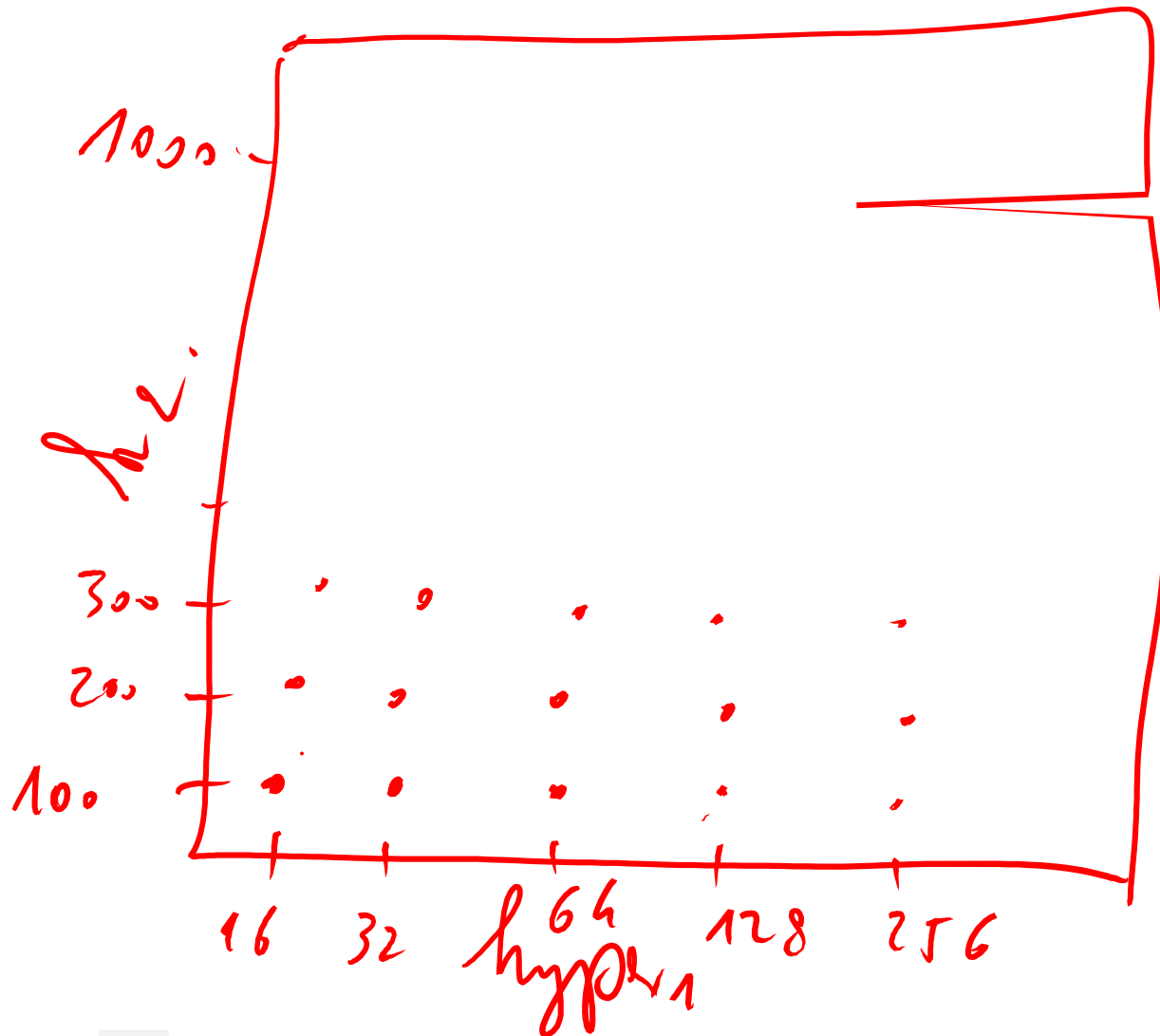


Hiperparaméterek beállítása

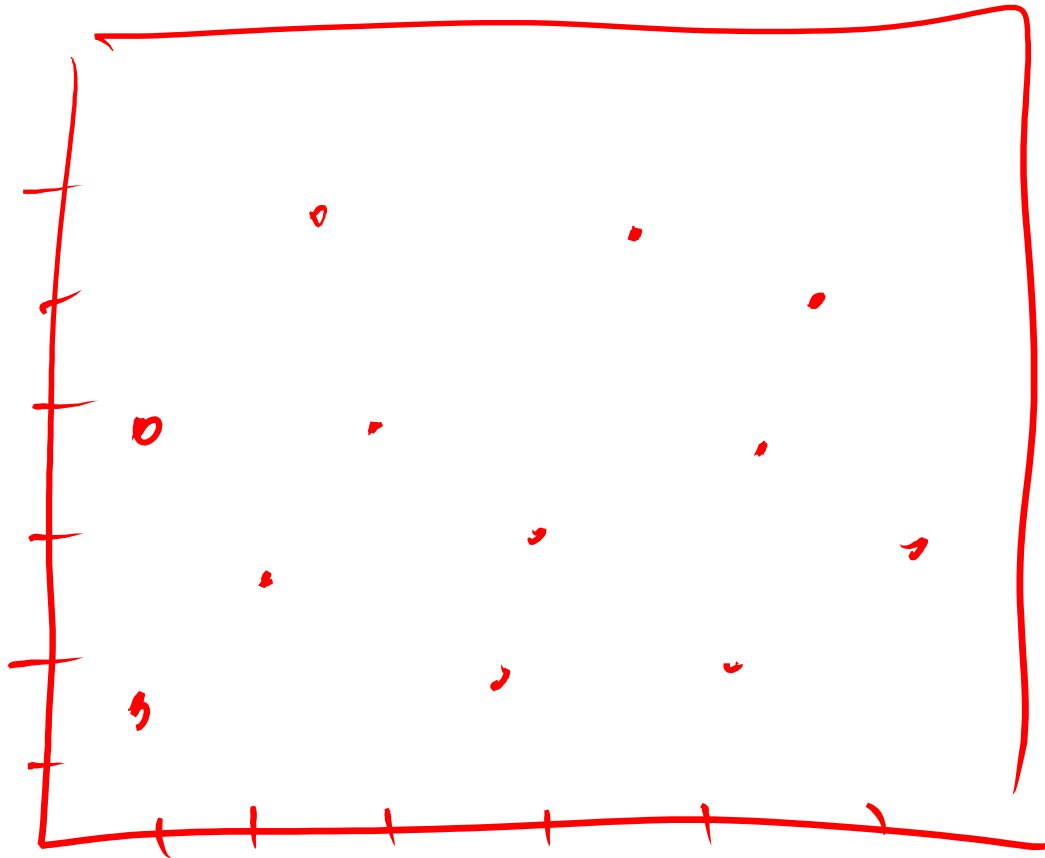
- Alpha Learning rate
- Beta Momentum
- Optimizáció
- Rétegek száma
- Neuronok száma
- Aktivációs függvények
- Batch méret

- Melyik fontosabb???

Grid search

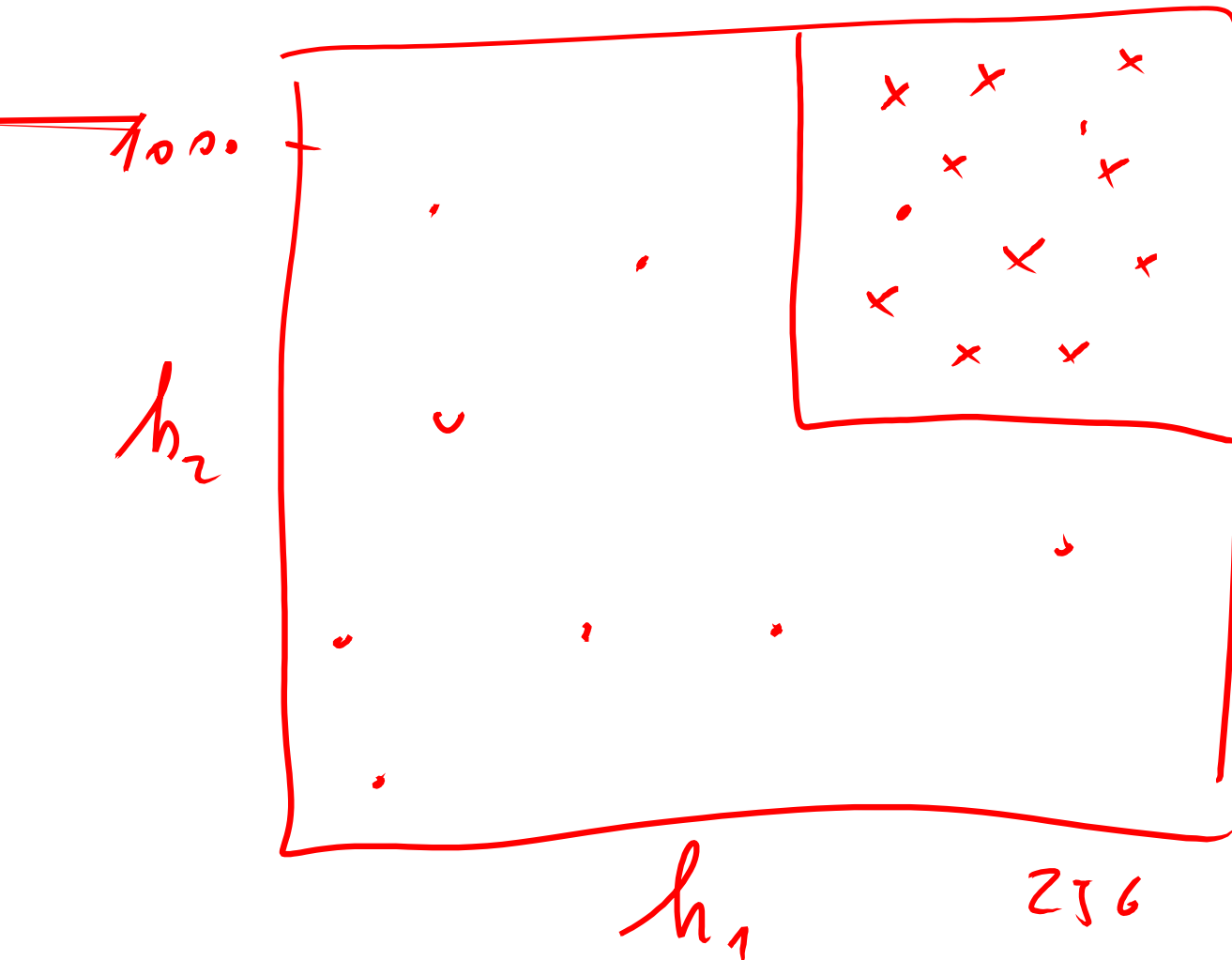


Random search



Forrás: <https://www.coursera.org/learn/deep-neural-network/home/welcome>

Először durva, majd finom részletek



Hiperparaméter, lineáris skála

rejtőrétegek: 2, 3, 4

neuronok: 100, 200, 300, ..., 1000

Hiperparaméter: milyen skálán?

$\alpha : 0,0001 \dots 1$

$\log$

$0,0001 / 0,001 / 0,01 / 0,1$

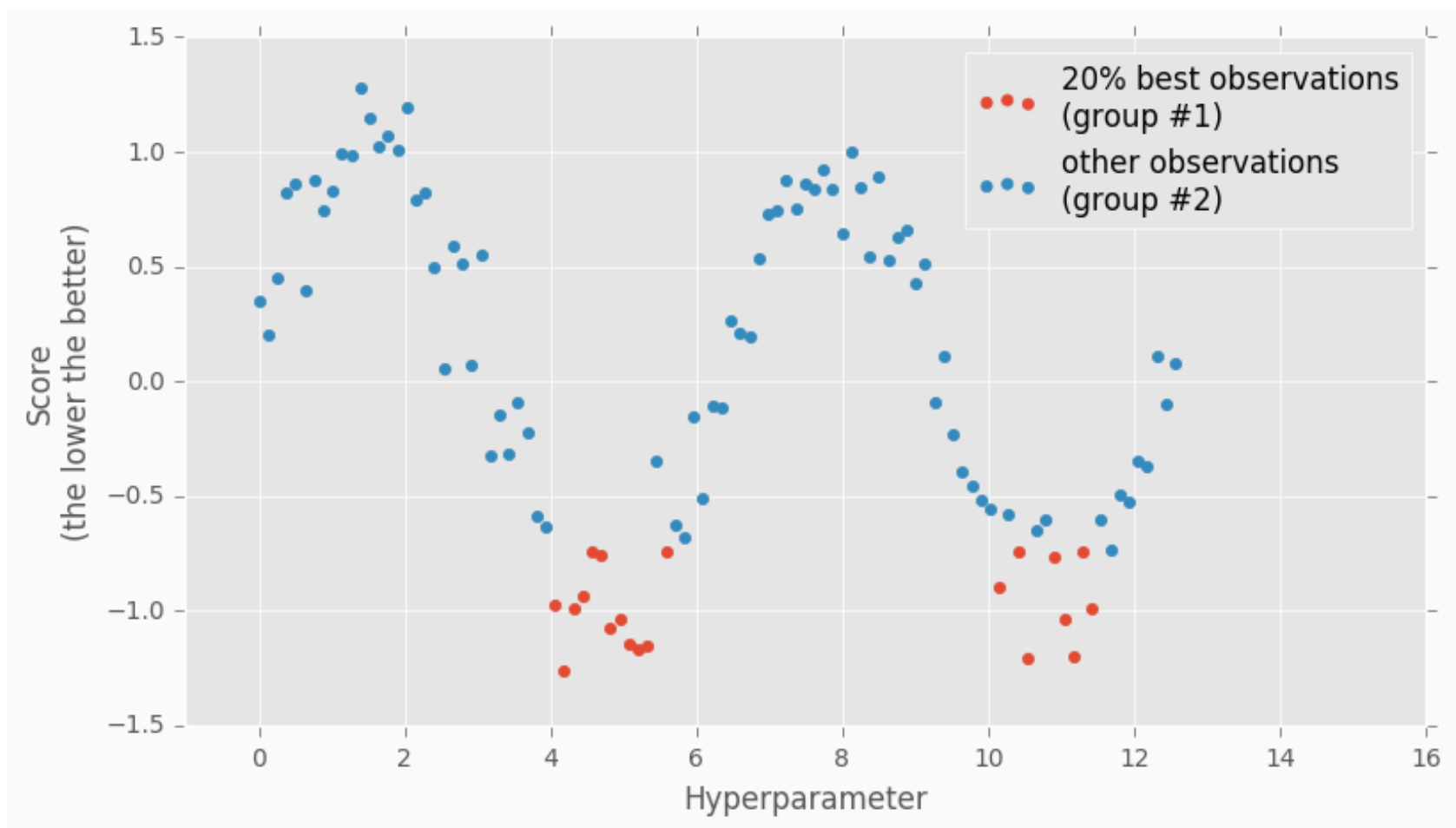
Bayesian hyperopt:

TPE : Tree-structured Parzen Estimators

- Kezdeti állapot: egyenletes vagy véletlen
- Minden iterációban eldönti, hogy melyik hiperparaméter halmazzal érdemes folytatni
- Iteráció után két csoport:
 - 1) Tesztadaton nagy pontosságot elérő paraméterek (pl. felső 10-25%)
 - 2) Maradék
 - Cél: olyan irányba folytassuk, ami 1)-et javítja
 - Nem csak egy „legjobb” paraméter csoport, hanem hiperparaméterek eloszlása

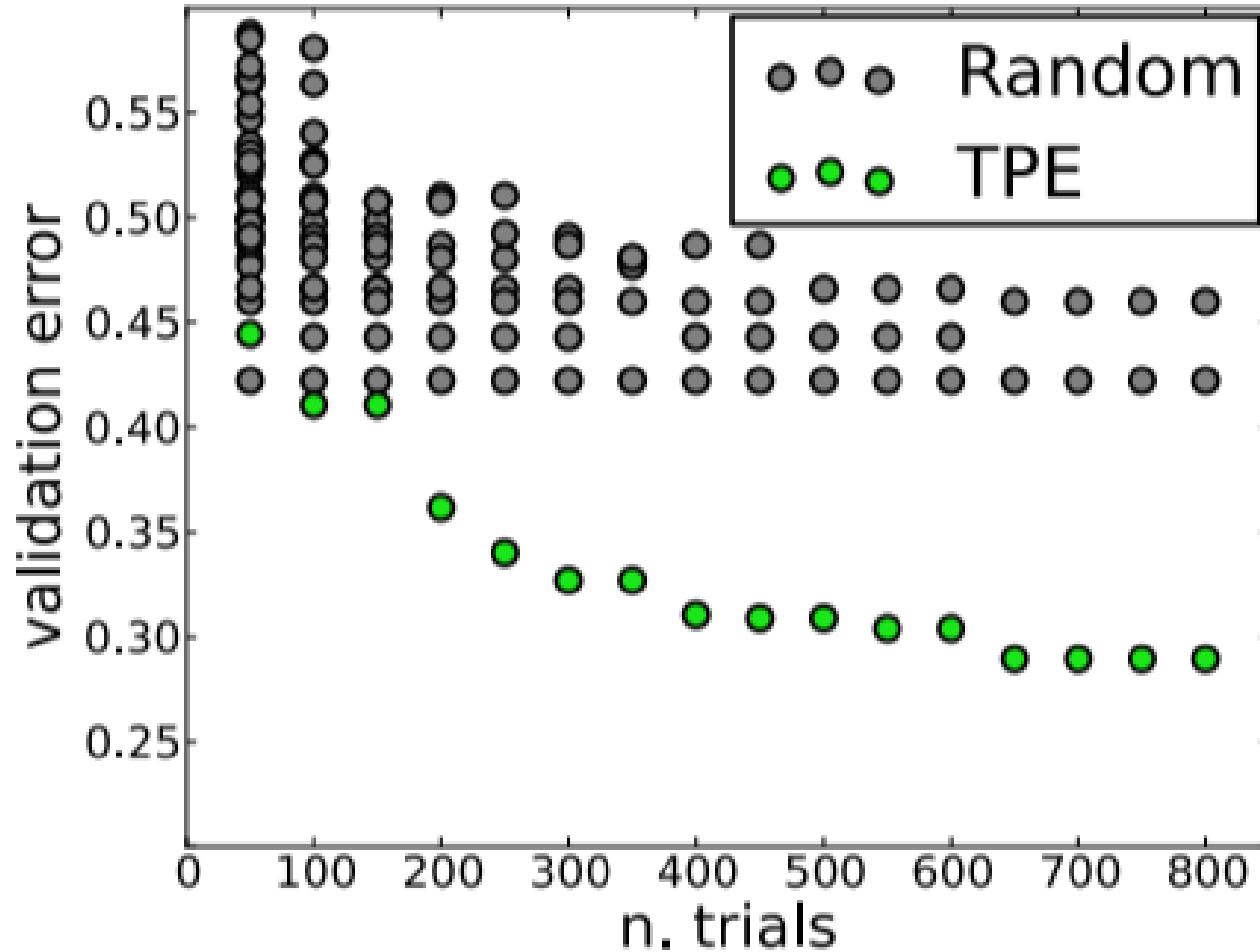
Forrás: http://neupy.com/2016/12/17/hyperparameter_optimization_for_neural_networks.html#tree-structured-parzen-estimators-tpe

TPE



Forrás: http://neupy.com/2016/12/17/hyperparameter_optimization_for_neural_networks.html#tree-structured-parzen-estimators-tpe

TPE vs random

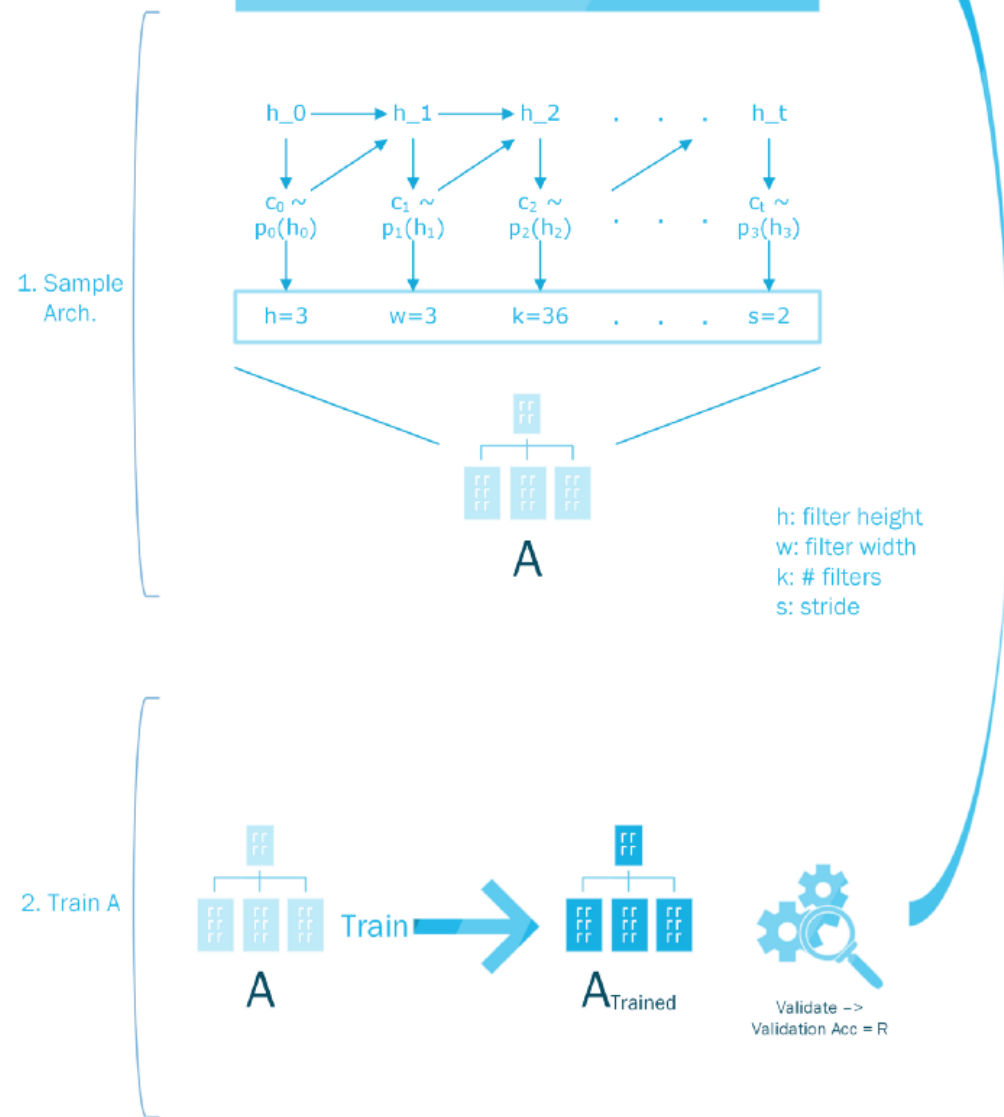


Forrás: <https://towardsdatascience.com/a-conceptual-explanation-of-bayesian-model-based-hyperparameter-optimization-for-machine-learning-b817227>

Neural Architecture Search (NAS)

- AutoML: optimális architektúra keresés & hyperopt
- NAS (ENAS, PNAS, DARTS, ...)
- betanítunk egy RNN vezérlőt RL-lel, ami automatikusan generál háló architektúrákat
- tanítás, majd kiértékelés a validációs halmazon, aminek eredménye a jutalom / büntetés
- analógia: manuális hiperopt, amikor érzése próbáljuk beállítani a paramétereket

NAS



Forrás: <https://towardsdatascience.com/neural-architecture-search-limitations-and-extensions-8141bec7681f>

Kahoot!

Névnek a NEPTUN
kódodat add meg!

<https://kahoot.it/>

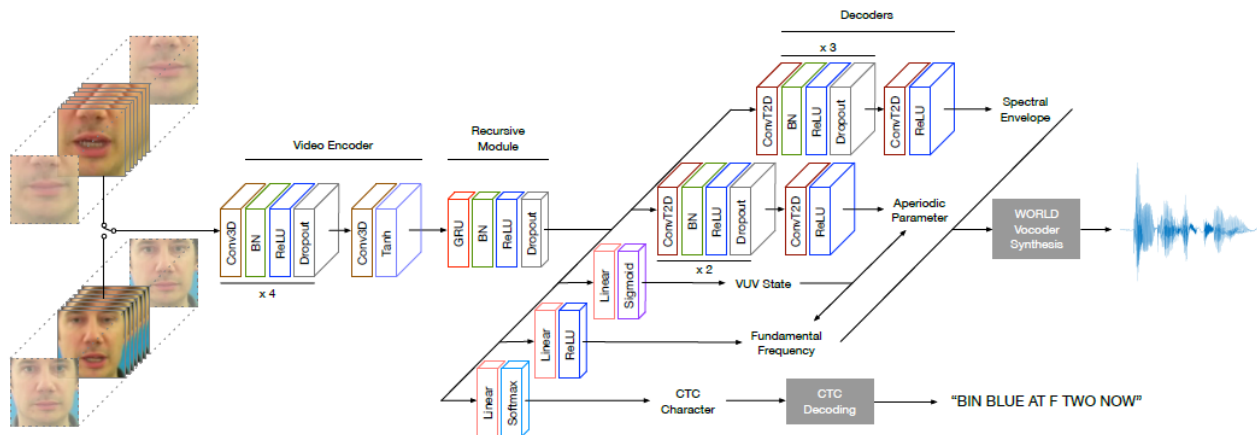


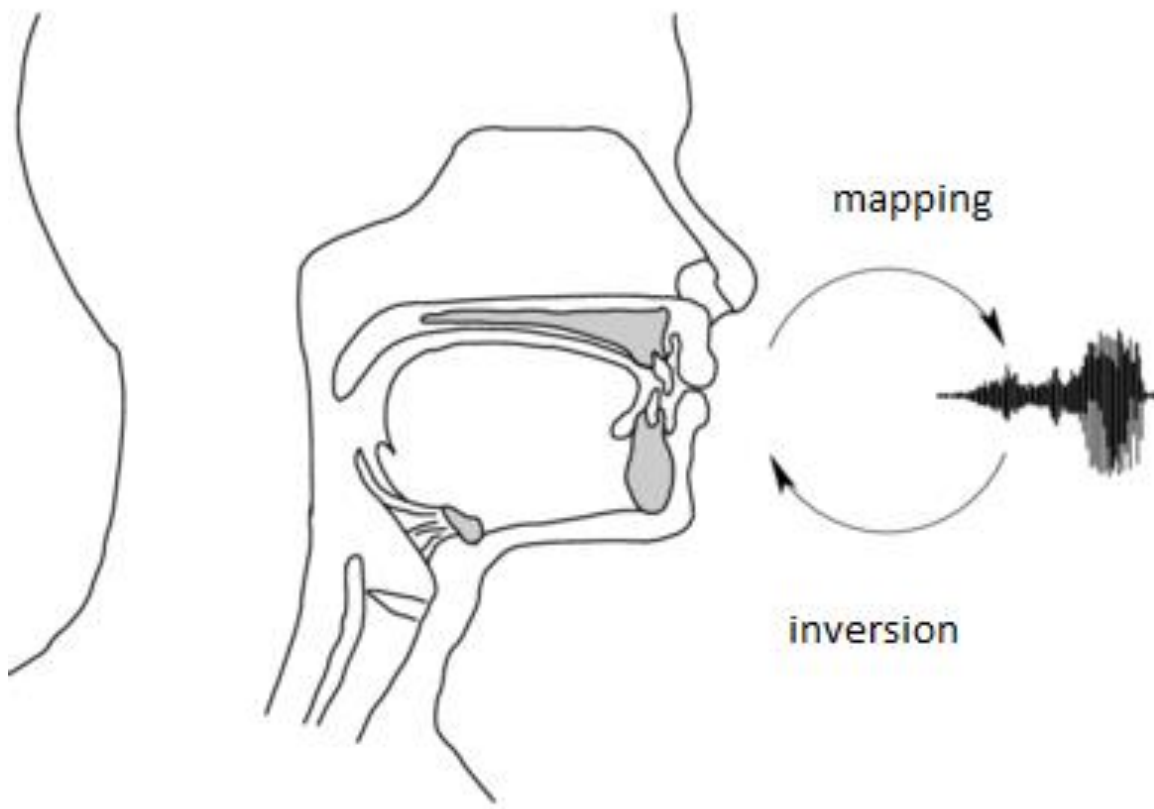
Figure 1: Pipeline of our system. Conv3D: 3-D convolution. BN: Batch normalisation. GRU: Gated recurrent unit. ConvT2D: 2-D transposed convolution. VUV: Voiced-unvoiced. CTC: Connectionist temporal classification.

HyperOpt esettanulmány: artikuláció -> beszéd szintézis

Artikuláció -> beszéd szintézis

- cél (Silent Speech Interface = SSI)
 - az artikulációs szervek mozgását felvesszük
 - a gépi rendszer ebből beszédet szintetizál
 - artikuláció-akusztikum becslés
- eszközök az artikuláció vizsgálatára
 - Ultrasound Tongue Imaging (UTI) / nyelvultrahang
 - Electromagnetic Articulography (EMA)
 - Permanent Magnetic Articulography (PMA)
 - surface Electromyography (sEMG)
 - Non-Audible Murmur (NAM)
 - lip video, multimodal, ...

Artikuláció -> beszéd szintézis



Forrás: <http://www.cstr.ed.ac.uk/research/projects/inversion/>

Speech Synthesis from Silent Videos

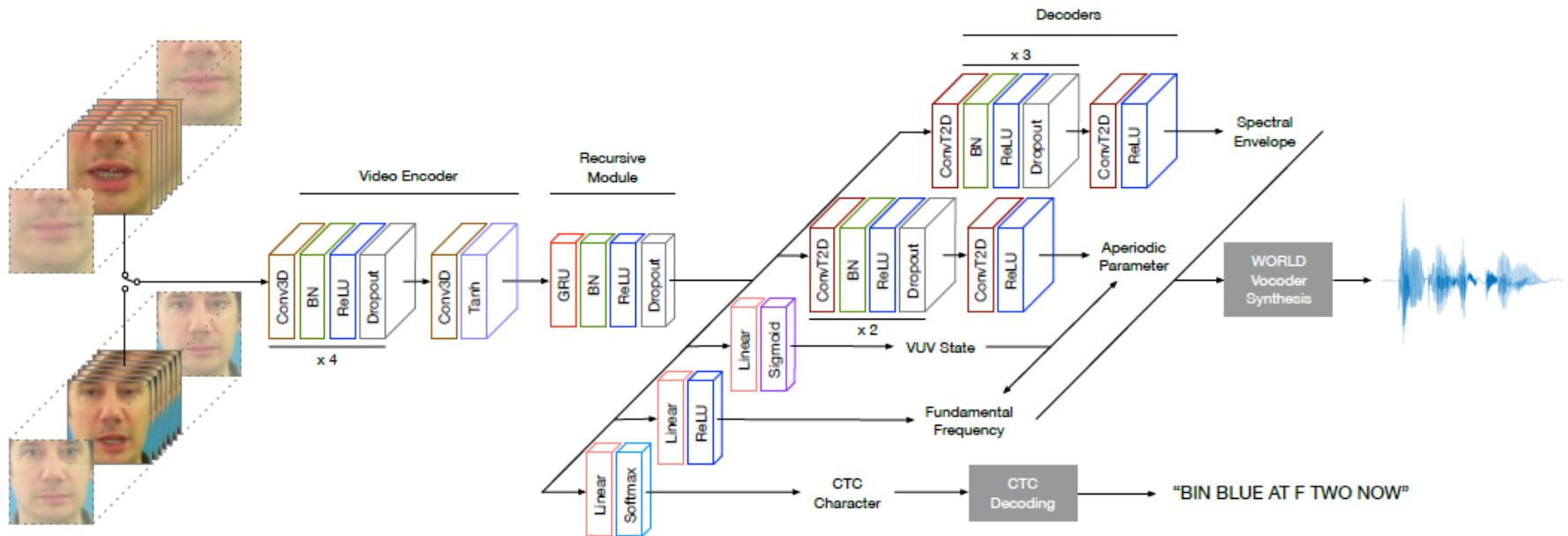
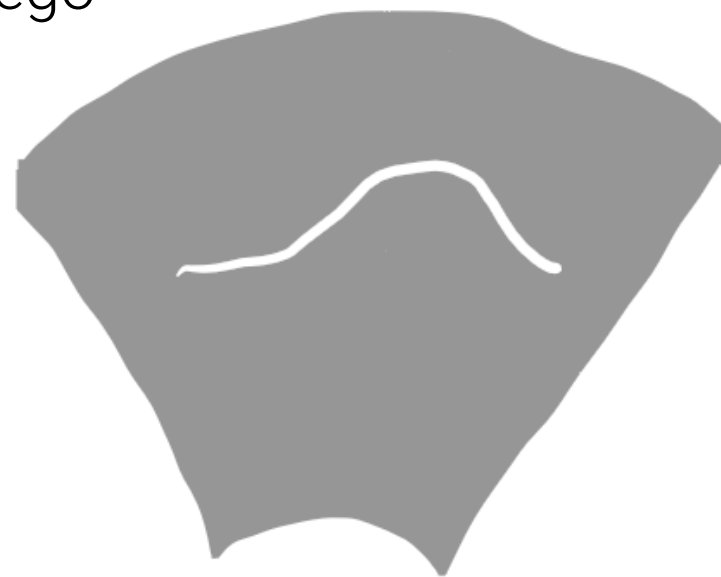


Figure 1: Pipeline of our system. Conv3D: 3-D convolution. BN: Batch normalisation. GRU: Gated recurrent unit. ConvT2D: 2-D transposed convolution. VUV: Voiced-unvoiced. CTC: Connectionist temporal classification.

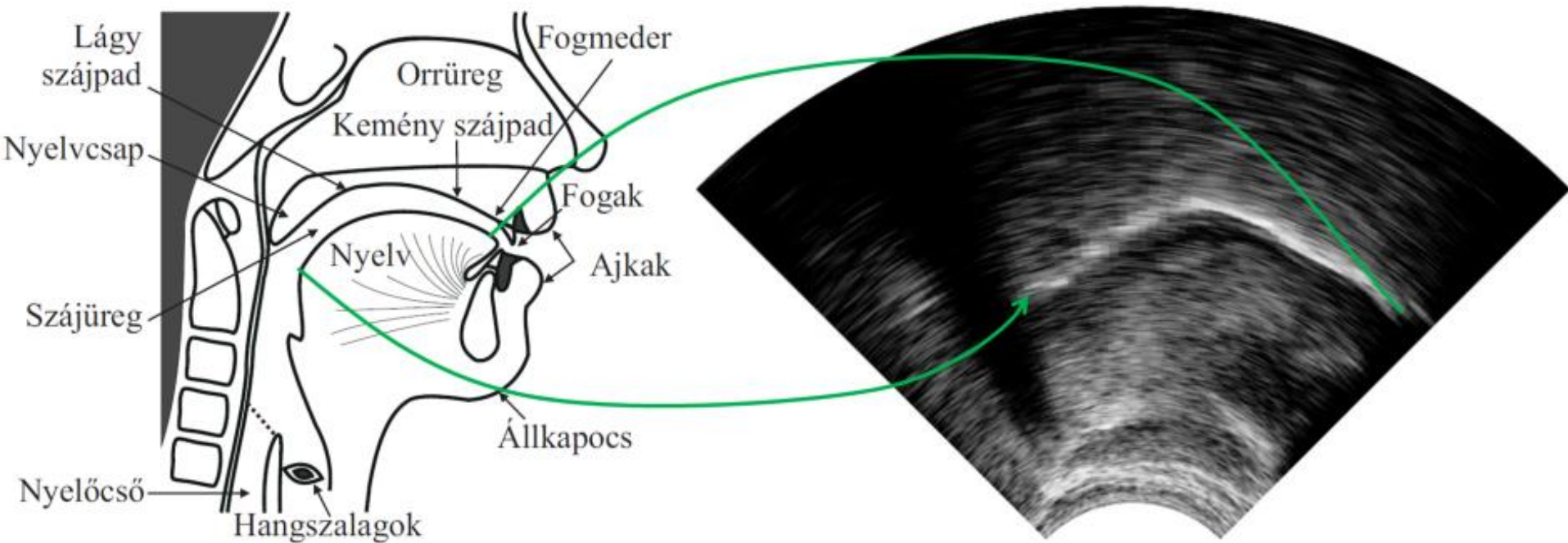
Forrás:Michelsanti, D., Slizovskaia, O., Haro, G., Gómez, E., Tan, Z., Jensen, J. (2020) Vocoder-Based Speech Synthesis from Silent Videos. Proc. Interspeech 2020, 3530-3534, DOI: 10.21437/Interspeech.2020-1026.

Artikuláció vizsgálata

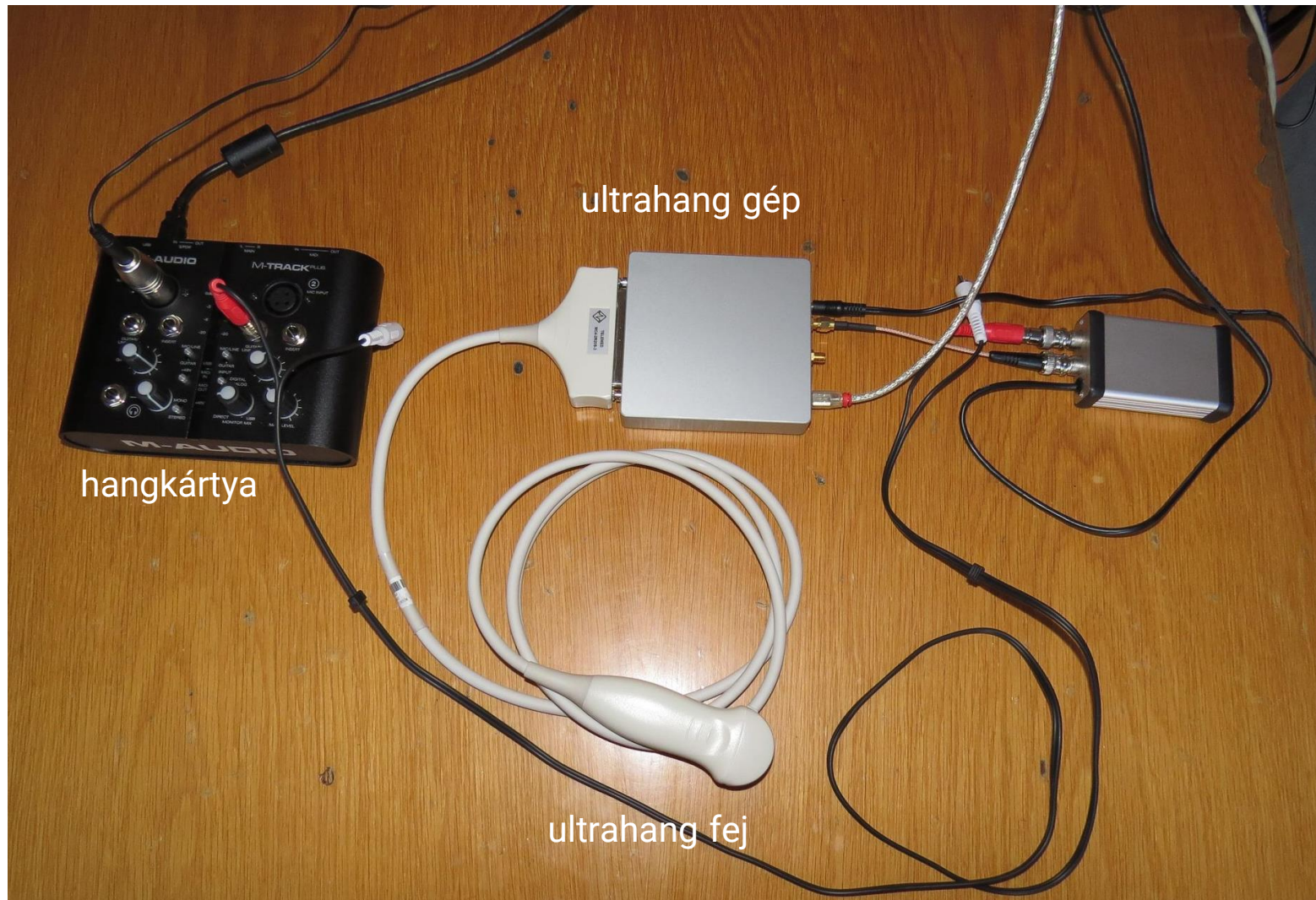
- nyelvultrahang
 - beszéd kutatásban korai '80-as évek óta
 - UH fej az állkapocs alatt beszéd közben
 - nyelv mozgása videón látszik (akár 100 képkocka/másodperc)
 - nyelv felső része világosabb, mint a környező szövetek és a levegő
 - viszonylag jó időbeli és térbeli felbontás



- Az ultrahang kép iránya



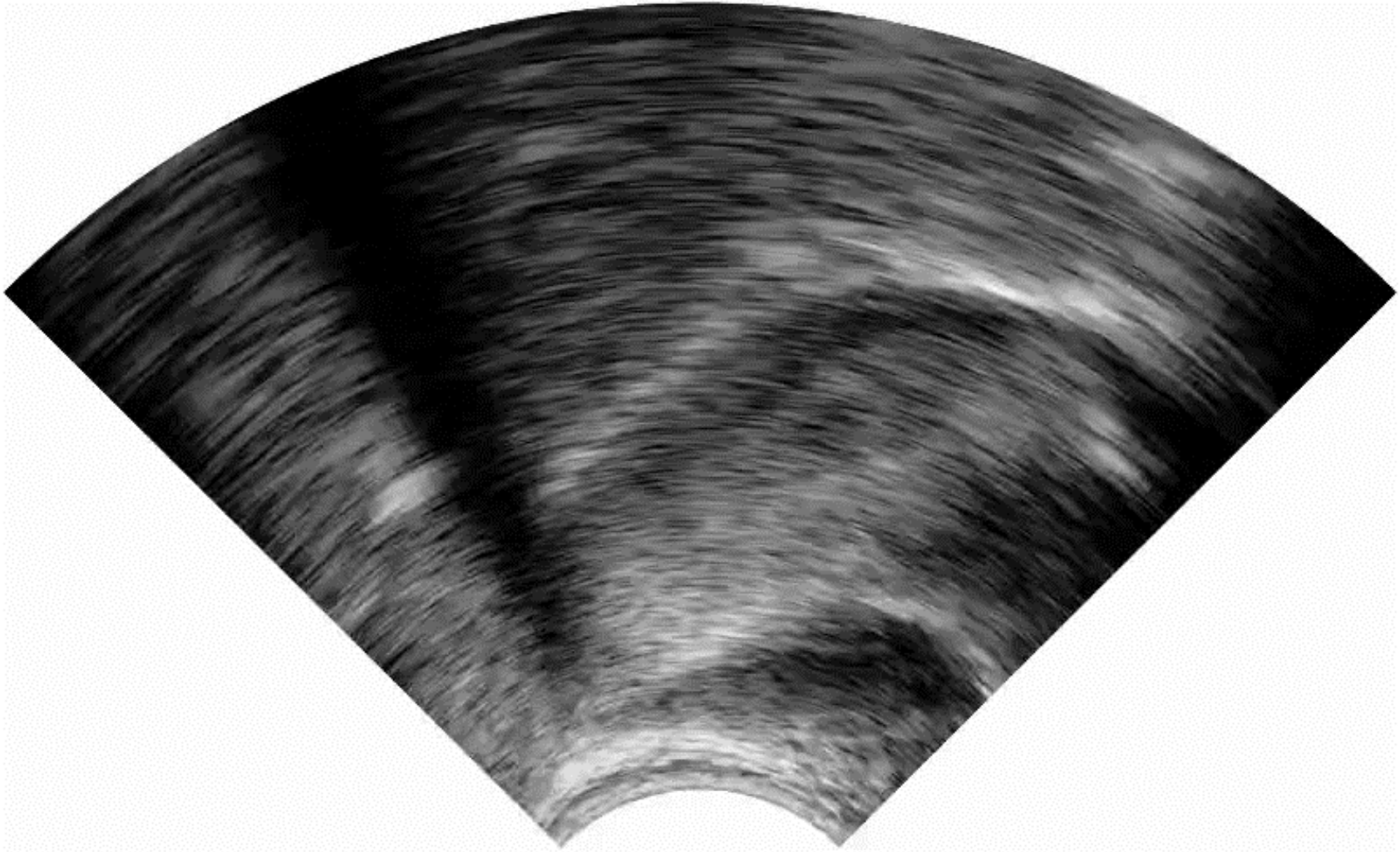
A felvételekhez használt eszközök



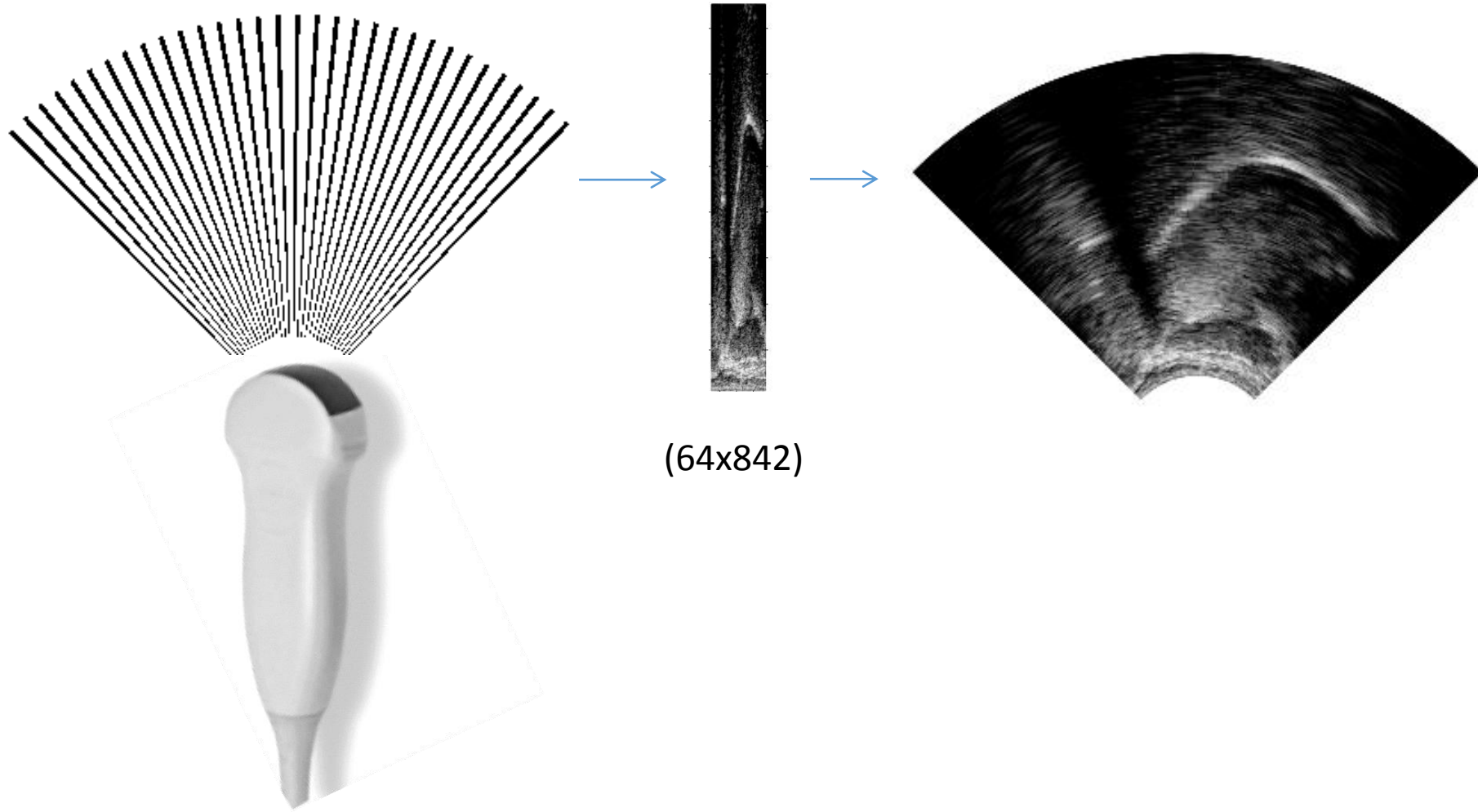
Sisak



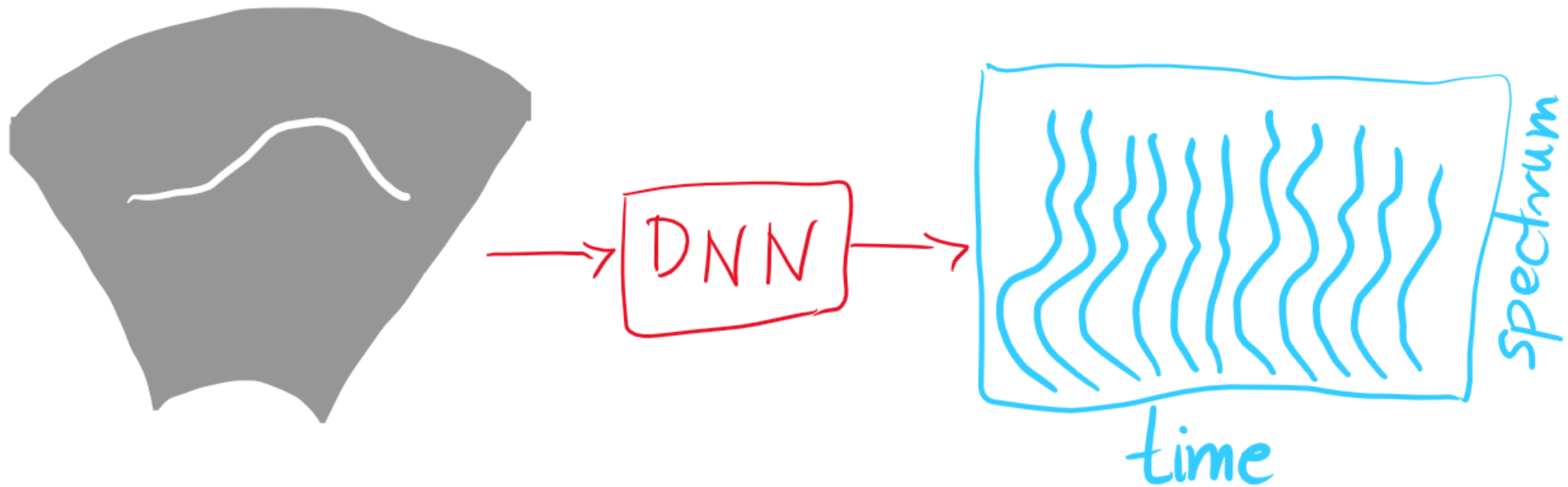
Példa: beszéd és ultrahang



„Nyers” ultrahang kép



Nyelvultrahang -> beszéd szintézis



Felmerülő kérdések

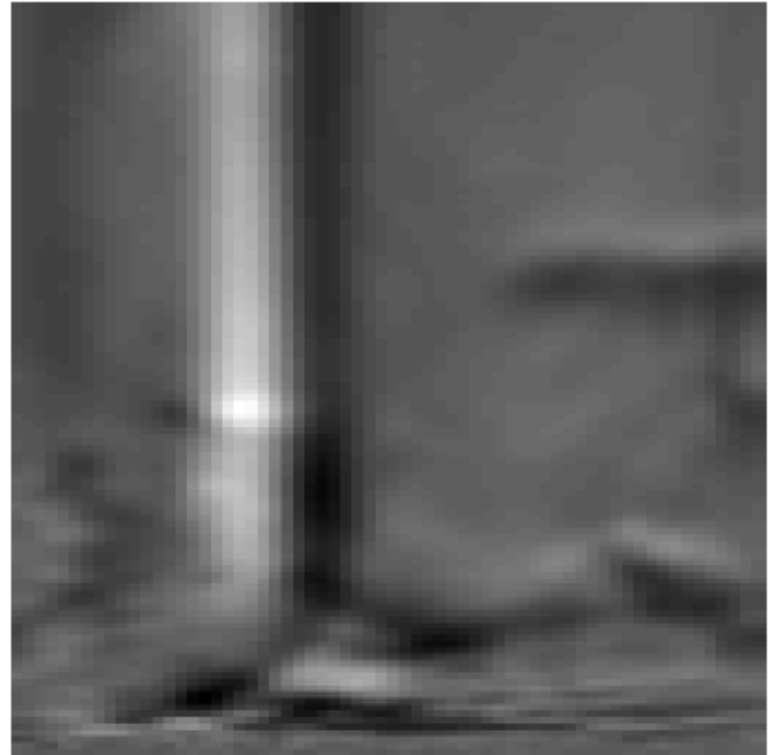
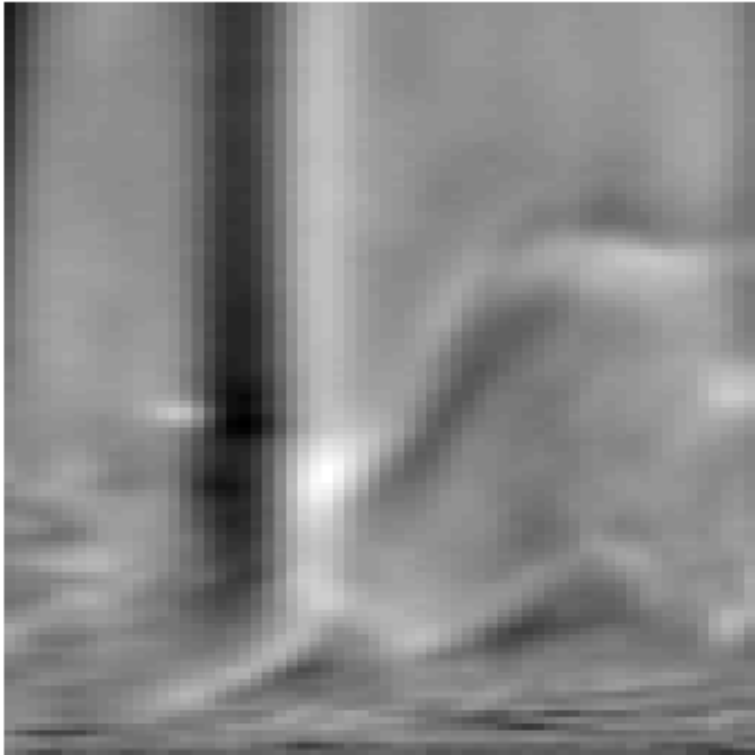
- Ultrahang bemenet milyen reprezentációban legyen?
- Beszéd / spektrális kimenet milyen reprezentációban legyen?
- Milyen típusú hálót tanítsunk?
- Hogyan végezzük a hálózat kiértékelését?

Bemenet: ultrahang

- 1) 64x842 pixeles kép -> 53888 méretű bemenet
 - Túl nagy, sok irreleváns információ
- 2) 64x96-ra átméretezés -> 6144 méretű bemenet
 - Képfeldolgozás, kettős köbös interpoláció
- 3) PCA, „EigenTongues”
 - Jó tömörítés, 6144 -> 20% -> kb. 1200 bemenet
- 4) Korreláció alapú
 - minden pixelre kiszámítjuk annak korrelációját a kimenő paraméterekkel
 - vesszük ezen korrelációk abszolút értékének maximumát, és küszöbölünk (korreláció egy küszöbérték fölött)
- 5) Szétterített ultrahang kép (interpolált)

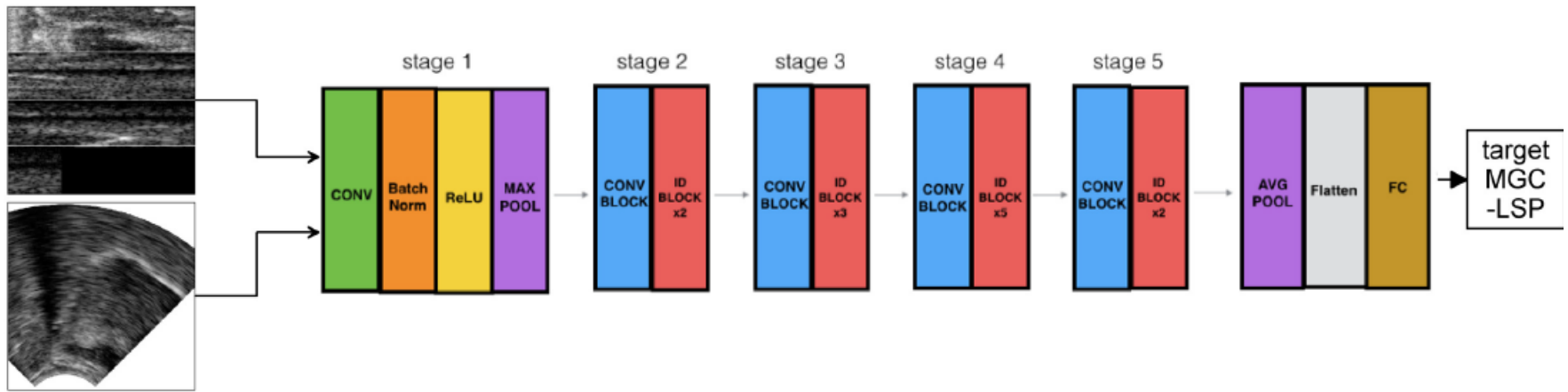
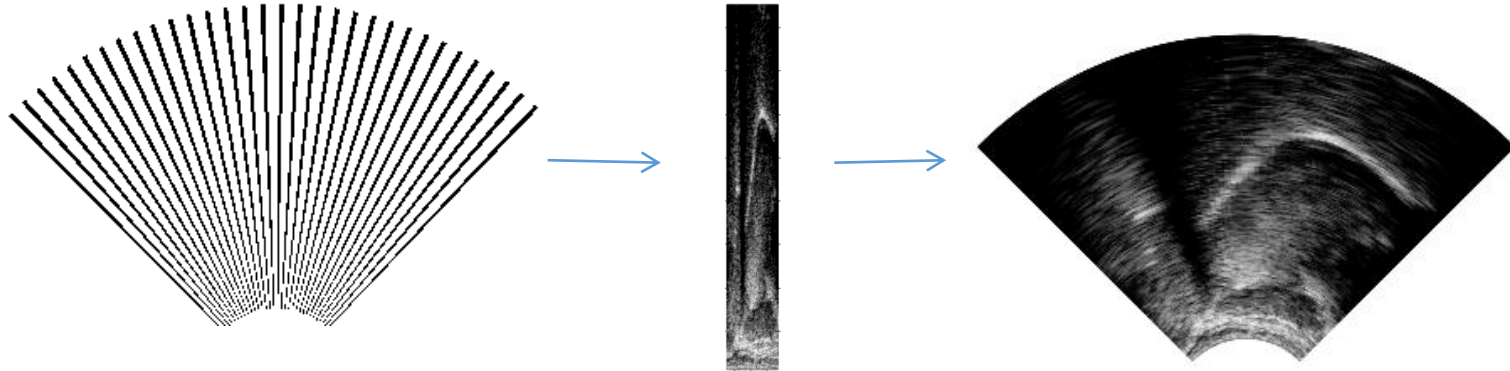
Bemenet: ultrahang

- 3) PCA, „EigenTongues”



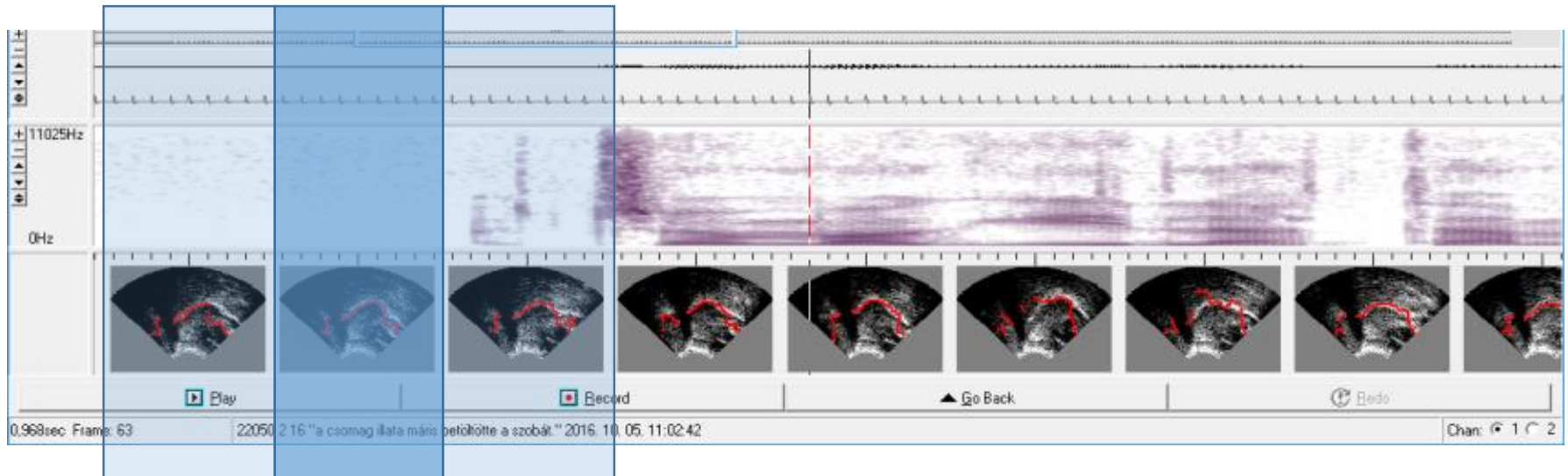
Bemenet: ultrahang

- 5) Szétterített ultrahang kép (interpolált)

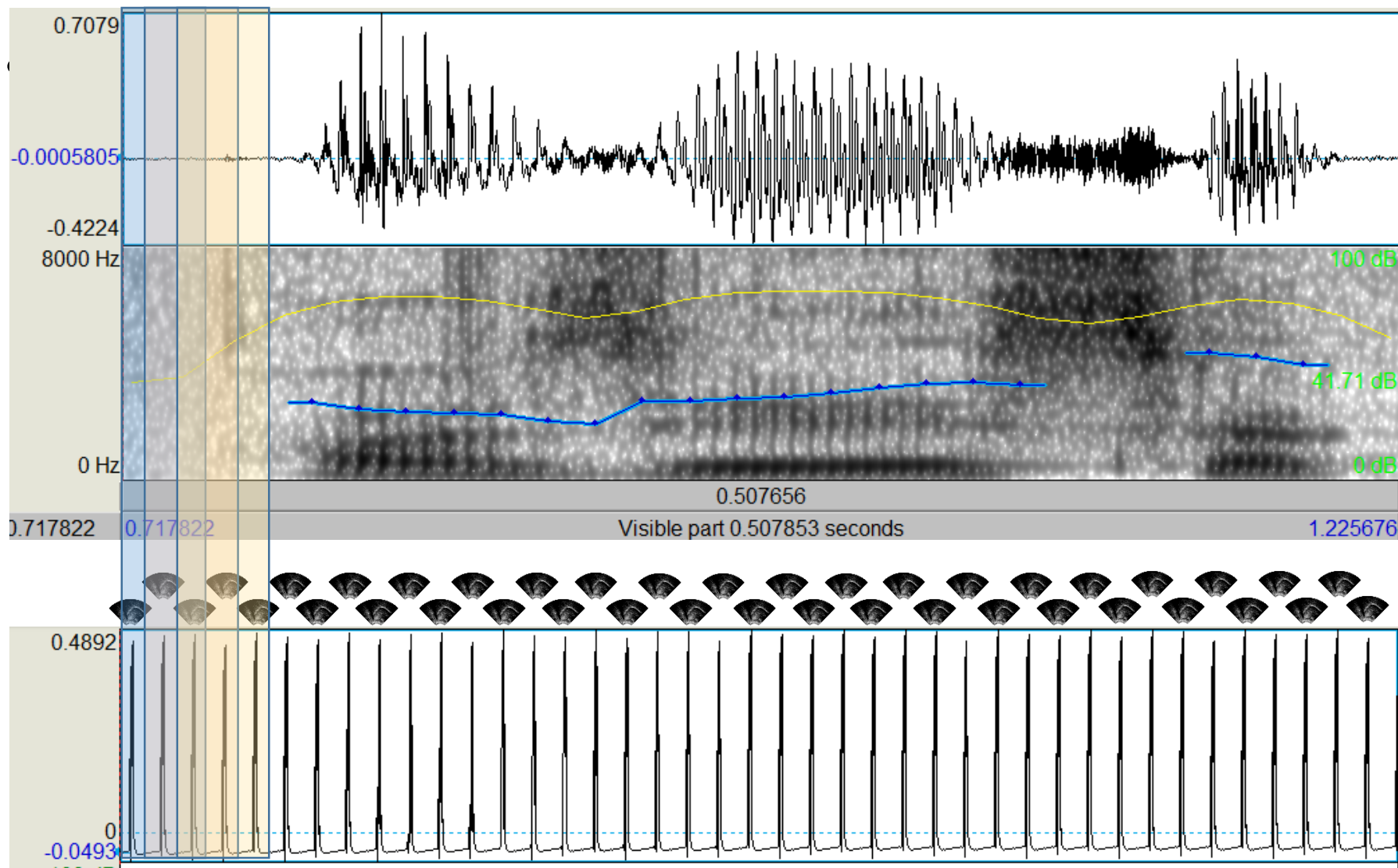


Bemenet: ultrahang

- 6) Több egymás utáni kép felhasználása
 - 1-1 / 5-5 / 8-8

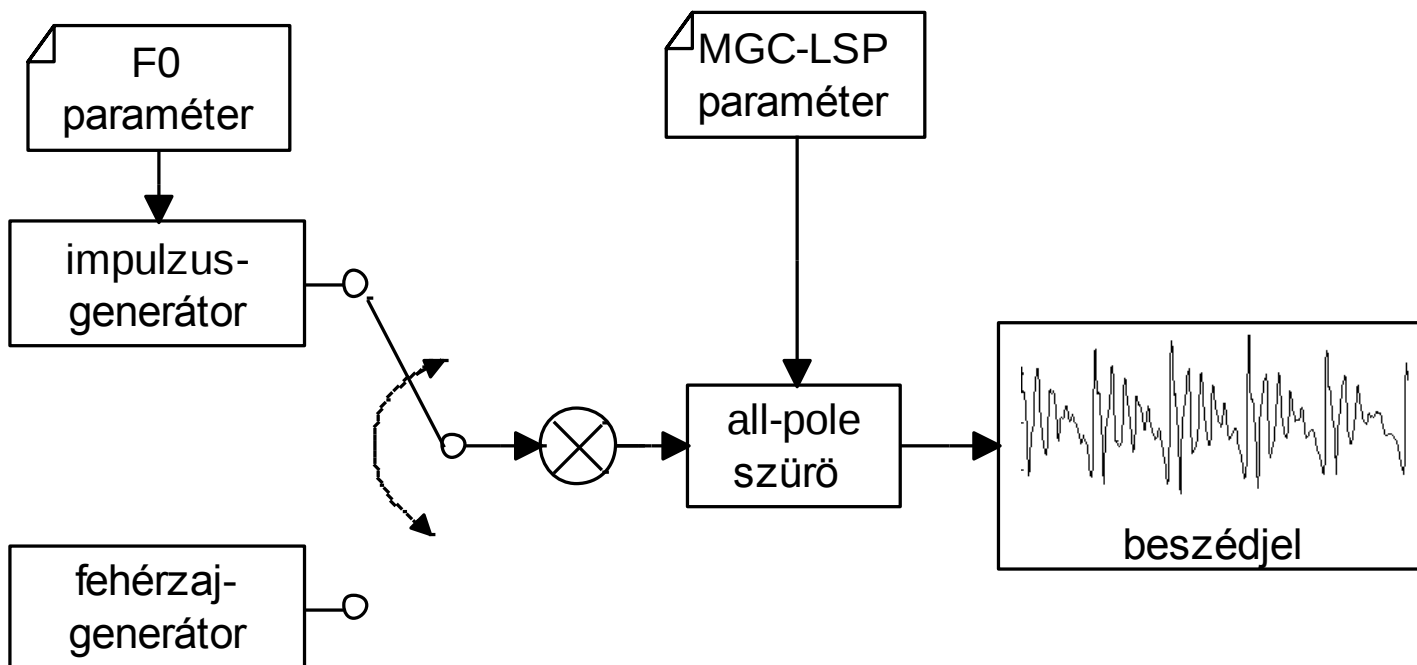


Kimenet: spektrális jellemzők



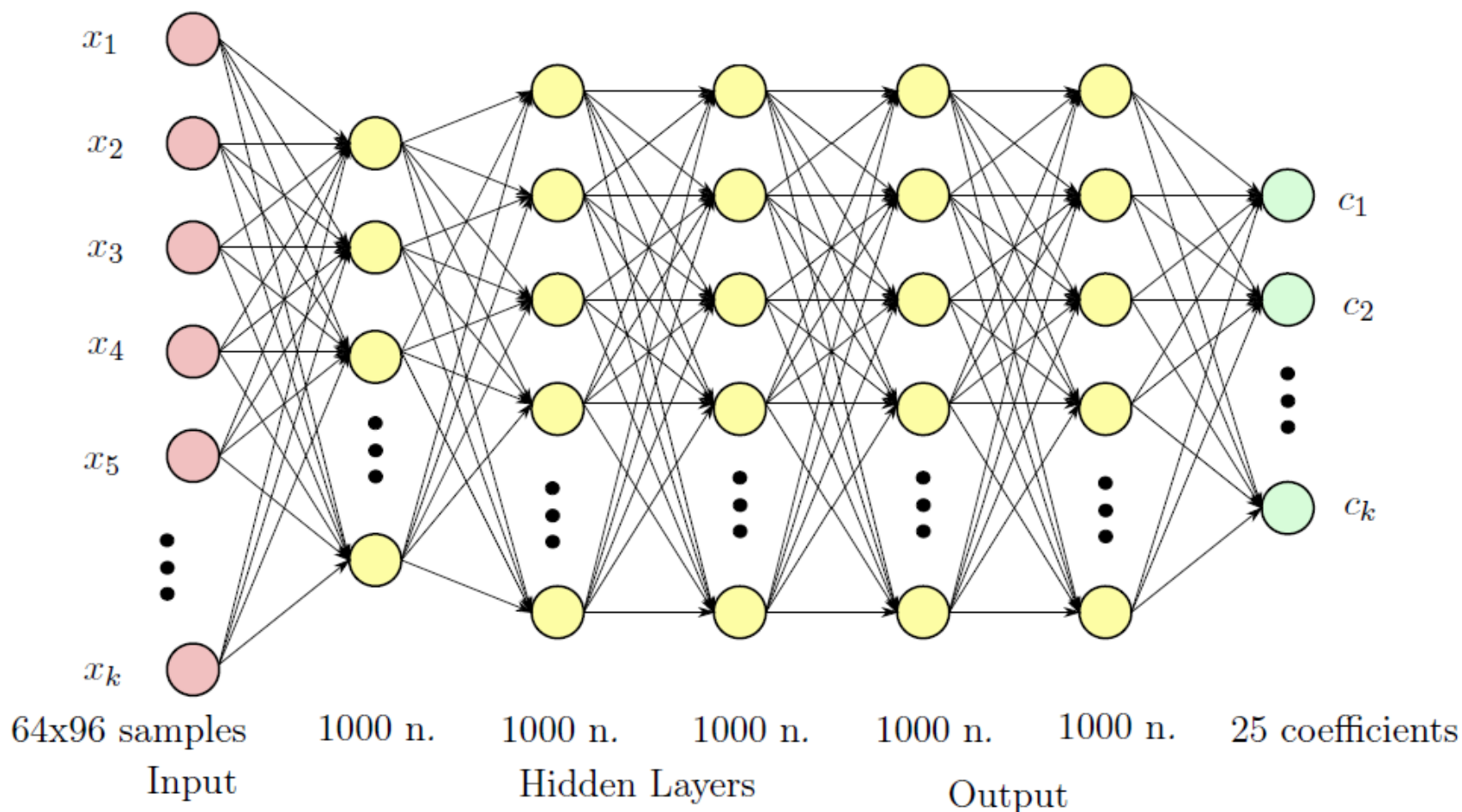
Kimenet: spektrális jellemzők

- MGC-LSP reprezentáció, 25 dimenzió
- Vokóder: MGC-LSP (és F0) -> beszéd



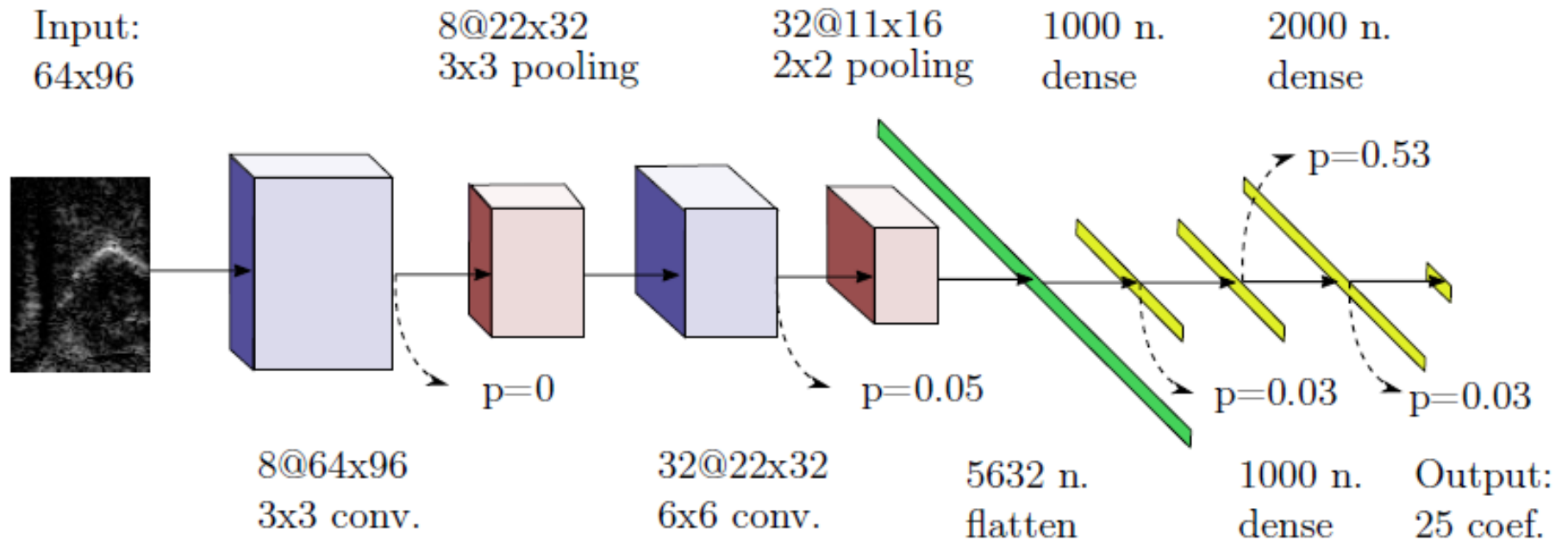
Hálózat/1: FC

- Első tipp: FC, 5 rejtett réteg, 1000 neuron, ReLU



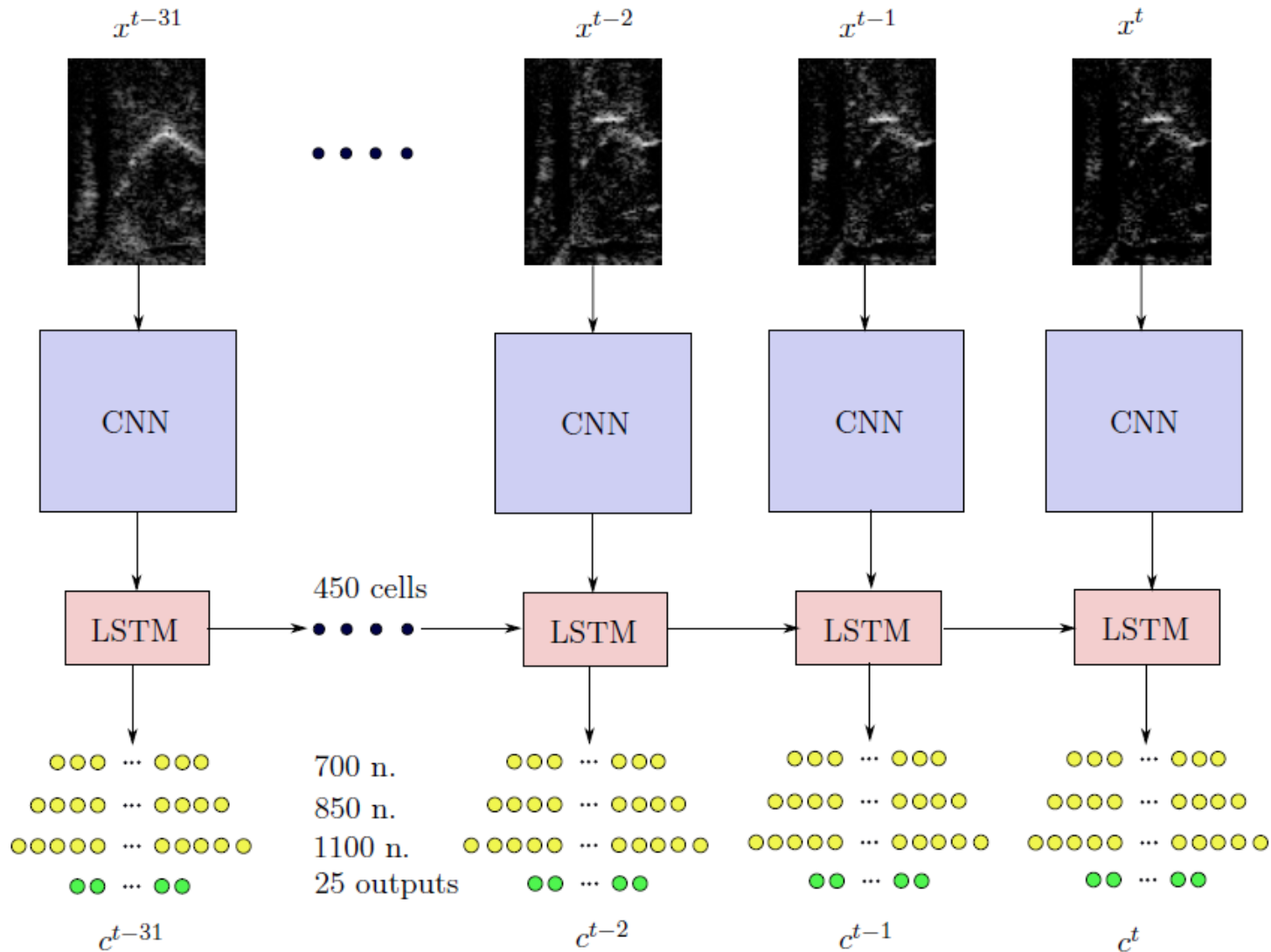
Forrás: Eloi Moliner Juanpere, „Ultrasound-based Silent Speech Interface using Deep Learning”, BSc thesis, BME, 2018

Hálózat/2: CNN



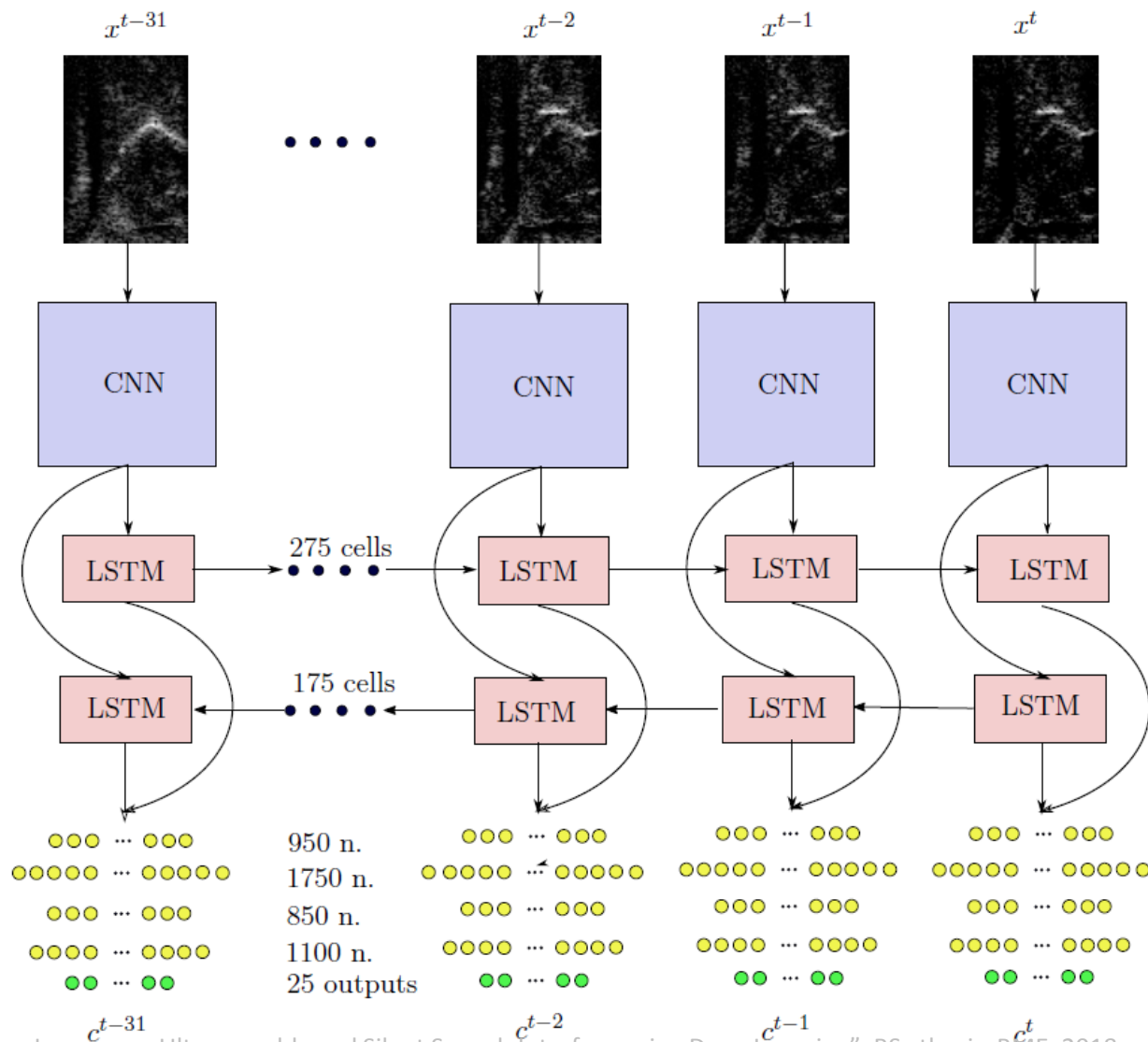
Forrás: Eloi Moliner Juanpere, „Ultrasound-based Silent Speech Interface using Deep Learning”, BSc thesis, BME, 2018

Hálózat/3: CNN-LSTM



Forrás: Eloi Moliner Juanpere, „Ultrasound-based Silent Speech Interface using Deep Learning”, BSc thesis, BME, 2018

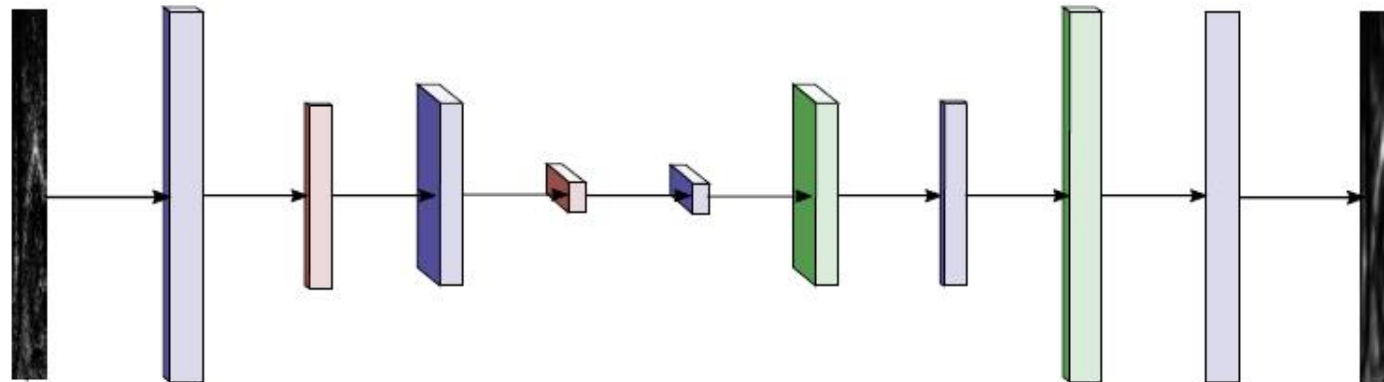
Hálózat/4: CNN-biLSTM



Forrás: Eloi Moliner Juanpere, „Ultrasound-based Silent Speech Interface using Deep Learning”, BSc thesis, BME, 2018

Hálózat/5: Conv AutoEncoder

Input: 64x768 4@32x194 32@16x32 32@32x192 32@64x768 Output: 64x768
2x4 pooling 2x6 pooling. 2x6 upsampling. 2x4 upsampling.



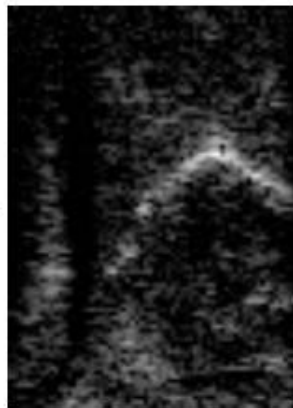
4@64x768
7x7 conv.

32@32x192
6x6 conv.

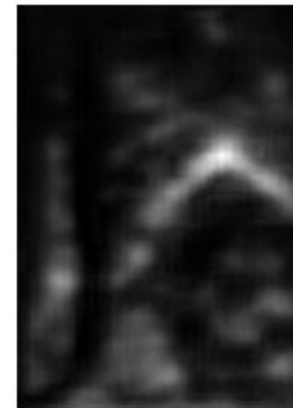
32@16x32
6x6 conv.

4@32x192
7x7 conv.

1@64x768
5x5 conv.



Input image decimated by 8

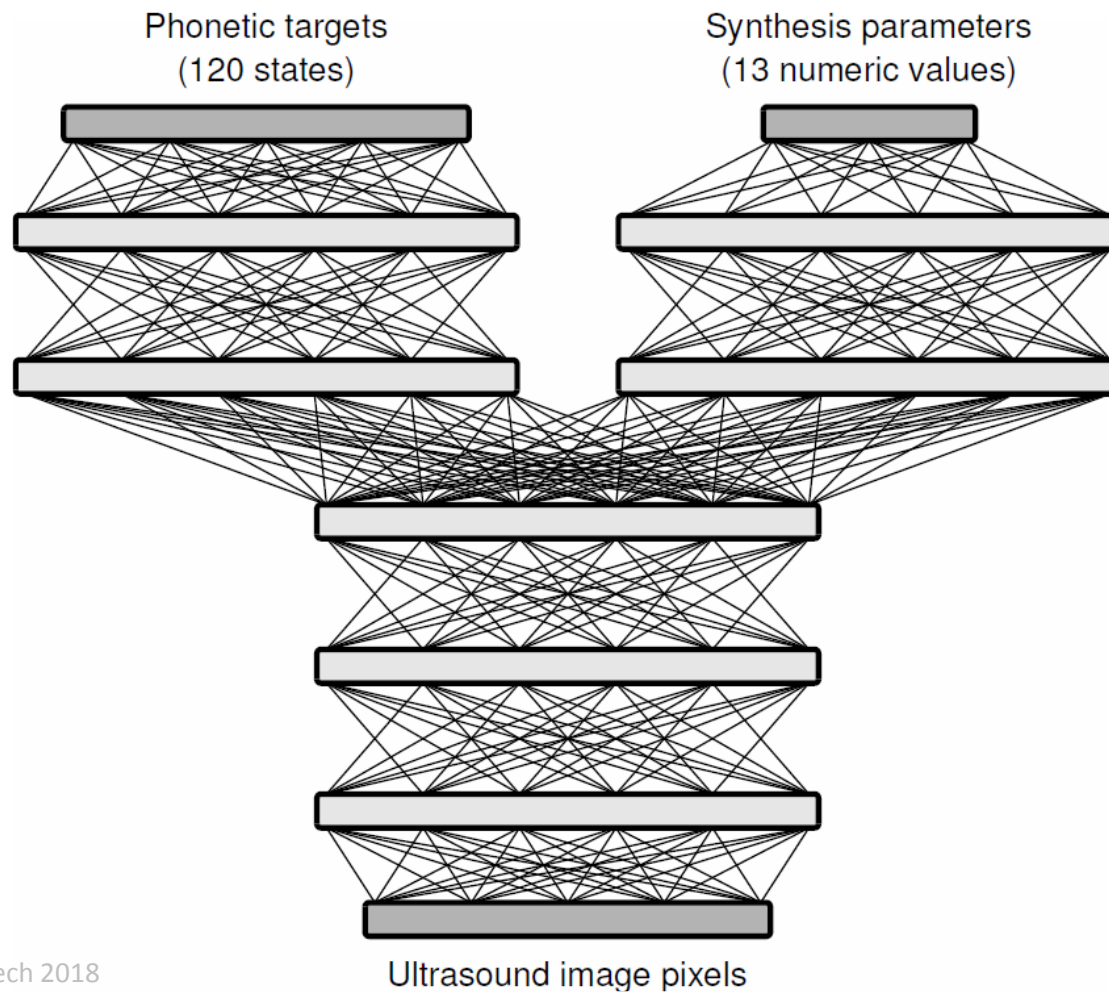


Denoised image decimated by 8

Forrás: Eloi Moliner Juanpere, „Ultrasound-based Silent Speech Interface using Deep Learning”, BSc thesis, BME, 2018

Hálózat/6: FC, multi-task

- Két feladat, megosztott rétegek



Forrás: Tóth et al, Interspeech 2018

Hálózat/7: Conditional GAN

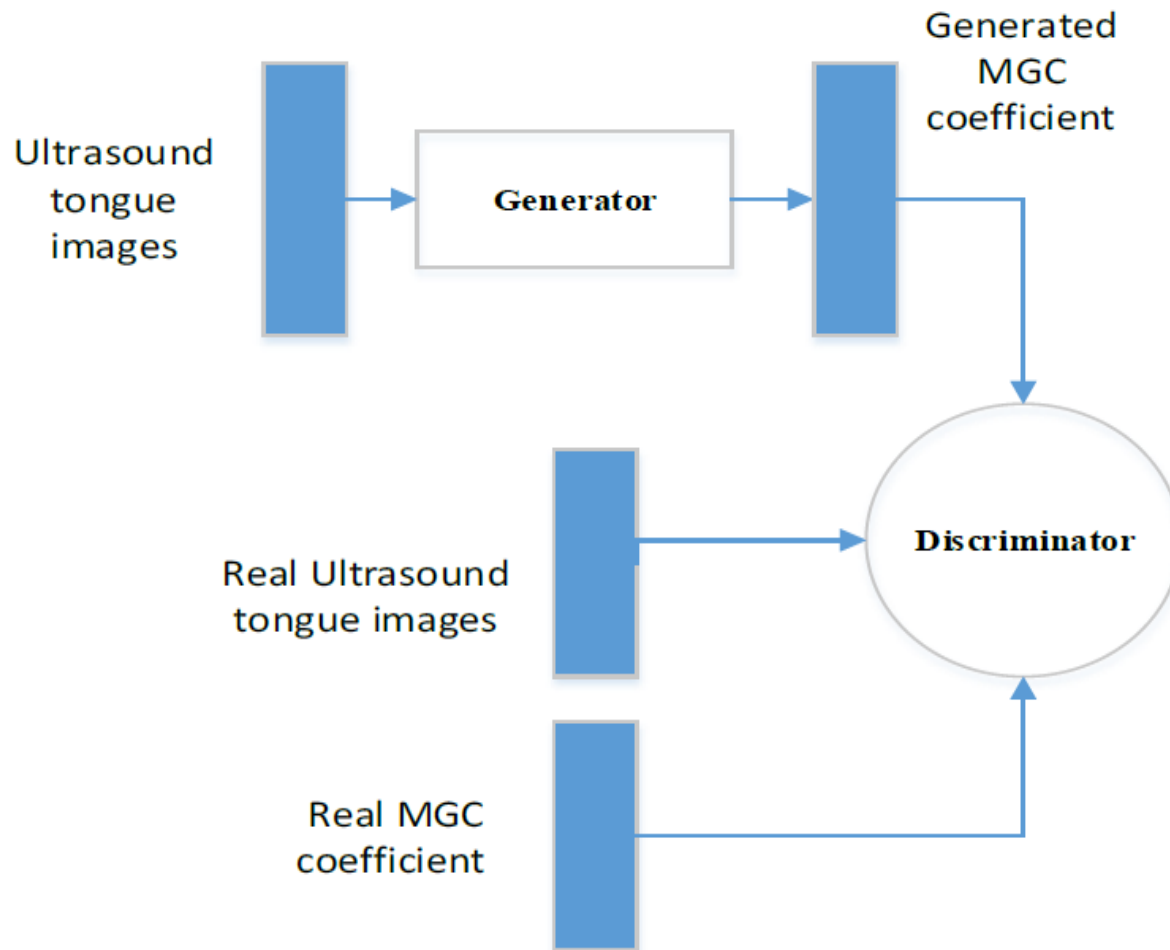


Figure 11 : GAN for UTI speech mapping

Melyik hálózat jobb?

- Bemenet: 5 féle reprezentáció
- Hálózat: 7 féle hálózat
- Adott hálón belül rengeteg paraméter...
 - Hiperparaméter optimalizálás!

DNN hyperopt

Hyperparameter	Range
Number of hidden layers	[1-4]
Number of Neurons in the first layer	{250, 500, 750, 1000}
Number of Neurons in the second layer	{250, 500, 750, 1000}
Number of Neurons in the third layer	{250, 500, 750, 1000}
Number of Neurons in the fourth layer	{250, 500, 750, 1000}
Dropout probability after the first hidden layer	[0-0.8]
Dropout probability after the second hidden layer	[0-0.8]
Dropout probability after the third hidden layer	[0-0.8]
Dropout probability after the fourth hidden layer	[0-0.8]
Optimizer	{adam, rmsprop, adamax}
Activation	{tanh, relu}
Batch size	{16, 32, 64, 128, 256}

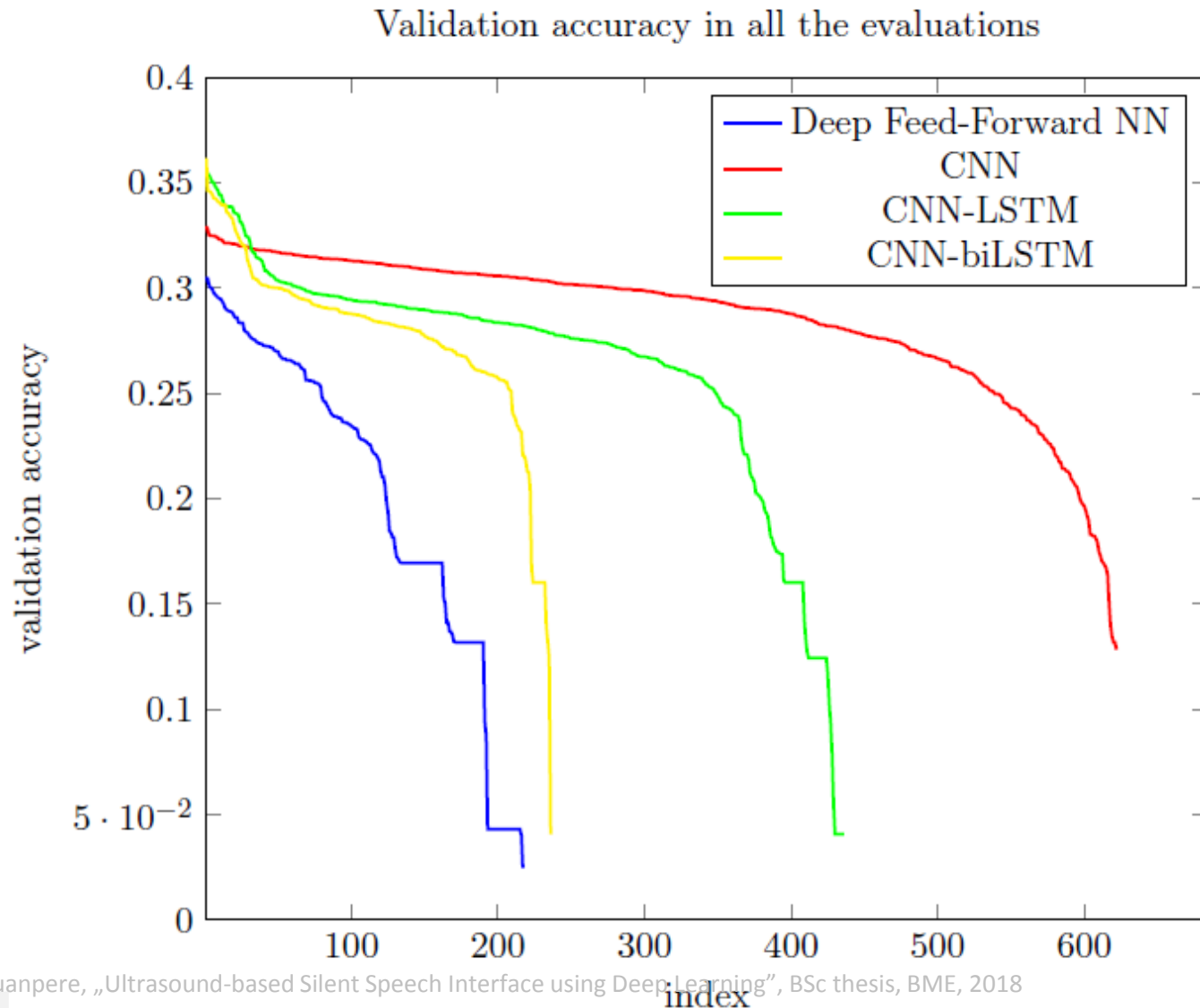
Forrás: Eloi Moliner Juanpere, „Ultrasound-based Silent Speech Interface using Deep Learning”, BSc thesis, BME, 2018

DNN hyperopt eredmény

Hyperparameter	Optimal
Number of hidden layers	4
Number of Neurons in the first layer	500
Number of Neurons in the second layer	1000
Number of Neurons in the third layer	1000
Number of Neurons in the fourth layer	1000
Dropout probability after the first hidden layer	0.09
Dropout probability after the second hidden layer	0.07
Dropout probability after the third hidden layer	0.15
Dropout probability after the fourth hidden layer	0.07
Optimizer	adamax
Activation	relu
Batch size	64

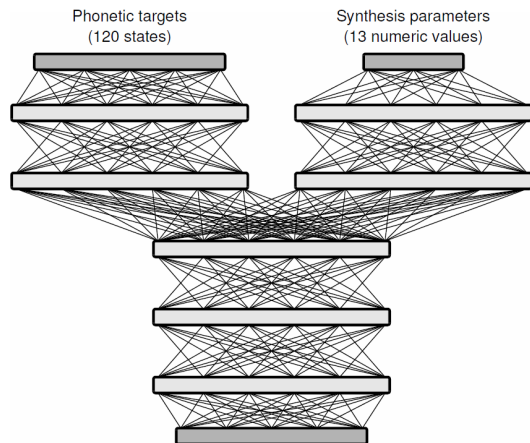
Forrás: Eloi Moliner Juanpere, „Ultrasound-based Silent Speech Interface using Deep Learning”, BSc thesis, BME, 2018

HyperOpt eredmény

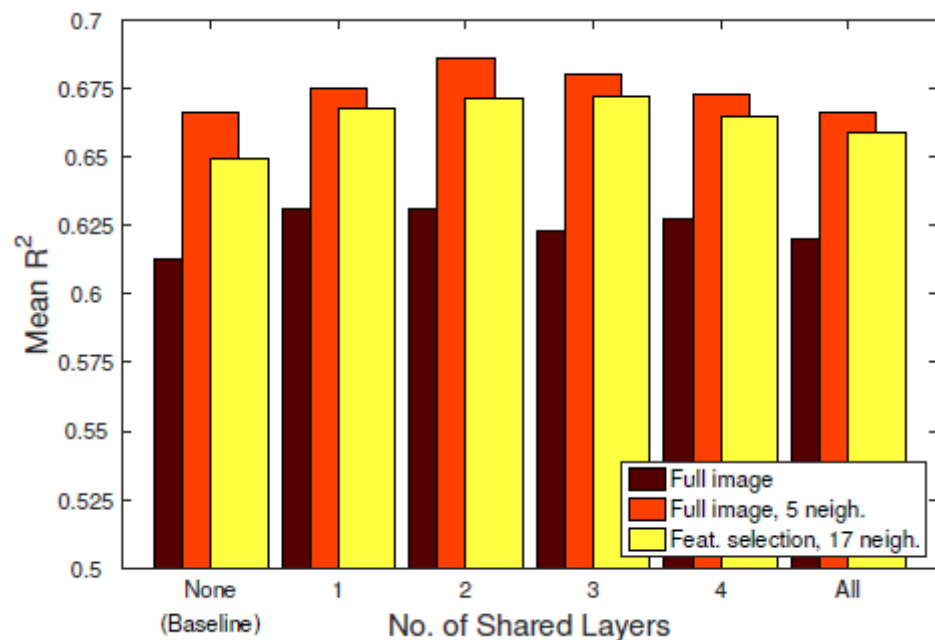
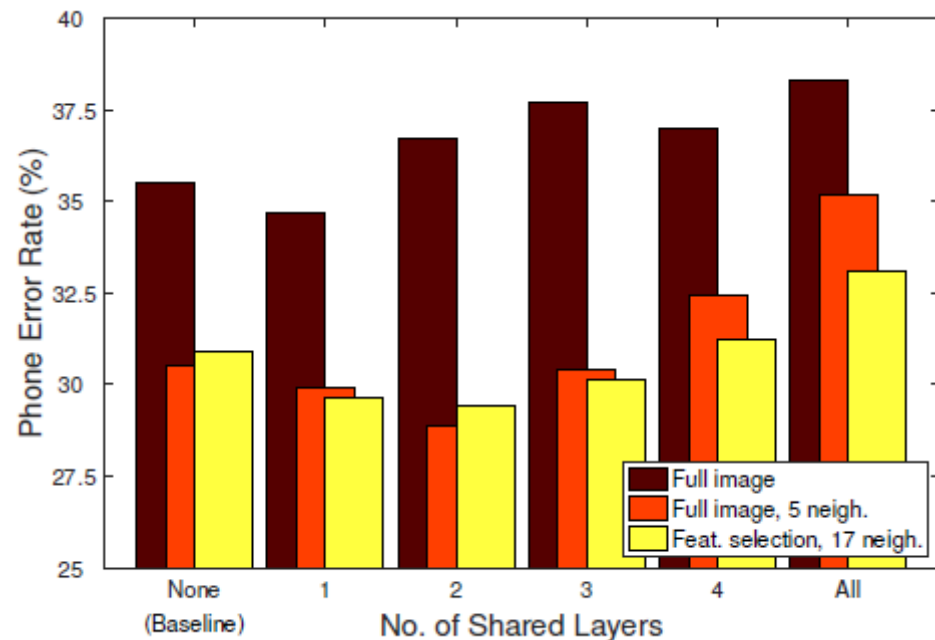


Forrás: Eloi Moliner Juanpere, „Ultrasound-based Silent Speech Interface using Deep Learning”, BSc thesis, BME, 2018

Legvégén: objektív kiértékelés



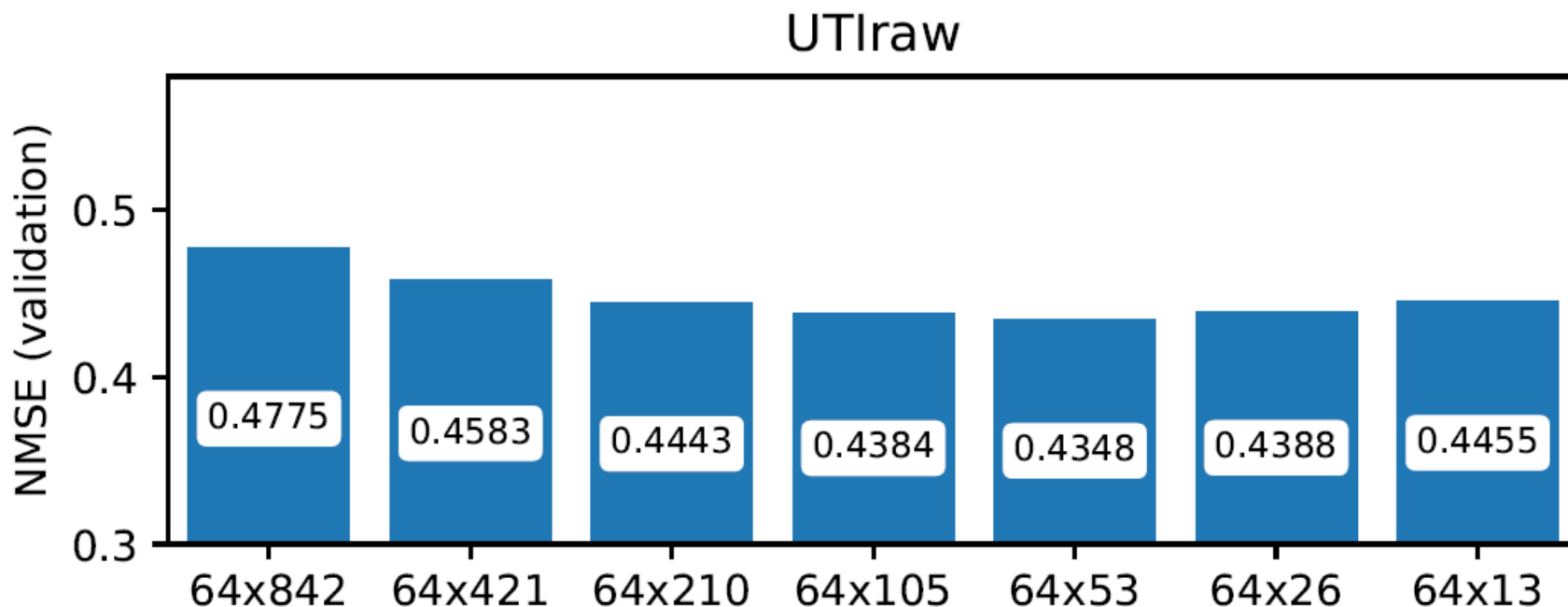
Hány megosztott réteg legyen?



Forrás: Tóth et al, Interspeech 2018

Legvégén: objektív kiértékelés

- Mi az optimális bemeneti reprezentáció?



Forrás: Csapó et al, Interspeech 2020

Legvégén: szubjektív kiértékelés

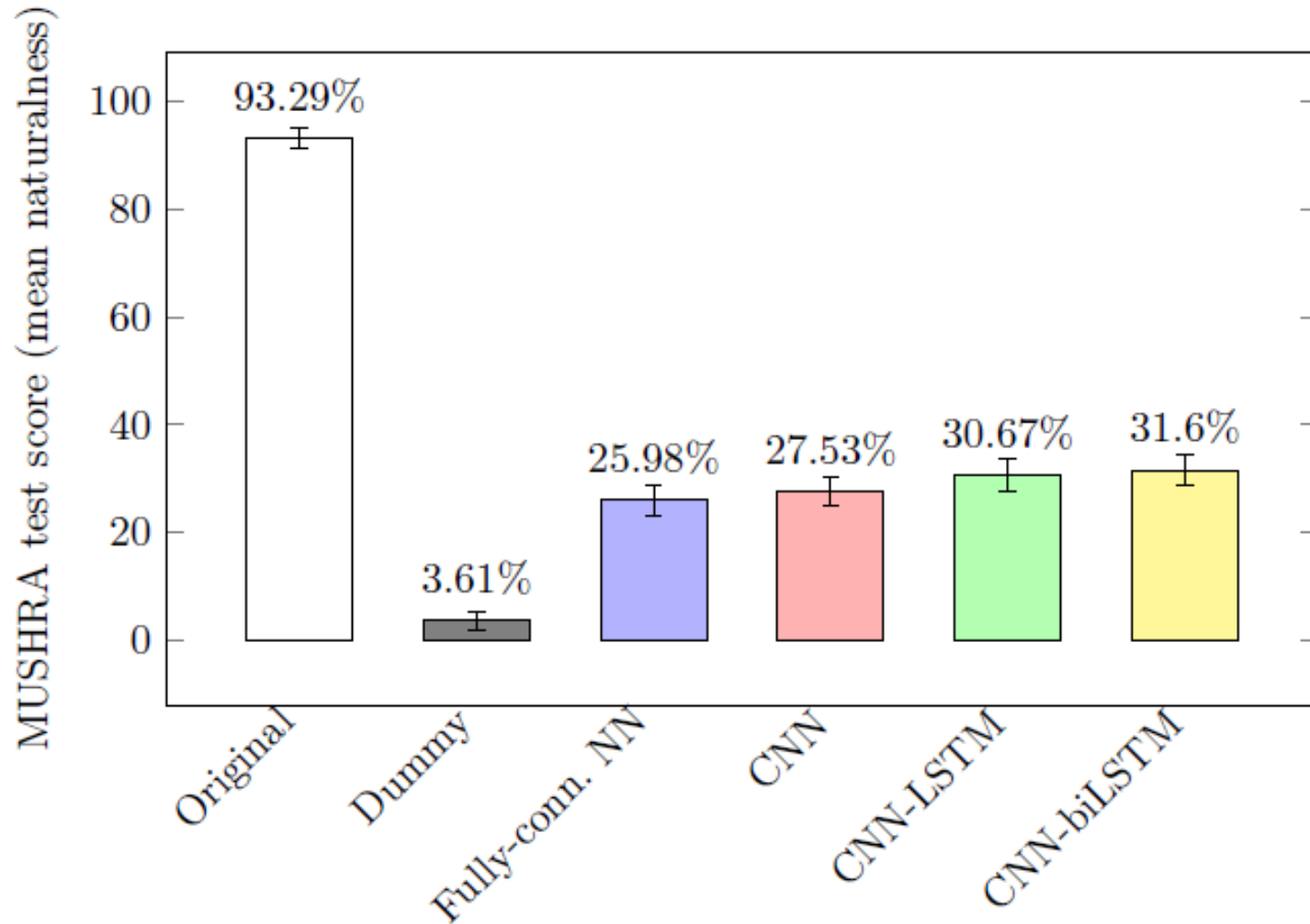
Sample group no. 1 out of 9.

**Please adjust the naturalness of each sample
compared to the reference
by moving the horizontal slider between 1-100!**

	Recording identifier & sample	Highly unnatural	Unnatural	Intermediate	Natural	Highly natural
reference						
a						
b						
c						
d						
e						
f						

Submit & next sample

Legvégén: szubjektív kiértékelés



Forrás: Eloi Moliner Juanpere, „Ultrasound-based Silent Speech Interface using Deep Learning”, BSc thesis, BME, 2018

Idő

- 1 tanítás kb. 15 perc
 - DNN: 1000 tanítás
 - CNN: 600 tanítás
 - CNN-LSTM: 450 tanítás
 - CNN-biLSTM: 250 tanítás
 - x 5-féle bemenet reprezentáció
-
- $15 \times (1000+600+450+250) \times 5 =$
 $= \text{kb. 2 hét, 5 gépen}$

Interspeech 2020

- Tamás Gábor Csapó, Csaba Zainkó, László Tóth, Gábor Gosztolya, and Alexandra Markó, „Ultrasound-based Articulatory-to-Acoustic Mapping with WaveGlow Speech Synthesis”, *Interspeech 2020*.
[arXiv:2008.03152](https://arxiv.org/abs/2008.03152)
- Tamás Gábor Csapó, „Speaker dependent articulatory-to-acoustic mapping using real-time MRI of the vocal tract”, *Interspeech 2020*.
[arXiv:2008.00889](https://arxiv.org/abs/2008.00889)
- Tamás Gábor Csapó, „Speaker dependent acoustic-to-articulatory inversion using real-time MRI of the vocal tract”, *Interspeech 2020*.
[arXiv:2008.02098](https://arxiv.org/abs/2008.02098)

HyperOpt összefoglalás

- Fontos a kezdeti intuíció
- Logfájlok
- Random search
- Bayesian / TPE hyperopt
- Neurális háló alapú (NAS)

HyperOpt keretrendszer

- <https://github.com/maxpumperla/hyperas>
- <https://github.com/keras-team/keras-tuner>
- <https://github.com/autonomio/talos>
- <https://github.com/sherpa-ai/sherpa>

- <https://github.com/hyperopt/hyperopt>
- <https://github.com/Avsecz/kopt>
- <https://github.com/tobegit3hub/advisor>
- <https://github.com/Alworx-Labs/chocolate>
- <https://machinelearningmastery.com/grid-search-hyperparameters-deep-learning-models-python-keras/>

5. kis házi feladat:

Hiperparaméter optimalizálás

Készíts kísérletsorozatot, melyben automatikus hiperparaméter optimalizálást végzel a CIFAR-10 adatbázison képosztályozási feladatra. Legyen legalább 100 tanítás, melynek eredményeit elemzed is (pl. hálózat mérete és az osztályozás pontossága hogyan függ össze?).

Tetszőleges deep learning keretrendszert és github forráskódokat (megfelelően hivatkozva) használhatsz. A megoldást részletesen kommentezd és a tanítás kimenetét (log fájlját) is mellékelj. A leírás és kommentek angolul legyenek!

<https://github.com/BME-SmartLab-Education/vitmav45/blob/master/kishazi/5-kis-hazi-feladat.md>

Beadási határidő: 2020. december 8. 23:59

Köszönöm a figyelmet!

`csapot@tmit.bme.hu`



Kutatási pályázatok (NKFIH) :

- *Artikulációs mozgás alapú beszédgenerálás*
- *Némabeszéd-interfészek kidolgozása és a beszélőfüggőség vizsgálata nyelvultrahang és elektromágneses artikulográf eszközökkel*