

GAN – Generatív versengő hálózatok és felhasználási lehetőségeik az aviatikában

Gyarmati Ádám*¹ Horváth Krisztina*² Dr. Rohács Dániel*³

* HungaroControl Magyar Légiforgalmi Szolgálat Zrt., Budapest, Magyarország

¹ (kutatás-fejlesztési szakértő, e-mail: adam.gyarmati@hungarocontrol.hu)

² (kutatás-fejlesztési csoportvezető, e-mail: krisztina.horvath@hungarocontrol.hu)

³ (szakmai fejlesztési osztályvezető, e-mail: daniel.rohacs@hungarocontrol.hu)

Absztrakt: A mesterséges intelligencia mindennapjaink szereplőjévé vált és további folyamatos fejlődésen megy keresztül. A generatív versengő hálózatok megjelenésével képesek vagyunk új, hihető adatot generálni. A folyamat során két hálózat verseng mindaddig, amíg a generáló hálózat nem tud olyan mintaelemet alkotni, amivel átveri a diszkriminatív hálózatot. Az adat lehet hanganyag, kép vagy egyéb adatstruktúra. Megtehetjük a versengő hálózatok segítségével például, hogy éjszakai képből nappali képet alkotunk. Mindehhez rengeteg alapadatra van szükségünk és egy, a modellek tanítását végző erős számítógépre. Ezt a tudást akár a repülés világában is kamatoztathatjuk. Született már cikk ahol a repülőgéptípusok meghatározására használják vagy repülő-felület tervezésnél, aerodinamikai optimalizációhoz. Továbbgondolva, vizionálhatunk képfeljavító modellt távoli tornyunkhoz (remote tower), vagy akár repülőgép anomáliák detektálására szolgáló alkalmazást.

1. BEVEZETÉS

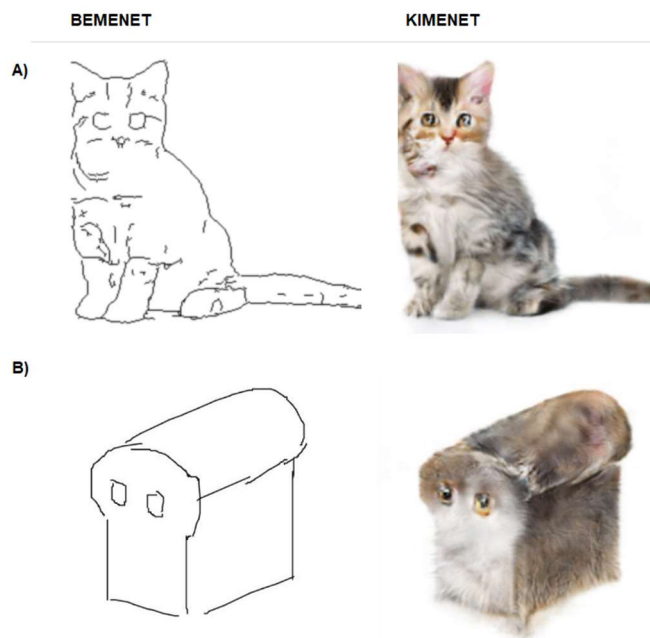
A mesterséges intelligencia (MI) tág fogalom. Általános használatban a kifejezés olyan gépet jelent, amely utánozza az emberi gondolkodást. Tehát képes tanulni és problémákat megoldani, habár nem ugyanúgy mint mi, emberek (Russell és Norvig, 2003).

A mesterséges intelligenciának sokféle kategorizálása létezik, valamint e kategóriák határai eléggé elmosódtak. Egyes értelmezések alapján, már egy egyszerű lineáris regresszió becslései is mesterséges intelligenciának tekinthetők (Hastie et al., 2008). Ha belegondolunk, a magyarázó változó értékei mellé olyan együtthatókat rendel a gépünk ilyenkor, amelyek lineáris kombinációja kiadja a célváltozó értékét. Más szavakkal a megfelelő inputok mellett okos döntést képes hozni a modell.

A gépi tanulás és így a neurális hálózatok már-már megkerülhetetlenek a mindennapi életünkben. A gmail-es fiókunkban egy modell segít automatikusan rendet rakni és a spamet elkülöníteni. Vagy amikor az útvonalunkat tervezzük, mesterséges intelligencia veti össze a lehetséges verziókat és választja ki a leoptimalisabbat. A repülésben is számtalan helyen működik emberi munkát könnyítő MI technológia, elég a robotpilótára gondolni.

Kezdetben a diszkriminatív, osztályozó modellek térnyerése és fejlődése volt jellemző. A legnagyobb sikernek a nagy felbontású, sokdimenziós inputok klasszifikációja tekinthető (Krizhevsky et al., 2012, Hinton et al., 2012). A generatív modellek fellendülése 2014-ben, Ian Goodfellow és csapata

ötlete alapján indult el robbanásszerűen (Goodfellow et al., 2014). Mára rengetegen foglalkoznak a témával. Elérhetővé váltak könnyen alkalmazható modellek az átlag felhasználók számára is. Így az interneten is futótűszerűen terjednek az egyes felhasználók szórakoztató alkotásai (1. ábra).



1. ábra: pix2pix edges2cats modell (Isola et al., 2017) alkalmazva. A) élő demo Christopher Hesse-től, B) Ivy Tsai kenyérmacskája

2. GAN (GENERATIVE ADVERSARIAL NETWORKS) – GENERATÍV VERSENGŐ HÁLÓZATOK

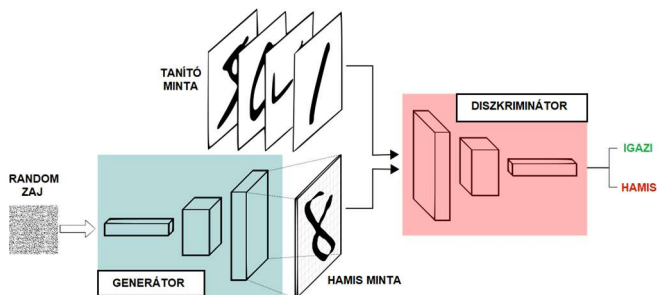
2.1 Generatív mesterséges intelligencia

Míg a diszkriminatív modellek arra alkalmazhatók, hogy a mintánk elemeit bizonyos kategóriákba soroljuk, a generatív modellek mintaelemek generálására fejlődtek ki. Miért is van erre szükség, ha a diszkriminatív modellek paramétereivel is tudunk generálni új céltváltozót? A válasz az, hogy osztályozásnál, vagy egy szimpla regresszióval a célunk egy egyenes vagy görbe illesztése az adatainkra. Bármilyen magyarázó változó értékek mellett kíváncsiak vagyunk a céltváltozó értékére, akkor kiderül, hogy az mindig a becsült egyenesre vagy görbére fog esni. Könnyen belátható, hogy így az új adatunk mesterkélt lesz, senki sem fogja elhinni. A generátornak teljesen más rendeltetése van. Az eredménye egy sűrűségfüggvény lesz, ami jól leírja a tanító mintánkat. Így ha generátorral alkotunk új mintaelemet, az nem egy egyenesre vagy görbére fog illeszkedni, hanem egy bizonyos eloszlást fog követni. A nagy különbség az, hogy így az adatainkban lesz egy afféle sztochaszticitás, ami élőbbé, hihetőbbé teszi őket és pont ez a célunk. A változatoság részét a folyamatnak egy random zaj adja, amiből a generátor megtanul megalkotni egy olyan mintaelemet, ami illeszkedik a céleloszlásba.

2.2 Versengés a hálózatok között

„A generatív versengő hálózat az elmúlt tíz év legérdekesebb ötlete a gépi tanulásban” – jelentette ki Yann Lecun, a Facebook MI igazgatója.

A generatív versengő hálózatok (Goodfellow et al., 2014, Salimans et al., 2016) különlegességét az adja, hogy nem egy, hanem kettő hálózatot tanítunk egyszerre, ráadásul ezek egymás versenytársai. Az egyik hálózat egy diszkriminátor $D(x)$, aminek a feladata, hogy megtanulja megkülönböztetni a tanító minta adatait, $p_{\text{tanító}}(x)$, a hamis, modell által generált mintáktól $p_{\text{modell}}(x)$. A másik hálózat pedig egy generátor $G(z)$, aminek az előbb említett diszkriminátor átverése a célja. Random z zajból igyekszik valósnak tűnő mintaelemeket létrehozni: $x = G(z)$ (2. ábra).



2. ábra: generatív versengő hálózatok tanítási folyamata (Thalles Silva ábrája).

A tanítási folyamat során a két hálózatot párhuzamosan tanítjuk, de felváltva frissítjük a paramétereiket. A folyamat leírható egy egyszerű játékelméleti modellel, ahol a pénzhamisító ($G(x)$) és a rendőrség ($D(z)$) próbálják a saját akaratukat érvényesíteni. A pénzhamisító hamis pénzt készít, a rendőr pedig próbálja lebuktatni. A tanítási folyamat elején még egyikük sem végzi olyan jól a dolgát. Viszont ahogy fejlődik a diszkriminátor hálózat és egyre jobban ki tudja szűrni a generált elemeket, úgy a generátornak is fejlődnie kell, hogy sikeres legyen. A folyamat során úgy tanítjuk a diszkriminátort, vagyis a rendőrséget, hogy maximalizáljuk az esélyét annak, hogy jól ítélje meg az igazi inputokat $\log D(x)$, vagyis az igazi pénzt és a hamis, generált elemeket $\log(1 - D(G(z)))$, vagyis a hamis pénzt. A generátort ellenben arra tanítjuk, hogy minimalizálja annak lehetőségét, hogy kiszúrják a hamisítást, tehát $\log(1 - D(G(z)))$ -t. Így eljutunk addig, hogy a probléma leírható egy kétjátékos minimax játékként:

$$\min_G \max_D V(D, G) = E_{x \sim p_{\text{tanító}}(x)} [\log D(x)] + E_{z \sim p_z(z)} [\log(1 - D(G(z)))]. \quad (1)$$

A tanítási folyamat végére, optimális esetben a diszkriminátor fele-fele arányban fogja az elemekre azt mondani, hogy igazi vagy hamis, teljesen függetlenül az eredetétől. Más szavakkal, nem tudja eldönteni melyik igazi és melyik nem, csak tippel. Ekkor a generátor már olyan elemeket generál amelyek nem tűnnek ki a tanító minta eloszlásából, így akár az emberi érzékszerveknek is valósnak tűnhetnek az eredmények. Ez egy könnyed példával élve jelentheti azt, hogy tanító mintának macskás képeket használva, generálhatunk emberi szemnek hihető, de még sosem látott, új macskás képeket (1. ábra). A generátor ez esetben megtanulja mi tesz egy macskát macskává és milyen variancia van a képekben. Majd a random zaj sztochaszticitását alapul véve, random méretű, pozíciójú, színű macskát kapunk a képen.

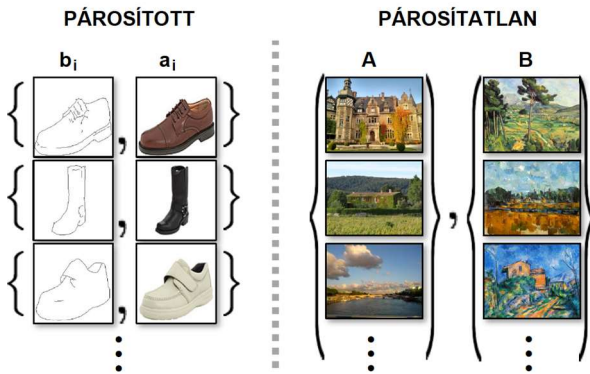
2.3 GAN általános felhasználása

A GAN-t sok esetben vizuális célokra használják. Már régebben is remek képek generálására volt alkalmas a módszer (Radford et al., 2015, Denton et al., 2015) azonban napjainkban még kifinomultabbakat, még egyszerűbben tud létrehozni.

A mára már híressé vált cycleGAN (Zhu et al., 2017) és pix2pix (Isola et al., 2017) dizájnoknak számtalan felhasználási lehetősége van kép↔kép transzformációval kapcsolatban. Már az alkotók is élénk tártak olyan alapfunkciókat, mint: nappal↔éjszaka, nyár↔tél, fotó↔festmény, fekete-fehér↔színes, körvonal↔táska. Csak azon múlik felhasználásuk, hogy milyen adatokkal tanítjuk meg a modellünknek a képalkotást.

A pix2pix párosított mintákat fogad el a tanításhoz (3. ábra). Így meg kell lennie mind a két verzióknak (b_i, a_i) az adott képből és abból fogja megtanulni, hogyan kell az egyik verzióból (b_i) a másik felét (a_i) megalkotni. Más szavakkal ez egy feltételes (conditional) verziója a generatív versengő hálózatoknak, így angolul CGAN-ként hivatkoznak rá. Ez esetben kijelenthetjük, hogy nem a random zajból készül a generátor hamis képe, hanem a kiinduló képből. Mivel mindkét kép rendelkezésre áll, így a hálózat könnyen el tudja dönteni, hogy mennyire hasonló az eredeti képhez a végkimenet: $G(b_i) \approx a_i$, és büntetni tudja az eltéréseket. Az adott eltéréseket általában nem pixel szinten, hanem nagyobb foltokban (PatchGAN) vizsgálja a rendszer. Ez a vizsgálat, ha szép eredményt akarunk kapni, általában azt is tartalmazza, hogy az adott kép a céleloszlásba illik-e, és hogy mennyire hasonlít az eredeti képre, kombináltan.

A másik közkedvelt, általános felhasználású dizájn, a cycleGAN előnye, hogy párosítatlan mintákat használ (3. ábra). Ez azt jelenti, hogy nem kell azzal bajlódni, hogy például ugyanaz a tájkép meglegyen nyáron és télen is, nyugodtan dolgozhatunk más-más tájakkal. Viszont ez azzal jár, hogy persze tudja hogyan kell átalakítani az egyik mintából származó képet ($a_i \in A$), hogy úgy tűnjön a másik mintából származik ($b_i \in B$), de nem tudja vizsgálni, hogy mennyire tér el az eredeti felépítésétől. Erre az a megoldás, hogy nem kettő, hanem négy hálózatot használ a program. Ebből egy G_{AB} végzi az $A \rightarrow B$ átalakítást és egy másik generátor, G_{BA} a visszafelé, $B \rightarrow A$ alakítást. Így a D_B meg tudja vizsgálni, hogy az első átalakítás i. e. $G_{AB}(a_i)$ utáni kép követi-e a B minta eloszlását, valamint D_A , hogy a második átalakítás i. e. $G_{BA}(b_i)$ után a végeredmény az A mintába illik-e. Ez a körfolyamat azért fontos, mert így a rendszer meg tudja vizsgálni és büntetni tudja, ha az eredeti kép nem hasonlít a folyamat végén kapott képre: $G_{BA}(G_{AB}(a_i)) \approx a_i$.

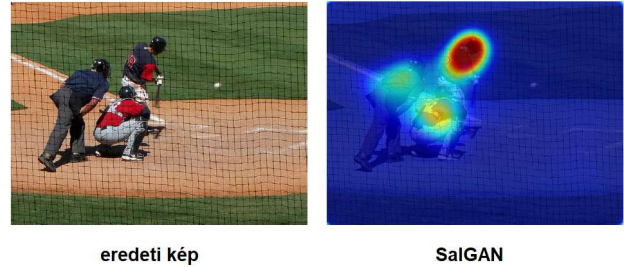


3. ábra: párosított és párosítatlan minták (Zhu et al., 2017).

A GAN-nak már számtalan másik egyedi verzióját dolgozták ki és ezek más-más célokat szolgálnak. Például felhasználható hiányos képek kiegészítésére (Pathak et al., 2016), vagy esős képekből lehet tiszta képeket generálni (Zhang, Sindagi, et al., 2017). Az arkifejezések módosítása is népszerű témakör (Perarnau et al., 2016). Kidolgozták például a SalGAN-t, ami a kitüntetett jelentőségű részleteket jelzi (4. ábra) egy képen

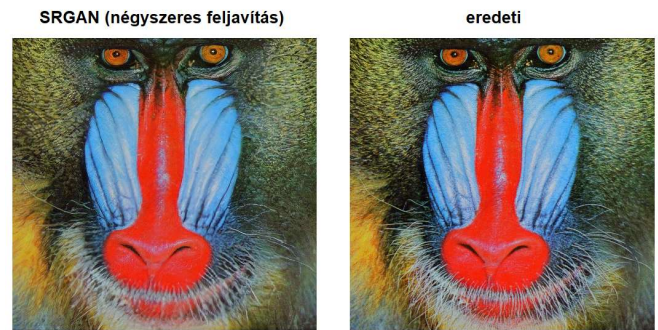
vagy felvételen (Pan et al., 2017). Született továbbá a fájlok elrejtésére, szteganográfiára alkalmazható GAN is (Shi et al., 2017).

Pózzok felvételére is rábírhathat minket virtuálisan a GAN (Ma et al., 2017). Az eredeti képből és egy hozzáadott pózból – mint metaadattól – egy feltételes GAN dizájn a képen látható embert adott pózban képes rekonstruálni. A program egy két fokozatú generátorból és a szokásos diszkriminátorból áll.



4. ábra: eredeti kép és SalGAN segítségével feltüntetett kulcsfontosságú részek ugyanazon a képen, hőterképként (Pan et al., 2017).

A twitter csapata kidolgozott egy olyan GAN-t, ami képes felbontásjavításra (Ledig et al., 2017). Ez egy olyan terület, ahol a GAN rendkívül bámulatos eredményeket produkál és azonnali kereskedelmi potenciállal rendelkezik. A fejlesztők SRGAN-nak, azaz super resolution GAN-nak nevezték el alkotásukat. A modell képes akár négyszeres feljavításra is (5. ábra).



5. ábra: képfelbontás-javítás SRGAN segítségével. Az eredeti kép és a generált szinte megkülönböztethetetlen (Ledig et al., 2017). [négyyszeres felskálázás]

Létezik olyan alkalmazás is, ami képes csupán szöveg alapján megalkotni egy képet (Zhang, Xu, et al., 2017). A dizájnt stackGAN-nak nevezték el, ami két fázisban alkot képeket. Az elsőben kontextusba helyezi és nagyjából felvázolja a leírt dolgot, majd a második fázisban kidolgozza a részleteket. Ezek a képek nagy részletességgel bírnak (6. ábra).

Sokszor nehéz valós mintákat találni a modell tanításához. Ilyenkor egyszerű megoldásnak ígérkezik, hogy szintetikus

képeket használunk. Viszont ez általában, a szintetikus és a valós képek eloszlásának különbségei miatt, rosszabb végeredményt produkál. Ezt a problémát kiküszöbölő GAN is született már, a SimGAN (*Shrivastava et al., 2017*). Itt a szintetikus kép input alapján kapunk realiztikus végeredményt.

Ennek a madárnak sárga hasa és lába van, szürke háta, szárnya, és barna torka, tarkója, fekete arca

Ez a madár fehér egy kis feketével a fején és szárnyán, és hosszú narancssárga csőre van

Ennek a virágnak átfedésben lévő, hegyes, rózsaszín szirmai vannak amik rövid, sárga filamentumokat vesznek körbe

fázis I.



fázis II.



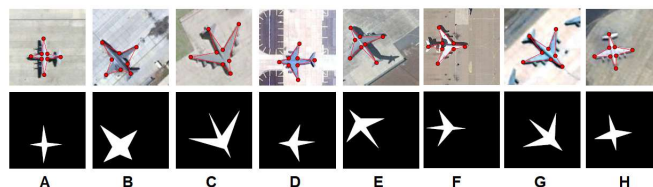
6. ábra: a stackGAN képet generál szöveg alapján, két fázisban (*Zhang, Xu, et al., 2017*).

A GAN technológia nem csak képek, hanem akár hangok generálására is képes. A diszkriminátor hálózatnak megfelelő melódiákat adva, a generátor igyekszik azokhoz hasonló, még sosem hallott zenéket komponálni. A MidiNet névre hallgató GAN (*Yang et al., 2017*) a tesztelő felhasználók szerint izgalmas, kellemes, reális dallamokat alkotott.

3. GAN FELHASZNÁLÁSA AZ AVIATIKÁBAN

A repülőgépek típusának meghatározása rendkívül hasznos, ha távoli (remote) környezetben dolgozunk fel képeket. Viszont a tradicionális megoldások nehezen általánosíthatóak, valamint a mély tanuláshoz rengeteg címkével ellátott képre volna szükség, így elég költséges és időigényes volna. A probléma kiküszöbölésére egy repülőgéptípus felismerő GAN-t dolgozott ki Yuhang Zhang és csapata (*Zhang et al., 2018*). A dizájn több fázisú és inputként repülőgépekről készült címke nélküli képeket igényel. Az első fázisban a képekre egy homokóra hálózat (*Newell et al., 2016*) felhelyezi a nyolc kulcsfontosságú pontot, amelyek a gép orra, farka, jobb és bal szárnyvége és négy pont a törzsön, ahonnan a szárnyak erednek. Ezekből a pontokból a következő lépésben elkészül egy-egy körvonal (7. ábra). A feltételes GAN-nak ezek a körvonalak a bemenetei és igyekszik ezekből egy új mintát létrehozni, szigorúan odafigyelve a kulcsfontosságú pontokra. A dizájn több diszkriminátort használ, hogy a modell különböző méretarányú és felbontással rendelkező képeket is kezelni tudjon. A diszkriminátorokból ezután kiszűrjük a

megfelelő, reprezentatív információkat a típusok felismeréséhez, majd azokat főkomponens analízissel csökkentik. Végül a feldolgozott információból, ami az egyes gépeket írja le a képek alapján, egy osztályozó szupport-vektor hálózat készül. Ez a hálózat az új képeken látható gépeket típusok alapján el tudja különíteni címke nélkül.



7. ábra: a felső sorban repülőgépek a nyolc kulcsfontosságú pontjaikkal, míg az alsó sorban ezekből alkotott körvonalaik. A-H-ig a betűk különböző repülőgép típusokat jelölnek (*Zhang et al., 2018*).

Az utasok adatai nagyon fontosak az utazási iparban. Amint létrejön egy út lefoglalása, az adatokat elmentik. A légitársaságok és egyéb ipari szereplők sokszor ezeket az adatokat cseréltetik. A kihívás az, hogy ezeket sensu stricto nem lehet felhasználni jogsértés nélkül. Ez különösen fennáll az európai piacra a GDPR miatt, ami szigorú szabályozásokat tartalmaz. Például kimondja, hogy az adatokat törölni kell az eredeti céljuk beteljesedése után. A probléma megoldásaként kifejlesztettek egy GAN-t, ami realiztikus, szintetikus utazási adatokat generál (*Mottini et al., 2018*). Ez a GAN így nem képet vagy hangot, hanem számokat, szavakat és hiányzó értékeket alkot. Az alap elképzelés az, hogy a modell alkot egy olyan mintát, ami eloszlásában hasonlít az eredetire, de mindezt az eredeti mintaelemek ismétlése vagy memorizálása nélkül. A célja, hogy egy reálisnak tűnő adatot lehessen generálni és felhasználni különböző adatvezérelt üzleti alkalmazásokhoz. A megoldáshoz Cramér GAN-t használtak (*Bellemare et al., 2017*), amely nevéhez hűen a Cramér távolságon alapul.

A repülőgépek tervezésénél kardinális kérdés az aerodinamikai optimalizáció. Ehhez általában rengeteg tervezés és rendkívüli adatbázis szükséges. A feladat szűk keresztmetszetét a számtalan szimuláció jelenti, amin eddig a dimenziók csökkentésével próbáltak javítani. Ezt a problémát is meg lehet oldani generatív versengő hálózatokkal (*Chen et al., 2019*). A GAN a repülő-felület adatbázisból (UIUC) tanulta meg az eddigi forma variációkat. A vektorgrafikában a Bézier-görbéket szabadon alakítható sima görbék modellezésére használják, így a formák realiztikusabbá, simább görbületűvé tételéhez Bézier-GAN-t használtak (*Chen és Fuge, 2018*). Így a modell képes realiztikus szárnyakat, propellereket és egyéb repülő-felületeket generálni (8. ábra), amik remek alapul szolgálnak további tervezésekhez, ezzel rengeteg időt spórolva. A Bézier-GAN további sima felületű járművek, így akár hajók tervezésénél, azok hidrodinamikai optimalizációjánál is megállja a helyét.



8. ábra: repülő-felületek Bézier-GAN felhasználásával és a Bézier réteg felhasználása nélkül (*Chen et al., 2019*).

Igaz, nem jelent még meg sok irodalom arról, miként használható a GAN az aviatikában, de mivel ez egy viszonylag új témakör, rengeteg kiaknázatlan lehetőség rejlik benne.

A légiforgalomban például rengeteg érzékeny, titkos információ létezik. A légitársaságok nem adják ki a gépek felszállási tömegét, utaslistáját, üzemanyagszintjét. Igaz rengeteg információ megtalálható a gépekről általánosságban például a BADA adatbázisban, viszont azok csak pontbecslések vagy jobb esetben tartományok, de nem reprezentatívak. Sokkal realiztikusabb, szintetikus adatokat lehetne GAN-nal generálni, amik nem tartalmazzák a valós értékeket, de követik az eloszlásukat. Ezek felhasználásával sokkal realiztikusabb szimulációkat, gyakorlatokat lehetne tervezni a légiforgalmi irányító gyakorlatoknak és álpilótáknak.

Különböző egyedi GAN-okkal távoli torony kameráképek felbontását, képét is lehetne javítani. Például ki lehetne szűrni vele az esőt a felvételekről, így nagyobb helyzet tudatosságot adva az irányítóknak. A kamerákra akár a kulcsfontosságú pontokat is fel lehetne vinni hő térképként, ezzel a megfelelő pontokra irányítva a figyelmet.

A repülőgépek meghibásodásairól, bármilyen anomáliájáról is készíthetünk előrejelzést generatív versengő hálózatokkal. Ehhez viszont kellő számú rögzített jelenségre volna szükség.

4. ÖSSZEFOGLALÁS

A generatív versengő hálózatok segítségével többek között realiztikus adatokat tudunk generálni. Erősségük, hogy a generált adat realiztikusnak hat, mivel a hálózat célja olyan adat generálása ami az adott céleloszlásba illeszkedik. Számptalan irodalomban felhasználták már különböző célokra a GAN-t. Ilyen például a képfelbontás-javítás, képmódosítás, zeneszerzés, titkosítás, tervezés, vagy kategorizálás. A technológiát már az aviatikában is használják repülőgéptípus meghatározására kép alapján, titkosított adatok átadására GDPR megszegése nélkül és repülő-felületek tervezésére. Úgy gondoljuk, hogy ez egy nagyon ígéretes terület és érdemes volna további fejlesztésekhez is alkalmazni a generatív mesterséges intelligenciát.

5. IRODALOMJEGYZÉK

- Bellemare, M. G. és I. Danihelka, W. Dabney, S. Mohamed, B. Lakshminarayanan, S. Hoyer, R. Munos (2017). The cramer distance as a solution to biased wasserstein gradients. arXiv preprint arXiv:1705.10743v1
- Chen, W. és M. Fuge (2018). BézierGAN: automatic generation of smooth curves from interpretable low-dimensional parameters. arXiv preprint arXiv:1808.08871
- Chen, W. és K. Chiu, M. D. Fuge (2019). Aerodynamic design optimization and shape exploration using generative adversarial networks. *AIAA Scitech 2019 Forum*. doi: 10.2514/6.2019-2351
- Denton, E. és S. Chintala, A. Szlam, R. Fergus (2015). Deep generative image models using a laplacian pyramid of adversarial networks. *Advances in Neural Information Processing Systems 28 (NIPS 2015)*. arXiv:1506.05751
- Goodfellow, I. és J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio (2014). Generative adversarial nets. *Advances in Neural Information Processing Systems 27 (NIPS 2014)*
- Hastie, T. és R. Tibshirani, J. Friedman (2008). *The elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Science & Business Media. Stanford, California: Springer.
- Hinton, G. és L. Deng, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, B. Kingsbury (2012). Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Processing Magazine*, **29(6)**, 82-97.
- Isola, P. és J.-Y. Zhu, T. Zhou, A. A. Afros (2017). Image-to-image translation with conditional adversarial networks. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1125-1134
- Krizhevsky, A. és I. Sutskever, G. Hinton (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems 25 (NIPS 2012)*
- Ledig, C. és L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, W. Shi (2017). Photo-realistic single image super-resolution using a generative adversarial network. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4681-4690
- Ma, L. és X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, L. Van Gool (2017). Pose guided person image generation. *Advances in Neural Information Processing Systems 30 (NIPS 2017)*
- Mottini, A. és A. Lhéritier, R. Acuna-Agost (2018). Airline passenger name record generation using generative adversarial networks. arXiv preprint arXiv:1807.06657
- Newell, A. és K. Yang, J. Deng (2016). Stacked hourglass networks for human pose estimation. *Computer Vision-ECCV*; B. Leibe, J. Matas, N. Sebe, M. Welling, eds. Springer International Publishing: Cham, Switzerland. pp. 483-499

- Pan, J. és C. Canton-Ferrer, K. McGuinness, N. E. O'Connor, J. Torres, E. Sayrol, X. Giro-i-Nieto (2017). SalGAN: visual saliency prediction with adversarial networks. *CVPR*
- Pathak, D. és P. Krähenbühl, J. Donahue, T. Darrell, A. A. Efros (2016). Context encoders: feature learning by inpainting. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2536-2544
- Perarnau, G. és J. van de Weijer, B. Raducanu, J. M. Álvarez (2016). Invertible conditional GANs for image editing. arXiv preprint arXiv:1611.06355
- Radford, A. és L. Metz, S. Chintala (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434
- Russell, Stuart J. és Norvig, Peter (2003). *Artificial intelligence: a modern approach*. 2nd ed. Upper Saddle River, New Jersey: Prentice Hall. ISBN 0-13-790395-2
- Salimans, T. és I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen (2016). Improved techniques for training gans. *Advances in Neural Information Processing Systems 29 (NIPS 2016)*
- Shi, H. és J. Dong, W. Wang, Y. Qian, X. Zhang (2017). SSGAN: secure steganography based on generative adversarial networks. In: Zeng B., Huang Q., El Saddik A., Li H., Jiang S., Fan X. (eds) *Advances in Multimedia Information Processing – PCM 2017. Lecture Notes in Computer Science, vol 10735*. Springer, Cham
- Shrivastava, A. és T. Pfister. O. Tuzel, J. Susskind, W. Wang, R. Webb (2017). Learning from simulated and unsupervised images through adversarial training. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2107-2116
- Yang, L-C. és Sz-Y. Chou, Y-H. Yang (2017). MidiNet: a convolutional generative adversarial network for symbolic-domain music generation. arXiv preprint arXiv:1703.10847
- Zhang, Han és T. Xu, H. Li, S. Zhang, X. Huang, X. Wang, D. Metaxas (2017). StackGAN: text to photo-realistic image synthesis with stacked generative adversarial networks. *The IEEE International Conference on Computer Vision (ICCV)*, pp. 5907-5915
- Zhang, He és V. Sindagi, V. M. Patel (2017). Image de-raining using a conditional generative adversarial network. arXiv preprint arXiv:1701.05957
- Zhang, Y. és H. Sun, J. Zou, H. Wang, G. Xu, X. Sun (2018). Aircraft type recognition in remote sensing images based on feature learning with conditional generative adversarial networks. *Remote Sensing*, **10**, 1123. doi: 103390/rs10071123
- Zhu, J.-Y. és T. Park, P. Isola, A. A. Efros (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. *The IEEE International Conference on Computer Vision (ICCV)*, pp. 2223-2232