PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

ESCUELA DE POSGRADO

Generación de imágenes de acciones específicas de una persona utilizando aprendizaje profundo

Tesis para optar el grado académico de Maestro en Informática con mención en Ciencias de la Computación que presenta:

Jose Ulises Morales Pariona

Asesor:

Dr. César Armando Beltrán Castañón

Lima, 2024

Informe de Similitud

Yo, **César Armando Beltrán Castañón**, docente de la Escuela de Posgrado de la Pontificia Universidad Católica del Perú, asesor del trabajo de investigación titulado "**Generación de imágenes de acciones específicas de una persona utilizando aprendizaje profundo**", del autor **José Ulises Morales Pariona**, dejo constancia de lo siguiente:

- El mencionado documento tiene un índice de puntuación de similitud de **10%**. Así lo consigna el reporte de similitud emitido por el software Turnitin el **18/08/2023**.
- He revisado con detalle dicho reporte y la Tesis, y no se advierte indicios de plagio.
- Las citas a otros autores y sus respectivas referencias cumplen con las pautas académicas.

Lugar y fecha:

Lima, 18 de agosto de 2023.

<table>
<tr><td colspan="2">Apellidos y nombres del asesor:
Beltrán Castañón, César Armanado</td></tr>
<tr><td>DNI: 29561260</td><td rowspan="2">Firma  </td></tr>
<tr><td>ORCID: 0000-0003-4001-8072</td></tr>
</table>

Resumen

Desde que aparecieron las redes GAN, se han realizado varias investigaciones sobre cómo generar imágenes en diversos ámbitos, como la generación de imágenes, conversión de imágenes, síntesis de videos, síntesis de imágenes a partir de textos y predicción de cuadros de videos. Basándose mayormente en mejorar la generación de imágenes de alta resolución y la reconstrucción o predicción de datos.

El propósito de este trabajo es implementar las redes GAN en otros ámbitos, como la generación de imágenes de entidades realizando una acción. En este caso se consideró 3 acciones de personas, que son los ejercicios de Glúteo, Abdomen y Cardio. En primer lugar, se descargaron y procesaron las imágenes de YouTube, el cual incluye una secuencia de imágenes de cada acción. Posteriormente, se separó dos grupos de imágenes, de una sola persona, y de personas diferentes realizando las acciones. En segundo lugar, se seleccionó el modelo InfoGAN para la generación de imágenes, teniendo como evaluador de rendimiento, la Puntuación Inicial (PI). Obteniendo como resultados para el primer grupo, una puntuación máxima de 1.28 y en el segundo grupo, una puntuación máxima de 1.3.

En conclusión, aunque no se obtuvo el puntaje máximo de 3 para este evaluador de rendimiento, debido a la cantidad y calidad de las imágenes. Se aprecia, que el modelo si logra diferenciar los 3 tipos de ejercicios, aunque existen casos donde se muestran incorrectamente las piernas, los brazos y la cabeza.

Abstract

Since the appearance of GAN networks, various investigations have been carried out on how to generate images in various fields, such as image generation, image conversion, video synthesis, image synthesis from text, and video frame prediction. Based mostly on improving the generation of high resolution images and the reconstruction or prediction of data.

The purpose of this work is to implement GAN networks in other areas, such as the generation of images of entities performing an action. In this case, 3 actions of people were considered, which are the Gluteus, Abdomen and Cardio exercises. First, the images from YouTube were downloaded and processed, which includes a sequence of images of each action. Subsequently, two groups of images were separated, of a single person, and of different people performing the actions. Secondly, the InfoGAN model was selected for image generation, having the Initial Score (PI) as a performance evaluator. Obtaining as results for the first group, a maximum score of 1.28 and in the second group, a maximum score of 1.3.

In conclusion, although the maximum score of 3 was not obtained for this performance tester, due to the quantity and quality of the images. It can be seen that the model is able to differentiate the 3 types of exercises, although there are cases where the legs, arms and head are shown incorrectly.

Tabla de contenidos

Índice de imágenes

Índice de tablas

Introducción

Las Redes Generativas Adversariales (GAN), desde sus inicios han sido aplicadas con éxito en la generación de imágenes (Sage, Timofte, Agustsson, y Gool, 2018), la reconstrucción de imágenes (Chandak, Saxena, Pattanaik, y Kaushal, 2019a), la generación de videos (Clark, Donahue, y Simonyan, 2019), la detección de objetos (Prakash y Karam, 2021), la segmentación semántica (Chandak, Saxena, Pattanaik, y Kaushal, 2019b) y en el procesamiento del lenguaje natural (Subramanian, Rajeswar, Dutil, Pal, y Courville, 2017). De acuerdo con el estudio de las redes GAN y sus aplicaciones (Calcagni, 2020), aunque las redes GAN no fueron los primeros modelos en generar datos, son estudiados ampliamente en la actualidad, dado que, se ha conseguido resultados considerables desde su aparición. Manifestando que entre sus aplicaciones mas exitosas, se encuentran en el área de imágenes y visión por computadora, debido a que los modelos GAN se ajustan a la distribución de las muestras de datos originales. Es decir, pueden generar imágenes sintéticas realistas, independientemente del dominio, que representa al conjunto de datos de entrenamiento original.

Actualmente no existen investigaciones sobre generación imágenes de una entidad realizando una acción, debido a la dificultad de encontrar datos estructurados para realizar la experimentación, como a su vez, el tiempo y poder computacional que conlleva el entrenamiento y afinación de un modelo de aprendizaje automático. Desde la posición del estudio sobre la generación de personas con XingGAN (Tang, Bai, Zhang, Torr, y Sebe, 2020), la finalidad de generar imágenes de personas es poder generar estas imágenes condicionadas a una imagen de entrada y varias poses deseadas, debido a su uso en la formación de nuevas imágenes, videos e identificación de personas.

Como consecuencia, los estudios posteriores, se basan en mejorar la generación de imágenes de personas con la colección de datos existente. Por tal razón, no se amplía el alcance hacia otros estudios, como la generación de acciones de una persona.

La presente investigación propone generar imágenes de una persona realizando acciones específicas, que en este caso son ejercicios de cardio, glúteo y abdomen, mediante un modelo de aprendizaje profundo. Se seleccionó estos tipos de imágenes porque se necesitaba que estas 3 acciones no difieran mucho en contexto entre ellas, por la complejidad del entrenamiento del modelo de generación de imágenes. Es decir, no se buscó acciones como comer, dormir y nadar, porque no tienen relación entre sí. También se consideró la factibilidad para obtener las imágenes que, gracias a la pandemia, se obtuvo de muchos videos de YouTube de personas realizando ejercicio en casa. Se considero todo esto, porque actualmente no existe muchas bases de datos específicas para acciones de personas y si existe, no en grandes cantidades.

En primer lugar, se buscó videos de personas realizando estos ejercicios en YouTube, con palabras claves como, cardio, glúteo, abdomen en diferentes idiomas como el español, inglés, japones, portugués y ruso. Sabiendo que estos tipos de ejercicios se practican en muchos países del mundo. Así mismo se seleccionó videos en resolución HD, que se encuentra entre las resoluciones de SD y FHD, por mantener la calidad al escalar la imagen y porque la mayoría de videos se encontraban máximo en HD. En segundo lugar, se descargaron estos videos de 60 fotogramas con el programa aTube Catcher, para obtener imágenes por cada 30 fotogramas por segundo del video. En tercer lugar, se seleccionó las imágenes donde se mostraban las personas realizando las acciones en cuerpo completo, ya que estos ejercicios se aprecian y diferencia, si

se muestra a la persona completamente. En cuarto lugar, se procesaron estos datos, realizando una segmentación, recorte, orientación y escalamiento de imágenes. Acotar que se formó dos grupos de imágenes, uno con imágenes de una sola persona realizando cada acción, para formar un modelo base y otro grupo de personas diferentes realizando estas acciones.

Fundamentalmente indicar, que se seleccionó el modelo InfoGAN(Chen y cols., 2016), para entrenar el modelo de generación de imágenes, debido a que aprende características de cada clase (Gluteos, Cardio y Abdomen) de imagen sin supervisión. Se entreno el modelo en una instancia de Google Colaboratory, haciendo uso de sus herramientas gratuitas. Finalmente, se evaluó el rendimiento del modelo, haciendo uso de Puntuación Inicial (Barratt y Sharma, 2018), porque de acuerdo con la investigación sobre la Puntuación Inicial, este se correlaciona bien con el juicio humano sobre la calidad de imágenes generadas por los modelos generativos. Para esta investigación la puntuación máxima es de tres, determinado por la cantidad de clases de imágenes que genera el modelo. Es decir, si fueran cinco clases, la puntuación máxima seria de cinco.

Como resultado de esta investigación, la Puntuación Inicial máxima obtenida en el primer grupo, fue de 1.28 y un 1.3 para el segundo grupo. No se logró, estar cerca del puntaje máximo, debido a que algunas imágenes de entrenamiento tenían algo de ruido y falto diversidad de imágenes, de igual modo que sucedió en la investigación de generación de rostros (Z. Liu, Luo, Wang, y Tang, 2015) realizado por Liu. A pesar de los puntajes obtenidos, las imágenes generadas por modelo InfoGAN, muestra que el modelo si diferencia las clases de ejercicios que se utilizaron para esta investigación,

aunque las imágenes no siempre se muestren correctamente, ya que existen casos donde se muestran incorrectamente las piernas, los brazos y la cabeza.

En definitiva, esta investigación servirá de referencia para dar inicio a mejorar o realizar nuevas investigaciones sobre la generación de acciones de personas. Como mejora de esta investigación, se podría desarrollar un modelo de selección y segmentación de imágenes de personas realizando ejercicios y otro modelo para mejorar la calidad de imagen en secciones de ruido. Como nuevas investigaciones, se puede desarrollar otros modelos, como la CGAN o DCGAN. Además de generar videos, en base a las imágenes generadas y su modificación de espacio latente en combinación con redes recurrentes. Finalmente implementar otra métrica de rendimiento, como la distancia de Fréchet (FID).

Este documento consta de 5 capítulos: El capítulo 1 define el problema y el enfoque adoptado en este trabajo para darle solución. El capítulo 2 describe la disciplina de la generación de imágenes en las arquitecturas de Aprendizaje Profundo y los trabajos previos que busquen generar imágenes mediante Redes Generativas Adversariales (GAN). El capítulo 3 define las bases para el proceso de desarrollo y experimentación. El capítulo 4 describe el procedimiento desarrollado, así como los resultados obtenidos en la aplicación del mismo. Finalmente, el capítulo 5 trata sobre las conclusiones y recomendaciones obtenidas como producto de este trabajo.

Capítulo I

Generalidades

1.1. Problemática

Desde el 2014 que aparecieron las redes GAN (Goodfellow y cols., 2014), se han realizado varias investigaciones sobre cómo generar imágenes en diversos ámbitos, por ejemplo, en el ámbito social, está la generación de posturas de personas (Tang y cols., 2020) y rostros (Karras, Aila, Laine, y Lehtinen, 2018); en el ámbito científico y biológico, está la generación de datos biológicos utilizando redes generativas de confrontación (Dong y Zhang, 2020), paisajes (Sun, Fang, y Schwing, 2020), plantas (Kola, 2019) y animales (Sendik, Lischinski, y Cohen-Or, 2020); asimismo, en el ámbito médico, las tomografías (Han y cols., 2018). En estas investigaciones las redes GAN (Goodfellow y cols., 2014) se enfocan en procesos específicos, como la generación de imágenes para completar datos de entrenamiento de los modelos de aprendizaje automático (Celik, 2018), procesos de restauración de imágenes o en procesos de generación de fragmentos de videos (M. Y. Liu, Huang, Yu, Wang, y Mallya, 2021).

De acuerdo con el estudio de las redes GAN y sus aplicaciones (Calcagni, 2020) , Calcagni manifiesta que este se aplica en la generación de imágenes, síntesis de imágenes a partir de textos, conversión de imagen-imagen , generación de imágenes de alta resolución, síntesis de videos, predicción de cuadros de videos y mapeo de imágenes 3D a partir de cortes en 2D. Basándose mayormente en mejorar la generación de imágenes de alta resolución y la reconstrucción o predicción de datos, porque todavía se busca resolver o mejorar las principales dificultades de las redes GAN (Calcagni, 2020). La no convergencia, cuando no se logra un equilibrio entre el generador y discriminador. El colapso modal, que ocurre cuando se generan muestras similares, aunque las entradas tengan diversas características. Por último, la perdida informativa, aunque se infiere que cuanto menos sea la pérdida del generador, mayor será la calidad, este no siempre es así. En cambio, esta investigación no trata de mejorar los procesos existentes. Sino de seguir implementando las redes GAN en otros ámbitos, como la generación de imágenes de entidades realizando una acción.

El propósito de este trabajo de investigación es generar imágenes de personas realizando acciones específicas, que en este caso son ejercicios de abdomen, glúteos y cardio. Se seleccionó estos tipos de imágenes porque se necesitaba que estas 3 acciones no difieran mucho en contexto entre ellas, por la complejidad del entrenamiento del modelo de generación de imágenes. Es decir, no se buscó acciones como comer, dormir y nadar, porque no tienen relación entre sí. También se consideró la factibilidad para obtener las imágenes, que gracias a la pandemia, se obtuvo de muchos videos de YouTube (Bienestar, 2020) de personas realizando ejercicio en casa. Obteniendo las imágenes de YouTube, se definió y desarrollo un modelo GAN. A su vez, se validó la generación de imágenes con una métrica de rendimiento, que en este caso es, la

Puntuación Inicial (Barratt y Sharma, 2018). La investigación busca comenzar y dar una visión de lo que se puede hacer con estas imágenes de personas realizando una acción, sentando el comienzo a futuras investigaciones. Mejorar la selección y segmentación de imágenes de personas realizando una acción. También en el desarrollo de nuevos modelos utilizando la CGAN o DCGAN. Finalmente en la implementación de otra métrica de rendimiento, como la distancia de Fréchet (FID). En caso se desarrollen más trabajos en el área, se podría desarrollar un modelo de generación de imágenes que represente acciones de una persona a partir de textos.

1.2. Objetivos

1.2.1. Objetivo General

Generar imágenes de acciones específicas de una persona realizando ejercicios físicos utilizando aprendizaje profundo.

1.2.2. Objetivos Específicos

1. Definir un modelo base de aprendizaje automático y su respectivo conjunto de datos de entrenamiento.
2. Desarrollar un modelo de aprendizaje automático para generar imágenes de una persona realizando una acción específica utilizando la arquitectura de aprendizaje profundo.
3. Validar el modelo de aprendizaje automático desarrollado utilizando métricas que evalúen su rendimiento.

1.3. Alcance

Existen múltiples trabajos de generación de imágenes, especialmente de animales y plantas (Kola, 2019), radiografías (Han y cols., 2018) o personas (Günel, Erdem, y Erdem, 2018) pero con un enfoque especifico como, por ejemplo los rostros. En el presente estudio, tiene como objetivo, generar imágenes de una persona realizando una acción específica.

En este trabajo de investigación se consideró, generar acciones de Glúteos, Cardio y Abdomen, por las imágenes que se podía obtener de YouTube. Se considera estas 3 acciones para que las acciones tengan similitud y un buen reflejo del dominio para entrenar los modelos de aprendizaje automático. Se implementó el modelo InfoGAN (Chen y cols., 2016) como generador de imágenes de personas realizando acciones específicas. Así mismo, se utilizó una variable de control discreta y dos continuas que se añaden al modelo, en conjunto con el BatchNorm (Ioffe y Szegedy, 2015) y LeakyRelu (Maas, 2013). Para evaluar el rendimiento se empleó la Puntuación Inicia (Barratt y Sharma, 2018) ya que se asemeja a una evaluación humana (Barratt y Sharma, 2018).

El entrenamiento de modelo InfoGAN (Chen y cols., 2016), se realizó con grupos de imágenes obtenidas de videos de YouTube de personas realizando los tres tipos de acciones mencionados anteriormente. Se considero para este estudio, imágenes de personas con fondo blanco, con un tamaño de 64x64 píxeles. No solo se utilizó imágenes de una posición de cada ejercicio, sino que se consideró una secuencia de imágenes que representan una acción, para que el modelo aprenda que una acción tiene más de una posición.

Capítulo II

Marco Conceptual y Estado del Arte

En este capítulo se explora la generación de imágenes describiendo algunas de las principales técnicas existentes y se revisan los desarrollos recientes orientados a realizar la generación de imágenes. Así mismo, se describe el enfoque utilizado de estas investigaciones, indicando las imágenes seleccionadas, los resultados obtenidos y complejidad.

2.1. Estado del Arte

2.1.1. Trabajos previos en generación de imágenes utilizando aprendizaje profundo

Jyoti y Yanqing (Han y cols., 2018) ante la dificultad de obtener imágenes para el desarrollo de un modelo robusto de diagnóstico de enfermedades. Propuso un modelo GAN novedoso en la generación de imágenes PET (Tomografía por emisión de positrones) cerebral para tres etapas diferentes de la enfermedad Alzheimer, que son el control normal, deterioro cognitivo leve y enfermedad de Alzheimer. Se desarrolló un

modelo DCGAN, con 411 tomografías PET cerebrales de 479 pacientes de la base de datos de la iniciativa de neuroimagen de la enfermedad de Alzheimer (ADNI, 2020). El generador tuvo como entrada un vector de 100 números aleatorios extraídos de una distribución uniforme, unido a 5 capas de transposición convolucional con zancada conocidas como ”deconv”, teniendo como salida, una imagen PET cerebral de tamaño 128 * 128 * 3. El discriminador consistió en una arquitectura CNN que toma una imagen de tamaño 128 * 128 * 3 como entrada para su aprendizaje. Así mismo el discriminador analiza las imágenes PET generadas por el generador y decide si es real o falsa. Así mismo, en el proceso de capacitación, el discriminador se capacitó para maximizar la probabilidad de asignar etiquetas correctas a los ejemplos de capacitación y las muestras generadas. Finalmente se compararon cuantitativamente los resultados pronosticados en términos de la proporción máxima de señal de ruido (PSNR) y el índice de similitud estructural (SSIM) para medir la calidad de las imágenes generadas, además de histogramas para comparar las imágenes reales y falsas. En conclusión, luego de la evaluación cualitativa y cuantitativa del modelo propuesto demostraron que las imágenes generadas por el modelo DCGAN están cerca de las imágenes de PET cerebral real de diferentes etapas de la enfermedad de Alzheimer. Sin embargo, se tuvo que entrenar modelos GANs (Goodfellow y cols., 2014) por separado, para cada grupo de imágenes de la etapa de la enfermedad de Alzheimer, lo que aumentó la complejidad del entrenamiento.

Ramyasree Kola (Kola, 2019) Refirió que las imágenes generadas con el modelo DCGAN para una base de datos de flores, incluían ruido y errores visibles. Por tal motivo planteó realizar la generación de imágenes de plantas usando el modelo Style GAN (Karras, Laine, y Aila, 2019), y usar técnicas de pre procesamiento a los datos

para reducir el ruido en las imágenes. La investigación también se enfocó en identificar varias métricas de evaluación del desempeño del modelo Style GAN para los conjuntos de datos de imágenes generados. La implementación del modelo Style GAN, fue entrenado con el conjunto de datos de Leafsnap (Kumar y cols., 2012) que tenía 7719 imágenes de hojas de plantas. En el pre procesamiento de imágenes, se ejecutó un re escalamiento de imagen a 512 x 512 px, también se trabajó en un mismo espacio de color (RGB) y formato de imagen (JPG), a su vez se hizo un recorte y centrado de imagen. El desarrollo y entrenamiento del modelo se basó en el estudio original de Style GAN (Karras, Laine, y Aila, 2019) con un ajuste en los parámetros de entrenamiento. Para evaluar el modelo de aprendizaje Style GAN, utilizaron la medida de distancia de inicio de fréchet. Como resultado, las muestras regeneradas fueron muy similares a las originales y convincentes para el juicio humano. Sin embargo, la puntuación FID no fue el esperado. La principal dificultad del estudio, fue los tiempos de entrenamiento por cada cambio en los parámetros de entrenamiento.

Wei Ren Tan (Tan, Chan, Aguirre, y Tanaka, 2019) propone una serie de nuevos enfoques para mejorar la red de confrontación generativa (GAN) para la síntesis de imagen condicional denominado ArtGAN (Tan, Chan, Aguirre, y Tanaka, 2018). Una de las principales innovaciones es que, el gradiente de la función de pérdida se propaga desde el discriminador categórico al generador. El generador logra aprender de manera más eficiente y generar imágenes con mejor calidad. Inspirado en trabajos recientes, se incorpora un codificador automático en el discriminador categórico para obtener información complementaria adicional. Por último, presenta una estrategia novedosa para mejorar la calidad de la imagen generado por el modelo. En los experimentos, evalúan ArtGAN en el conjunto de datos de CIFAR-10 y STL-10 a través de

estudios de ablación. Todas las redes están entrenadas con el optimizador Adam con una tasa de aprendizaje inicial de 0.0002, b1 de 0.5 y un tamaño de minibatch igual a 100. La tasa de aprendizaje disminuye en un factor de 10 después de la iteración 30 000. El vector de ruido de entrada z es una Variable aleatoria multivariante de 100 dimensiones muestreada usando un generador aleatorio distribuido uniforme U de -1 a 1. Para la evaluación, se adopta la puntuación inicial (Salimans y cols., 2016). Los resultados empíricos mostraron que el modelo propuesto supera los resultados del estado del arte en CIFAR-10 en términos de puntaje de inicio. Cualitativamente, demostraron que ArtGAN es capaz de generar imágenes de aspecto plausible en Oxford-102 y CUB-200, así como también puede dibujar obras de arte realistas basadas en estilo, artista y género.

Simiao Yu, H. D (Yu y cols., 2019) propone SIMGAN, que puede generar imágenes fotorrealistas de tamaño 256 * 256 de alta resolución para SIM (manipulación de imágenes semánticas) (Dong, Yu, Wu, y Guo, 2017). Para lograr esto se tuvo que crear un generador capacitado G que primero tomaba como entrada una imagen y características de una descripción de texto objetivo. Las características extraídas de la imagen y el texto se concatenan y se alimentarán del bloque residual. El uso del bloque residual no solo ayudó a retener la estructura subyacente como lo requiere SIM (Dong y cols., 2017), sino que también permitió que el modelo a través de un proceso de codificación más profundo aprenda mejores mapeos entre las características visuales y textuales. La salida del bloque residual fue como la entrada del módulo decodificador, a partir del cual se generaron múltiples y diversas imágenes para SIM. En la etapa de entrenamiento, se emplearon varios términos de pérdida designados para permitir que SIMGAN genere imágenes fotorrealistas de alta resolución para SIM. Finalmente se

demuestra la efectividad de SIMGAN y su superioridad sobre los métodos existentes a través de la evaluación cualitativa y cuantitativa de los conjuntos de datos Caltech-200 y Oxford-102.

Han Zhang (Zhang y cols., 2017) propone que las Redes Adversarias Generativas Apiladas (StackGAN) están destinadas a generar imágenes fotorrealistas de alta resolución. Primero, proponen una arquitectura de red de confrontación generativa de dos etapas, StackGAN-v1, para la síntesis de texto a imagen. En segundo lugar, se propone una arquitectura avanzada de red de confrontación generativa de múltiples etapas, StackGAN-v2, para tareas generativas tanto condicionales como incondicionales. De acuerdo al artículo, el StackGAN-v2 que desarrollaron, consta de múltiples generadores y discriminadores, estructurados en forma de árbol. Las imágenes a múltiples escalas correspondientes a la misma escena se generan a partir de diferentes ramas del árbol. StackGAN-v2 muestra un comportamiento de entrenamiento más estable que StackGAN-v1 al aproximar conjuntamente múltiples distribuciones. Extensos experimentos demuestran que las redes de confrontación generativa apiladas propuestas superan significativamente a otros métodos de vanguardia en la generación de imágenes fotorrealistas.

2.2. Marco Conceptual

2.2.1. Aprendizaje Profundo

El aprendizaje profundo (Schmidhuber, 2015) hace uso de las redes neuronales para mejorar el aprendizaje automático. Una red neuronal, realiza un proceso parecido al del cerebro humano, haciendo uso de capas de neuronas. A diferencia del apren-

dizaje automático que obtiene conocimiento por medio de experiencia supervisada. El aprendizaje profundo lo obtiene casi sin supervisión.

El aprendizaje profundo es parte de muchos procedimientos de aprendizaje automático que se basan en entender datos representados. Por ejemplo, una imagen puede ser interpretada de muchas maneras (un vector de píxeles), pero algunas interpretaciones dan facilidad de aprender tareas focalizadas, por ejemplo, ”¿es esta imagen un gato?”sobre la muestra de datos, y la investigación en este ámbito trata de definir qué interpretaciones son mejores y cómo crear modelos para reconocer estas interpretaciones.

Existen muchas arquitecturas de aprendizaje profundo, como redes neuronales recurrentes, redes convolucionales, redes neuronales artificiales y redes neuronales de retroalimentación. Cada una para un caso de uso específico, estos se han estudiado en áreas, como la visión computacional, reconocimiento del habla, reconocimiento de audio y música, revelando resultados de vanguardia en varias áreas. En la Figura 2.1: se visualiza la diferencia entre una red neuronal simple y profunda, mostrando que la red neuronal profunda tiene más capas ocultas.

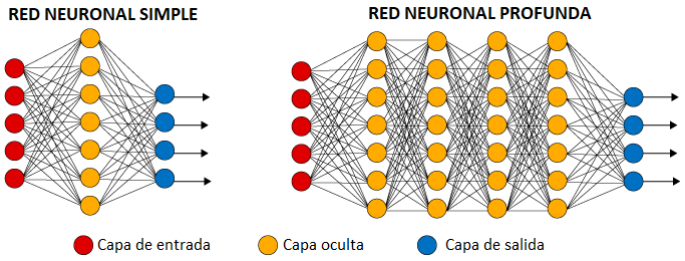


Figura 2.1: : Aprendizaje Profundo (Torres, 2017)

2.2.2. Aprendizaje supervisado y no supervisado

El aprendizaje automático (Kotsiantis, 2007) supervisado es el desarrollo de algoritmos a partir de datos de entrada y de salida(etiquetas), que luego pueden hacer predicciones en base a otros datos de entrada, que no necesariamente hayan existido inicialmente en el entrenamiento del modelo. El propósito del aprendizaje supervisado es componer un modelo de la distribución de los datos etiquetados para realizar predicciones genéricas. El clasificador resultante se usa para asignar etiquetas de clase a las instancias de prueba donde se conocen los valores de las características del predictor, pero se desconoce el valor de la etiqueta de clase.

El aprendizaje automático no supervisado (Ghahramani, 2004) tiene como objetivo extraer características de los conjuntos de datos, a fin de simplificar su representación. En síntesis, la agrupación de estas características separa conjuntos de datos en grupos que luego forman un resumen de los datos iniciales. Una descripción sencilla y precisa debería contener suficiente información sobre las inclinaciones presentes en el conjunto de datos, pero también sobre los comportamientos atípicos, para medir la medida en que los grupos son representativos de todo el conjunto de datos.

2.2.3. Redes Adversarias Generativas

Redes Adversarias Generativas o también llamadas redes GAN (Goodfellow, 2016) configura un juego entre dos actores. El primer actor es el generador G(z) donde (z) representa los datos de entrada que se inicializa como un ruido. Creando muestras de la misma distribución que los datos de entrenamiento. El segundo actor es el discriminador D(x), donde (x) representa la imagen de entrada del modelo. El discriminador

examina estas imágenes para decidir si son reales o falsas. El discriminador aprende utilizando procedimientos tradicionales de aprendizaje supervisado, dividiendo las entradas en clases reales y falsas. El generador está entrenado para engañar al discriminador. Es decir que el generador es como un estafador tratando de vender pinturas falsas de Picasso, y que el discriminador es como el experto en pinturas, tratando de diferenciar entre una pintura falsa y una real. Para tener éxito en este juego, el estafador debe aprender a crear pinturas como las de Picasso, hasta tal punto, en que el discriminador no pueda distinguir entre uno real y falso. En la Figura 2.2: se ilustra una GAN de rostros, donde la función del generador es G(x) y del discriminador es D(x). El generador busca generar imágenes que se asemejen a un rostro, mientras que el discriminador busca reconocer correctamente si una imagen de rostro es real o falsa. Cada vez que el discriminador evalúe la imagen como falsa, el generador aprende de sus errores generando nuevas imágenes de rostros aún más reales. Dicho de otra manera, ambas están en constante evolución mejorando y siendo más eficientes. Como resultado se tiene un generador que aprende a desarrollar un conjunto de imágenes más realistas y un discriminador que aprende a identificar como falsas incluso aquellas imágenes que tienen una apariencia más real.

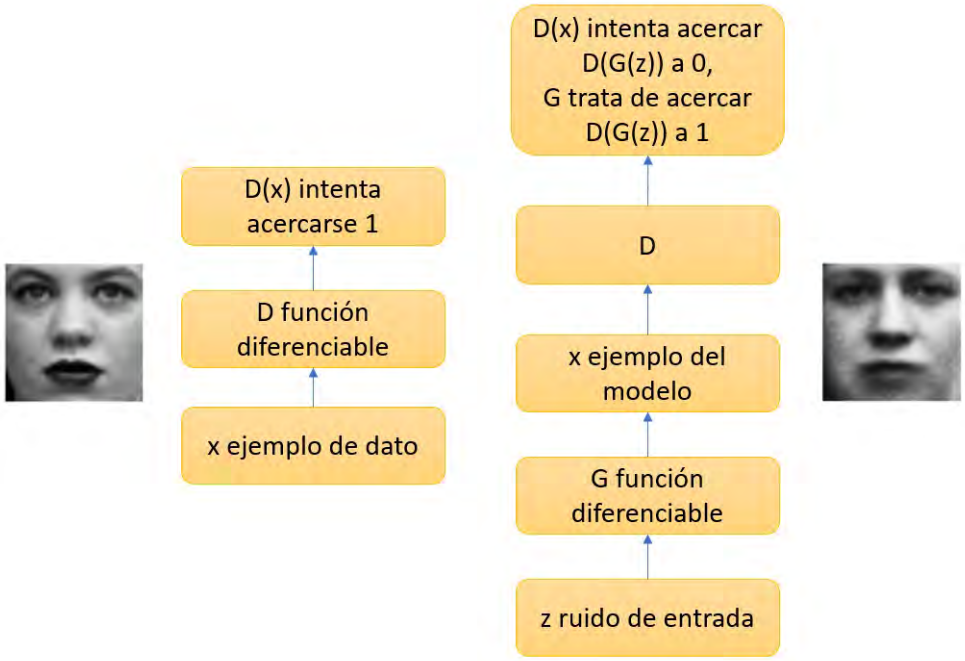


Figura 2.2: : Arquitectura GAN (Goodfellow, 2016)

En términos formales, la función objetivo de este juego corresponde a un juego entre el generador(G) y discriminador(D) con una función de costos V(D,G) regida por la siguiente fórmula :

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

En este caso $D(x)$ representa al discriminador y $G(z)$ al generador. La variable x representa los datos reales, mientras que $p_{data}(x)$ a la distribución de datos originales. La variable z es ruido con el cual se alimenta al generador para sintetizar los datos y $p_z(z)$ a la distribución del ruido.$\mathbb{E}_x$ es el valor esperado sobre todas las instancias de datos reales. $\mathbb{E}_z$ es el valor esperado sobre todas las entradas aleatorias al generador.

GAN (Goodfellow y cols., 2014) demuestra que este modelo tiene un óptimo global

cuando $p_g = p_{data}$, donde p_g, que es la distribución generada por el modelo, alcanza un equilibrio de Nash en el dominio de las funciones.

2.2.4. Equilibro Nash

Al ser las redes GAN como un juego entre el generador y discriminador, se busca alcanzar el equilibro de Nash, donde los jugadores seleccionan la estrategia óptima, dada la estrategia que seleccionan los demás. Es decir, elegir la mejor jugada, independientemente de la elección del otro jugador, teniendo en cuenta que cada jugador individual (Generador y Discriminador) no gana nada, modificando su estrategia mientras los otros mantengan las suyas. Así, cada jugador está ejecutando el mejor movimiento posible teniendo en cuenta los movimientos de los demás jugadores. Una estrategia que maximiza sus ganancias dadas las estrategias de los otros.

Es importante tener presente que un equilibrio de Nash no implica que se logre el mejor resultado conjunto para los jugadores, sino solo el mejor resultado para cada uno de ellos considerados individualmente. Goodfellow (Goodfellow, 2016) explica que en la práctica la red puede no necesariamente converger ni mucho menos alcanzar el equilibrio de Nash.

2.2.5. Redes generativas adversarias convolucionales profundas

Las redes adversarias generativas convolucionales profundas o DCGAN (Radford y cols., 2015) son una mejora de las redes GAN (Goodfellow y cols., 2014). Ya que pueden generar imágenes de mejor calidad y tener más estabilidad durante la etapa de entrenamiento. En el proceso de generar imágenes, hay dos fases al igual que con

las redes GAN: una fase de entrenamiento y una fase de generación. En la Figura 2.3: se muestra el modelo generador, que parte desde un ruido de 100 dimensiones, pasando por 4 convoluciones, hasta producir la imagen de 64 x 64 que será tratada por el discriminador. En esencia, no se utilizan capas completamente conectadas o agrupadas.

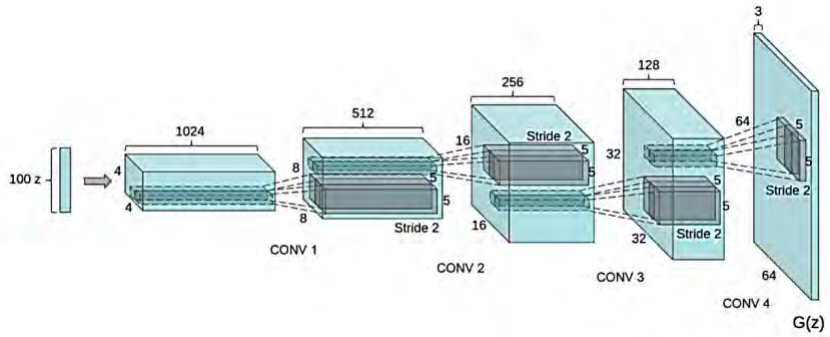


Figura 2.3: : Arquitectura DCGAN (Radford y cols., 2015)

Dos características importantes de este tipo de red son el uso de BatchNorm (Ioffe y Szegedy, 2015) para regular la escala de características extraídas, y LeakyRelu (Maas, 2013) para prevenir gradientes muertos. Este también reemplaza toda agrupación máxima con zancada convolucional y utiliza la convolución transpuesta para el muestreo ascendente. Elimina capas completamente conectadas, usa la normalización por lote y ReLU en el generador, excepto la salida que usa Tanh y LeakyReLU en el discriminador.

2.2.6. Redes generativas adversarias convolucionales profundas condicionales

Cuando se desea condicionar la generación de imágenes por clases, es cuando se habla de Redes generativas adversarias convolucionales profundas condicionales o

DCGAN (Mirza y Osindero, 2014) propuesta en el año 2014. En esta situación tanto el generador como el discriminador están condicionados por alguna información adicional que podría ser de cualquier tipo, etiquetas de clase u otro tipo de dato. Se realiza el acondicionamiento, dirigiendo esta clase o dato, a la capa de entrada del generador y discriminador. Por ejemplo, supongamos que el modelo CDCGAN genera imágenes de varios tipos de animales, pero se desea generar solo imágenes de gatos, entonces se pasan los datos de la etiqueta gatos al generador, junto con el ruido de entrada para producir la imagen de un gato. Integrando esta variable a la formula origina de una GAN, se tendría lo siguiente:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data(x)}}[logD(x|y)] + \mathbb{E}_{z\sim p_z(z)}[log(1-D(G(z|y)))]$$

En este caso $D(x|y)$ representa al discriminador y $G(z|y)$ al generador. La variable x representa los datos reales, $p_{data}(x)$ a la distribución de datos originales. La variable z es ruido con el cual se alimenta al generador para sintetizar los datos. La variable y representa la etiqueta de la clase y $p_z(z)$ a la distribución del ruido. $\mathbb{E}_x$ es el valor esperado sobre todas las instancias de datos reales. $\mathbb{E}_z$ es el valor esperado sobre todas las entradas aleatorias al generador.

En la Figura 2.4: se ilustra la estructura de este tipo de red, donde como ejemplo se agrega la clase "y" al generador y discriminador, unido a los datos de entrada, para condicionar el aprendizaje.

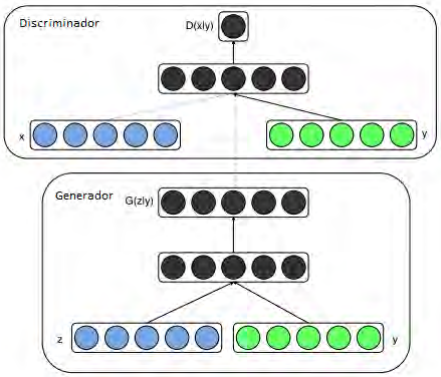


Figura 2.4: : Arquitectura Generativa Condicional (Recognizer, 2021)

2.2.7. Aprendizaje de representación interpretable por información que maximiza las redes generativas adversarias

Aprendizaje de representación interpretable por información que maximiza las redes generativas adversarias o InfoGAN (Chen y cols., 2016) es una extensión de las redes GAN (Goodfellow y cols., 2014) donde agrega el término de maximización de información.

Aunque en una GAN el generador aprenda a separar espacialmente propiedades de imagen en el espacio latente, no existe control sobre ellas porque están entrelazadas. InfoGAN (Chen y cols., 2016) busca desenredar estas propiedades. Para lograr esto, se agregan variables de control junto con el ruido de entrada del generador, entrenando el modelo con una función de perdida de información mutua. Información mutua, es la cantidad de información aprendida de una variable en base a otra variable. InfoGAN (Chen y cols., 2016) agrega el término de maximización de información, maximizando los códigos latentes y las muestras guiadas por estos códigos a la función objetivo del GAN. Es decir, agrega variables de control al modelo, que son aprendidas

automáticamente, permitiendo controlar la imagen generada, como el grosor, estilo o tipo. En experimentos, el artículo demuestra la efectividad del modelo para desenredar factores visuales, utilizando los códigos latentes.

Esto inicia introduciendo información mutua para inducir códigos latentes

$$\min_G \max_D V(D,G) = V(D,G) - \lambda I(c; G(z,c))$$

λ es un hiperparámetro adicional al modelo de regularización constante y es típicamente configurado en uno para códigos latentes discretos. Cuando son códigos latentes continuos, se usa el λ más pequeño para garantizar que el $\lambda I(c; G(z,c))$, este en la misma escala que los objetivos GAN. El termino $I(c; G(z,c))$ es la información mutua entre el código latente c y la salida del generador $G(z,c)$.

Al no ser práctico calcular la información mutua de forma explícita, entonces un límite inferior se aproxima utilizando argumentos variacionales estándar. Introduciendo una distribución auxiliar $Q(c|x)$, que es modelada por una red neuronal parametrizada, y que pretende aproximarse a la $P(c|x)$ real, el cual representa la probabilidad del código c dada la entrada generada x.

Variables de control categóricas

Estas variables se utilizan para controlar el tipo o la clase de imagen que se quiere generar del modelo. El número de valores está limitado por un grupo fijo. Por ejemplo, la variable ”País”, con valores: Perú, Chile y Uruguay, cada uno representan una clase diferente.

Los modelos GAN (Goodfellow y cols., 2014) usan la codificación one-hot a la representación de estas clases categóricas. Es decir, si se tiene 3 clases, entonces el

código de control sería una de las clases, por ejemplo, la primera clase, la entrada del vector de control categórico al modelo del generador sería un vector de 3 elementos, 2 valores ceros con un valor 1 para la clase, por ejemplo [0, 0, 1]. No se necesita elegir las variables de control categóricas al entrenar el modelo; ya que, se genera aleatoriamente, cada uno con una probabilidad uniforme para cada muestra.

Variables de control continuo

Se utiliza para controlar el estilo de la imagen, se muestrean a partir de una distribución uniforme, entre -1 y 1, y se proporcionan como entrada al modelo del generador. El modelo InfoGAN puede implementar la predicción de variables de control continuas con una distribución gaussiana, donde la capa de salida está configurada para tener un nodo, la media que genera la red de reconocimiento, y un nodo para la desviación estándar del gaussiano, por ejemplo, se requieren dos salidas para cada variable de control continuo. La función de pérdida debe calcularse como la información mutua sobre los códigos de control gaussianos, lo que significa que deben reconstruirse a partir de la desviación estándar y media antes de calcular la pérdida.

2.2.8. Normalización por Lotes

La normalización es un método de preprocesamiento usada para uniformar los datos. Es decir, tener diferentes orígenes de datos en un mismo rango. Si no se normaliza los datos antes del entrenamiento de un modelo, puede causar, dificultad en el entrenamiento y lentitud en el aprendizaje.

Normalización de lotes (Ioffe y Szegedy, 2015), es un método de normalización, realizada en medio de las capas de una red neuronal, facilitando el aprendizaje. Se

caracteriza de hacer que la normalización sea parte de la arquitectura del modelo y realizar la normalización para cada mini lote de entrenamiento. La normalización por lotes permite utilizar tasas de aprendizaje mucho más altas y disminuir el efecto de una pobre inicialización. También actúa como un regularizador, en algunos casos eliminando la necesidad de abandono. Además, la normalización por lotes permite que cada capa de la red aprenda por sí misma un poco más independientemente de otras capas.

2.2.9. Funciones de Activación ReLU y LeakyReLU

La función de activación se ocupa de retornar una salida dada un valor de entrada o conjunto de entradas. Normalmente la salida está en un rango de (0,1) o (-1,1).

En la Figura 2.5 se muestra la función de activación ReLU muy utilizada debido al aprendizaje rápido en las redes neuronales, que solo retorna valores positivos. Retornando el valor cero si es un ingreso negativo. Sin embargo, ReLU tiene una limitación, que a veces es frágil durante el entrenamiento, cuando la función es cero y su derivada también lo es, se genera lo que se llama, muerte de neuronas, lo que dificulta el aprendizaje, ya que las neuronas muertas dan una activación cero.

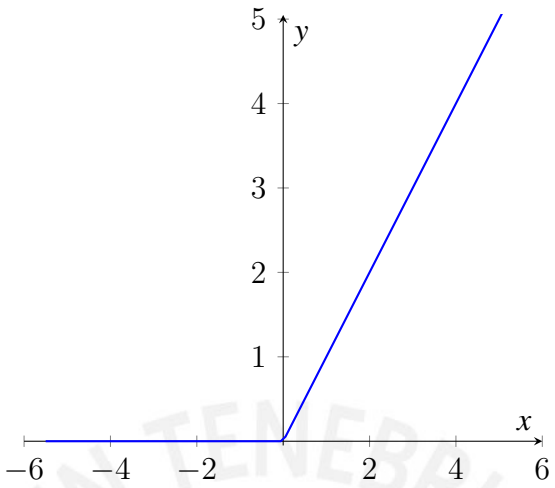


Figura 2.5: : Función de Activación ReLU

Características de la función ReLU:

- Activación Sparse – solo se activa si son positivos.
- No está acotada.
- Se pueden morir demasiadas neuronas.
- Se comporta bien con imágenes.
- Buen desempeño en redes convolucionales.

En la Figura 2.6 se muestra la función de activación Leaky ReLU, es una versión mejorada de la función ReLU, para dar solución a la muerte de neuronas. Basándose en ReLU, cambia los valores ingresados, multiplicando los negativos por un coeficiente rectificativo y abandonando los positivos según entran. Al realizar esta modificación para los valores de entrada negativos, el gradiente del lado izquierdo del gráfico muestra ser un valor distinto de cero. Por eso, ya no encontramos neuronas muertas en esa región.

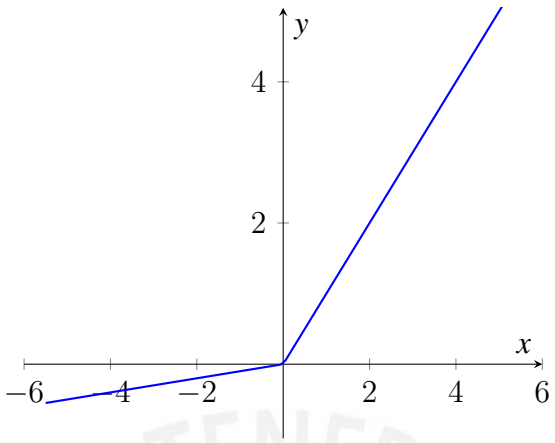


Figura 2.6: : Función de Activación Leaky ReLU

Características de la función Leaky ReLU:

- Similar a la función ReLU.
- Penaliza los negativos mediante un coeficiente rectificador.
- No está acotada.
- Se comporta bien con imágenes.
- Buen desempeño en redes convolucionales.

2.2.10. Medidas de rendimiento

Puntuación Inicial

Puntuación Inicial o IS (Barratt y Sharma, 2018) es una métrica objetiva para determinar la calidad de imágenes generadas por modelos GAN. Se desarrollo con el intento de dejar atrás la valoración humana de las imágenes.

La puntuación inicial involucra el uso de un modelo pre entrenado para clasificar las imágenes, precisamente el modelo Inception v3 (Szegedy, Vanhoucke, Ioffe, Shlens, y Wojna, 2015). Este modelo clasifica las imágenes, es decir, predice la probabilidad de que la imagen corresponda a cada clase. La puntuación inicial mide el rendimiento en base a 2 criterios, la calidad de la imagen generada y la diversidad. Esta correlaciona bien con la puntuación humana de imágenes generadas a partir de un conjunto de datos. Como resultado, dispone de un valor más bajo de 1.0 y un valor más alto del número de clases aceptadas por el modelo clasificador. Se calcula utilizando el modelo Inception v3 entrenado con antelación para pronosticar las probabilidades de clase para cada imagen generada. La fórmula utilizada es la siguiente:

$$IS(G) = \exp(\mathbb{E}_{x \sim p_g} D_{KL}(P(y|x)||p(y))))$$

Donde $x \sim p_g$ indica que x es una imagen muestreada de p_g, $D_{KL}(p(y|x)||p(y))$ es la divergencia KL, esta es una medida de cuan similares o diferentes son dos distribuciones de probabilidad. KL es alta cuando las distribuciones son diferentes. Es decir, alto cuando cada imagen generada tiene una clase distinta y el conjunto general de imágenes generadas tiene una amplia gama de clases. Mientras que $P(y|x)$ es la distribución de clases condicional, $p(y)$ es la distribución de clases marginal. La puntuación inicial (PI) se obtiene, de la exponencial de la divergencia KL entre las distribuciones $p(y)$ y $p(y|x)$.

Una deficiencia de IS es que puede tergiversar el rendimiento si solo genera una imagen por clase. p(y) seguirá siendo uniforme, aunque la diversidad sea baja.

Distancia de inicio de Fréchet o FID

El puntaje FID (Heusel, Jan, y Hochreiter, 2017) es una métrica para evaluar la calidad de las imágenes generadas y desarrollada para evaluar el rendimiento de las redes adversas generativas. Se planteo como una mejora de la Puntuación Inicial (PI). Sin embargo, FID, no compara las imágenes sintéticas con las imágenes reales. Su objetivo es evaluar en base a las estadísticas de una colección de imágenes sintéticas en comparación con las estadísticas de una colección de imágenes reales del dominio objetivo. Es decir, verifica cuán similares son los dos grupos en términos de estadísticas sobre las características de visión por computadora de las imágenes calculadas, utilizando el modelo de Inception v3 (Szegedy y cols., 2015) para la clasificación de imágenes. Los puntajes más bajos indican que los dos grupos de imágenes son más similares o tienen estadísticas más similares con un puntaje perfecto de 0.0 que indica que los dos grupos de imágenes son idénticos.

La puntuación FID se calcula utilizando la siguiente ecuación:

$$FID = ||\mu_r - \mu_g||^2 + T_r(\sum_r + \sum_g - 2(\sum_r \sum_g)^{1/2})$$

Donde,

T_r: Operación de álgebra lineal de trazas (suma de los elementos a lo largo de la diagonal principal de la matriz cuadrada),

r: Distribución de datos reales,

g: Distribución de datos generados,

μ_r, μ_g: Media de datos reales y generados respectivamente,

$\sum_r, \sum_g$: Covarianza de datos reales y generados respectivamente

Capítulo III

Metodología

En la Figura 3.1: se muestra el flujo de procesos realizado para lograr el objetivo planteado para este estudio. El cual consiste en generar imágenes de personas realizando una acción.

En primer lugar, se buscó y descargó videos de personas realizando ejercicios de abdomen, cardio y glúteos en YouTube. Estos videos tenían resolución HD, a su vez, mostrar a la persona realizar ejercicios en cuerpo completo. En segundo lugar, se obtuvo las imágenes de los videos y se realizó la selección de imágenes. Las imágenes seleccionadas, tenían que ser de personas realizando un ejercicio en diferentes posiciones, no repetitivos. Luego se agruparon, según el tipo de ejercicio. En tercer lugar, se segmentó las imágenes con el modelo YOLACT pre entrenado con los datos de Microsoft COCO para obtener solo a la persona, quitando el fondo de las imágenes. Además, se recortaron las imágenes para adquirir la sección donde se encuentra la persona. El cuarto lugar, se definió una orientación de las imágenes, para los ejercicios de glúteos y abdomen. A su vez se cambió el tamaño de las imágenes a 64x64, utilizando el método de interpolación INTERCUBIC.

Finalmente, se desarrolló y ajusto el modelo InfoGAN, hasta conseguir el mejor

resultado en la evaluación de Puntuación Inicial para este modelo.

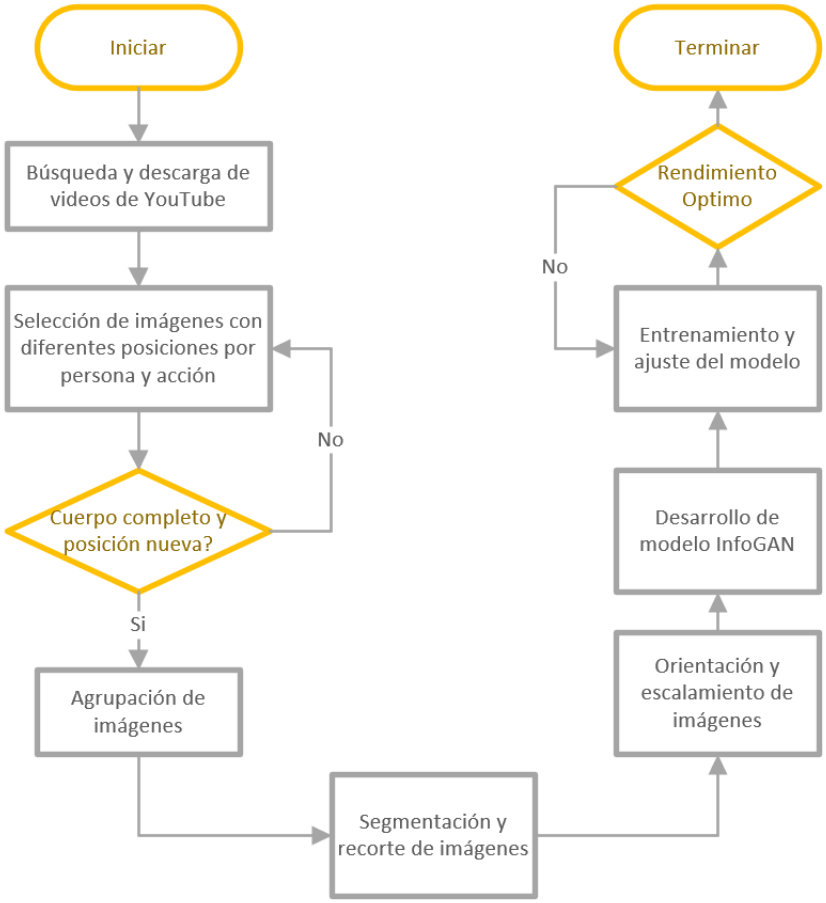


Figura 3.1: : Metodología

3.1. Recopilación de datos

Para el propósito de esta investigación, no se usó un conjunto de datos de imágenes existentes, porque no existe. Sin embargo, se recopilo imágenes por cuenta propia de una y varias personas que realizan 3 tipos de ejercicios: glúteo, abdomen y cardio.

3.1.1. Búsqueda de datos

Para que una GAN (Goodfellow y cols., 2014) aprenda a generar imágenes sintéticas, es necesario obtener una cierta cantidad de imágenes de entrenamiento. En este estudio, se trabajó con 2 grupos de imágenes de personas realizando los 3 tipos de ejercicios. El primer grupo de una sola persona realizando los 3 tipos de ejercicio, se recopiló un total de 3900 imágenes, dividiéndose en partes iguales de 1300 imágenes por clase, con la finalidad de realizar entrenamientos rápidos y tener un modelo base con hiperparametros ajustados, para el siguiente grupo. El segundo grupo de personas diferentes realizando los 3 tipos de ejercicios, se recopiló un total de 15000 imágenes, dividiéndose en partes iguales de 5000 por clase. Se tomo como referencia la cantidad de imágenes del conjunto de datos de CIFAR-10 (Krizhevsky, Nair, y Hinton, s.f.) utilizados para entrenar un modelo GAN. Estas imágenes fueron recolectadas de videos de YouTube con aTube Catcher, donde existen muchas personas que realizan estos tipos de ejercicios. Así mismo, se realizó una selección de videos donde se pueda visualizar a la persona en cuerpo completo. Las figuras 3.2: , 3.3: y 3.4: muestran un ejemplo de los videos de YouTube utilizados en esta investigación.

Figura 3.2: : Videos de Abdomen

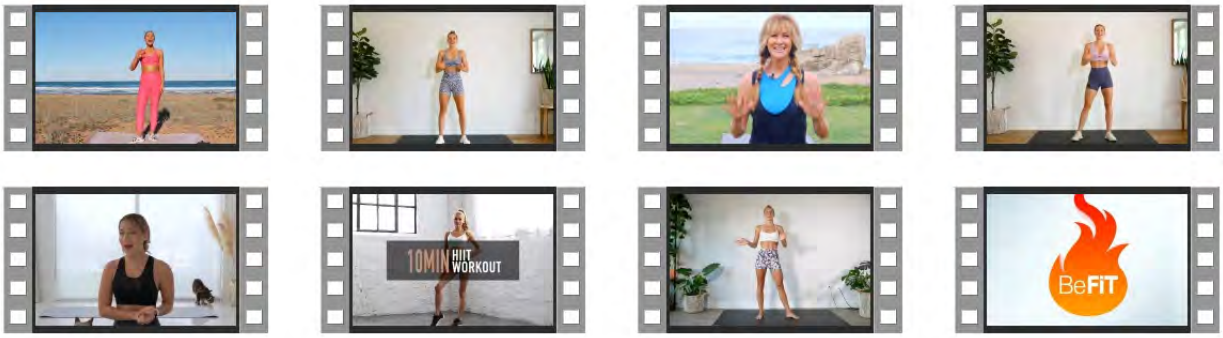


Figura 3.3: : Videos de Cardio

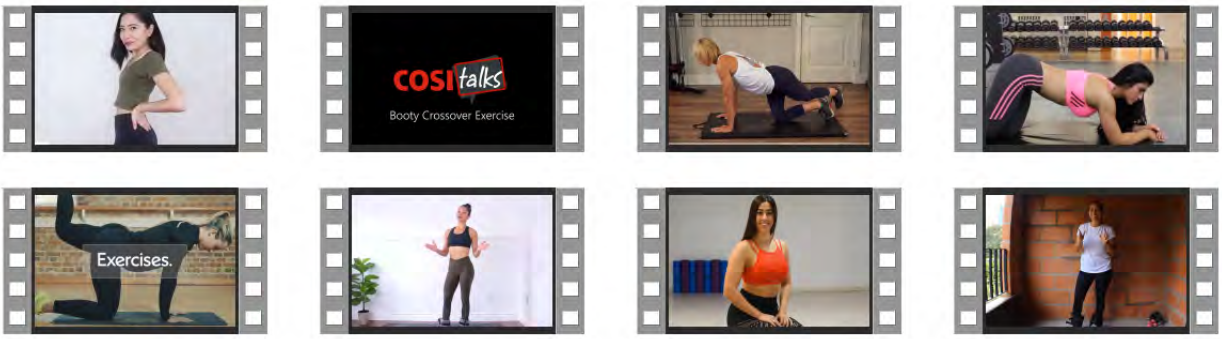


Figura 3.4: : Videos de Gluteo

Por otra parte, los videos tuvieron una duración, entre 3 a 45 minutos, algunos videos de mayor duración, tenían los 3 tipos de ejercicio. Así mismo se descargó los videos con resolución HD (1920x1080) en formato mp4. Por otra parte, los videos fueron grabados con una velocidad, entre 25 y 60 cuadros por segundo. El estudio de formatos de video de alta velocidad de fotogramas (Mackin, Zhang, y Bull, 2019) argumenta, que la velocidad de fotogramas por segundo, mejora la calidad de percepción de los videos hasta los 120 cuadros por segundo. La nitidez de cada fotograma, depende de la cámara que se use, es decir, la cantidad de fotogramas del video no influye en la calidad de la imagen. Para obtener los fotogramas, se usó OpenCV (Bradski, 2000), la librería permite obtener los cuadros por segundo y guardar las imágenes en

una carpeta específica.

3.1.2. Selección de datos

Esta selección de imágenes se realiza de manera manual, esto quiere decir que se utiliza el criterio del ojo humano para la selección.

La selección se basa según las posturas de los ejercicios mostrados en DAREBEE (Darebee, 2021), que es un recurso de fitness global independiente. Una organización sin fines de lucro, donde toda la información se ha investigado y probado para ofrecerse de manera gratuita. En la Figura 3.5: se muestra las posturas del ejercicio de Abdomen, Cardio y Glúteos obtenidos de DAREBEE, que se usaron como base para seleccionar imágenes de cada tipo de ejercicio, pero que se diferencien en las posturas de cada una. Se tiene que verificar que se visualice el cuerpo completo de la persona y que esta imagen sea la más nítida posible. Esto con el fin de que nuestro modelo de entrenamiento GAN (Goodfellow y cols., 2014) tenga mejores imágenes para su aprendizaje. No se utiliza un proceso automático, porque requeriría entrenar otro modelo para el dominio de este estudio.

(a) Abdomen

(b) Cardio

(c) Gluteos

Figura 3.5: : Imágenes de posturas de ejercicios

Para el entrenamiento del modelo de aprendizaje automático, se divide las imágenes en dos grupos. El primer grupo de la Tabla 3.1, es un grupo de imágenes donde se considera a una sola persona. Este criterio se basa en el estudio de datos MNIST (LeCun y Cortes, 2010). Este grupo fue usado para desarrollar el modelo InfoGAN y

tomar los parámetros ajustados como base para el siguiente grupo de imágenes.
El segundo grupo de la Tabla 3.2, se tiene a personas diferentes realizando los ejercicios, los cuales se usaron para lograr los objetivos de este estudio.

Tabla 3.1: : Imágenes de una misma persona

Ejercicio de Gluteo	Ejercicio de Abdomen	Ejercicio de Cardio
		
$1920x1080$	$1920x1080$	$1920x1080$
1300	1300	1300

Tabla 3.2: : Imágenes de varias personas

Ejercicio de Gluteo	Ejercicio de Abdomen	Ejercicio de Cardio
		
$1920x1080$	$1920x1080$	$1920x1080$
5000	5000	5000

3.2. Procesamiento de datos

Cuando se trata de desarrollar un modelo GAN, el procesamiento de datos es el primer paso que marca el inicio del proceso. Por lo general los datos reales están incompletos, inconsistentes e inexactos. El procesamiento, ayuda a limpiar, formatear y organizar los datos, dejando listo estos datos para el entrenamiento del modelo.

3.2.1. Segmentación y recorte de imágenes

Con la finalidad de que el modelo de aprendizaje se enfoque en la generación de acciones, solo de la persona y no su entorno que la rodea, se retira el fondo de las imágenes a través de un proceso de segmentación de imágenes, en este caso se usó el modelo YOLACT (Bolya, Zhou, Xiao, y Lee, 2019) con los datos de Microsoft COCO (Lin y cols., 2014), ya que se adecúa mejor a las imágenes. En la Figura 3.6: se muestra un ejemplo del resultado que se logró con la segmentación de imágenes.

Figura 3.6: : Ejemplo de segmentación

El resultado de la segmentación que se muestra en la Figura 3.6: , se obtuvo, gracias a la modificación interna del proyecto original YOLACK (Bolya y cols., 2019). La primera modificación del YOLACK se realiza en el archivo de evaluación llamado eval.py, se remplaza el método de enmascaramiento que se encuentra en la línea 183. Con esto se realiza la segmentación con fondo negro.

Luego, en el mismo archivo eval.py se agrega varias líneas de código después de la línea 224 con el fin de realizar el recorte de las imágenes, según la detección de YOLACT. En este caso inicialmente se verifica que se haya detectado una persona, si esto se cumple se realiza un recorte rectangular con ciertos márgenes en cada lado, siempre verificando de que estos márgenes que se están agregando no sobrepasen el tamaño de la imagen original, este margen es diferente para cada grupo de imágenes.

Es importante también que solo se detecten a las personas de cada imagen, omitiendo otros posibles objetos que se encuentren en la imagen original. Para lograr este objetivo en el archivo detection.py se agregará una línea de código después del código en la línea 83.

```
cur_scores[1:] *= 0
```

3.2.2. Orientación de imágenes

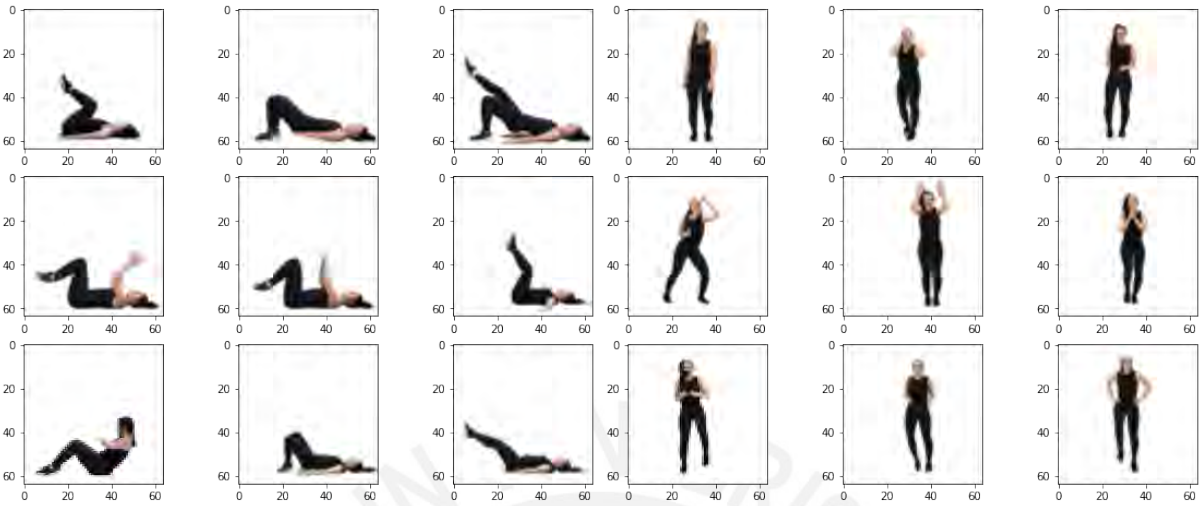
Los grupos de imágenes de Abdomen y Glúteo, tienen imágenes con orientación de la cabeza hacia la izquierda y derecha. En las pruebas iniciales del modelo con el primer grupo de imágenes, mostró dificultad del modelo en aprender ambas orientaciones de la cabeza, debido a la poca cantidad de imágenes. Por consiguiente, se procesaron las imágenes para que tengan la misma orientación, en este caso hacia la derecha. Esto debido a la limitante de datos con lo que se cuenta para el proceso de experimentación, con el fin de que la InfoGAn (Chen y cols., 2016) evite confusiones en el aprendizaje.

(a) O. Izquierda (b) O.Derecha

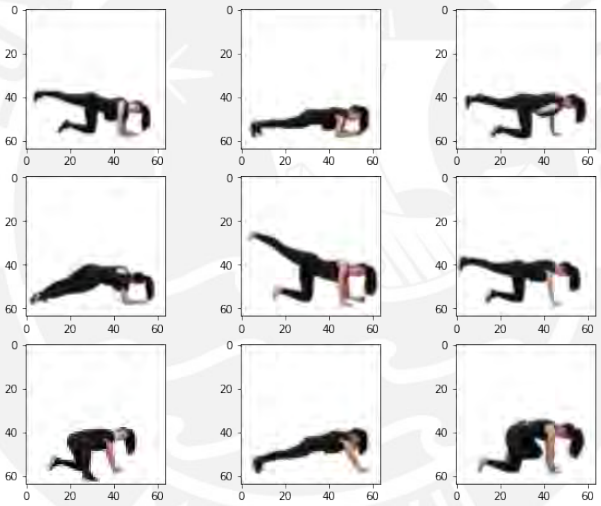
Figura 3.7: : Cambio de orientación de imágenes

3.2.3. Escalamiento y Extracción de características

Las imágenes serán cambiadas a 64x64 píxeles utilizando el método de interpolación INTER CUBIC (Keys, 1981) que, a diferencia de una interpolación bilineal, contempla 16 píxeles de 4 × 4. Las imágenes procesadas con interpolación bicúbica son más suaves demostrando que puede mantener la calidad de la imagen para esta investigación. Luego de realizar el escalamiento de imagen, el siguiente paso es la extracción de características de las imágenes, en este caso se usara las características de color (Salau y Jain, 2019), para que pueda ser procesado en los modelos del Generador y Discriminador. Finalmente se guardaron estos datos en formato NPZ que es apropiado para este caso, ya que admite una compresión nativa NumPy. En la Figura 3.8: se muestra las imágenes que se usaron en el entrenamiento del modelo.



(a) Abdomen (b) Cardio



(c) Gluteos

Figura 3.8: : Imágenes de entrenamiento 64x64

Normalización y Barajamiento de datos

El objetivo de normalizar los datos, es la de utilizar una escala general sin alterar la divergencia en los intervalos de valores sin perder información. En este caso se utiliza la normalización Min-Max, el cual consiste en transformar linealmente los datos originales, en valores mínimos y máximos, es decir en un intervalo de valores, en este

caso para los píxeles de imágenes tienen valor entre -1 y 1.

Por otro lado, para garantizar que todas las imágenes no estén en orden según las clases de ejercicio, barajamos el orden de las imágenes. Esto se logra mediante la función shuffle de la biblioteca numpy.

3.3. Arquitectura GAN

Para este proyecto, se utiliza el modelo InfoGAN (Chen y cols., 2016), que es la red adversaria generativa convolucional con etiquetas condicionales. No se usa una DC-GAN (Mirza y Osindero, 2014), ya que el mapeo estructurado por el generador durante el proceso de entrenamiento es algo aleatorio. Aunque el modelo aprende a separar espacialmente las propiedades de las imágenes generadas en el espacio latente, no existe control, las propiedades están enredadas. InfoGAN (Chen y cols., 2016) justamente busca desenredar y controlar las propiedades de las imágenes generadas haciendo uso de las variables de control discretas y continuas, entrenadas por medio de la función de perdida de información mutua. Por esta razón, InfoGAN (Chen y cols., 2016) es la arquitectura que se usa en este proyecto.

El modelo desarrollado tiene como código base el repositorio de Comparación-rendimiento-de-GAN-en-cifar-10 (Aria, 2019), desarrollado en Python con TensorFlow 1.5, el cual fue adaptado para el propósito de este proyecto. Este modelo baso su aprendizaje y generación de imágenes en tamaño 64x64, haciendo uso de variables de control, tipo discreto y continuo, enfocados en controlar el tipo y estilo de la imagen. Como el modelo realiza un aprendizaje supervisado, cada imagen tiene como salida un código vector de dimensión 5 (3 Etiquetas + 2 características), esta salida controla

la acción generada. El modelo toma el vector de ruido (z) y el vector de control (c) como entradas separadas y concatenarlos antes de usarlos como base para generar la imagen.

Se uso un espacio latente de 128 dimensiones para hacer coincidir el papel de InfoGAN, es decir, en este caso, cada vector de entrada al modelo del generador será de 128 (variables gaussianas aleatorias) mas 3 (una variable de control categórica codificada en caliente). Él modelo generador utilizado se muestra en la Figura 3.9: , teniendo como principales actores el uso de Normalización de Lotes y Relu con una salida Sigmoide. El modelo discriminador se entrenó de forma independiente en imágenes reales y falsas, según una GAN normal como se muestra en la Figura 3.10: , haciendo uso de LeakyRelu, teniendo como salida otra sigmoide. También se cuenta con un modelo auxiliar que tiene una salida de nodo para cada valor en la variable categórica y usa una función de activación softmax.

El modelo no se compila ya que no se utiliza de forma independiente. La función de pérdida a utilizar en el generador y discriminador es de una entropía cruzada sigmoidal de logits. Para el proceso de la función de pérdida de información mutua del modelo auxiliar se usa un SoftMax. El optimizador a utilizado en el generador, discriminador y clasificador es un Adam.

Figura 3.9: : Modelo Generador

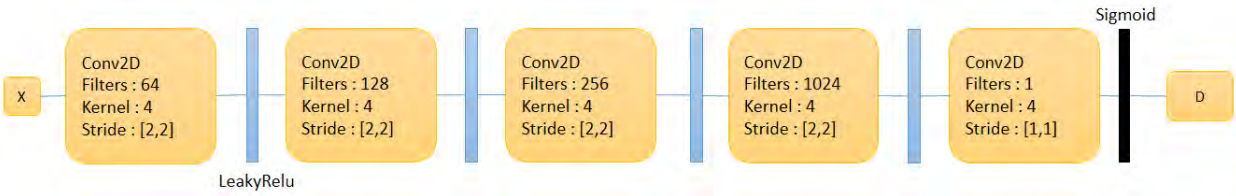


Figura 3.10: : Modelo Discriminador

Del repositorio Comparación-rendimiento-de-GAN-en-cifar-10 (Aria, 2019) se modificó los parámetros de entrada y entrenamiento del modelo. El tamaño de imágenes es de 64x64, a su vez, se cambió la configuración original que generaba 10 clases a 3 clases para este proyecto en el parámetro que indica la cantidad de clases del modelo. La tasa de aprendizaje se mantuvo según el proyecto original, cada 64 imágenes generadas se guardó una imagen en una carpeta externa, para su posterior revisión y visualización.

Se modificó el clasificador, generador, discriminador y cálculo de rendimiento del modelo original para generar y extraer características de imágenes en dimensiones de 64x64. El clasificador, generador y discriminador mantuvo la cantidad de capas y funciones de activación que tenían originalmente, pero se cambió las dimensiones para que se maneje las imágenes según las dimensiones establecidas. En el cálculo del rendimiento por Puntuación Inicial, se cambiaron las dimensiones y parámetros clasificador a 3, que son los tipos de imágenes que se usó para este proyecto. En el archivo de utilidades se modificó el método de guardar la imagen, se reemplazó el uso de la librería ”scipy”por ïmageio”.

3.4. Entrenamiento

En primera instancia se utiliza el entorno local para la recopilación y procesamiento de datos. Por otra parte, para el entrenamiento del modelo se necesita buena capacidad computacional, ya que estos modelos GAN demoran en su proceso de aprendizaje, por tal razón se utilizó la plataforma de Google Colaboratory, creando una instancia virtual.

Los parámetros iniciales del modelo de entrenamiento se tomaron del repositorio de Comparación-rendimiento-de-GAN-en-cifar-10 (Aria, 2019). Luego de cada entrenamiento y evaluación de la gráfica de rendimiento del modelo, se optimizo los hiperparámetros, como la época, el tamaño del lote y la dimensión de ruido del generador, en función del avance o retroceso del puntaje de rendimiento de la Puntuación Inicial (Barratt y Sharma, 2018). Obteniendo el mayor rendimiento con 100 épocas, 32 lotes y una dimensión de ruido de 128.

3.5. Evaluación de rendimiento

El objetivo de este trabajo es generar imágenes de personas realizando una acción específica. Por consiguiente, es necesario evaluar el rendimiento del modelo. En esta ocasión se utiliza la Puntuación Inicial (Barratt y Sharma, 2018), que determina la calidad y diversidad de las imágenes. Internamente en el modelo, cada época de entrenamiento realizo una evaluación de las muestras generadas por el generador, colocando un puntaje entre 0 y 3. Siendo 3 el máximo, determinado por la cantidad de clases de imágenes del modelo. Terminando la experimentación, se mostró una gráfica de líneas de los puntajes obtenidos en cada época.

Capítulo IV

Experimentación

Lo siguiente son los resultados obtenidos luego de ejecutar la metodología para lograr los objetivos planteados. También se indica los parámetros usados en esta fase para cumplir con el objetivo del proyecto.

4.1. Generación de persona realizando acciones específicas

4.1.1. Procesamiento de datos

Se descargaron un total de 600 videos para los 3 tipos de acciones, estos videos tenían una duración entre 3 y 45 minutos. De estos videos se obtuvieron aproximadamente 600 mil imágenes. Se obtuvo una imagen de cada 30 fotogramas de los videos descargados de YouTube. El proceso de selección de imágenes tomo mucho tiempo debido que se realizó de manera manual usando el ojo humano, procesar cada 60 mil imágenes tomaba entre 5 y 6 horas. En la Figura 4.1: se muestra un ejemplo de

cómo se realizó la evaluación y selección de 159 imágenes de glúteos. El área roja no se consideró, porque no forma parte de los ejercicios de glúteos, no se muestra correctamente la imagen o no se visualiza a la persona en cuerpo completo. En el área verde, se muestra las 4 imágenes que consideramos correcta en esta sección. En el área amarilla, se muestra posiciones repetitivas que ya fueron consideradas. Es decir, se seleccionó 4 imágenes de 159 en 2 o 3 segundos. Considerar que luego de tener las 5000 primeras imágenes para cada tipo de ejercicio, se revisó nuevamente, para descartar o reemplazar alguna imagen.

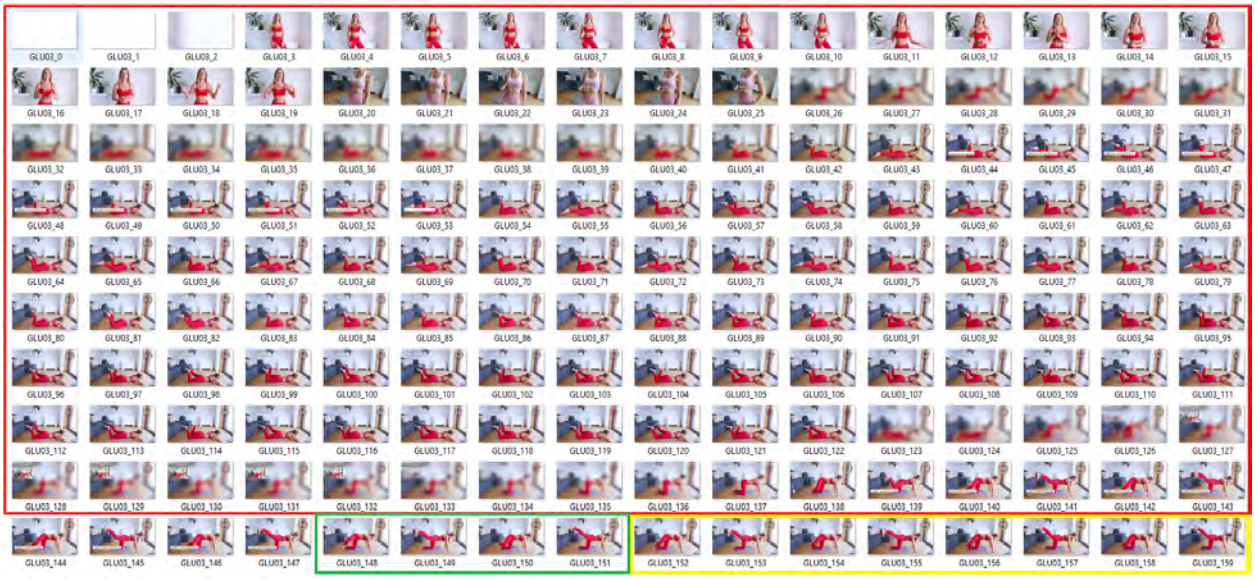


Figura 4.1: : Ejemplo de selección de imágenes de Glúteos

En la segmentación de imágenes, cuando se realizó la segmentación de la imagen con el modelo MaskRCNN (He, Gkioxari, Dollár, y Girshick, 2020), este no fue muy eficiente, la segmentación era imperfecto y tomaba mucho tiempo, se perdía parte de las extremidades de las personas. A diferencia de YOLACT (Bolya y cols., 2019), que si realizaba mejor la segmentación y era más rápido para procesar las imágenes.

4.1.2. Entrenamiento y ajuste del modelo

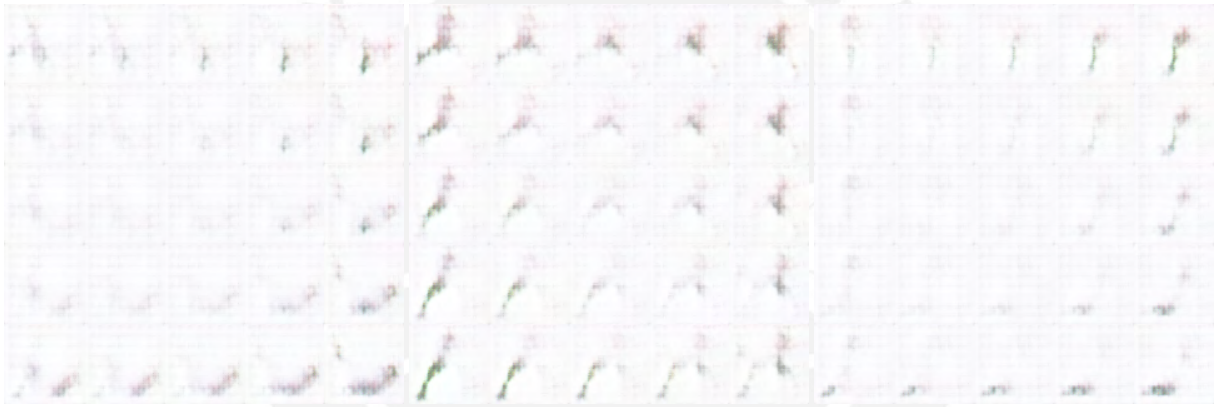
El modelo InfoGAn (Chen y cols., 2016) fue desarrollado con una configuración que se adapta al conjunto de datos, que inicialmente empieza con ruido de datos con dimensión de 128. El entrenamiento se realizó en 100 épocas con micro lotes de 32. Para el modelo del generador se definió una tasa de aprendizaje de 0.002, el modelo discriminador una tasa de aprendizaje de 0.0002 y el modelo auxiliar de 0.0001.

El modelo se entrenó con los 2 grupos de datos, uno de 3600 y el otro de 15 000 imágenes. El primero con imágenes de una misma persona con la finalidad de realizar la experimentación en tiempos cortos para definir los hiperparámetros y tener resultados preliminares. Esto sirvió de base para el segundo grupo de imágenes de personas diferentes. En ambos casos con posturas diferentes en cada acción. El entrenamiento total de ambos grupos de imágenes duró entre 4 y 5 horas. Se hizo uso de Google Colaboratory para realizar la experimentación en ambos. Este cuenta con GPU 1xTesla K80, compute 3.7, having 2496 CUDA cores , 12GB GDDR5 VRAM Y CPU 1xsingle core hyper threaded Xeon Processors @2.3Ghz i.e(1 core, 2 threads)

Se muestra en la Figura 4.2: y Figura 4.3: como inicia el aprendizaje por cada tipo de acción, en estas imágenes se visualiza cómo el modelo separa correctamente el tipo de imagen, y como este cambia si se modifica el espacio latente.

(a) Abdomen (b) Cardio (c) Glúteos

Figura 4.2: : 1º Grupo de Imágenes - Espacio latente epoca 1



(a) Abdomen (b) Cardio (c) Glúteos

Figura 4.3: : 2º Grupo de Imágenes - Espacio latente epoca 1

En la Figura 4.4: muestra imágenes generadas por clase, la primera columna muestra la clase de cardio, la segunda columna la clase de abdomen y la tercera columna la clase de glúteos, cada fila es una muestra de cada clase. Se puede apreciar que en la primera época existe mucho ruido en las imágenes, no se puede distinguir una diferencia entre los 3 tipos de ejercicio.

(a) Imágenes de una persona (b) Imágenes de varias personas

Figura 4.4: : Aprendizaje clasificado época 1

En la Figura 4.5: y Figura 4.6: se observa una mejora en el aprendizaje por cada tipo de acción, se empieza a diferenciar las imágenes de cada tipo. También se nota una secuencia de imágenes en cada clase de acción.

(a) Abdomen (b) Cardio (c) Glúteos

Figura 4.5: : 1º Grupo de Imágenes - Espacio latente época 50

(a) Abdomen (b) Cardio (c) Glúteos

Figura 4.6: : 2º Grupo de Imágenes - Espacio latente época 50

Se muestra en la Figura 4.7: como mejora el aprendizaje en la época 50, generando las clases específicas y aleatorias. Se puede apreciar que existe una notable mejora en las imágenes, en este caso, se observa que el modelo entiende la diferencia entre cada clase de acción.

(a) Clases específicas (b) Clases Aleatorias

Figura 4.7: : Aprendizaje clasificado época 50

Se muestra en la Figura 4.8: y Figura 4.9: como mejora el aprendizaje por cada tipo de acción, en la época 100, se nota una diferencia clara entre cada clase de acción.

(a) Abdomen (b) Cardio (c) Glúteos

Figura 4.8: : 1º Grupo de Imágenes - Espacio latente época 100

(a) Abdomen (b) Cardio (c) Glúteos

Figura 4.9: : 2º Grupo de Imágenes - Espacio latente época 100

En la época 100 en la Figura 4.10: se observa finalmente que el modelo si entiende la diferencia entre cada clase de acción, pero que en ciertos momentos no logra concebir a la persona completa.

(a) Clases específicas (b) Clases Aleatorias

Figura 4.10: : Aprendizaje clasificado época 100

4.2. Evaluación de Rendimiento del modelo

En la sección de Metodología se revisó varias medidas cualitativas y cuantitativas para evaluar los modelos de aprendizaje GAN. Se decidió usar en esta ocasión la Puntuación Inicial (Barratt y Sharma, 2018) para evaluar el desempeño del modelo InfoGAN, que evalúa la calidad de imagen. Es decir, si las imágenes se parecen a un objeto especifico y la diversidad de imágenes, si se genera una amplia gama de objetos.

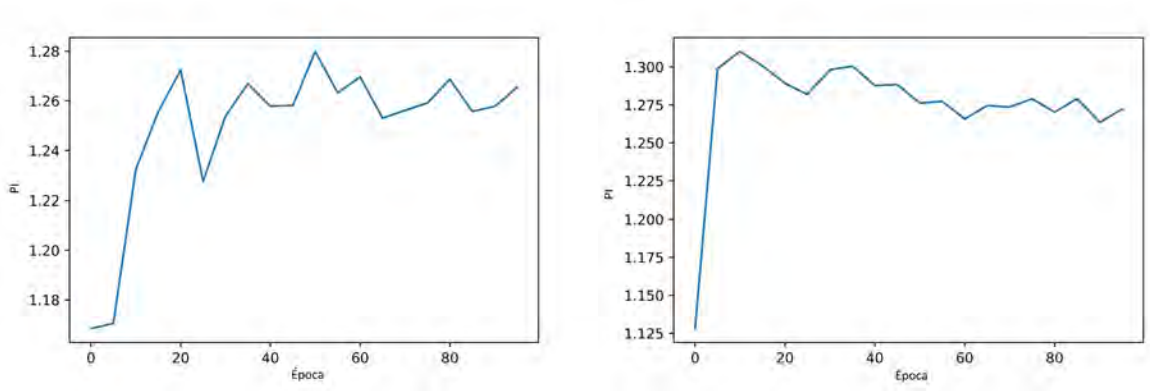
La puntuación inicial (Barratt y Sharma, 2018) tiene un valor más bajo de 1 y un máximo igual al número de clases respaldadas por nuestro modelo, en este caso seria 3. Tener en cuenta que las imágenes se dividen en grupos y la puntuación inicial se calcula en cada grupo de imágenes.

El cálculo de la puntuación inicial en un grupo de imágenes, implica primero utilizar el modelo inception v3 para calcular la probabilidad condicional para cada imagen $(p(y|x))$. Además, la probabilidad marginal se calcula como el promedio de las probabilidades condicionales de las imágenes del grupo $(p(y))$. Por otra parte, la divergencia KL se calcula para cada imagen como la probabilidad condicional multiplicada por el logaritmo de la probabilidad condicional menos el logaritmo de la probabilidad marginal. En esta sección presentamos los resultados del experimento para el tercer objetivo de investigación.

En la Figura 4.11: , se muestra la curva de evaluación de rendimiento por Puntuación Inicial, a lo largo del entrenamiento de cada época para cada grupo de imágenes. En el primer grupo, se manifiesta una subida en la curva hasta la época 20, luego existe

una baja hasta la época 25, para luego tener una tendencia de subida y bajada hasta la época 100. A pesar de ello, al final de la curva, se aprecia que posiblemente con más épocas, el puntaje subiría aún más, obteniendo como máximo puntaje de evaluación de 1.28. En el segundo grupo, se visualiza una rápida subida en la curva, hasta la época 10, para luego tener una tendencia a la baja de manera progresiva, obteniendo como máximo puntaje de evaluación de 1.30. Si comparamos las dos imágenes, se aprecia que, en el primer grupo, existe mayor diferencia entre los puntajes de cada época, en cambio, en el segundo grupo, la diferencia es menor.

Como resultado no se logró, estar cerca del puntaje máximo, debido a que algunas imágenes de entrenamiento tenían algo de ruido y falto diversidad de imágenes, de igual modo que sucedió en la investigación de generación de rostros (Z. Liu y cols., 2015) realizado por Liu. A pesar de los puntajes obtenidos, las imágenes generadas por modelo InfoGAN, muestra que el modelo si diferencia las clases de ejercicios que se utilizaron para esta investigación, aunque las imágenes no siempre muestren correctamente las piernas, los brazos y la cabeza.



(a) 1º Grupo de Imágenes (b) 2º Grupo de Imágenes

Figura 4.11: : Gráfico de Puntuación Inicial (PI)

Capítulo V

Conclusiones y Futuros Trabajos

En este capítulo se describen los principales hallazgos realizados como producto de la presente investigación, y mencionamos algunas recomendaciones para posibles investigaciones futuras.

5.1. Conclusiones

La generación de imágenes sintéticas es un problema complejo. El presente trabajo obtuvo datos de ejercicio de abdomen, cardio y glúteo, obteniendo un total de 18900 imágenes, entre el primer grupo de imágenes de 3900 y el segundo de 15000 imágenes, para que un modelo GAN en este caso, una InfoGAN, genere imágenes de manera específica entendiendo internamente la diferencia entre estas clases de acciones. En este caso se realizó una evaluación de rendimiento por Puntuación Inicial (PI), obteniendo puntaje máximo de 1.28 para el primer grupo de 3900 imágenes y un puntaje máximo de 1.3 para el segundo grupo de 15000 imágenes.

Aunque para el ojo humano se logró el objetivo principal para ambos grupos de imágenes. Para la función de evaluación de rendimiento, no fue de la misma manera, esta se vio afectada por la limitante en diversidad de imágenes y cierto ruido en algunas imágenes. Esto se debe a que a pesar de que existen muchas bases de datos de imágenes en el mundo, no necesariamente existe un grupo de imágenes sobre la investigación que se quiere realizar, es por ello que los datos utilizados en este proyecto fueron recolectados por cuenta propia a través de videos. Los resultados experimentales muestran que el modelo seleccionado, si logra diferenciar y separar las características de cada clase de imagen, y que puede generar estas imágenes a pedido, y no aleatoriamente como en un modelo GAN convencional. Asimismo, en la sección de entrenamiento y ajuste del modelo, donde se muestran grupos de imágenes para cada ejercicio modificando el espacio latente. Se aprecia para cada ejercicio, que es posible generar secuencias cortas de ejercicio de glúteo, cardio y abdomen, debido a que el modelo se entrenó con secuencia de posturas para cada ejercicio.

5.2. Trabajos Futuros

En esta investigación, se presenta la generación de imágenes de una persona realizando una acción específica. Luego de evaluar los resultados finales, se recomienda, obtener una mayor cantidad de imágenes de personas realizando ejercicios a fin de mejorar el entrenamiento del modelo. Asimismo, es importante mejorar el proceso de segmentación de la persona para cada imagen, desarrollando un modelo de segmentación de imágenes, de personas realizando un ejercicio, con la finalidad de mejorar el entrenamiento del modelo, utilizando imágenes que representen lo mejor posible una acción determinada. Luego de resolver estos dos puntos con respecto a los datos,

sería interesante, desarrollar otros modelos como la CGAN o DCGAN, ya que posiblemente, otros modelos tengan mejor resultado. Finalmente implementar otra métrica de rendimiento, como la distancia de Fréchet (FID).

Bibliografía

ADNI. (2020, diciembre). *The Alzheimer's Disease Neuroimaging Initiative.* Autor. Descargado de http://adni.loni.usc.edu/

Aria, A. (2019, diciembre). *Performance-comparison-of-GAN-on-cifar-10.* AliceAria. Descargado de https://github.com/AliceAria/Performance-comparison-of-GAN-on-cifar-10

Barratt, S., y Sharma, R. (2018). A Note on the Inception Score. *InceptionScore*. Descargado de http://arxiv.org/abs/1801.01973

Bienestar, A. (2020, diciembre). *Los canales de fitness en Youtube más populares este año.* Autor. Descargado de https://www.abc.es/bienestar/fitness/

Bolya, D., Zhou, C., Xiao, F., y Lee, Y. J. (2019). YOLACT: Real-time instance segmentation. *Proceedings of the IEEE International Conference on Computer Vision*, *2019-October*, 9156–9165.

Bradski, G. (2000). The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.

Calcagni, L. R. (2020). Redes Generativas Antagónicas y sus aplicaciones. , 72.

Celik, O. (2018, 09). A research on machine learning methods and its applications.

Chandak, V., Saxena, P., Pattanaik, M., y Kaushal, G. (2019a). Semantic Image Completion and Enhancement using Deep Learning. *2019 10th International Conference on Computing, Communication and Networking Technologies, ICCCNT*

2019. doi: 10.1109/ICCCNT45670.2019.8944750

Chandak, V., Saxena, P., Pattanaik, M., y Kaushal, G. (2019b). Semantic Image Completion and Enhancement using Deep Learning. *2019 10th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2019*. doi: 10.1109/ICCCNT45670.2019.8944750

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., y Abbeel, P. (2016). InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. *Advances in Neural Information Processing Systems*, 2180–2188.

Clark, A., Donahue, J., y Simonyan, K. (2019). Adversarial Video Generation on Complex Datasets. , 1–21. Descargado de http://arxiv.org/abs/1907.06571

Darebee. (2021, diciembre). *Darebee.* ADNI. Descargado de https://darebee.com/

Dong, H., Yu, S., Wu, C., y Guo, Y. (2017). Semantic image synthesis via adversarial learning. En *2017 ieee international conference on computer vision (iccv)* (p. 5707-5715).

Dong, S., y Zhang, Z. (2020). Joint optimization of cyclegan and cnn classifier for covid-19 detection and biomarker localization. En *2020 ieee international conference on progress in informatics and computing (pic)* (p. 112-118). doi: 10.1109/PIC50277.2020.9350813

Ghahramani, Z. (2004). Unsupervised learning. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *3176*, 72–112.

Goodfellow, I. (2016). NIPS 2016 Tutorial: Generative Adversarial Networks. *GAN*. Descargado de http://arxiv.org/abs/1701.00160

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., …

Bengio, Y. (2014, 06). Generative adversarial networks. *Advances in Neural Information Processing Systems*, *3*.

Günel, M., Erdem, E., y Erdem, A. (2018). Generating person images based on attributes. En *2018 26th signal processing and communications applications conference (siu)* (p. 1-4).

Han, C., Hayashi, H., Rundo, L., Araki, R., Shimoda, W., Muramatsu, S., … Nakayama, H. (2018). Gan-based synthetic brain mr image generation. En *2018 ieee 15th international symposium on biomedical imaging (isbi 2018)* (p. 734-738).

He, K., Gkioxari, G., Dollár, P., y Girshick, R. (2020). Mask R-CNN. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *42*(2), 386–397.

Heusel, M., Jan, L. G., y Hochreiter, S. (2017). GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. (Nips).

Ioffe, S., y Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *32nd International Conference on Machine Learning, ICML 2015*, *1*, 448–456.

Karras, T., Aila, T., Laine, S., y Lehtinen, J. (2018). Progressive growing of GANs for improved quality, stability, and variation. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, 1–26.

Karras, T., Laine, S., y Aila, T. (2019). A style-based generator architecture for generative adversarial networks. En *2019 ieee/cvf conference on computer vision and pattern recognition (cvpr)* (p. 4396-4405).

Karras, T., Laine, S., y Aila, T. (2019). A style-based generator architecture for generative adversarial networks. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2019-June*, 4396–4405.

Keys, R. (1981). Cubic convolution interpolation for digital image processing. *IEEE*

Transactions on Acoustics, Speech, and Signal Processing, *29*(6), 1153-1160.

Kola, R. (2019). Generation of synthetic plant images using deep learning architecture. (June).

Kotsiantis, S. (2007, 10). Supervised machine learning: A review of classification techniques. *Informatica (Ljubljana)*, *31*.

Krizhevsky, A., Nair, V., y Hinton, G. (s.f.). Cifar-10 (canadian institute for advanced research).

Kumar, N., Belhumeur, P. N., Biswas, A., Jacobs, D. W., Kress, W. J., Lopez, I. C., y Soares, J. V. B. (2012). Leafsnap: A computer vision system for automatic plant species identification. En A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, y C. Schmid (Eds.), *Computer vision – eccv 2012* (pp. 502–516). Berlin, Heidelberg: Springer Berlin Heidelberg.

LeCun, Y., y Cortes, C. (2010). MNIST handwritten digit database. http://yann.lecun.com/exdb/mnist/. Descargado 2016-01-14 14:24:11, de http://yann.lecun.com/exdb/mnist/

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., … Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *8693 LNCS*(PART 5), 740–755.

Liu, M. Y., Huang, X., Yu, J., Wang, T. C., y Mallya, A. (2021). Generative Adversarial Networks for Image and Video Synthesis: Algorithms and Applications. *Proceedings of the IEEE*, *109*(5), 839–862.

Liu, Z., Luo, P., Wang, X., y Tang, X. (2015, December). Deep learning face attributes in the wild. En *Proceedings of international conference on computer vision (iccv).*

Maas, A. L. (2013). Rectifier nonlinearities improve neural network acoustic models..

Mackin, A., Zhang, F., y Bull, D. R. (2019). A study of high frame rate video formats. *IEEE Transactions on Multimedia*, *21*(6), 1499-1512.

Mirza, M., y Osindero, S. (2014). Conditional Generative Adversarial Nets. , 1–7. Descargado de http://arxiv.org/abs/1411.1784

Prakash, C. D., y Karam, L. J. (2021). It Gan do better: GaN-based detection of objects on images with varying quality. *IEEE Transactions on Image Processing*, *30*, 9220–9230. doi: 10.1109/TIP.2021.3124155

Radford, A., Metz, L., y Chintala, S. (2015, 11). Unsupervised representation learning with deep convolutional generative adversarial networks.

Recognizer, D. (2021). *Conditional generative adversarial network.* kaggle. Descargado de https://www.kaggle.com/arpandhatt/conditional-generative-adversarial-network

Sage, A., Timofte, R., Agustsson, E., y Gool, L. V. (2018). Logo Synthesis and Manipulation with Clustered Generative Adversarial Networks. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 5879–5888. doi: 10.1109/CVPR.2018.00616

Salau, A. O., y Jain, S. (2019). Feature extraction: A survey of the types, techniques, applications. En *2019 international conference on signal processing and communication (icsc)* (p. 158-164).

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., y Chen, X. (2016, 06). Improved techniques for training gans.

Schmidhuber, J. (2015). Deep Learning in neural networks: An overview. *Neural Networks*, *61*, 85–117.

Sendik, O., Lischinski, D., y Cohen-Or, D. (2020). Unsupervised K-modal styled content

generation. *ACM Transactions on Graphics*, *39*(4).

Subramanian, S., Rajeswar, S., Dutil, F., Pal, C., y Courville, A. (2017). Adversarial generation of natural language. *Proceedings of the 2nd Workshop on Representation Learning for NLP, Rep4NLP 2017 at the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017*(2016), 241–251. doi: 10.18653/v1/w17-2629

Sun, R., Fang, T., y Schwing, A. (2020). Towards a Better Global Loss Landscape of GANs. (NeurIPS). Descargado de http://arxiv.org/abs/2011.04926

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., y Wojna, Z. (2015). Rethinking the inception architecture for computer vision. *CoRR*, *abs/1512.00567*. Descargado de http://arxiv.org/abs/1512.00567

Tan, W. R., Chan, C. S., Aguirre, H. E., y Tanaka, K. (2018). ArtGAN: Artwork synthesis with conditional categorical GANs. *Proceedings - International Conference on Image Processing, ICIP*, *2017-September*, 3760–3764.

Tan, W. R., Chan, C. S., Aguirre, H. E., y Tanaka, K. (2019). Improved artgan for conditional synthesis of natural image and artwork. *IEEE Transactions on Image Processing*, *28*(1), 394-409.

Tang, H., Bai, S., Zhang, L., Torr, P. H., y Sebe, N. (2020). XingGAN for Person Image Generation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *12370 LNCS*, 717–734.

Torres, G. V. (2017, noviembre). *Qué es deep learning.* Openwebinars. Descargado de https://openwebinars.net/blog/que-es-deep-learning/

Yu, S., Dong, H., Liang, F., Mo, Y., Wu, C., y Guo, Y. (2019). Simgan: Photo-realistic semantic image manipulation using generative adversarial networks. En *2019*

ieee international conference on image processing (icip) (p. 734-738).

Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., y Metaxas, D. (2017, 10). Stackgan++: Realistic image synthesis with stacked generative adversarial networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *PP*.