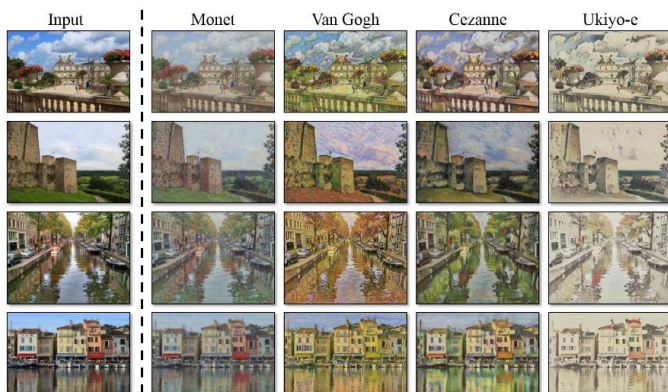


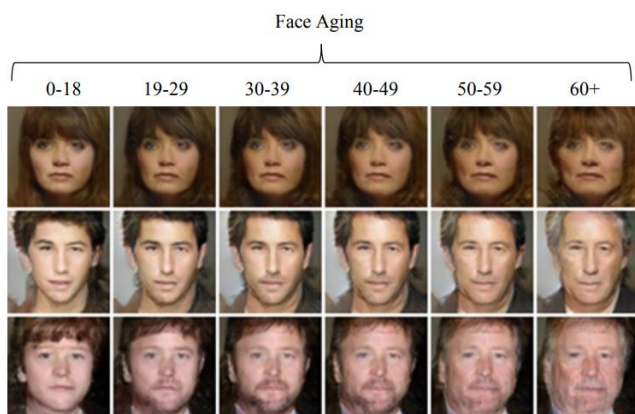
Na začiatku sa rekonštruuje pôvodný obrázok pomocou G na obrázok nejakého štýlu napríklad Van Gogh-a. Následne sa vygenerovaný obrázok porovna v D s reálnymi obrázkami Van Gogh-a. Keďže chceme použiť rekonštruovaný Van Gogh-ov

štýl na náš pôvodný obrázok tak sa pridáva ešte jeden generator (F), kde na jeho výstupe je pôvodný obrázok už so štýlom Van Gogh-a. V druhom systéme sú navzájom vymenené generátory.



Obrázok 2 Výstupy CycleGAN s použitím rôznych vstupov a štýlov [4].

Ďalšou zaujímavou aplikáciou je Face Aging [5]. Kde GAN vytvára sekvenciu obrázkov ľudských tvári a predpovedá ako budú vyzerat' v ďalších rokoch života.



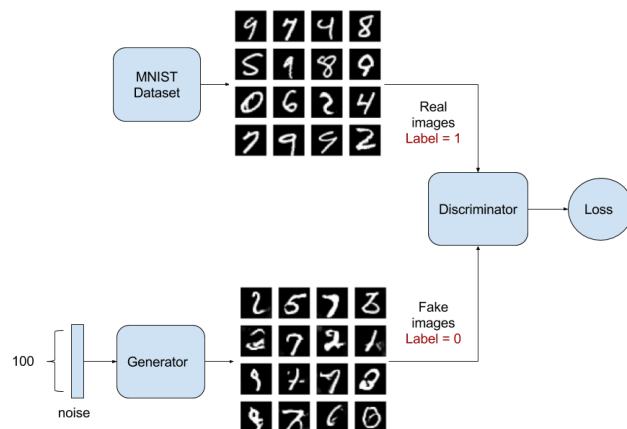
Obrázok 3 Výstup Face Aging-u [5].

Jednou z prvých aplikácií s použitím textu je syntetizácia obrázku z textu [6]. Vedcom sa podarilo vytvoriť pomocou GAN obrázok s rozlíšením 256x256 pixelov len z textového popisu. Pričom je to lepší výsledok ako state-of-the-art metódy, ktoré generujú obrázky s polovičným rozlíšením.

GAN sa môže aplikovať taktiež na rekonštrukciu tváre (TP GAN) z rôznych uhlov pričom neurónová sieť pozná len jednu fotografiu tváre [7].

### III. TOPOLOGIA A POPIS SIETE

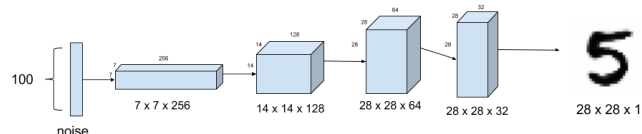
Tento systém sa skladá z dvoch hlavných podsystémov. A to z hlbokých sietí, ktoré sa nazývajú Generator a Discriminator.



Obrázok 4 Topológia GAN [8].

#### A. Generator

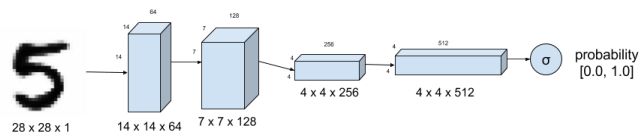
Model Generator je v princípe autoencoder, ktorý má na vstupe náhodné zašumené dáta vytvorené pomocou neakej distribúcie. Napríklad jednou z možností je pomocou normálneho rozdelenia vytvoriť zašumený vektor hodnôt, ktorá bude reprezentovať vstup. A na výstupe sa bude snažiť rekonštruovať digitálny obraz, pomocou upsampling-u. Tento model môže vyzerat' takto:



Obrázok 5 Model Generator-a [8].

#### B. Discriminator

Druhým podsystémom je Discriminator. Discriminator je klasická Konvolučná neurónová sieť, kde vstup reprezentuje digitálny obraz a výstup je apriórna pravdepodobnosť aký je obraz "real" resp. "fake". V poslednej vrstve tak figuruje sigmoidálna aktivačná funkcia, ktorej výstup reprezentuje "real" obraz a to hodnotou 1, resp. "fake" = 0. Takže úlohou tejto siete je vedieť rozpoznať či vstup je reálny z datasetu alebo vytvorený pomocou modelu Generator.



Obrázok 6 Konvolučná neurónová sieť, ktorá reprezentuje model Discriminator-a [8].

Najzaujímavejšou časťou tohto systému je samotný spôsob trénovania.

Na začiatok začneme pomocou malej vzorky z trénovacích dát (mini-batch) trénovať Discriminator. Kde na vstup privedieme obrázky z datasetu a na výstupe očakávame hodnotu, ktorú nazveme *real loss*. Následne necháme vygenerovať vstup Generator-a a na výstupe očakávame obrázok. Tento obrázok nastavíme na vstup nášho Discriminator-a a očakávame pravdepodobnosť *fake loss*. V princípe pri jednej epoche sa Generator spustí raz ale Discriminator dva-krát

Loss funkcia Discriminator-a predstavuje vzťah:

$$\text{real loss} - \text{fake loss}$$

Čo predstavuje ako odlišné sú si vygenerované dáta voči reálnym z datasetu. Tieto hodnoty sú znovu vrátane do svojich pôvodných modelov, v ktorých sa následne upravujú parametre podľa nich.

Týmto spôsobom sa snaží Generator oklamať Discriminator, že aktuálny vstup sú realne obrázky z datasetu.

Ak sa parametre nastavujú správne tak hodnota *fake loss* má narastať a *real loss* zase klesať. Pokiaľ je *real loss* rovná 0.5. Čoho dôsledkom je to, že generátor je dokázal naučiť generovať podobné dáta ako na nachádzajú v datasete. Je nutné podotknúť, že každý model má vlastnú Loss funkciu.

#### IV. HYPERPARAMETRE A PARAMETRE

##### A. Hyperparameter GAN

Neurónová sieť GAN je veľmi náročná na stabilitu trénovania.

Parametre učenia jednotlivých systémov sú veľmi malé. Rádovo sú to hodnoty  $1 \times 10^{-6}$  a menšie.

Kvôli penalizácii veľkých synaptických váh sa v tomto prípade môže použiť weight decay.

Váhy sa menia po 64 vzorkách – batch size.

Vstupný vektor (šum) do Generátora má veľkosť  $1 \times 100$ .

Počet epoch je na pomery klasických neurónových sietí aj konvolučných hlbokých sietí veľmi veľký. A to v rádovo niekoľko desiatok tisíc.

##### B. Parametre Discriminator-a

Vstupný obraz  $28 \times 28 \times 1$

Filter 1  $5 \times 5 \times 64$ , stride 1 – 64 príznakových platní  
 Filter 2  $5 \times 5 \times 126$ , stride 2 – 128 príznakových platní  
 Filter 3  $5 \times 5 \times 256$ , stride 2 – 256 príznakových platní  
 Filter 4  $5 \times 5 \times 512$ , stride 1 – 512 príznakových platní

##### C. Parametre Generator-a

Filter 1  $5 \times 5 \times 128$ , padding  
 Filter 2  $5 \times 5 \times 64$ , padding

Filter 3  $5 \times 5 \times 32$ , padding  
 Filter 4  $5 \times 5 \times 1$ , padding

#### V. VÝBER L FUNKCIE A JEJ MATEMATICKÝ POPIS

Kedže každý systém má vlastnú L funkciu, ktorá môže byť rozna, tak som sa rozhodol popísať L funkciu celého systému, ktorá sa nazýva teoretická loss funkcia. Je to vlastne minimax pravidlo, ktoré sa používa v teórii hier.

Teoretická loss funkcia:

$$E(G, D) = \frac{1}{2} \mathbb{E}_{x \sim p_t} [1 - D(x)] + \frac{1}{2} \mathbb{E}_{z \sim p_z} [D(G(z))]$$

$$= \frac{1}{2} (\mathbb{E}_{x \sim p_t} [1 - D(x)] + \mathbb{E}_{x \sim p_g} [D(x)])$$

$z$  – zašumený vektor.

$G(z)$  – výstup generátora.

$x$  – trénovacia vzorka.

$D(x)$  – výstup discriminator-a pre reálnu trénovaciu vzorku.

$D(G(z))$  – výstup discriminator-a pre vstup od generátora.

Optimalizačným kritériom je nájsť také parametre, ktoré sme minimalizovali túto funkciu podľa  $G$  a maximalizovali podľa  $D$ .

Pre Discriminator platí:

$D(x)$  – má byť maximálne.

$D(G(z))$  – má byť minimálne.

Pre Generator platí:

$D(G(z))$  – má byť maximálne.

#### REFERENCES

- [1] GoodFellow, I. (2014). Generative Adversarial Nets. Available at: <https://arxiv.org/pdf/1406.2661.pdf> [Accessed 12 May 2019].
- [2] Radford, A., Metz, L. and Chintala, S. (2016). Unsupervised representation learning with deep convolutional generative adversarial networks. *ICLR 2016*. [online] Available at: <https://arxiv.org/pdf/1511.06434.pdf> [Accessed 12 May 2019].
- [3] Ledig, C., Theis, L., Huszar, F. and Caballero, J. (2017). Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. [online] Available at: <https://arxiv.org/pdf/1609.04802.pdf> [Accessed 12 May 2019].
- [4] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks", in IEEE International Conference on Computer Vision (ICCV), 2017.
- [5] Antipov, G., Baccauche, M. and Degelay, J. (2018). Face Aging with Identity-Preserved Conditional Generative Adversarial Networks. *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [6] Zhang, H., Xu, T., Li, H., Zhang, S., Huang, X., Wang, X. and Metaxas, D. (2016). StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks. [online] Available at: <https://arxiv.org/pdf/1612.03242v1.pdf> [Accessed 12 May 2019].
- [7] Huang, R., Zhang, S., Li, T. and He, R. (2017). Beyond Face Rotation: Global and Local Perception GAN for Photorealistic and Identity Preserving Frontal View Synthesis. [online] Available at: <https://arxiv.org/pdf/1704.04086.pdf> [Accessed 12 May 2019].
- [8] Towards Data Science. (2019). *GAN by Example using Keras on Tensorflow Backend*. [online] Available at: <https://towardsdatascience.com/gan-by-example-using-keras-on-tensorflow-backend-1a6d515a60d0> [Accessed 4 May 2019].

