



UNIVERSITAT DE  
BARCELONA

**Treball final de grau**

**GRAU DE MATEMÀTIQUES**

**Facultat de Matemàtiques i Informàtica  
Universitat de Barcelona**

---

**Com podem programar l'art?  
Introducció a les Generative  
Adversarial Networks**

---

**Autora: Clara Rivadulla Duró**

**Director: Dr. Eloi Puertas Prats  
Realitzat a: Departament de Matemàtiques i Informàtica  
Barcelona, 13 de juny de 2022**

## Abstract

Generative art and computational creativity are two areas that have been very popularized between artists, engineers, and theorists of philosophy and psychology in the past few years. Computational advances and their recent applications challenge the creative exclusivity of the human being. But can a computer be creative?

We still can't find an answer and maybe we never will, being art something so personal and subjective, it avoids all sorts of agreement and convention. Nevertheless, we can sow doubt. We will do so by implementing a Generative Adversarial Network, a recent trend in the world of artificial intelligence and deep learning, and training it with the work of a renowned artist. The results will be analyzed, always subjectively, through comparisons between executions, and surveys.

## Resum

L'art generatiu i la creativitat computacional són dues àrees popularitzades en els darrers anys tant entre artistes, com enginyers, com teòrics de la filosofia i la psicologia. Els avenços computacionals i les seves aplicacions recents desafien l'exclusivitat creativa de l'ésser humà. Però pot un ordinador ser creatiu?

Encara no podem trobar una resposta i potser mai no ho podrem fer, essent l'art quelcom tan subjectiu i personal, que defuig de tot acord i convenció. Malgrat això, sí que podem sembrar el dubte. Ho farem implementant una xarxa generativa antagònica, tendència recent al món de la intel·ligència artificial i el *deep learning* [1], i entrenant-la amb l'obra d'una artista reconeguda. Els resultats s'analitzaran, sempre de manera subjectiva, mitjançant comparacions entre execucions i enquestes.

## Resumen

El arte generativo y la creatividad computacional son dos áreas muy popularizadas en los últimos años tanto en artistas, como ingenieros, como en teóricos de la filosofía y la psicología. Los avances computacionales y sus recientes aplicaciones desafían la exclusividad creativa del ser humano. Pero ¿puede un ordenador ser creativo?

Todavía no podemos hallar una respuesta y tal vez nunca lo hagamos, pues el arte es algo tan subjetivo y personal que rehúye de todo acuerdo y convención. Sin embargo, sí podemos sembrar la duda. Lo haremos implementando una red generativa antagónica, reciente tendencia en el mundo de la inteligencia artificial y el *deep learning*, y entrenándola con la obra de una artista reconocida. Los resultados se analizarán, siempre de forma subjetiva, mediante comparaciones entre ejecuciones y encuestas.

# Índex

<b>1</b>	<b>Introducció</b>	<b>1</b>
<b>2</b>	<b>Objectius</b>	<b>3</b>
<b>3</b>	<b>Planificació</b>	<b>5</b>
<b>4</b>	<b>Estat de l'art</b>	<b>6</b>
4.1	Creativitat computacional . . . . .	6
4.2	Art generatiu . . . . .	9
4.2.1	L'aleatorietat . . . . .	9
4.2.2	AARON . . . . .	10
4.2.3	Processing . . . . .	11
4.2.4	AI Art . . . . .	12
4.3	Conceptes previs de <i>machine learning</i> . . . . .	15
4.3.1	Supervised learning . . . . .	15
4.3.2	Discriminative modeling . . . . .	15
4.3.3	Unsupervised learning . . . . .	16
4.3.4	Generative modeling . . . . .	16
4.3.5	Artificial Neural Networks . . . . .	16
4.3.6	Convolutional Neural Networks . . . . .	18
4.3.7	Latent Space . . . . .	20
4.3.8	Binary Cross-Entropy Loss Function . . . . .	21
4.3.9	Adam Optimizer . . . . .	22

4.4	Generative Adversarial Networks . . . . .	23
4.4.1	Extensions i aplicacions de les GANs . . . . .	26
4.4.1.1	Conditional Generative Adversarial Network . . . . .	26
4.4.1.2	Pix2Pix . . . . .	26
4.4.1.3	Cycle-Consistent Generative Adversarial Network . . . . .	27
4.4.1.4	Style-Based Generative Adversarial Network . . . . .	28
4.4.1.5	Super Resolution Generative Adversarial Network . . . . .	28
4.4.1.6	Stacked Generative Adversarial Networks . . . . .	30
<b>5</b>	<b>Implementació d'una GAN</b>	<b>32</b>
5.1	Codi base . . . . .	32
5.2	Entorn de programació . . . . .	32
5.3	Set de dades . . . . .	33
5.4	Discriminador . . . . .	34
5.5	Generador . . . . .	35
5.6	Entrenament de la GAN . . . . .	36
<b>6</b>	<b>Resultats</b>	<b>37</b>
6.1	Execucions . . . . .	37
6.2	Enquestes . . . . .	39
6.2.1	Resultats de l'enquesta A . . . . .	39
6.2.2	Resultats de l'enquesta B . . . . .	42
6.2.3	Anàlisi final . . . . .	44
<b>7</b>	<b>Treball futur</b>	<b>45</b>

<b>8 Conclusions</b>	<b>46</b>
<b>Annexos</b>	<b>48</b>
<b>A Estructura del discriminador i les seves CNNs</b>	<b>48</b>
<b>B Estructura del generador i les seves CNNs</b>	<b>50</b>
<b>Referències</b>	<b>50</b>

# 1 Introducció

Fruit d'un fort interès personal per la tecnologia i l'art en totes les seves formes, neix la necessitat d'explorar certes àrees de la informàtica recentment popularitzades entre artistes i teòrics tant de la intel·ligència artificial com de l'art, la filosofia i la psicologia: l'art generatiu i la creativitat computacional.

La creativitat computacional s'introdueix amb una finalitat merament contextual. En aquest treball no es pretén determinar què és la creativitat, sinó desafiar la creença tradicional que només pot ser una qualitat humana. Aquesta creença es posa en dubte exposant el concepte d'art generatiu, del qual es fa una breu explicació teòrica, amb exemples des del segle passat fins a l'actual, que evidencien la seva evolució en paral·lel amb la tecnologia i la computació.

Posem èmfasi a l'AI Art, un subgrup d'art generatiu que fa ús de tècniques d'intel·ligència artificial en el seu procés, més concretament, en l'art que es genera amb l'ús d'una *Generative Adversarial Network* [2], o GAN per abreujar. Abans d'explicar en profunditat el funcionament d'una GAN, però, introduïm alguns dels conceptes clau de *machine learning* [3], com les xarxes neuronals convolucionals, l'espai latent, la funció de pèrdua i l'optimitzador, que utilitzen la majoria de les GANs.

A la secció 4.4 *Generative Adversarial Networks*, s'exposa l'arquitectura i el funcionament general d'una GAN, composta de dues xarxes neuronals: un discriminador i un generador. A més a més, es presenten algunes de les extensions més populars de les GANs, amb les seves respectives aplicacions.

Com a pràctica, s'implementa una GAN i s'entrena amb l'obra de l'artista Georgia O'Keeffe, deixant que la mateixa intel·ligència artificial faci art per nosaltres a partir d'obres ja existents. La seva implementació s'explica en profunditat a la secció 5 *Implementació d'una GAN*, passant per cada pas del procés.

A la secció **6 Resultats**, fem una valoració, tan subjectiva com ho és l'art, dels resultats obtinguts. Recorrem a l'opinió popular mitjançant dues enquestes: una on les persones enquestades saben que les imatges han estat generades amb l'ús d'una GAN i una on es diu que l'obra és d'un artista contemporani anònim. Els resultats de les enquestes no sorprenen: la IA, quan tracta de substituir una funció o qualitat humana, és durament jutjada.

Finalment, s'exposen algunes idees futures d'investigació i s'extreuen les conclusions d'haver realitzat aquest treball que, si més no, desperta la curiositat d'explorar els límits estètics, tècnics i morals de la intel·ligència artificial.



## 2 Objectius

Els objectius d'aquest treball es poden diferenciar en objectius teòrics i objectius pràctics.

### Objectius teòrics

- Saber què és la creativitat computacional i què pretén aconseguir.
- Exposar les definicions de creativitat de diversos teòrics i entendre la dificultat de distingir què és creatiu i què no.
- Explorar l'art generatiu amb exemples i veure la seva evolució des del seu sorgiment fins a l'actualitat.
- Entendre la importància de l'aleatorietat en l'art generatiu.
- Veure exemples d'AI Art que utilitza GANs en el seu procés creatiu.
- Introduir alguns conceptes previs de *machine learning* que són clau per la comprensió de les GANs.
- Descriure el problema que tracta de resoldre cadascuna de les xarxes neuronals de la GAN: la modelació generativa, en cas del generador; i la modelació discriminatòria, en cas del discriminador.
- Comprendre les tècniques de *deep learning* que utilitzen les xarxes neuronals convolucionals de què es compon una GAN: convolucions, funcions de pèrdua i optimitzadors.
- Explicar en profunditat el funcionament d'una *Generative Adversarial Network* per després implementar-la a la part pràctica.

## Objectius pràctics

- Trobar un entorn de programació computacionalment potent que ens permeti executar el codi a implementar.
- Implementar una *Generative Adversarial Network* amb Python.
- Determinar com ha de ser el set de dades amb què entrenarem la GAN.
- Entrenar la GAN amb l'obra de l'artista Georgia O'Keeffe.
- Entendre la utilitat de cadascuna de les llibreries utilitzades.
- Analitzar els resultats obtinguts i sotmetre'ls a l'opinió popular mitjançant enquestes A/B.

### 3 Planificació

La feina a realitzar per tal de complir amb els objectius exposats es divideix en les següents tasques:

1. Lectura i recerca.
2. Redacció de la part teòrica a partir de les lectures i els coneixements adquirits durant la recerca.
3. Implementació d'una Generative Adversarial Network.
4. Testatge de la GAN implementada amb l'obra de l'artista Georgia O'Keeffe.
5. Enquestes i anàlisi de resultats
6. Redacció de la part pràctica.
7. Revisió i perfeccionament del codi i la memòria.

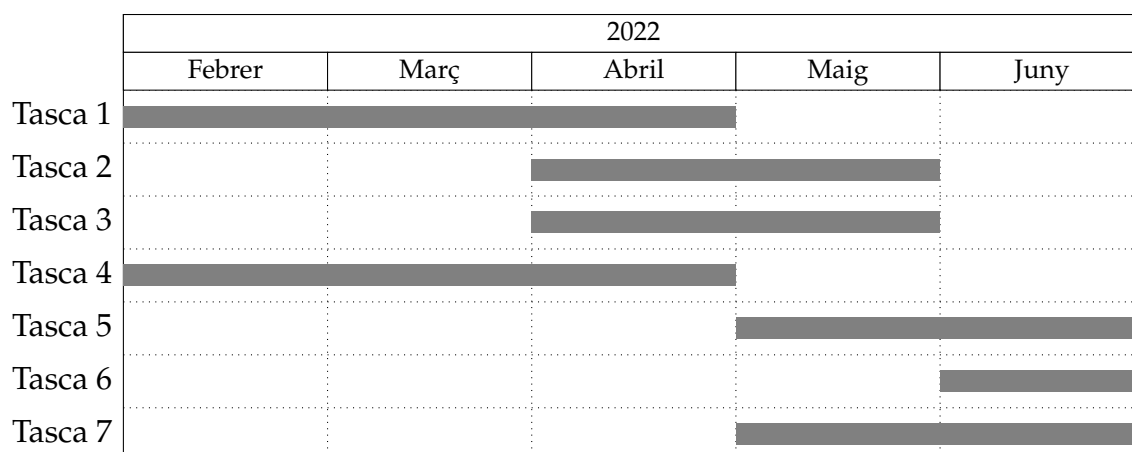


Figura 1: Diagrama de Gantt amb la planificació de les tasques a realitzar

## 4 Estat de l'art

### 4.1 Creativitat computacional

D'acord amb l'*Association for Computational Creativity* [4], la creativitat computacional és l'àrea multidisciplinària que resulta de la intersecció de la intel·ligència artificial, la psicologia cognitiva, la filosofia i l'art.

El seu principal objectiu és el de modelar o replicar la creativitat, mitjançant un ordinador, per tal de:

- Construir un programa o ordinador capaç de ser creatiu al nivell humà.
- Entendre millor la creativitat humana i formular una perspectiva algorítmica sobre el nostre comportament creatiu.
- Dissenyar programes que puguin millorar la creativitat humana sense ser necessàriament creatius ells mateixos.

L'estudi de la creativitat computacional suposa tant el treball teòric que es realitza sobre la naturalesa de la creativitat com el treball pràctic d'implementació de sistemes que puguin ser creatius. Aquests sistemes tenen l'objectiu d'utilitzar-se en tasques creatives autònomes, com ara inventar teories matemàtiques, escriure poemes, pintar quadres o compondre música [5].

Tanmateix, el desenvolupament de la creativitat computacional es troba obstaculitzat pel fet que és quasi impossible definir objectivament què és la creativitat. És un estat mental, cert talent o habilitat o només un procés?

Més concretament, ens formulem les següents qüestions [6]:

- Pot programar-se la creativitat? En sistemes creatius, és la creativitat del mateix sistema o del seu programador?

- Com podem avaluar la creativitat computacional? Què és creatiu en un sistema computacional?
- Si un ordinador només pot fer allò pel qual ha estat programat, com pot considerar-se que el seu comportament és creatiu?

El 1958, a la seva publicació *The Process of Creative Thinking* [7], A. Newell, J. C. Shaw i H. A. Simon determinen els següents quatre criteris per determinar si certa solució és o no creativa:

- És innovadora i útil (sigui per l'individu o per la societat)
- Fa que es rebutgin idees prèviament acceptades
- Resulta d'una persistència i motivació intensa
- Prové d'aclarir un problema que originalment era imprecís

Margaret Boden, investigadora en ciències cognitives al departament d'informàtica de la Universitat de Sussex, distingeix dos tipus de creativitat dependent de la seva rellevància [8]:

- **Creativitat P** (o psicològica): aquella que resulta innovadora només per l'agent que la produeix.
- **Creativitat H** (o històrica): aquella que resulta innovadora per tota la societat.

Així mateix, Boden divideix la creativitat en tres grans tipus, segons el seu origen i la seva naturalesa:

- **Creativitat exploratòria**: és la creativitat que sorgeix d'una exploració dins d'un espai conceptual establert.

- **Creativitat transformacional:** és la creativitat que sorgeix d'una transformació deliberada o que transcendeix a l'espai conceptual establert. Aquest tipus de creativitat és molt més radical i menys comú.
- **Creativitat combinatòria:** és la creativitat que sorgeix de la combinació innovadora d'objectes o idees preexistents. Algunes estratègies comuns són:
  - Situar un objecte familiar dins d'un context desconegut, com feia Marcel Duchamp, o un objecte desconegut en un context familiar.
  - Barrejar dos objectes o gèneres diferents, com cowboys i robots a la sèrie de televisió *Westworld*.
  - Afegir una característica nova i inesperada a un concepte existent, com la càmera quan es va integrar als telèfons mòbil.

La creativitat computacional s'aplica a multitud de disciplines artístiques i en totes elles s'han pogut aconseguir grans resultats: *Deep-speare* [9], un model neuronal per generar poemes, mètrica i rima; *Jukebox* [10], una xarxa neuronal per generar música, incloent el cant i la instrumental de diversos estils; o *Sassbook* [11], per generar textos sobre temes concrets que es donen com a input.

Però totes aquestes eines són realment creatives? Si bé es cert que el que generen és innovador i compleix amb diverses de les definicions de creativitat donades, és això suficient per considerar-ho art?

Al següent apartat, veurem alguns exemples d'art pictòric generat amb l'ús de sistemes computacionals que només accentuaran les preguntes que ens ha guem pogut fer fins ara.

## 4.2 Art generatiu

L'art generatiu [12] és aquell que, parcialment o totalment, ha estat creat amb l'ús d'un sistema autònom.

És, també, un procés de generació algorítmica de noves idees, formes, colors i patrons on l'artista generatiu estableix certes regles que l'ordinador seguirà per crear noves obres. Amb aquestes limitacions artístiques, el resultat s'aproxima al desig de l'artista generatiu que, a diferència de l'artista "convencional", podrà generar milers d'obres en només mil·lisegons.

### 4.2.1 L'aleatorietat

La majoria d'art generatiu es basa en l'aleatorietat [13]. Vera Molnár, una de les artistes generatives més prolífiques del segle 20, la compara amb la intuïció [14]:

"Ara que treballem amb ordinadors, som moderns i diem que la intuïció ha passat de moda, però hi ha quelcom capaç de reemplaçar-la: l'aleatorietat. Perquè una màquina ens mostra bilions de possibilitats a les quals, amb la nostra limitada imaginació, no hi podríem arribar. L'aleatorietat enriqueix els sentits".

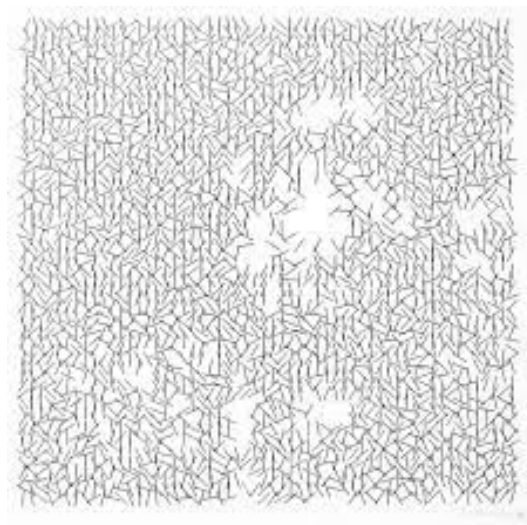


Figura 2: *Interruptions* (1968-69), Vera Molnár

I és que, certament, en incorporar l'atzar en el codi generador, s'obté una obra d'art diferent i única cada vegada que s'executa. A la Figura 2, una de les obres de Molnár, amb traços únics i inesperats fruit de l'aleatorietat mencionada.

#### 4.2.2 AARON

L'artista Harold Cohen va ser considerat un dels primers impulsors de l'art generatiu a finals dels anys 60. Cohen utilitzava robots controlats per ordinador per generar obres pictòriques. Algunes de les seves obres estan exposades a la Tate Gallery de Londres o el MOMA de Nova York. La seva més destacada, però, és AARON (1973) [15]: el primer robot a la història en crear obres d'art originals, entrenat amb eines d'intel·ligència artificial durant molts anys de recerca.

Cohen va desafiar per primera vegada la creença que l'art és una activitat exclusiva de l'ésser humà. És evident que partint de la idea que l'art requereix consciència d'un mateix, el que pugui generar un ordinador no ho serà. Però què és, si no? AARON genera obres que presenten un estil consistent, amb una identitat tan clara com la de qualsevol artista humà, sense cap intervenció [16]. A la Figura 3 veiem un clar exemple de la plausibilitat de l'obra d'AARON.



Figura 3: *TCM #2* (1995) 45×58 — Tinta sobre paper, AARON



Cohen, però, no intenta provar ni el pensament, ni la creativitat, ni la consciència de les màquines; el que vol és demostrar el seu poder de fer algunes coses que havíem assumit que requerien el pensament, la creativitat i l'autoconsciència propis d'un ésser humà.

### 4.2.3 Processing

A partir del 2001, amb el naixement de l'eina *Processing* [17], l'art generatiu va passar a ser accessible per tothom qui tingués un ordinador a la seva disposició (Figura 4).

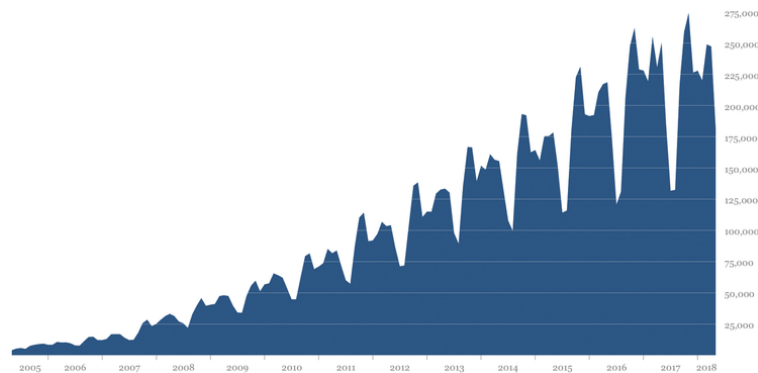


Figura 4: Nombre de vegades que *Processing* s'obre en ordinadors únics cada mes des del 2005 fins a principis del 2018 [18]

Processing va ser creat amb la finalitat de servir com a *sketchbook* i ensenyar els conceptes bàsics de programació en el context visual. Durant aquests darrers vint anys, però, ha anat evolucionant fins a esdevenir una eina de desenvolupament professional. A més a més, és gratuïta, open source i està disponible tant per Mac com per Linux i Windows.

A la Figura 5, una obra de l'artista Matt Pearson, realitzada amb l'ús de Processing. Pearson va publicar, l'any 2011, un manual pràctic d'art generatiu amb Processing: *Generative Art: A practical Guide Using Processing* [19].

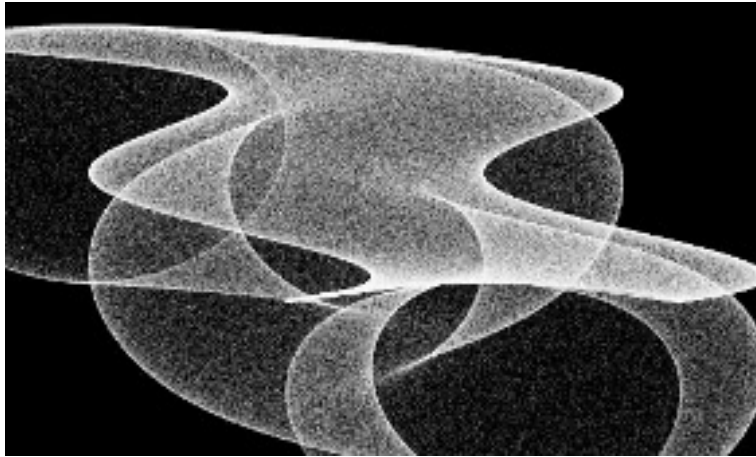


Figura 5: *100 Abandoned Artworks* (2008-2010), Matt Pearson

Malgrat que una eina com Processing, que barrejava l'art amb la programació i ho posava a disposició de tothom, va resultar molt innovadora en el seu temps, ha anat quedant enrere en relació amb els avenços tecnològics i computacionals.

#### 4.2.4 AI Art

L'AI Art (*Artificial Intelligence Art*) [20] és un subgrup d'art generatiu que fa ús de tècniques d'intel·ligència artificial en el seu procés de creació.

L'any 1989, ja s'utilitzaven xarxes neuronals artificials per modelar certs aspectes de la creativitat. Peter Todd va entrenar una xarxa neuronal per reproduir melodies musicals d'un conjunt de peces d'entrenament [21]. Més endavant, podent modificar alguns dels paràmetres d'entrada, la xarxa era capaç de generar música nova aleatòriament, però de forma controlada.

Avui dia, la majoria d'AI Art es fonamenta en l'ús de *Generative Adversarial Networks*, que explicarem en profunditat en el següent apartat (4.4 Generative Adversarial Networks) i posarem a prova a la part pràctica del treball (5 Implementació d'una GAN). Veiem-ne alguns exemples.

## Memories of Passersby I

*Memories of Passersby I* [22] és una obra pionera autònoma que utilitza diverses GANs per generar un flux interminable de retrats [23]. La instal·lació es presenta amb un ordinador allotjat dins d'un armari de fusta, connectat a dues pantalles emmarcades (Figura 6).

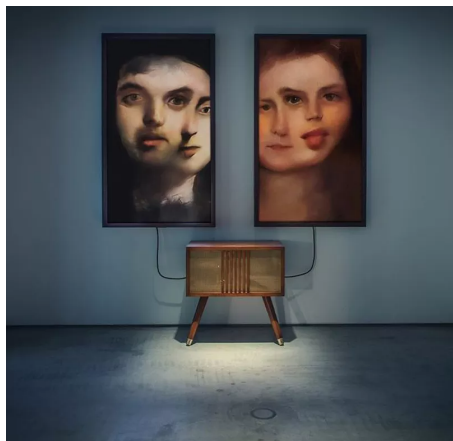


Figura 6: Instal·lació de *Memories of Passersby I*, Mario Klingemann

A diferència d'altres instal·lacions d'art generatiu, *Memories of Passersby I* no conté una base de dades. És un cervell d'IA, desenvolupat i entrenat per Mario Klingemann, que crea nous retrats, píxel a píxel, a temps real. Les sortides que es mostren a la pantalla no són combinacions aleatòries o programades d'imatges existents, sinó obres d'art úniques generades per IA.

Per desenvolupar *Memories of Passersby I*, Klingemann va entrenar el seu model d'IA utilitzant milers de retrats dels segles XVII al XIX i va entrenar la màquina perquè prengués les seves pròpies decisions estètiques, influenciades per figures surrealistes com Max Ernst.

L'obra es subhastava per 46.450 euros l'any 2019 [24].

## Edmond de Belamy

*Edmond de Belamy* [25] és un quadre generat l'any 2018 pel col·lectiu d'artistes

*Obvious* [26] amb l'ús d'una GAN entrenada amb més de 15.000 retrats del segle XIV fins al segle XIX disponibles a WikiArt.

L'obra, impresa sobre tela, pertany a una sèrie d'imatges generatives anomenada *La Famille de Belamy* [27]. El nom Belamy és un homenatge a Ian Goodfellow, creador de les GANs. En francès, "bel ami" significa "bon amic", i en anglès "good fellow".

L'obra va ser subhastada i venuda per 432,500\$ l'octubre del mateix any [28].



Figura 7: *Edmond de Belamy* (2018), 70 x 80 cm, *Obvious*

### 4.3 Conceptes previs de *machine learning*

Per tal d'entendre bé què és i com funciona una xarxa antagònica generativa, cal tenir clars primer alguns dels conceptes bàsics de *machine learning*.

#### 4.3.1 Supervised learning

Molts dels problemes habituals d'aprenentatge automàtic impliquen l'ús d'un model que faci una predicció. D'aquesta aproximació predictiva en diem *supervised learning* [29]. Com podem veure a la Figura 8, amb un *training set* (o set d'entrenament) amb *samples* (o mostres), cadascuna amb el seu input (o entrada)  $X$  i el seu output (o sortida)  $y$  (*labels* o etiquetes), s'entrena i es corregeix el model perquè els seus outputs s'aproximin als outputs reals.

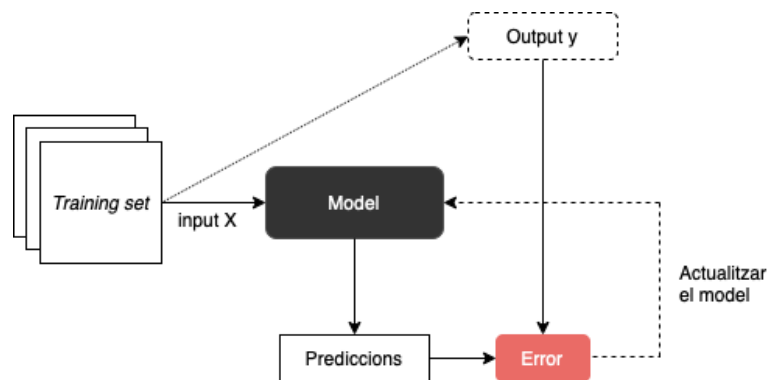


Figura 8: Diagrama d'un model de *supervised learning*

#### 4.3.2 Discriminative modeling

La tasca predictiva rep el nom de classificació, *classification* [30], o modelatge discriminatori, *discriminative modeling* [31], ja que el model ha de discriminar, triar i decidir si un exemple correspon o no a certa classe. Com veurem més endavant, aquesta és precisament la tasca que realitza el discriminador en una GAN.

### 4.3.3 Unsupervised learning

L'altre paradigma conegut d'aprenentatge automàtic és l'*unsupervised learning* [32], on només es dona l'input  $X$ , sense l'output  $y$ , al model. La clusterització (*clustering* [33]) i el *generative modeling* [34] són alguns dels seus problemes més coneguts.

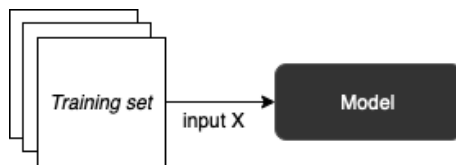


Figura 9: Diagrama d'un model de *unsupervised learning*

### 4.3.4 Generative modeling

En el *generative modeling*, els exemples d'entrenament  $x$  s'extreuen d'una distribució desconeguda  $p_{data}(x)$  i el seu objectiu és aprendre un  $p_{model}(x)$  que s'aproximi a  $p_{data}(x)$  el màxim possible.

Existeixen altres aproximacions de models generatius, com els VAEs (*Variational Autoencoders* [35]) o els autoregressius [36], però, a més de ser més exitoses en la generació d'imatges realistes d'alta qualitat, les GANs resulten molt interessants perquè, a diferència d'altres models basats en l'optimització, estan basades en *game theory* [37].

### 4.3.5 Artificial Neural Networks

Les *Artificial Neural Networks* [38] (o ANNs, per abreviar) són la base dels algorismes de deep learning i tant el seu nom com la seva estructura estan inspirats en el cervell humà, imitant la forma en què les neurones biològiques es comuniquen entre elles.

Les ANNs estan formades per capes de nodes, que contenen una capa d'entrada, una o més capes ocultes i una capa de sortida (Figura 10). Cada node es connecta a un altre i té un pes i un llindar associats. Si la sortida de qualsevol node individual està per sobre del valor llindar especificat, aquest node s'activa, enviant dades a la següent capa de la xarxa; en cas contrari, no es transmeten dades a la següent capa de la xarxa [39].

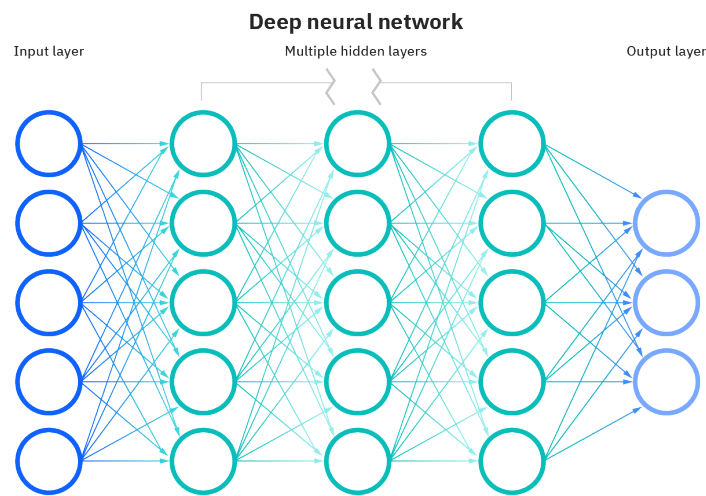


Figura 10: Capes d'una *Deep Neural Network* [39]

Cada node es pot representar amb un model de regressió lineal, amb les seves dades d'entrada, els seus pesos (o *weights* [40]), un llindar (*threshold* [41]) associat i una sortida.

$$\sum_{i=1}^m w_i x_i + bias = w_1 x_1 + w_2 x_2 + w_3 x_3 + bias$$

$$output = f(x) = \begin{cases} 1 & \text{if } \sum w_1 x_1 + b \geq 0 \\ 0 & \text{if } \sum w_1 x_1 + b < 0 \end{cases}$$

Un cop es determina la capa d'entrada, s'assignen els pesos. Aquests pesos ajuden a determinar la importància de qualsevol variable donada, i les més grans contribueixen de manera més significativa a la sortida. Totes les entrades es multipliquen pels seus respectius pesos i se sumen. Després, la sortida passa per una funció d'activació. Si aquesta sortida supera un determinat llindar, s'activa el node, i es passen les seves dades a la següent capa de la xarxa.

El perceptró [42] és la xarxa neuronal més antiga i senzilla, té una única neurona i va ser creada per Frank Rosenblatt l'any 1958 (Figura 11).

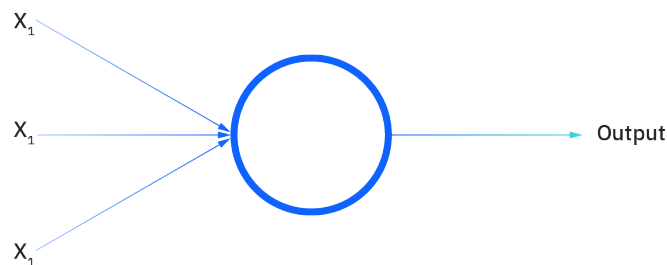


Figura 11: Representació d'un perceptró [39]

Les xarxes neuronals definides fins ara i que utilitzarem en els models d'una GAN, amb una *input layer*, diverses *hidden layers* i una *output layer*, també poden anomenar-se *multi-layer perceptrons* (MLPs) o perceptrons multi-capa.

#### 4.3.6 Convolutional Neural Networks

Les GANs sovint treballen amb imatges i utilitzen xarxes neuronals convolucionals (*Convolutional Neural Networks* [43], o CNNs per abreviar) com a models generadors i discriminadors.

Una convolució és l'aplicació d'un filtre a una entrada que resulta en una activació. L'aplicació reiterada d'un mateix filtre esdevé en un *feature map* [44] (mapa de característiques), que indica les ubicacions i la intensitat d'una característica



detectada a la imatge d'entrada [45].

Les CNNs estan dissenyades per imatges bidimensionals, però també es poden utilitzar amb imatges unidimensionals i tridimensionals.

Quan realitzem una convolució, estem fent una operació lineal: la multiplicació d'un conjunt de pesos amb les imatges d'entrada. Si s'aplica en imatges bidimensionals, la multiplicació es dona entre un array d'input i un array bidimensional amb els pesos (filtre o *kernel* [46]). Per imatges en color, el filtre serà un array tridimensional [47]. La profunditat del filtre ha de coincidir amb el nombre de canals de color. Si treballem amb imatges RGB, el filtre tridimensional tindrà una profunditat de 3 (Figura 12).

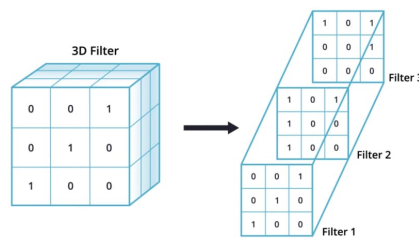


Figura 12: Profunditat d'un filtre per imatges RGB

El *kernel* és més petit que les dades d'entrada i el tipus de multiplicació aplicat entre una part de l'entrada de la mida d'un filtre i el filtre és un producte escalar [48]. Multipliquem el *kernel* per la matriu d'entrada diverses vegades en diferents punts de l'entrada, d'esquerra a dreta i de dalt a baix (Figura 13).

La sortida de multiplicar el filtre amb la matriu d'entrada una vegada és un valor únic. Com que el filtre s'aplica diverses vegades a la matriu d'entrada, el resultat és una matriu bidimensional de valors que representen un filtratge de l'entrada (*feature map* [49]).

Un cop creat el mapa de característiques, podem passar cada valor per la funció d'activació  $\text{ReLU}$  (*rectified linear activation function* [50]), que és la funció d'activació predeterminada per la majoria de xarxes neuronals perquè resulta

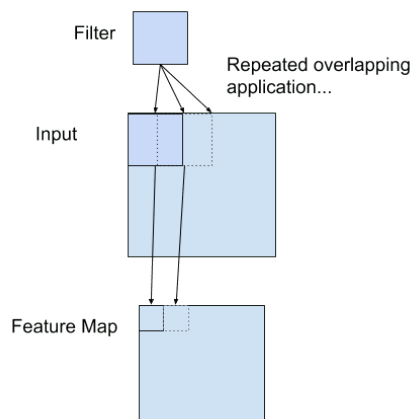


Figura 13: Exemple d'aplicar un filtre a un input bidimensional per crear un mapa de característiques [45]

fàcil d'entrenar i té un millor rendiment [51].

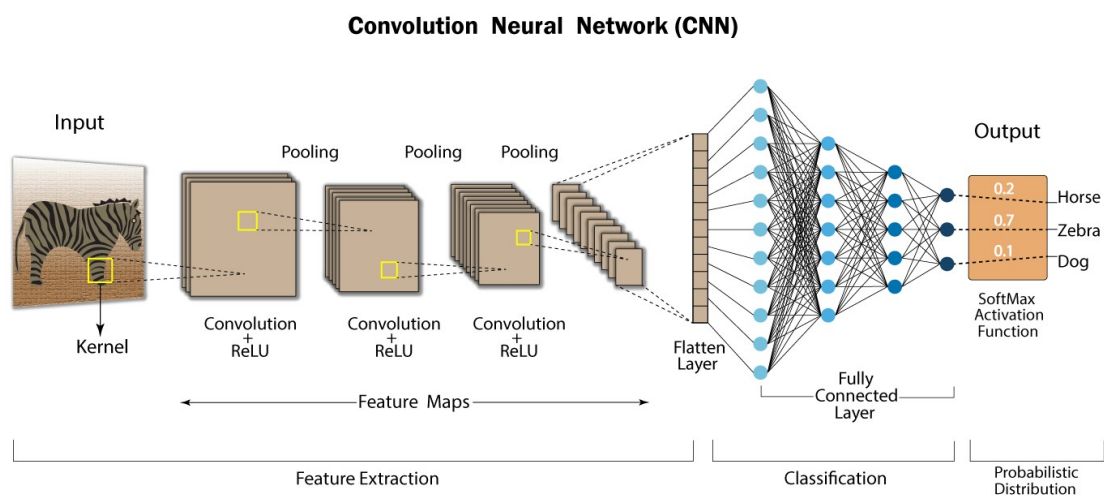


Figura 14: Diagrama d'una xarxa neuronal convolucional

### 4.3.7 Latent Space

L'espai latent o *latent space* [52] es defineix com un espai multi-dimensional abstracte que codifica una representació interna significativa dels esdeveniments observats externament [53]. En resum, és una representació de dades comprimides.

La compressió és necessària quan les dades d'entrada tenen grans dimensions, ja que és impossible aprendre la informació important d'aquestes dades di-

rectament. Per això, per exemple, en una tasca de classificació d'imatges de  $512 \times 512 \times 3$ , que correspon 786.432 píxels d'entrada, és necessari codificar l'espai d'entrada en un espai latent de reduïdes dimensions mitjançant una *deep neural network* [54].

A la Figura 15, podem observar com seria l'espai latent d'un model de classificació d'animals. Els punts verds corresponen al vector latent de cada imatge extreta de l'última capa del model. Com és d'esperar, els vectors relatius als mateixos animals estan més a prop en l'espai latent.

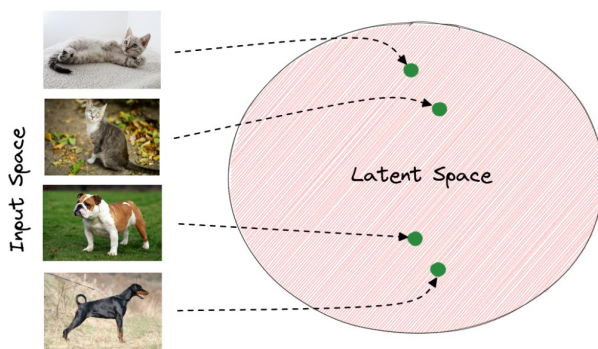


Figura 15: Espai latent d'un problema de classificació d'imatges d'animals [53]

#### 4.3.8 Binary Cross-Entropy Loss Function

Les xarxes neuronals de *deep learning* s'entrenen mitjançant el descens de gradient, mètode iteratiu d'optimització de primer ordre per a trobar el mínim d'una funció [55].

Com a part de l'algorisme d'optimització, l'error de l'estat actual del model s'ha d'estimar repetidament. Això requereix l'elecció d'una funció d'error, anomenada convencionalment *loss function* [56], que es pot utilitzar per estimar la pèrdua del model de manera que els pesos es puguin actualitzar per reduir la pèrdua en la següent avaluació [57].

L'entropia creuada, *Cross-Entropy* [58], és la funció de pèrdua predeterminada per als problemes de classificació binària (valors dins del conjunt 0, 1) i calcula una puntuació que resumeix la diferència mitjana entre les distribucions de probabilitat real i predita, on el valor ideal és 1.

Com que el discriminador d'una GAN realitza una tasca de classificació binària, utilitza l'entropia creuada com a funció de pèrdua.

### 4.3.9 Adam Optimizer

En un model de *deep learning*, un algorisme d'optimització troba el valor dels paràmetres (pesos o *weights*) que minimitzen l'error en mapejar els inputs als outputs. Aquests optimitzadors afecten tant a la precisió com a la velocitat del model [59].

L'algorisme d'optimització *Adam* [60] és una extensió del descens del gradient estocàstic que recentment s'ha popularitzat en aplicacions de *deep learning*, *computer vision* i *natural language processing*.

Adam va ser presentat al 2015 per Diederik Kingma i Jimmy Bae al seu paper titulat "*Adam: A Method for Stochastic Optimization*" [61]. Els investigadors destaquen els següents avantatges: el mètode és senzill d'implementar, és computacionalment eficient, té pocs requisits de memòria i és molt adequat per a problemes que són grans en termes de dades i/o paràmetres.

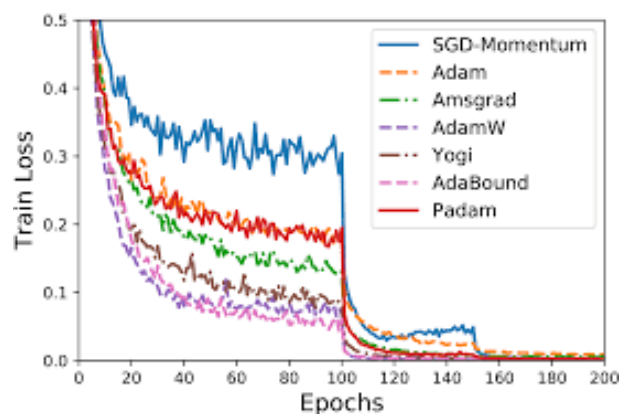


Figura 16: *Train loss* d'Adam vs. altres algorismes d'optimització amb el dataset CIFAR-10 i una ResNET [62]

## 4.4 Generative Adversarial Networks

Les xarxes generatives antagòniques (*Generative Adversarial Networks* [2], o GANs per abreviar) van ser presentades per Ian Goodfellow al 2014.

D'acord amb la seva definició, les GANs són un tipus d'algorisme d'intel·ligència artificial que, amb mètodes de *deep learning*, estan dissenyades per resoldre el problema de la modelització generativa. Aquest té com a finalitat estudiar una col·lecció d'exemples, *training set*, i aprendre'n la probabilitat de distribució que els genera. D'aquesta forma, les GANs són capaces de generar més exemples que es podrien haver extret del conjunt de dades original a partir de la distribució de probabilitat estimada.

L'arquitectura d'una GAN comprèn dos submodels: el **generador** i el **discriminador**. Els dos models són adversaris en un joc de suma zero (*zero-sum game* [63]). La competència en aquest joc impulsa els dos models a millorar els seus mètodes fins que les falsificacions no es distingeixen dels elements genuïns.

El generador intenta generar mostres falses, amb una distribució similar als exemples del conjunt de dades real i continua millorant la seva xarxa per enganyar el discriminador. Alhora, el discriminador intenta distingir entre les mostres reals i les mostres falses (generades).

La suma zero significa que quan el discriminador identifica amb èxit mostres reals i falses, se'l premia sense necessitat de fer cap canvi en els paràmetres del model, mentre que el generador és penalitzat actualitzant els paràmetres del model. Quan el generador enganya el discriminador, se'l recompensa i no cal canviar els paràmetres del model, però es penalitza el discriminador i s'actualitzen els seus.

A la Figura 17, observem el flux d'una GAN: introduïm soroll aleatori en el generador, que s'encarrega de generar mostres falses, mentre que les mostres reals provenen del dataset desitjat. Les mostres reals i les mostres falses (generades) van al discriminador, que ha de determinar l'origen de les mateixes (són reals o no?) realitzant una classificació binària. El generador "aprèn" del discriminador i, a cada iteració, les mostres falses se semblen cada cop més a les reals.

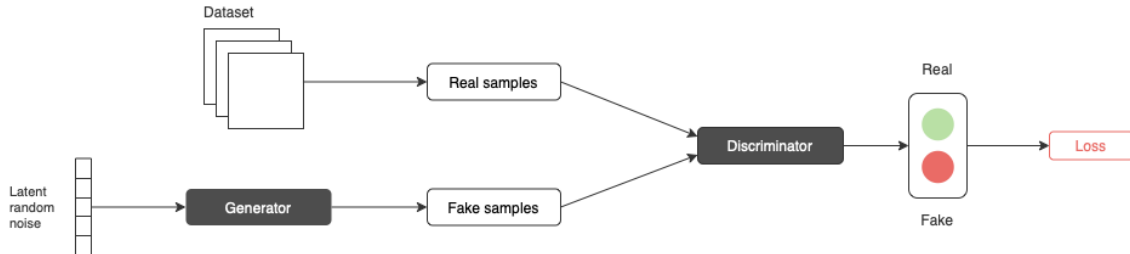


Figura 17: Diagrama de l'arquitectura d'una GAN

Per conèixer la distribució del generador  $p_g$  sobre les dades  $x$ , definim el soroll d'entrada  $p_z(z)$  i representem una correspondència a l'espai de dades com  $G(z; \theta_g)$ , on  $G$  és una funció representada per un perceptró multicapa amb paràmetres  $\theta_g$ . D'altra banda, definim un segon perceptró multicapa  $D(x; \theta_d)$  que produeix un únic escalar.  $D(x)$  representa la probabilitat que  $x$  provingui de les dades en lloc de  $p_g$ . Entrenem  $D$  per maximitzar la probabilitat d'assignar l'etiqueta correcta tant a exemples d'entrenament com a mostres de  $G$ . Simultàniament, entrenem  $G$  per minimitzar  $\log(1 - D(G(z)))$ .

En altres paraules,  $D$  i  $G$  juguen al següent joc *minimax* [64] per a dos jugadors amb la funció de valor  $V(G, D)$ :

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

Aquest joc *minimax* té un òptim global per a  $p_g = p_{data}$  [2].

El generador  $G$  defineix implícitament una distribució de probabilitat  $p_g$  com la distribució de les mostres  $G(z)$  obtingudes quan  $z \sim p_z$ . L'algorisme d'entrenament, que ha de convergir a un bon estimador de  $p_{data}$  amb proua capacitat i temps d'entrenament, es mostra a la Figura 18.

---

**Algorithm 1** Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator,  $k$ , is a hyperparameter. We used  $k = 1$ , the least expensive option, in our experiments.

---

**for** number of training iterations **do**

**for**  $k$  steps **do**

- Sample minibatch of  $m$  noise samples  $\{z^{(1)}, \dots, z^{(m)}\}$  from noise prior  $p_g(z)$ .
- Sample minibatch of  $m$  examples  $\{x^{(1)}, \dots, x^{(m)}\}$  from data generating distribution  $p_{data}(x)$ .
- Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^m \left[ \log D(x^{(i)}) + \log(1 - D(G(z^{(i)}))) \right].$$

**end for**

- Sample minibatch of  $m$  noise samples  $\{z^{(1)}, \dots, z^{(m)}\}$  from noise prior  $p_g(z)$ .
- Update the generator by descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^m \log(1 - D(G(z^{(i)}))).$$

**end for**

The gradient-based updates can use any standard gradient-based learning rule. We used momentum in our experiments.

---

Figura 18: Algorisme d'entrenament d'una GAN [2]

#### 4.4.1 Extensions i aplicacions de les GANs

Des de l'aparició de les GANs, el seu èxit ha desembocat en moltíssims models avançats i extensions que en deriven. Podem trobar centenars d'articles on s'exposen aquestes variacions, la seva aplicació i els seus avantatges. Veiem-ne algunes.

##### 4.4.1.1 Conditional Generative Adversarial Network

La xarxa generativa antagònica condicional (*Conditional Generative Adversarial Network* [65], o cGAN per abreujar) és una extensió d'una GAN que fa ús d'informació addicional com a *input*. D'aquesta forma, a més d'un set d'imatges, podem tenir també les etiquetes de les classes a les que pertanyen com a entrada.

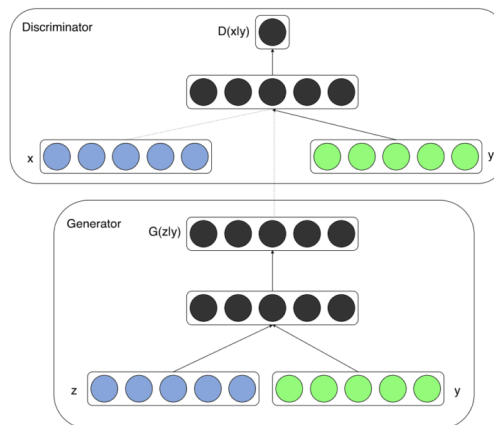


Figura 19: Model de l'arquitectura d'una cGAN [65]

##### 4.4.1.2 Pix2Pix

El model *Pix2Pix* [66] és una extensió de la cGAN que té el propòsit de generar imatges condicionades a imatges, que es coneix com a *Image-to-Image Translation* (Figura 20). Pel model generador, s'utilitza una arquitectura de model *U-Net*; pel model discriminador, una arquitectura de model *PatchGAN*, un classificador



convolucional [67].

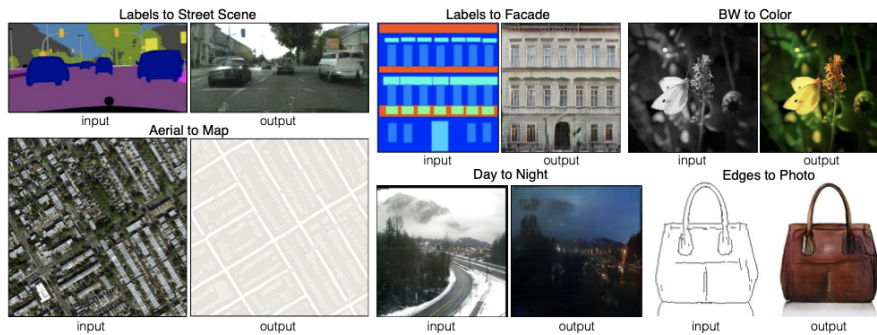


Figura 20: *Image-to-Image Translation* amb una cGAN [66]

#### 4.4.1.3 Cycle-Consistent Generative Adversarial Network

Una *Cycle-Consistent Generative Adversarial Network* [68], o CycleGAN per abreviar, és una extensió de les GANs per fer *Image-to-Image Translation* sense dades d'imatge aparellades (Figura 21). Això vol dir que no es requereixen exemples de la imatge d'interès com a les GANs condicionals com Pix2Pix [67].

Aquest model explota la propietat que la traducció ha de ser *cycle-consistent* (‘coherent al cicle’): per exemple, si es tradueix una frase de l'anglès al francès, i després es tradueix de nou del francès a l'anglès, s'hauria d'arribar a la frase original.

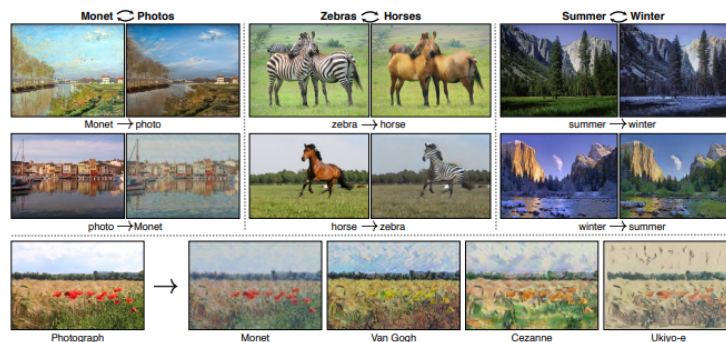


Figura 21: Exemples d'*Image-to-Image Translation* amb una CycleGAN

Les CycleGANs disposen de dos models generadors: un per a la translació X

a  $Y$  i un altre per reconstruir  $X$  donat  $Y$ . També tenen dos discriminadors,  $D_X$  i  $D_Y$ , on  $D_X$  pretén distingir entre imatges  $x$  i imatges traduïdes  $F(y)$ ; de la mateixa manera,  $D_Y$  pretén discriminar entre  $y$  i  $G(x)$ .

#### 4.4.1.4 Style-Based Generative Adversarial Network

Una *Style-Based Generative Adversarial Network* [69], o StyleGAN per abreujar, és una extensió d'una GAN que proposa grans canvis al model generador, com l'ús d'una xarxa de mapes per mapejar punts de l'espai latent a un espai latent intermedi, l'ús de l'espai latent intermedi per controlar l'estil en cada punt del model del generador, i la introducció del soroll com a font de variació en cada punt del model del generador [70].

El model és capaç de generar fotos realistes d'alta qualitat de rostres humans (Figura 22) i ofereix control sobre l'estil de la imatge generada mitjançant la variació dels vectors d'estil i soroll.



Figura 22: Imatges generades amb una GAN des de <https://thispersondoesnotexist.com>

#### 4.4.1.5 Super Resolution Generative Adversarial Network

Una *Super Resolution Generative Adversarial Network* [71], o SRGAN per abreujar,

conté, com una GAN, un generador i un discriminador. El generador (Figura 23), però, utilitza una *residual network* [72] en lloc d'una xarxa neuronal convolucio-

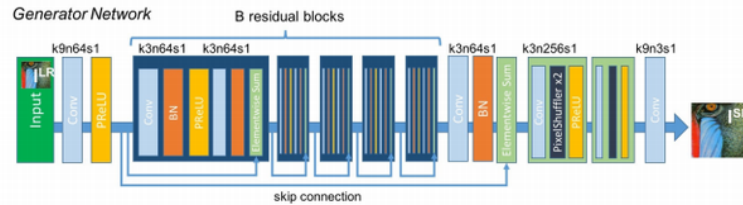


Figura 23: Arquitectura del generador d'una SRGAN [73]

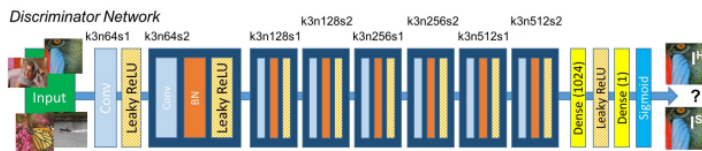


Figura 24: Arquitectura del discriminador d'una SRGAN [73]

Durant l'entrenament, una imatge d'alta resolució (HR) es redueix a una imatge de baixa resolució (LR). El generador intenta augmentar la imatge de baixa resolució a *super-resolution*. Llavors, la imatge passa pel discriminador (Figura 24) i aquest intenta distingir entre les imatges de super-resolution i les d'alta resolució. La pèrdua, *loss*, es retropropaga a l'arquitectura del generador (Figura 25) [73].

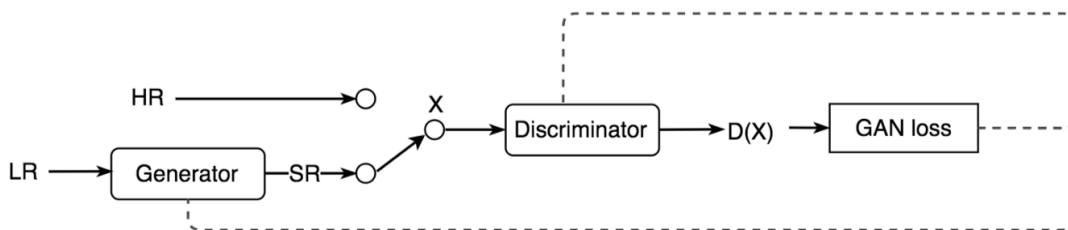


Figura 25: Arquitectura d'una SRGAN [74]

La possibilitat de millorar la qualitat d'una imatge ens permet implementar

les GANs dins de l'àmbit mèdic per tenir imatges realistes en *super-resolution* (Figura 26).

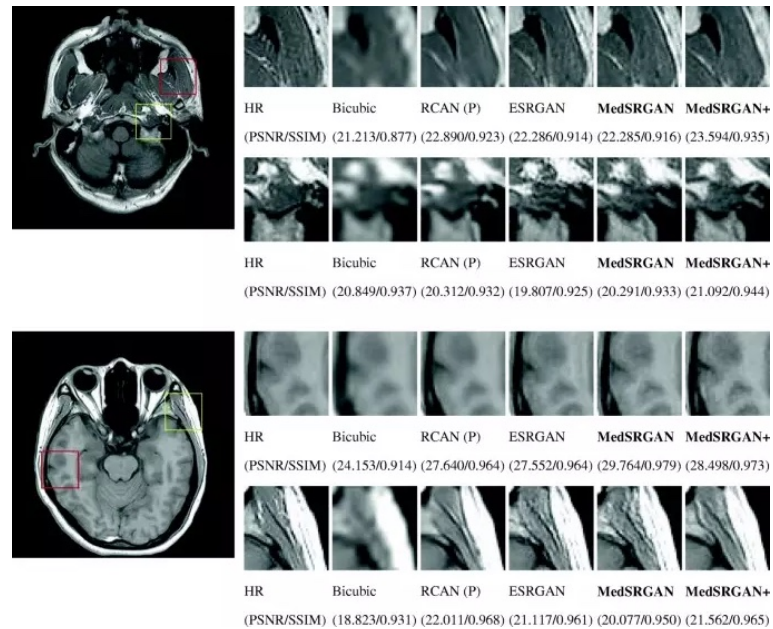


Figura 26: Aplicació mèdica d'una SRGAN. Comparativa de diverses variacions de l'algorisme [75]

Les imatges d'alta resolució poden ser difícils d'obtenir en certs protocols mèdics i una SRGAN eliminar el soroll i millorar considerablement la qualitat de les imatges obtingudes. Tanmateix, la introducció de les GANs dins de l'àmbit mèdic és bastant lenta, ja que s'han de realitzar molts experiments i assaigs per qüestions de seguretat. Cal assegurar que l'eliminació del soroll no distorsiona el contingut real de les proves d'alguna manera que pugui conduir a un diagnòstic incorrecte [76].

#### 4.4.1.6 Stacked Generative Adversarial Networks

Una *Stacked Generative Adversarial Networks* [77], o StackGAN per abreviar, es compon de dues GANs que s'apilen per a formar una xarxa capaç de generar imatges d'alta resolució a partir d'una descripció de text (Figura 27). Té dues etapes: l'*Stage-I* i l'*Stage-II*. La GAN de l'*Stage-I* dibuixa la forma primitiva i

els colors de l'objecte basant-se en la descripció del text donat, retornant imatges de baixa resolució. La GAN de l'*Stage-II* pren els resultats de l'*Stage-I* i les descripcions de text com a entrada i genera imatges d'alta resolució amb detalls fotorealistes, sent capaç de rectificar els defectes dels resultats de l'*Stage-I* i afegir altres detalls per tal de perfeccionar la imatge.



Figura 27: Text-to-Image Translation amb una StackGAN [77]

## 5 Implementació d'una GAN

### 5.1 Codi base

El procediment seguit en la implementació de la nostra GAN, que es pot trobar correctament comentat al *notebook* adjunt a la memòria, es basa sobretot en l'article *How to Develop a GAN to Generate CIFAR10 Small Color Photographs* [78], publicat per Jason Brownlee a la web *Machine Learning Mastery*, que, a diferència d'altres algorismes, permet generar imatges en color.

Originalment, Brownlee testeja la GAN amb el set de dades CIFAR-10 Small Object Photograph Dataset, que consta de 60,000 fotografies d'objectes de 10 classes (ocells, gats, vaixells, avions, granotes, etc.) en color de  $32 \times 32$  píxels.

No obstant això, la nostra implementació estarà optimitzada per un set de dades diferent: l'obra pictòrica de l'artista Georgia O'Keeffe, imatges RGB de  $128 \times 128$  píxels.

### 5.2 Entorn de programació

És ben sabut que treballar en projectes de *machine learning* és computacionalment molt costós. Per això, la implementació exposada a continuació no es podrà executar en el nostre ordinador com ho faríem amb qualsevol *Jupyter Notebook*.

Escriurem i executarem el nostre codi a *Google Colab* (<https://colab.research.google.com/>), un producte de *Google Research* que permet a qualsevol usuari escriure i executar codi *Python* des del navegador que s'utilitza sobretot per tasques d'aprenentatge automàtic, anàlisi de dades i educació. Tots els quaderns de *Colab* s'emmagatzemen a *Google Drive*.

La versió gratuïta de *Colab* permet l'ús de TPUs i GPUs, però aquests recursos

no estan garantits i els límits d'ús poden variar. El codi s'executa en una màquina virtual que s'elimina quan passa cert temps inactiva o quan s'esgota el seu cicle de vida màxim (12h aproximadament).

Per tal de tenir un accés més fiable als recursos, hem adquirit la subscripció mensual de Colab Pro, que ens dona accés prioritari a GPU més ràpides. Amb Colab Pro és possible tenir accés a una GPU T4 o P100, mentre que la majoria dels usuaris de la versió gratuïta només poden utilitzar una GPU K80, que és més lenta. Així, podrem garantir millors resultats.

### 5.3 Set de dades

Per tal d'entrenar la GAN, cal comptar amb un conjunt d'imatges o set de dades. Per a la nostra GAN, són en format RGB i de mida 128 x 128. El nombre d'imatges òptim és elevat, d'unes 60.000 imatges aproximadament. Tanmateix, com que els nostres recursos són limitats, haurem de reduir considerablement aquesta xifra, ja que amb 24.000 imatges aproximadament només aconseguim executar un parell d'epochs i amb 2.416, 17 epochs. Hem de limitar també les dimensions de les imatges a 128 x 128, ja que només amb dimensions de 256 x 256 l'execució s'atura immediatament.

Obtenim els resultats que es mostren a la secció 6 Resultats, utilitzant les següents primitives: 46 obres de l'artista Georgia O'Keeffe (Figura 28). Totes les imatges del set de dades original han estat extretes de WikiArt [79].

Per tal d'ampliar el set de dades, realitzem rotacions de 90°, 180° i 270° a les imatges i les anem retallant amb una finestra de mida 100. Així, el training set passa de tenir 46 imatges a 2.416, una mida computacionalment assumible amb l'ús de Colab.

Si comptéssim amb GPUs més avançades i temps d'execució més llargs, l'ús d'un set de dades més nombrós i amb imatges de majors dimensions i qualitat

seria possible i els resultats generats per la GAN molt més gratificants.

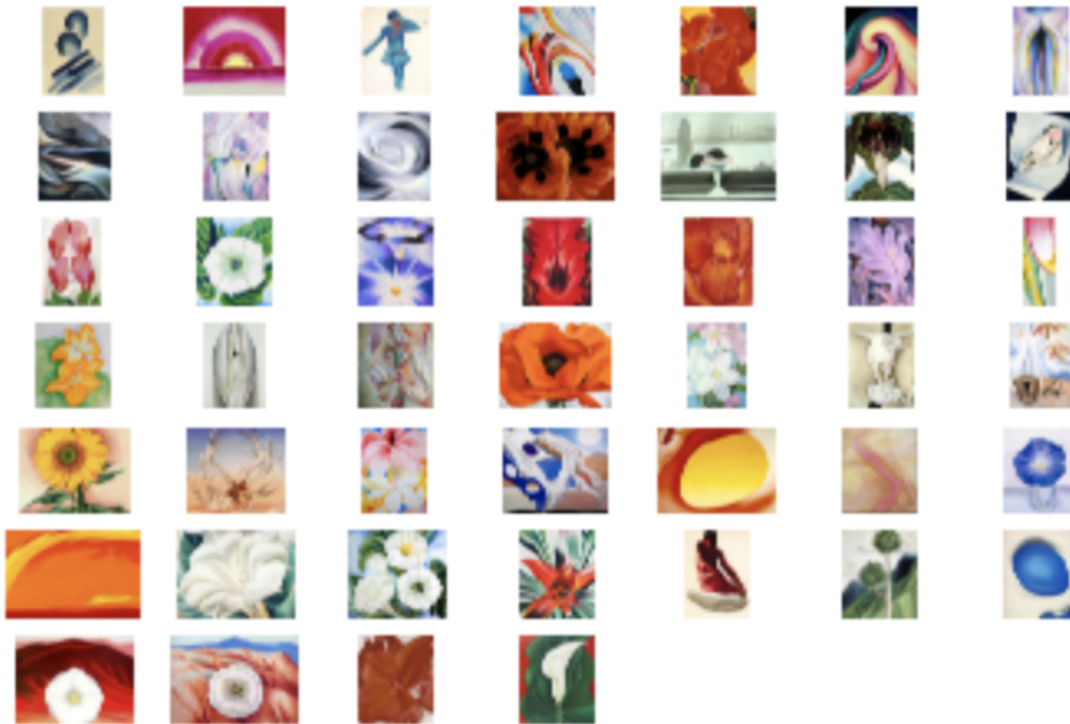


Figura 28: 46 obres de l'artista Georgia O'Keeffe.

## 5.4 Discriminador

El model discriminador pren com a input una imatge del dataset i retorna com a output si la mostra és real o falsa (*binary classification* [80]).

Definim el discriminador a partir d'un model seqüencial, al qual hi afegim, ordenadament, les següents capes: primer, Conv2D [81] per aplicar una convolució de 64 filtres a la imatge i LeakyReLU [82]; després, capes Conv2D i LeakyReLU per fer el *downsampling* [83] de  $128 \times 128$  fins a  $8 \times 8$ ; a continuació, Flatten [84] i Dropout [85]; i, finalment, una capa Dense [86] amb un únic node i la funció d'activació *sigmoid* [87] per predir si la imatge és real o falsa.

El model s'entrena per minimitzar la funció de pèrdua *Binary Cross-Entropy*



*Loss Function*, apropiada per la classificació binària, i es compila amb l'optimitzador *Adam* [61] amb una *learning rate* de 0.0002 i un *momentum* de 0.5.

L'estructura del discriminador pot veure's a l'**Annex A**. *Estructura del discriminador i les seves CNNs*.

## 5.5 Generador

El generador realitza la tasca de crear noves imatges, falses però plausibles. Ho fa prenent un punt de l'espai latent (un vector de 100 nombres gaussians aleatoris) com a input i retornant una imatge quadrada en color com a output.

Definim el generador seguint la mateixa estructura que el discriminador, creant un model seqüencial i afegint les capes necessàries: primer, una capa `Dense` amb suficients nodes per representar diverses versions de baixa resolució de la imatge de l'output, una capa `LeakyReLU` per activar els nodes i una capa `Reshape` [88] per redimensionar la imatge a  $8 \times 8$ ; després, capes `Conv2DTranspose` [89] i `LeakyReLU` per anar fent l'*upsampling* [83] o *deconvolution* [90] de la imatge generada des d'una dimensió  $8 \times 8$  fins a  $128 \times 128$ , tamany desitjat dels samples. La capa de sortida del model és una `Conv2D` amb tres filtres per als tres canals necessaris i una mida del *kernel* de  $7 \times 7$  per crear un únic *feature map* de  $128 \times 128 \times 3$  i una funció d'activació *tanh* [91] per garantir que els valors de sortida estiguin dins l'interval desitjat de  $[-1,1]$ .

En el nostre cas, el desenvolupament d'un model generador requereix la transformació del vector l'espai latent amb 100 dimensions a una matriu 2D amb  $128 \times 128 \times 3$ .

L'estructura del generador pot veure's a l'**Annex B**. *Estructura del generador i les seves CNNs*.

## 5.6 Entrenament de la GAN

Definim la nostra GAN com a l'*stack* del generador i el discriminador, de forma que el generador rep com a input punts aleatoris a l'espai latent i genera mostres falses que el discriminador ha de classificar. Els pesos del model generador s'actualitzen d'acord amb el rendiment del discriminador. És a dir, quan el discriminador detecta correctament quines són les mostres falses, el generador s'actualitza més; però quan el discriminador es confon, el generador s'actualitza menys.

El model de la GAN utilitza la mateixa funció de pèrdua que el discriminador, *Binary Cross-Entropy Loss Function*, i l'optimitzador *Adam*.

Per tal d'entrenar la nostra GAN, actualitzem a cada *epoch* [92] els pesos del discriminador amb imatges reals i imatges falses o generades. A més a més, actualitzem els pesos del generador a partir del model de la GAN.

El nombre de *batches* [93] dins de cada *epoch* és igual a la mida del set de dades. El discriminador s'actualitza dues vegades a cada *batch*, una amb imatges reals i una amb imatges generades; el generador només una.

## 6 Resultats

Per tal d'analitzar els resultats obtinguts a les diferents execucions, realitzem una comparació visual entre les imatges obtingudes modificant el set de dades i alguns dels paràmetres dels models.

A més a més, elaborem dues enquestes amb *Google Forms* que enviem a participants diferents per valorar els resultats de forma subjectiva i veure quins biaixos existeixen.

### 6.1 Execucions

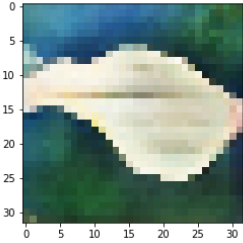
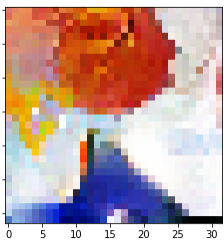
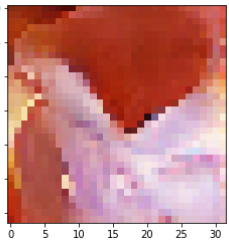
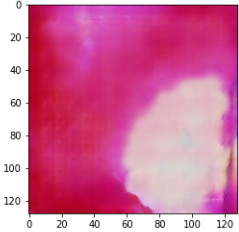
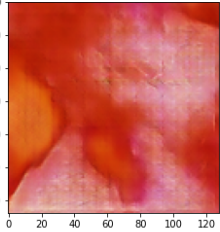
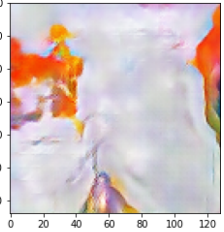
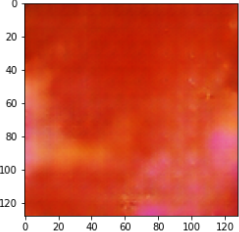
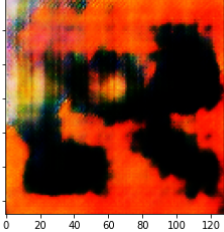
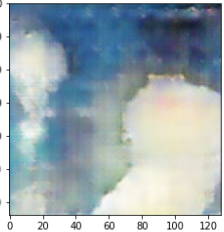
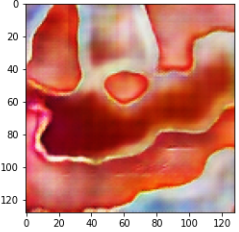
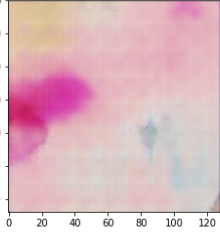
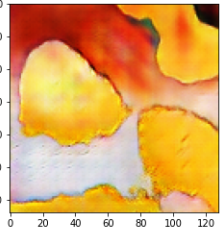
A la següent taula (Taula 1), es mostren algunes de les execucions realitzades amb la mida del set de dades i les dimensions de les imatges utilitzades, així com els *epochs* que s'han pogut acabar d'executar fins a obtenir-ne els resultats i els nombre de *batches* emprat.

Llistat d'execucions				
Execució	Mida del set de dades	Mida de les imatges	Epochs	Batches
[0]	6.476	32 x 32	45	128
[1]	2.416	128 x 128	16	128
[2]	1.148	128 x 128	23	64
[3]	4.976	128 x 128	4	256

Taula 1: Execucions de la GAN variant la mida del set de dades i de les imatges i el nombre d'*epochs* i de *batches*

A continuació, veiem algunes de les imatges associades a cadascuna de les execucions llistades anteriorment (Taula 2).

A primera vista, els resultats més plausibles són els de l'Execució [1], amb una mida del set de dades original de 2.416 imatges de 128 x 128, 16 *epochs* i 128 *batches*.

Resultats de les execucions			
Execució	Exemple 1	Exemple 2	Exemple 3
[0]			
[1]			
[2]			
[3]			

Taula 2: Imatges associades a les execucions mostrades a la Taula 1

## 6.2 Enquestes

Per tal d'avaluar millor els resultats, fem dues enquestes utilitzant la tècnica d'A/B testing:

- **A:** *Valoració de l'obra anònima inspirada en la de la Georgia O'Keeffe, on no es revela com s'han generat realment les obres.*
- **B:** *Art generatiu i creativitat computacional, on se sap que l'obra ha estat generada amb l'ús d'una GAN.*

Com és d'esperar, un 62,8% de les persones que saben que les obres han estat generades amb un ordinador veuen similituds entre l'obra original i la generada, mentre que els que no ho saben són un 87,3%. Això fa palesos els biaixos i prejudicis que existeixen amb l'art generat per una IA.

### 6.2.1 Resultats de l'enquesta A

En aquesta primera enquesta, que han respost un total de 55 persones, es mostren diverses imatges sota la premissa que han estat creades per un artista contemporani anònim i es basen en l'obra de la reconeguda artista Georgia O'Keeffe.

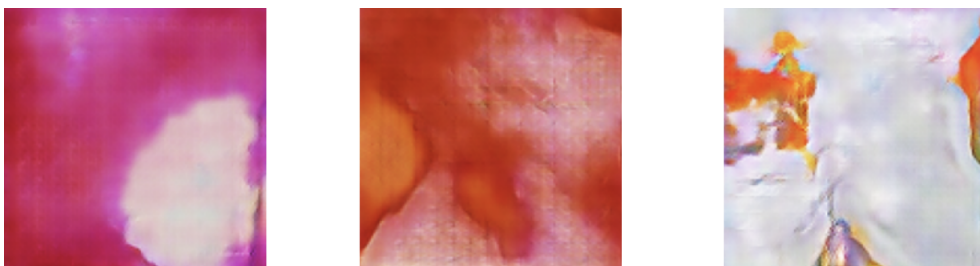


Figura 29: Tres exemples de l'obra generada per la nostra GAN i entrenada amb la de l'artista Georgia O'Keeffe

A partir d'aquestes tres primeres imatges (Figura 29), es fan diverses preguntes i s'obtenen les següents respostes:

- *Veus similituds entre l'obra de l'artista anònim i la de Georgia O'Keeffe?*

- *Respostes:* Un 87,3% dels enquestats veuen similituds entre l'obra original i la de l'artista contemporani (Figura 35).

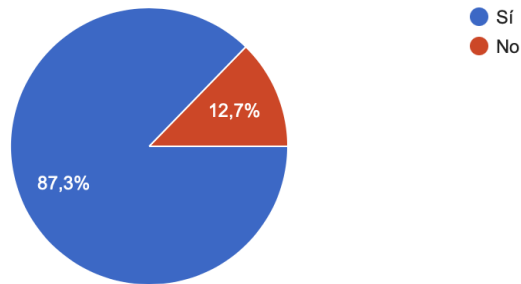


Figura 30: Percentatge d'enquestats que veuen o no veuen similituds entre l'obra original i la que s'hi basa

- *En cas afirmatiu, quines similituds trobes?*

- *Respostes:* Un 52,7% veu similituds només cromàtiques; un 34,5% cromàtiques i conceptuals; un 12,7% cap (Figura 36).

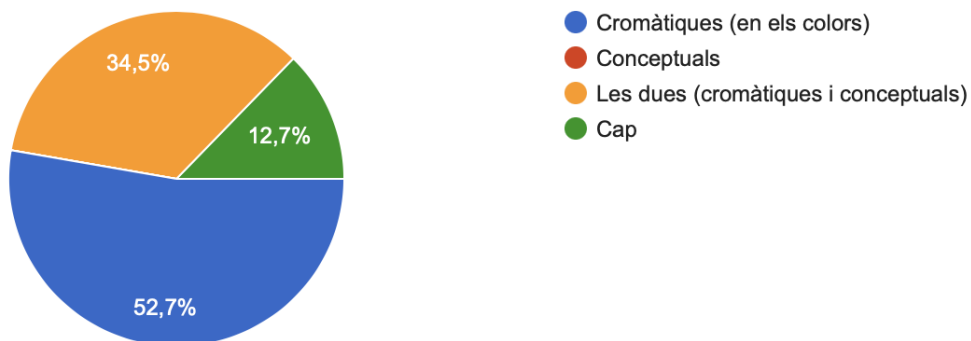


Figura 31: Percentatge de resposta segons la naturalesa de les similituds entre l'obra original i la que s'hi basa

- Quina tècnica/medi creus que ha utilitzat l'artista anònim per crear les obres inspirades?

– Respostes: Un 38,2% dels enquestats creuen que l'obra ha estat realitzada amb aquarel·la; un 23,6% amb acrílic; i un 36,4% amb eines digitals (Figura 32).

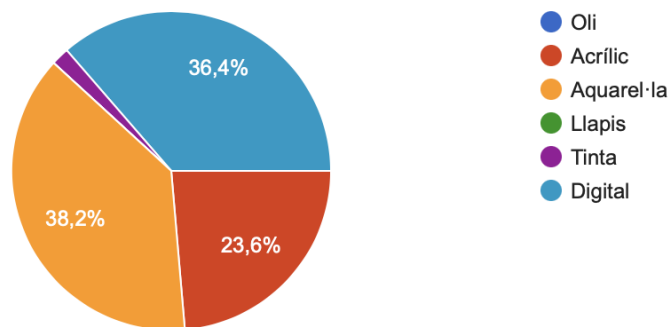


Figura 32: Percentatge de resposta segons la tècnica que es creu que ha utilitzat l'artista contemporani

A continuació, es mostren als enquestats les tres imatges d'abans i cinc imatges més, algunes amb resolució baixa però interessants visualment (Figura 33).

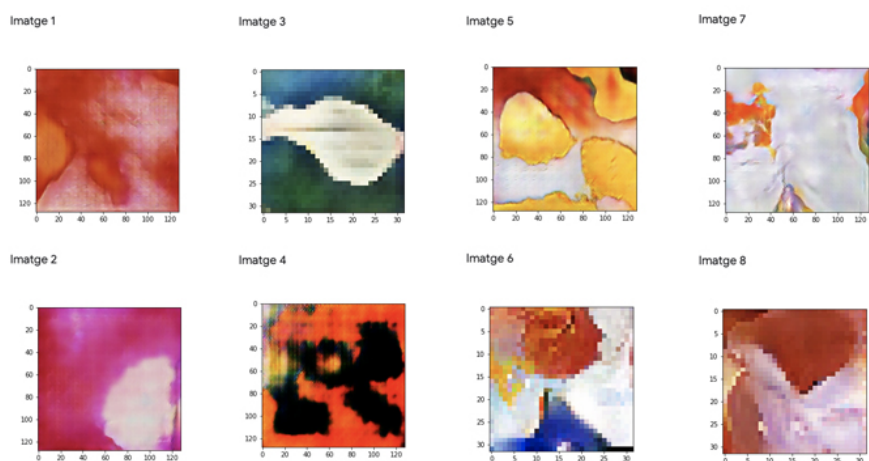


Figura 33: Imatges mostrades als enquestats, obra generada per la GAN amb diferents paràmetres i dimensions

Els preguntem quines imatges els recorden més a l'obra original i obtenim que la imatge més votada és la 2, amb un 83,6%; la segona, la 1, amb un 76,4%; i la tercera, la 7, amb un 54,5% (Figura 37).

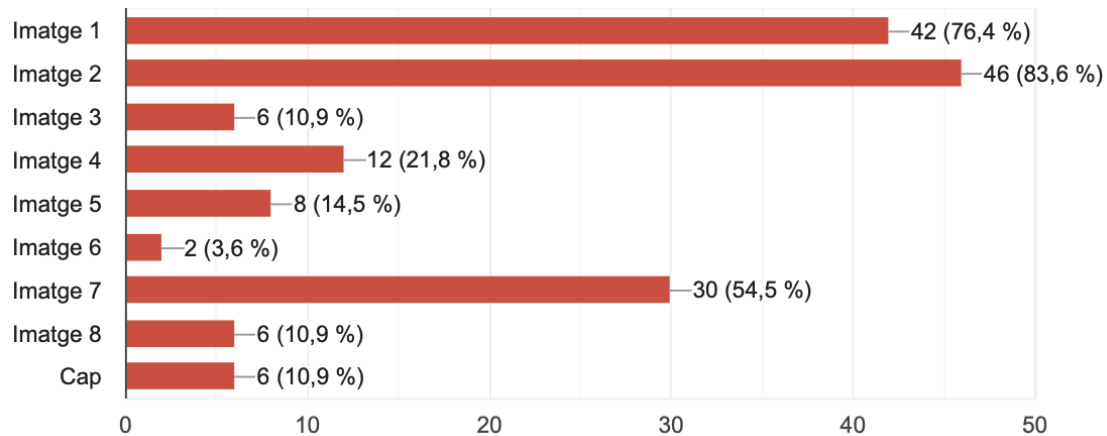


Figura 34: Percentatge de respostes segons la similitud de les vuit imatges inspirades i l'obra original

### 6.2.2 Resultats de l'enquesta B

A la segona enquesta, on han participat un total de 43 persones, diem la veritat als enquestats: l'obra mostrada ha estat generada amb l'ús d'una GAN entrenada amb l'obra de la Georgia O'Keeffe.

Mostrem les mateixes tres imatges (29), però ara obtenim resultats molt diferents quan preguntem sobre la similitud entre l'obra original i la generada.

- *Veus similituds entre l'obra generada i la de Georgia O'Keeffe?*
  - *Respostes:* Un 62,8% dels enquestats veuen similituds entre l'obra original i la generada (Figura 35).



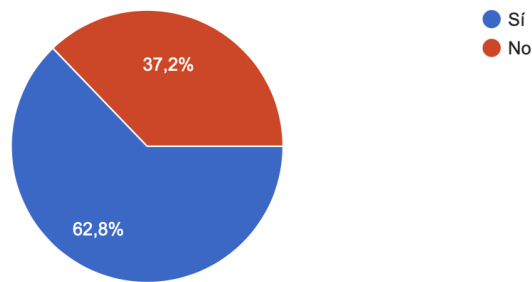


Figura 35: Percentatge d'enquestats que veuen o no veuen similituds entre l'obra original i la generada

- *En cas afirmatiu, quines similituds trobes?*

– *Respostes:* Un 39,5% veu similituds només cromàtiques; un 25,6% cromàtiques i conceptuals; un 32,6% cap (Figura 36).

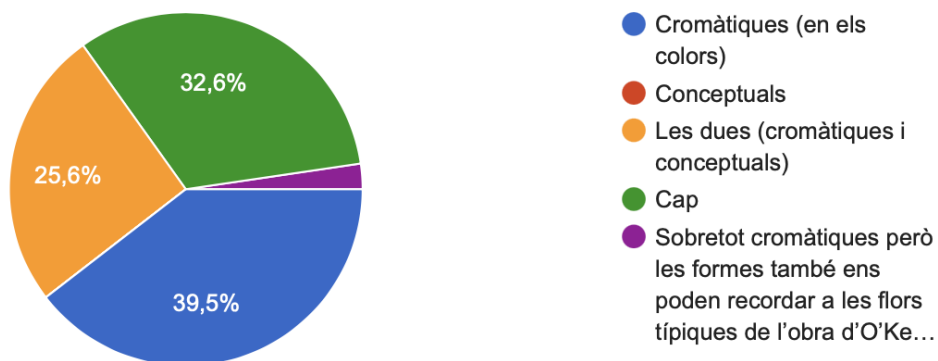


Figura 36: Percentatge de resposta segons la naturalesa de les similituds entre l'obra original i la que generada

Mostrem les vuit imatges seleccionades (Figura 33) als enquestats i obtenim els següents percentatges:

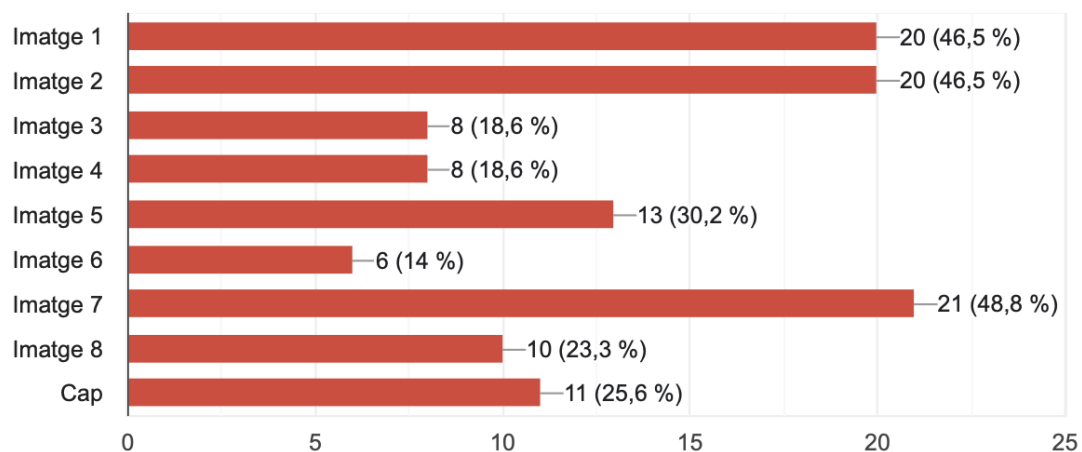


Figura 37: Percentatge de respostes segons la similitud de les vuit imatges generades i l'obra original

### 6.2.3 Anàlisi final

De les respostes a les dues enquestes, podem extreure les següents conclusions:

- L'obra generada és semblant a l'obra original, ja que en tots dos casos més del 50% dels enquestats troben similituds entre elles.
- Existeixen prejudicis quan se sap que l'obra ha estat generada per una IA, ja que, quan els enquestats no saben que ho ha estat, el percentatge de persones que troben semblants les obres creix un 24,5%.
- Les imatges que més se semblen a l'obra original de la Georgia O'Keeffe són les de l'Execució [1] de la Taula 2. Per tant, la mida del set d'imatges òptima per la nostra implementació i els nostres recursos computacionals ha estat de 2.416 imatges de 128 x 128, amb 45 *epochs* i 128 *batches*.

## 7 Treball futur

Fer el treball pràctic al voltant de les *Generative Adversarial Networks* no ha estat fàcil, sobretot per les limitacions computacionals a l'hora d'executar el codi i, consegüentment, pel poc marge que queda per fer variacions en els paràmetres entre execució i execució. Tanmateix, no deixa d'interessar-me el problema que pretenen resoldre: la modelització generativa. No puc evitar sorprendre'm per tots els límits que desafia i, malgrat que a mi també em genera certes contradiccions ètiques que un algorisme pugui simular la creativitat humana, vull que vagi més enllà i vull saber-ne més.

Per això, ara que al setembre cursaré el Màster d'Intel·ligència Artificial, no descarto enfocar-me a l'àrea de la creativitat computacional i experimentar amb GANs novament, però en molta més profunditat. Seria interessant, per què no, proposar un model que estengui la GAN a la generació d'altres tipus d'input i output, i explorar les alternatives a les GANs (per exemple, els VAEs) fent-ne una comparativa i continuant la recerca en el model que obtingui millors resultats a la tasca desitjada.

## 8 Conclusions

Malgrat els recursos computacionals limitats amb què es comptava i els obstacles amb què m'he pogut trobat, s'han aconseguit aconseguir tots els objectius que es van definir en iniciar aquest projecte.

En primer lloc, s'ha realitzat una breu introducció a l'art generatiu i a la creativitat computacional, exposant les seves controvèrsies i els seus interessos. Així mateix, s'han explicat tots els conceptes previs de *machine learning* necessaris per comprendre les *Generative Adversarial Networks*.

En segon lloc, s'ha desenvolupat la teoria relativa a les GANs, entenent el seu funcionament per la seva posterior implementació, explicant separadament el propòsit i model tant del discriminador com del generador que les conformen. A més a més, s'han donat exemples d'aplicacions i extensions de les GANs per aconseguir diversos propòsits.

A continuació, s'ha exposat el procediment seguit per la implementació de la GAN i s'han analitzat els resultats obtinguts amb comparacions entre execucions i enquestes A/B, que han fet palesos els prejudicis que existeixen envers l'IA i han demostrat que, en efecte, una GAN pot generar obres plausibles.

Finalment, es deixa oberta la possibilitat, en un futur, de la recerca en profunditat de la creativitat computacional i els diferents models generatius. Concloem, així doncs, que la introducció a aquests conceptes ha estat més que satisfactòria. La intel·ligència artificial, cada cop més propera a la nostra, cada cop més creativa (o només capaç de semblar-ho, pels escèptics), no deixarà mai de sorprendre'm.

Ara bé, quin és el propòsit, més enllà de desafiar els límits de la IA, de simular la creativitat humana? Si la creativitat és aplicada a avenços científics i s'utilitza amb fins beneficiosos per a la nostra salut o per a la humanitat (per exemple, si un algorisme aconsegueix trobar la cura a una malaltia o la solució a un proble-

ma matemàtic), per suposat haurà estat pròspera la investigació d'aquest camp. Tanmateix, amb la tècnica que s'està assolint també pot fer-se el mal: suplantar la identitat d'un artista o individu, generar i vendre art de forma massiva, o menystenir la màgia, només humana, que jeu darrere d'una obra original feta amb intencionalitat, consciència (o subconsciència) i emoció.

Tant de bo que en aquest camí es mantingui una ètica ferma i no s'oblidi mai el veritable origen de la creativitat: una ment que es pregunta i un cor que s'agita, com el nostre ara.

# Annexos

## A Estructura del discriminador i les seves CNNs

```
Model: "sequential"
```

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 128, 128, 128)	3584
leaky_re_lu (LeakyReLU)	(None, 128, 128, 128)	0
conv2d_1 (Conv2D)	(None, 64, 64, 128)	147584
leaky_re_lu_1 (LeakyReLU)	(None, 64, 64, 128)	0
conv2d_2 (Conv2D)	(None, 32, 32, 128)	147584
leaky_re_lu_2 (LeakyReLU)	(None, 32, 32, 128)	0
conv2d_3 (Conv2D)	(None, 16, 16, 128)	147584
leaky_re_lu_3 (LeakyReLU)	(None, 16, 16, 128)	0
conv2d_4 (Conv2D)	(None, 8, 8, 128)	147584
leaky_re_lu_4 (LeakyReLU)	(None, 8, 8, 128)	0
flatten (Flatten)	(None, 8192)	0
dropout (Dropout)	(None, 8192)	0
dense (Dense)	(None, 1)	8193

```
=====  
Total params: 602,113  
Trainable params: 602,113  
Non-trainable params: 0  
=====
```

Figura 38: *Summary* del model discriminador

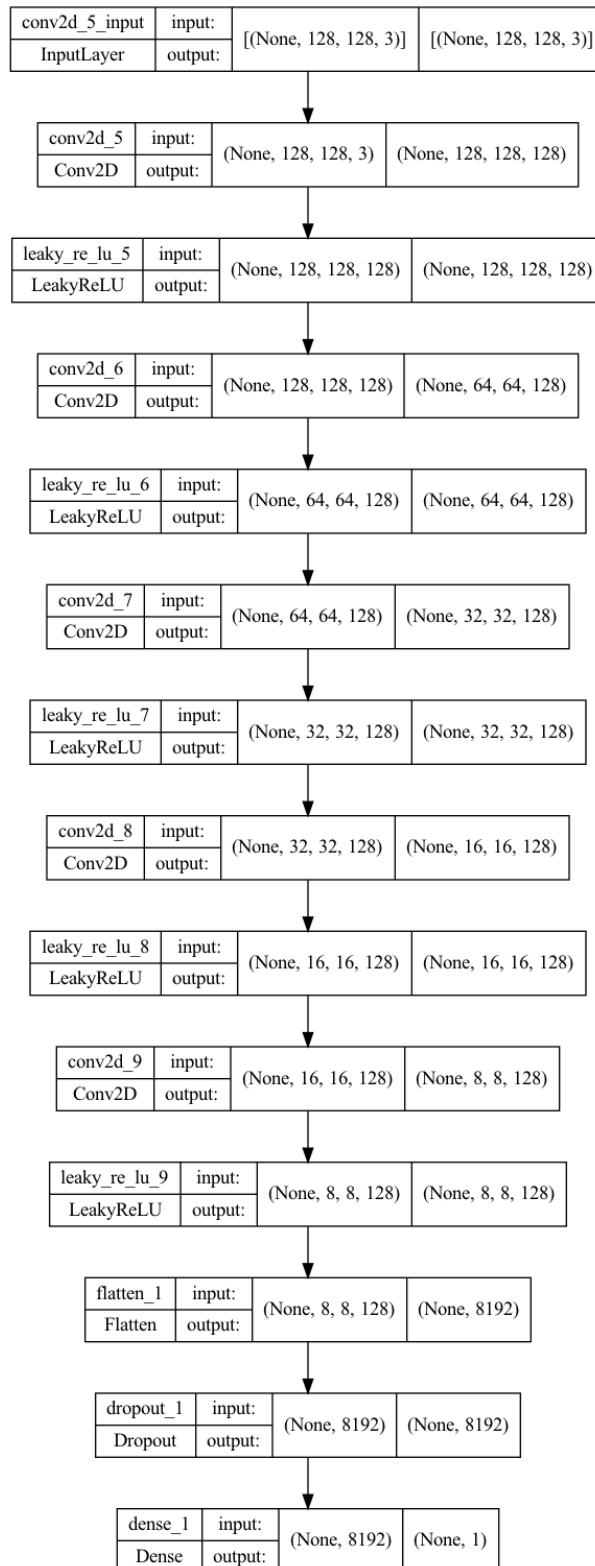


Figura 39: *Plot* del model discriminador

## B Estructura del generador i les seves CNNs

```

Model: "sequential_1"

```

Layer (type)	Output Shape	Param #
dense_1 (Dense)	(None, 16384)	1654784
leaky_re_lu_5 (LeakyReLU)	(None, 16384)	0
reshape (Reshape)	(None, 8, 8, 256)	0
conv2d_transpose (Conv2DTranspose)	(None, 16, 16, 128)	524416
leaky_re_lu_6 (LeakyReLU)	(None, 16, 16, 128)	0
conv2d_transpose_1 (Conv2DTranspose)	(None, 32, 32, 128)	262272
leaky_re_lu_7 (LeakyReLU)	(None, 32, 32, 128)	0
conv2d_transpose_2 (Conv2DTranspose)	(None, 64, 64, 128)	262272
leaky_re_lu_8 (LeakyReLU)	(None, 64, 64, 128)	0
conv2d_transpose_3 (Conv2DTranspose)	(None, 128, 128, 128)	262272
leaky_re_lu_9 (LeakyReLU)	(None, 128, 128, 128)	0
conv2d_5 (Conv2D)	(None, 128, 128, 3)	18819

```

=====
Total params: 2,984,835
Trainable params: 2,984,835
Non-trainable params: 0

```

Figura 40: *Summary* del model generador



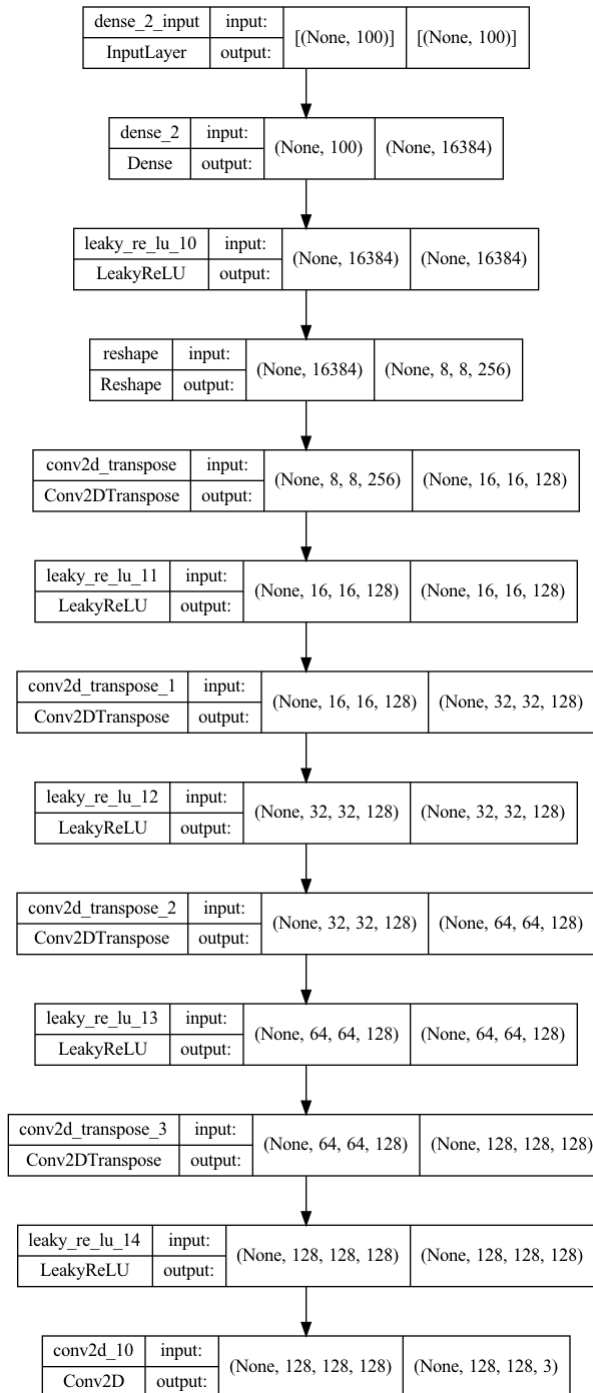


Figura 41: *Plot* del model generador

## Referències

- [1] *Aprenentatge profund*. Wikimedia Foundation, Jun 2022.
- [2] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. *Generative Adversarial Networks*. arXiv, 2014.
- [3] *Aprenentatge automàtic*. Wikimedia Foundation, Jun 2022.
- [4] *Association for Computational Creativity. About*.
- [5] Simon Colton, Ramon Mántaras, and Oliviero Stock. *Computational Creativity: Coming of Age*, volume 30. 09 2009.
- [6] *Creatividad computacional*. Wikimedia Foundation, May 2022.
- [7] Allen Newell, J. C. Shaw, and Herbert Alexander Simon. *The Processes of Creative Thinking*. RAND Corporation, Santa Monica, CA, 1959.
- [8] Margaret A. Boden. *The Creative Mind: Myths and Mechanisms*. Basic Books, Inc., USA, 1991.
- [9] Jey Han Lau, Trevor Cohn, Timothy Baldwin, Julian Brooke, and Adam Hammond. *Deep-speare: A joint neural model of poetic language, meter and rhyme*. Association for Computational Linguistics, Melbourne, Australia, July 2018.
- [10] Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. *Jukebox: A Generative Model for Music*. arXiv, 2020.
- [11] Sassbook. *AI writer, summarizer, story writer: Best ai writing tools*.

- [12] freeCodeCamp.org. *An introduction to generative art: What it is, and how you make it*. freeCodeCamp.org, Sep 2018.
- [13] Amy Goodchild. *What is generative art?* Amy Goodchild, Feb 2022.
- [14] Museum of Digital Art MuDA. *Vera Molnar: Randomness*. Jun 2022.
- [15] Paul Cohen. *Harold Cohen and Aaron*, volume 37. 2017.
- [16] Harold Cohen. *The Further Exploits of Aaron, Painter*, volume 4. Stanford Humanities Review, USA, jul 1995.
- [17] *Processing. Overview*.
- [18] Processing Foundation. *A modern prometheus*. Processing Foundation, Jun 2018.
- [19] Matt Pearson and Marius Watz. *Generative art: A practical guide using processing*. Manning, 2011.
- [20] *Artificial intelligence art*. Wikimedia Foundation, Jun 2022.
- [21] Jamshed J. Bharucha and Peter M. Todd. *Modeling the Perception of Tonal Structure with Neural Nets*, volume 13. The MIT Press, 1989.
- [22] Mario Klingemann. *Memories of Passersby I*. Dec 2018.
- [23] Art Dip. *Mario Klingemann Memories of Passersby I*. Medium, Jul 2019.
- [24] Teresa Sesé. *Sotheby's subasta Por 46.450 euros una obra creada por inteligencia artificial*. La Vanguardia, Mar 2019.
- [25] Obvious. *Edmond de Belamy*.
- [26] Obvious. *Obvious. About*.
- [27] Obvious. *La Famille de Belamy* Obviou.

- [28] Gabe Cohn. *Ai Art at Christie's sells for \$432,500*. The New York Times, Oct 2018.
- [29] *Supervised learning*. Wikimedia Foundation, Jun 2022.
- [30] *Classification*. Wikimedia Foundation, Apr 2022.
- [31] *Discriminative model*. Wikimedia Foundation, Apr 2022.
- [32] Thomas Wood. *Unsupervised learning*. Aug 2020.
- [33] *What is clustering? Clustering in machine learning*. Google developers. Google.
- [34] Background: What is a generative model? generative adversarial networks. google developers.
- [35] Joseph Rocca. *Understanding variational autoencoders (VAES)*. Towards Data Science, Mar 2021.
- [36] Thiel McClain. *An intuitive introduction to deep autoregressive networks*.
- [37] Pier Paolo Ippolito. *Game theory in artificial intelligence*. Towards Data Science, Sep 2019.
- [38] *Artificial Neural Network*. Wikimedia Foundation, Jun 2022.
- [39] By: IBM Cloud Education. *What are neural networks?*
- [40] DeepAI. *Weight (artificial neural network)*. DeepAI, May 2019.
- [41] *What is classification threshold*. Aug 2021.
- [42] DeepAI. *Perceptron*. DeepAI, May 2019.
- [43] Sumit Saha. *A comprehensive guide to Convolutional Neural Networks*. Towards Data Science, Dec 2018.
- [44] *Feature (machine learning)*. Wikimedia Foundation, Nov 2021.

- [45] Jason Brownlee. *How do convolutional layers work in Deep Learning Neural Networks?* Apr 2020.
- [46] Seb. *What is a kernel in machine learning?* Apr 2022.
- [47] Sandeep Balachandran. *Machine learning - convolution with color images*. DEV Community, Mar 2020.
- [48] *Produkte escalar*. Wikimedia Foundation, Oct 2021.
- [49] *Feature maps (activation maps)*.
- [50] DeepAI. *Relu*. DeepAI, May 2019.
- [51] Jason Brownlee. *A gentle introduction to the rectified linear unit (ReLU)*. Aug 2020.
- [52] *Latent space*. Wikimedia Foundation, Oct 2021.
- [53] Panagiotis Antoniadis. *Latent space in deep learning*. Mar 2022.
- [54] July 27 and Jonathan Johnson. *What's a deep neural network? Deep Nets explained*. Jul 2020.
- [55] *Algorithme del gradient descendent*. Wikimedia Foundation, Feb 2022.
- [56] Jason Brownlee. *Loss and loss functions for training Deep Learning Neural Networks*. Oct 2019.
- [57] Jason Brownlee. *How to choose loss functions when Training Deep Learning Neural Networks*. Aug 2020.
- [58] Kiprono Elijah Koech. *Cross-entropy loss function*. Towards Data Science, Nov 2021.
- [59] *A comprehensive guide on Deep learning optimizers*. May 2022.

- [60] Jason Brownlee. *Gentle introduction to the adam optimization algorithm for deep learning*. Jan 2021.
- [61] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. arXiv, 2014.
- [62] Jinghui Chen, Dongruo Zhou, Yiqi Tang, Ziyang Yang, Yuan Cao, and Quanquan Gu. *Closing the Generalization Gap of Adaptive Gradient Methods in Training Deep Neural Networks*. arXiv, 2018.
- [63] *Zero-sum game*. Wikimedia Foundation, Jun 2022.
- [64] *Minimax algorithm in Game theory: Set 1 (introduction)*. Jun 2022.
- [65] Mehdi Mirza and Simon Osindero. *Conditional Generative Adversarial Nets*. arXiv, 2014.
- [66] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. *Image-to-Image Translation with Conditional Adversarial Networks*. arXiv, 2016.
- [67] Jason Brownlee. *A tour of Generative Adversarial Network Models*. Jul 2019.
- [68] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. *Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks*. arXiv, 2017.
- [69] Tero Karras, Samuli Laine, and Timo Aila. *A Style-Based Generator Architecture for Generative Adversarial Networks*. arXiv, 2018.
- [70] Jason Brownlee. *A gentle introduction to stylegan the style generative Adversarial Network*. May 2020.
- [71] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes

Totz, Zehan Wang, and Wenzhe Shi. *Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network*. arXiv, 2016.

[72] Connor Shorten. *Introduction to resnets*. Towards Data Science, May 2019.

[73] *Super resolution gan (SRGAN)*. May 2022.

[74] Jonathan Hui. *GAN - Super Resolution GAN (SRGAN)*. Medium, Jul 2018.

[75] Zeng Z. Chen H. Gu, Y. *MedSRGAN: medical images super-resolution using generative adversarial networks*. 2020.

[76] Maksym Tatariants. *GAN technology: Use cases for business applications*. MobiDev, May 2022.

[77] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiao-olei Huang, and Dimitris Metaxas. *StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks*. arXiv, 2016.

[78] Jason Brownlee. *How to develop a GAN to generate CIFAR10 small color photographs*. Sep 2020.

[79] *Georgia O'keeffe - 237 Obras de Arte - Pintura*.

[80] *Binary classification*. Wikimedia Foundation, May 2022.

[81] Keras Team. *Keras documentation: Conv2d layer*.

[82] Keras Team. *Keras Documentation: Leakyrelu Layer*.

[83] *Handling imbalanced data- machine learning, Computer Vision, NLP*. Nov 2020.

[84] Keras Team. *Keras Documentation: Flatten Layer*.

[85] Keras Team. *Keras documentation: Dropout layer*.

[86] Keras Team. *Keras documentation: Dense layer*.

- [87] Sagar Sharma. *Activation functions in neural networks*. Towards Data Science, Jul 2021.
- [88] Keras Team. *Keras Documentation: Reshape layer*.
- [89] Keras Team. Keras documentation: Conv2dtranspose layer.
- [90] *Deconvolution*. Wikimedia Foundation, Apr 2022.
- [91] *Papers with code - tanh activation explained*.
- [92] Mayuresh. *Epoch in machine learning: A simple introduction*. Mar 2021.
- [93] Jason Brownlee. *Difference between a batch and an epoch in a neural network*. Oct 2019.