

Tecnológico de Costa Rica  
Escuela de Ingeniería Electrónica  
Programa de Licenciatura en Ingeniería Electrónica



## **Estrategia para la síntesis de imágenes de microscopía óptica digital utilizando técnicas de aprendizaje profundo**

Informe de Trabajo Final de Graduación para optar por el título de  
Ingeniero en Electrónica con el grado académico de Licenciatura

Olivier Josué Ramírez Morales

Cartago, 17 de agosto, 2021



Esta obra está bajo una [licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional](#).

Declaro que el presente documento de tesis ha sido realizado enteramente por mi persona, utilizando y aplicando literatura referente al tema e introduciendo conocimientos y resultados experimentales propios.

En los casos en que he utilizado bibliografía he procedido a indicar las fuentes mediante las respectivas citas bibliográficas. En consecuencia, asumo la responsabilidad total por el trabajo de tesis realizado y por el contenido del presente documento.

Olivier Josué Ramírez Morales

Cartago, 17 de agosto de 2021



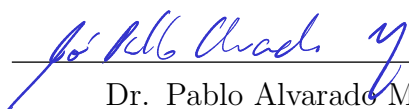
Céd: 6-0428-0299

Instituto Tecnológico de Costa Rica  
Escuela de Ingeniería Electrónica  
Proyecto de Graduación  
Acta de Aprobación

Defensa de Proyecto de Graduación  
Requisito para optar por el título de Ingeniero en Electrónica  
Grado Académico de Licenciatura

El Tribunal Evaluador aprueba la defensa del proyecto de graduación denominado *Estrategia para la síntesis de imágenes de microscopía óptica digital utilizando técnicas de aprendizaje profundo*, realizado por el señor Olivier Josué Ramírez Morales y, hace constar que cumple con las normas establecidas por la Escuela de Ingeniería Electrónica del Instituto Tecnológico de Costa Rica.

Miembros del Tribunal Evaluador

  
\_\_\_\_\_  
Ing. Juan José Montero Rodríguez  
Profesor Lector  
\_\_\_\_\_  
Ing. Aníbal Ruiz Barquero  
Profesor Lector  
\_\_\_\_\_  
Dr. Pablo Alvarado Moya  
Profesor Asesor

Cartago, 25 de agosto de 2021

# Resumen

El presente proyecto propone una estrategia de síntesis de imágenes para proporcionar una base de datos de mayor calidad al entrenamiento de algoritmos con modelos de aprendizaje profundo y así superar la escasez de datos. Este proyecto se enfoca en el aumentado de imágenes de microscopía óptica digital.

Se ha explorado un método de generación sin etiquetas basado en redes generativas adversarias y se han analizado métricas existentes para medir el desempeño de los modelos entrenados, así como la fidelidad de las imágenes generadas, mediante matrices de confusión y realismo perceptivo, respectivamente.

Partiendo de un proyecto existente llamado CycleGAN se desarrollan tres experimentos en los cuales se han entrenado las redes con una nueva base de datos generando imágenes realistas de microscopía. Además, en cada experimento se modifica el set de entrenamiento.

**Palabras clave:** aumentado de datos, aprendizaje profundo, nematodos, GAN, microscopía óptica.

# Abstract

The present project proposes an image synthesis strategy to provide a higher quality database for model training with deep learning algorithms and thus overcome the scarcity of data. This project focuses on the augmentation of digital optical microscopy images.

A labelless generation method based on adversarial generative networks has been explored and existing metrics have been analyzed to measure the performance of the trained models, as well as the fidelity of the generated images, using confusion matrices and perceptual realism, respectively.

Starting from an existing project called CycleGAN, three experiments are developed in which the networks have been trained with a new database, generating realistic microscopy images. In addition, in each experiment the training set is modified.

**Keywords:** data augmentation, deep learning, nematodes, GAN, optical microscopy.

*A mis queridos padres*

# Agradecimientos

El primera instancia, darle gracias a Dios, a la Virgen de Fátima y al Padre Pío de Pietrelcina, quienes han sido los bastiones en mi vida.

Al Dr.-Ing. Pablo Alvarado Moya, gracias por su paciencia, por compartir sus conocimientos de manera profesional e invaluable, y por su dedicación y disposición en la asesoría brindada.

Finalmente a mis padres Olivier Ramírez y Daisy Morales por ser ejemplos de superación.

Olivier Josué Ramírez Morales

Cartago, 17 de agosto de 2021



# Índice general

Índice de figuras	III
Índice de tablas	V
<b>1. Introducción</b>	<b>1</b>
1.1. Objetivos y estructura del documento . . . . .	3
<b>2. Marco teórico</b>	<b>5</b>
2.1. Aumentado de datos . . . . .	5
2.1.1. Mezcla de imágenes . . . . .	6
2.1.2. Aumentos de espacio de características . . . . .	7
2.1.3. Redes adversarias generativas . . . . .	7
2.1.4. CycleGAN . . . . .	9
2.1.5. Transferencia de estilo neuronal . . . . .	12
2.1.6. Resumen de aumentado de datos en literatura . . . . .	12
2.2. Modelo de forma . . . . .	12
2.3. Procesamiento de imágenes . . . . .	15
2.4. Métricas . . . . .	16
2.4.1. Realismo perceptivo . . . . .	16
2.4.2. Interpretabilidad semántica . . . . .	17
2.4.3. Consistencia de ciclo . . . . .	18
2.4.4. Matriz de confusión . . . . .	18
2.4.5. Optimalidad Pareto . . . . .	19
<b>3. Sistema de aumentado por síntesis de imagen</b>	<b>20</b>
3.1. Generación de siluetas . . . . .	21
3.2. Procesamiento de datos . . . . .	21
3.2.1. Limpieza de datos . . . . .	21
3.2.2. Incorporación de características de fondo y ruido . . . . .	22
3.2.3. Combinatoria de bases de datos . . . . .	23
3.2.4. Aumentado básico . . . . .	23
3.3. Aumentado de datos . . . . .	24
3.3.1. Red generativa . . . . .	26
3.3.2. Red discriminadora . . . . .	26

---

3.3.3. Entrenamiento y alternativas de evaluación . . . . .	27
<b>4. Resultados y análisis</b>	<b>29</b>
4.1. Datos de entrenamiento y prueba . . . . .	29
4.1.1. Experimento 1 . . . . .	30
4.1.2. Experimento 2 . . . . .	33
4.1.3. Experimento 3 . . . . .	36
4.2. Análisis comparativo . . . . .	39
4.2.1. Evaluación de discriminabilidad . . . . .	45
4.2.2. Evaluación de traducciones . . . . .	52
<b>5. Conclusiones</b>	<b>53</b>
<b>Bibliografía</b>	<b>55</b>

# Índice de figuras

1.1. Ejemplar de nematodo . . . . .	2
1.2. Diagrama de bloques para la <i>estrategia de síntesis de imágenes</i> . . . . .	3
2.1. Ejemplo de aumentado tradicional a imagen de un perro [21] . . . . .	6
2.2. Diagrama del método mezcla de imágenes [24] . . . . .	6
2.3. Ejemplos de imágenes de mezcla entre un gato y una llama [14] . . . . .	7
2.4. Diagrama del método aumentos de espacio de características [14] . . . . .	7
2.5. Instancias interpoladas en el espacio de características del carácter @ . . . . .	8
2.6. Diagrama del modelo GAN [14] . . . . .	8
2.7. Datos de entrenamiento emparejados y sin emparejar . . . . .	10
2.8. Ejemplo de imágenes sin emparejar con contenido común y estilo distinto . . . . .	10
2.9. Ejemplo de un aumentado en datos de tomografía computarizada . . . . .	11
2.10. Transferencia del estilo de una pintura a una fotografía . . . . .	12
2.11. Clasificación de aumentados de datos en imágenes . . . . .	13
2.12. Silueta de un nematodo segmentado por 40 hitos de frontera . . . . .	13
2.13. Proceso de obtención de los subdominios de forma . . . . .	15
2.14. Caminatas aleatorias entre subdominios vecinos [10] . . . . .	15
2.15. Ejemplo de matriz de confusión . . . . .	18
2.16. Ejemplo de frente de Pareto . . . . .	19
3.1. Diagrama de la <i>estrategia de síntesis de imágenes</i> . . . . .	20
3.2. Ejemplo de silueta generada por el modelo de forma . . . . .	21
3.3. Diagrama de flujo de las etapas del procesamiento de datos . . . . .	22
3.4. Ejemplo de dato crudo de microscopía ruidoso . . . . .	22
3.5. Ejemplo de silueta modificada . . . . .	23
3.6. Diagrama de bloques de la arquitectura CycleGAN (inspirado en [34]) . . . . .	25
3.7. Esquema de las funciones de pérdida (inspirado en [21]) . . . . .	25
3.8. Arquitectura de la red generadora [21] . . . . .	26
3.9. Parches de la red discriminadora [21] . . . . .	26
3.10. Adaptación del modelo CycleGAN a este proyecto . . . . .	27
3.11. Diagrama de la función de pérdida de identidad . . . . .	28
4.1. Selección de datos de microscopía . . . . .	29
4.2. Ejemplos de clases $S$ y $M$ de la base de datos 1 . . . . .	30
4.3. Imágenes sintetizadas durante el entrenamiento del modelo del experimento 1 . . . . .	31

4.4. Imágenes sintetizadas evaluando el modelo del experimento 1 . . . . .	32
4.5. Ejemplos de clases $S$ y $M$ de la base de datos 2 . . . . .	33
4.6. Imágenes sintetizadas durante el entrenamiento del modelo del experimento 2	34
4.7. Imágenes sintetizadas evaluando el modelo entrenado del experimento 2 . .	35
4.8. Ejemplos de clases $S$ y $M$ de la base de datos 3 . . . . .	36
4.9. Ejemplos de los tipos de aumentado básico . . . . .	37
4.10. Imágenes sintetizadas durante el entrenamiento del modelo del experimento 3	38
4.11. Imágenes sintetizadas evaluando el modelo entrenado del experimento 3 . .	39
4.12. Pérdida del discriminador para los experimentos 1, 2 y 3 . . . . .	40
4.13. Pérdida del generador para los experimentos 1, 2 y 3 . . . . .	41
4.14. Pérdida del generador filtrada para los experimentos 1, 2 y 3 . . . . .	42
4.15. Pérdida de consistencia de ciclo para los experimentos 1, 2 y 3 . . . . .	43
4.16. Pérdida de identidad para los experimentos 1, 2 y 3 . . . . .	44
4.17. Espacio precisión/exhaustividad y frente de pareto del experimento 1 . . .	47
4.18. Espacio precisión/exhaustividad y frente de pareto del experimento 2 . . .	47
4.19. Espacio precisión/exhaustividad y frente de pareto del experimento 3 . . .	48
4.20. Matrices de confusión del modelo discriminativo . . . . .	49
4.21. Salida del discriminador para 50 imágenes reales . . . . .	50
4.22. Salida del discriminador para 50 imágenes sintetizadas . . . . .	51
4.23. Imágenes de evaluación de consistencia de ciclo . . . . .	52

# Índice de tablas

2.1. Resumen, ventajas y desventajas de métodos de aumentado de datos . . .	14
2.2. Operaciones para el procesamiento de imagen . . . . .	16
3.1. Bases de datos conformadas por imágenes de siluetas y microscopía . . . .	23
3.2. Selección de las alternativas propuestas para el aumentado de datos . . . .	24
3.3. Selección de las alternativas propuestas para la evaluación de resultados . .	28
4.1. Umbrales de decisión del experimento 1 . . . . .	45
4.2. Umbrales de decisión de los experimentos 2 y 3 . . . . .	46
4.3. Métricas de exactitud y precisión de los modelos discriminativos . . . . .	50
4.4. Métrica SSIM para evaluar el nivel de consistencia de ciclo alcanzado . . .	52

# Lista de abreviaciones

## Abreviaciones

AMT	Amazon Mechanical Turk
CNN	Redes neuronales convolucionales
GAN	Redes generativas adversarias
SSIM	Índice de similitud estructural
VAE	Autocodificadores variacionales

# Capítulo 1

## Introducción

La economía en los países centroamericanos depende en gran medida de la agricultura, siendo esta una actividad económica que aporta cerca del 8,5 % del PIB de la región y genera, en promedio, el 19,8 % del empleo [1]. En 2016 el desempeño del sector agrícola en las exportaciones de Centroamérica destaca en productos como el café, aceite de palma, azúcar, banano, frutas, azúcar y crustáceos [2]. En Costa Rica, por ejemplo, hay un aproximado de 140,000 productores agropecuarios, los cuales, se dedican a la producción de granos, como maíz, frijol y arroz, cultivos de hortalizas, café y frutas [3]. El aprovechamiento de estos productos en el mercado local e internacional requiere investigación que permita aplicar nuevas tecnologías en la optimización de la producción [2], controlando plagas y contrarrestando los efectos del cambio climático [4].

Los nematodos edáficos son un grupo de invertebrados pseudocelomados y vermiformes que presentan abundancias de varios millones de individuos por metro cuadrado [5]. Los nematodos son diversos con diferentes funciones y presentan atributos que les convierte en valiosas herramientas para ser considerados indicadores biológicos y ecológicos. En la figura 1.1 se muestra un ejemplo de un nematodo. Los nematodos herbívoros que son parásitos de cultivos, también conocidos como fitoparásitos [4] se alimentan de las raíces de las plantas, incluso tallos y hojas [5], y son los causantes de un 12 % a 15 % de pérdidas del total de producción de las cosechas [6]. En Costa Rica hay numerosos estudios de nematodos en cultivos de interés económico como hortalizas, arroz y árboles frutales [7]. La mayoría de nematodos son microscópicos [8] de menos de 1 mm de largo [5], por lo que el desarrollo de microscopios ópticos digitales favoreció el estudio de sus comportamientos permitiendo dar con soluciones como utilizar otros nematodos beneficiosos para combatir a los dañinos [4].

Los nematólogos recurren a la ciencia de la morfología para lograr la identificación de diferentes clases de nematodos. Según [9], los nematodos que viven en el agua tienen colas largas que les ayudan a nadar bien, mientras los que viven parcialmente en las raíces de las plantas siempre tienen una cola corta. Otros presentan un movimiento particular. Los *Cricos* se mueven mediante contracciones y discontracciones de su cuerpo y el *Desmocolex* sp como un gusano medidor. Después de muertos algunos nematodos como el *Rotylenchus*



**Figura 1.1:** Ejemplar de nematodo

y el *Helicotylenchus* adquieren una forma de espiral. Además, conocer el funcionamiento de sus estructuras brinda valiosa información para predecir el comportamiento alimenticio. El *Longidorella* sp tiene un estilete en el sistema digestivo de 40 a 50  $\mu\text{m}$  de largo, lo cual indica que se alimentan de raíces de plantas.

El conteo e identificación de nematodos permite inferir información relacionada con estimación de densidades, estados de resistencia o quiescencia (estados que les permiten sobrevivir inactivos en condiciones de crecimiento no favorables), otros estados sensibles a los cambios ambientales, así como determinar los efectos de métodos de control de plagas por ejemplo los efectos tóxicos de productos químicos [5]. Sin embargo, alcanzar estos resultados implica que el investigador procese manualmente cientos de imágenes volviendo la tarea inviable y presenta el inconveniente de un trabajo de laboratorio altamente costoso, ya que el análisis de una sola muestra de suelo, sustrato o raíz de un cultivo, obtenida con un microscopio en busca de nematodos puede tardar de entre 15 minutos hasta varias horas. Surge entonces la necesidad de contar con herramientas automáticas que realicen estas tareas tediosas para un mayor aprovechamiento del tiempo e investigador.

Se han implementado sistemas de visión por computador para segmentación, detección, adaptación de forma y clasificación de nematodos [4], [8], [10], [11], [12], [13]. No obstante, estos sistemas necesitan imágenes etiquetadas para su entrenamiento. Entre mayor sea la cantidad y diversidad de imágenes, mejores serán sus predicciones. La dificultad radica en etiquetar manualmente cada imagen, realizando anotaciones de dónde exactamente están los nematodos, razón por la cual sólo se cuenta con una base de datos anotadas de 3260 imágenes [8].

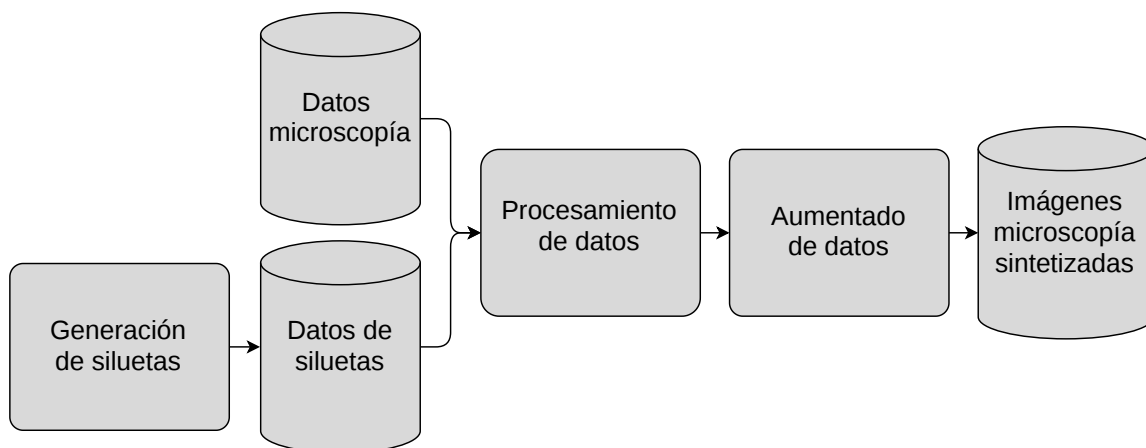
El aumento de datos es un método utilizado para solucionar el problema de datos limitados en tareas de visión por computador. Su principal función es mejorar el tamaño y la calidad de los conjuntos de datos de entrenamiento, de modo que se pueden construir mejores modelos de aprendizaje profundo con ellos [14]. Se basan en diversos algoritmos, desde manipulaciones de imagen básicas como transformación del espacio de color, borrado aleatorio y transformaciones geométricas.

Investigaciones realizadas en sistemas de segmentación de nematodos recurren al aumento de datos mediante manipulaciones básicas de rotaciones y traslaciones de muestras



de nematodos como de fondo [4], escalando y agregando corrupción aleatoria utilizando máscaras gaussianas [8]; sistemas de reconocimiento de especies de nematodos [15] usan giros aleatorios (vertical y horizontal) y el desenfoque gaussiano durante el entrenamiento; marcos para la detección de objetos con forma de gusano basados en redes neuronales convolucionales (CNN) emplean una corrección gamma para generar imágenes con diferente contraste [16]. Se necesita aún más variedad.

La solución desarrollada en este proyecto es una alternativa al aumento de datos tradicional y está basada en la implementación del método de traducción de imagen a imagen sin emparejar y la arquitectura CycleGAN propuesta por Zhu et al. [17], la cual está compuesta por dos redes adversarias generativas (GAN) construidas sobre la base de conceptos de las arquitecturas PatchGAN, U-Net y DCGAN. En esta arquitectura no existen datos de entrenamiento emparejados; solo se tiene una pila de imágenes de un estilo y otra pila de otro estilo. Así en una pila se introducen siluetas de nematodos (sintéticas) y en otra nematodos reales. Además, CycleGAN incorpora una función de pérdida de consistencia de ciclo que permite obtener resultados de transferencia de estilo neuronal y de esta manera, junto con las GAN, generar imágenes realistas de nematodos. La figura 1.2 ilustra las etapas generales de la investigación.



**Figura 1.2:** Diagrama de bloques para la *estrategia de síntesis de imágenes*

Un set de datos de siluetas es generado por medio de un modelo de forma que captura la forma verme intrínseca de los nematodos en imágenes digitales como se discute en la sección 2.2. La modificación de características en las siluetas, así como la limpieza de los datos de microscopía, se lleva a cabo usando técnicas de procesamiento de imágenes (sección 2.3). Finalmente, se sintetizan imágenes de microscopía por medio de un aumento de datos basado en el modelo de aprendizaje profundo CycleGAN (sección 2.1.4).

## 1.1. Objetivos y estructura del documento

El objetivo de este proyecto es implementar un sistema de procesamiento de imágenes que desarrolle un modelo neuronal convolucional para la síntesis de imágenes de micros-

copía anotadas con presencia de nematodos. En función de lo planteado, se muestra una metodología para preparar el set de datos, seleccionar e implementar el modelo neuronal convolucional para la transferencia de estilo, así como evaluar su desempeño en la generación de imágenes realistas de nematodos.

El documento se estructura de la siguiente forma: en el siguiente capítulo 2 se desarrolla un marco teórico con un análisis del estado del arte en las áreas de aumentado de datos para imágenes arbitrarias y los conceptos en los que esta investigación se fundamenta. En el capítulo 3 se detalla la estrategia desarrollada y el modelo implementado para la solución propuesta. En el capítulo 4 se exponen y analizan los resultados obtenidos. Finalmente, en el capítulo 5 se presentan las conclusiones y se sugieren trabajos futuros a desarrollar.

# Capítulo 2

## Marco teórico

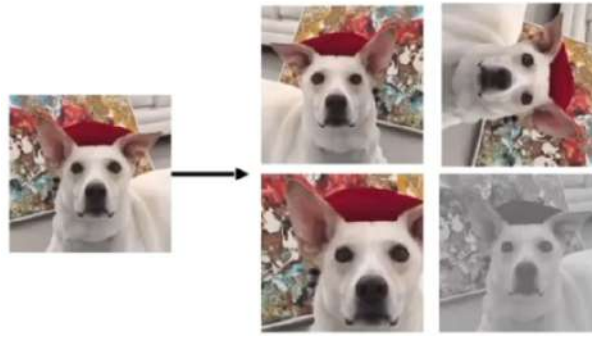
La cantidad de datos de entrenamiento es un factor crucial para la aplicación exitosa de modelos de aprendizaje profundo. Desafortunadamente, muchas de las tareas de visión por computador no cuentan con suficientes datos etiquetados, como el análisis de imágenes médicas debido a la rareza de las enfermedades, la privacidad del paciente, el requisito de los expertos médicos para el etiquetado y el gasto y esfuerzo manual necesarios [18]. El aumento de datos como herramienta para mejorar el tamaño y la calidad de los datos, es una solución eficaz para combatir desde la raíz el problema de sobreajuste en el aprendizaje profundo [14], una aplicación que demuestra su eficacia es el clasificador AlexNet [19], para la clasificación del conjunto de datos ImageNet. Además, según [20], el aumento de datos también es útil para tratar problemas de desequilibrio de clases: cuando no se tienen muchos ejemplos de una clase determinada.

### 2.1. Aumentado de datos

La idea básica del aumento de datos es generar un conjunto de datos sintéticos que cubra el espacio de entrada inexplorado mientras se mantienen las etiquetas correctas [19]. Suponiendo que se tiene la imagen de la figura 2.1 como parte de un conjunto de datos reales y la tarea es detectar si hay perros en imágenes usando un clasificador, pero el clasificador no tiene suficientes datos de entrenamiento, entonces lo que se puede hacer es aplicarle todo tipo de modificaciones a esa imagen y luego alimentar todas esas imágenes en el clasificador también como datos reales.

Según [14], las técnicas de aumento actuales consisten en la deformación de datos y el sobremuestreo. Los aumentados de deformación de datos han sido aplicados en CNN para la clasificación de dígitos escritos a mano [22] e imágenes del conjunto de datos ImageNet [23], logrando con ello reducir la tasa de error del modelo en más del 1 %.

Algunas de las técnicas por deformación se consideran manipulaciones básicas de imágenes ya que consisten de filtros, transformaciones geométricas y de color, borrado aleatorio y mezcla de imágenes; en adelante referido como aumento básico. Los aumentados por

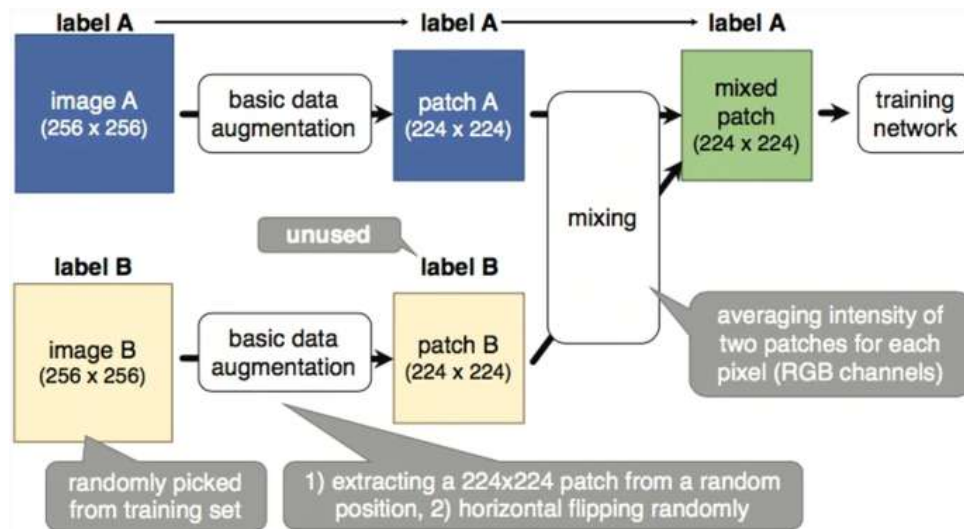


**Figura 2.1:** Ejemplo de aumentado tradicional a imagen de un perro [21]

sobremuestreo basados en aprendizaje profundo, como la transferencia de estilo neuronal y la traducción de imagen a imagen, son los únicos que crean instancias sintéticas para ser agregadas al conjunto de entrenamiento. Por supuesto que las manipulaciones básicas pueden combinarse con los enfoques de aprendizaje profundo.

### 2.1.1. Mezcla de imágenes

Esta técnica consiste en un emparejamiento de dos imágenes aplicando algún aumentado básico como recortes y volteos aleatorios, para luego promediar los valores de píxeles, por ejemplo para cada uno de los canales de RGB [24] lo que da como resultado una imagen mixta que tiene por etiqueta la misma que la primera imagen seleccionada al azar como puede apreciarse en la figura 2.2 mediante un diagrama de su arquitectura y en la figura 2.3 los posibles resultados. El método básico de mezcla de imágenes puede extenderse para sobremuestrear datos [14].



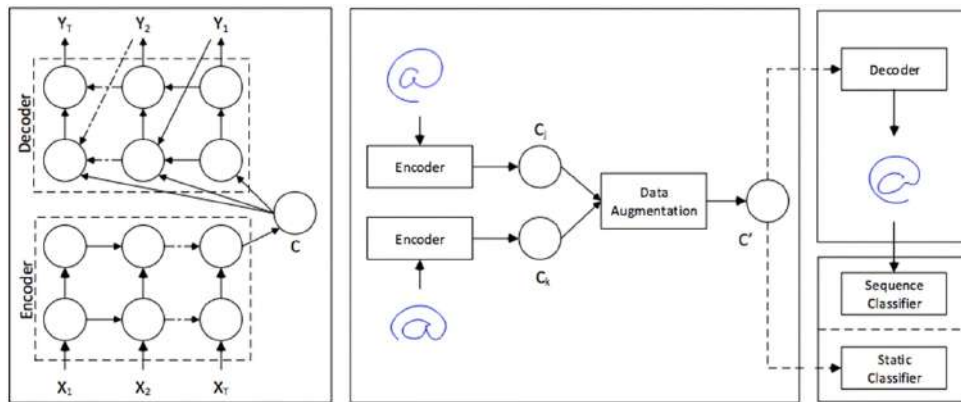
**Figura 2.2:** Diagrama del método mezcla de imágenes [24]



**Figura 2.3:** Ejemplos de imágenes de mezcla entre un gato y una llama [14]

### 2.1.2. Aumentos de espacio de características

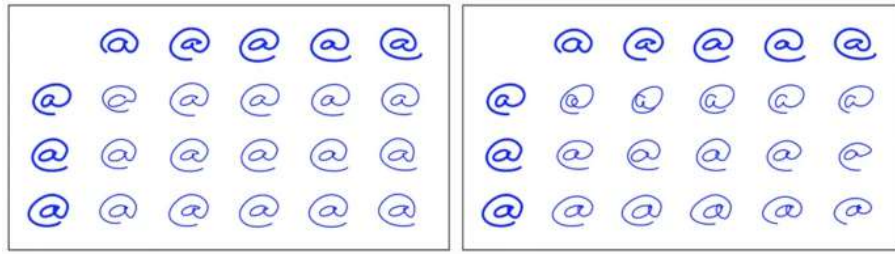
A diferencia de la mezcla de imágenes, el aumentado no se aplica en el espacio de entrada sino en el espacio de características. Las representaciones de dimensiones inferiores que se encuentran en capas de alto nivel de una CNN se conocen como espacio de características [14]. En este caso, se aprovecha la funcionalidad de los codificadores automáticos para mapear imágenes en representaciones vectoriales de baja dimensión a través de su bloque codificador. En dicha representación pueden aplicarse la adición de ruido, la interpolación o extrapolación [25]. Y así el siguiente bloque decodificador reconstruye estos vectores en la imagen original, como se aprecia en figura 2.4.



**Figura 2.4:** Diagrama del método aumentos de espacio de características [14]

### 2.1.3. Redes adversarias generativas

Las GAN se implementan mediante dos modelos: generativo y discriminativo, que son redes neuronales [18]. En [21] los modelos generativos aprenden a hacer una representación realista con características  $X$  dada alguna clase  $Y$  y/o ruido aleatorio, y modelan la siguiente distribución de probabilidad:



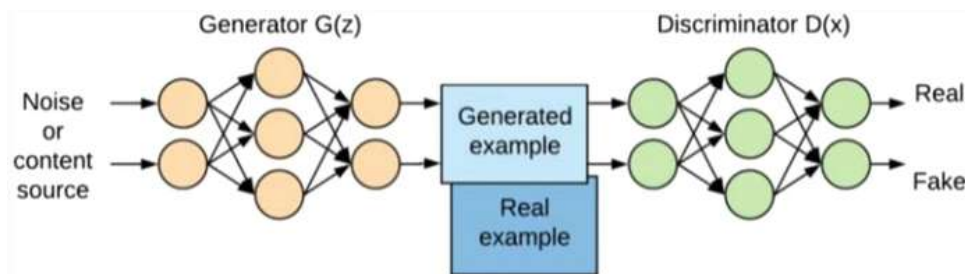
**Figura 2.5:** Ejemplos de instancias interpoladas en el espacio de características del carácter @ escrito a mano [14]

$$P(X|Y) \quad (2.1)$$

Los modelos discriminativos distinguen entre clases  $Y$  (clasificación) dado un conjunto de características  $X$  y modelan la siguiente distribución de probabilidad:

$$P(Y|X) \quad (2.2)$$

Un diagrama con los modelos se ilustra en la figura 2.6. La red generadora recibe como entrada ruido aleatorio, por ejemplo un vector con números reales y una clase opcional como puede ser la de un animal específico, y se entrena para generar imágenes que engañen a la red discriminadora. La red discriminadora estará mirando imágenes falsas (generadas) y reales, simultáneamente determina si son reales o falsas; tal como si un equipo de falsificadores (modelo generativo) intentara producir moneda falsa y utilizarla sin ser detectada, mientras que la policía (modelo discriminativo) intenta detectar la moneda falsa [26]. Estos modelos compiten entre sí, por lo que se les llama adversarios. De esta manera las GAN crean instancias artificiales a partir de un conjunto de datos de modo que conserven características similares al conjunto original [14], es decir, generan nuevos datos de entrenamiento, por lo que son denominados modelos generativos.



**Figura 2.6:** Diagrama del modelo GAN [14]

El entrenamiento de las redes generadoras presenta más complicaciones que el de las discriminadoras [18]. En la generación de imágenes pueden presentarse problemas de calidad de imágenes, diversidad o curva de aprendizaje. Algunas causas son las siguientes:

## Desbalance entre generador y discriminador

Según [21] ambos modelos deben mejorar juntos y deben mantenerse en niveles de habilidad similares desde el comienzo del entrenamiento. Si se tiene un discriminador que es superior al generador obtendrá predicciones diciendo que todos los ejemplos falsos son 100 % falsos. Esto no es útil para el generador porque no sabe cómo mejorar, no hay nada que le diga en qué dirección ir, no sabe de qué manera crecer y aprender. Mientras tanto, un generador que supera al discriminador obtendrá predicciones que indicarán que todas las imágenes generadas son 100 % reales. A esto se le llama el problema de desvanecimiento de gradiente [18].

## Colapso de modo

Un modo en una distribución de probabilidad de características es un área con una alta concentración de observaciones (picos). Por ejemplo, el valor medio en una distribución normal es el modo único en esa distribución, también pueden haber dos picos (bimodal) o múltiples picos (multimodal) [21]. En este sentido, los conjuntos de datos del mundo real tienen muchos modos relacionados con cada clase posible dentro de ellos.

El colapso de modo ocurre cuando el generador aprende a engañar al discriminador produciendo ejemplos de una sola clase a partir de todo el conjunto de datos de entrenamiento, algunas señales son: todas las salidas tienen las mismas características (no son tan diversas como las imágenes reales) y también presentan poco realismo.

## Convergencia

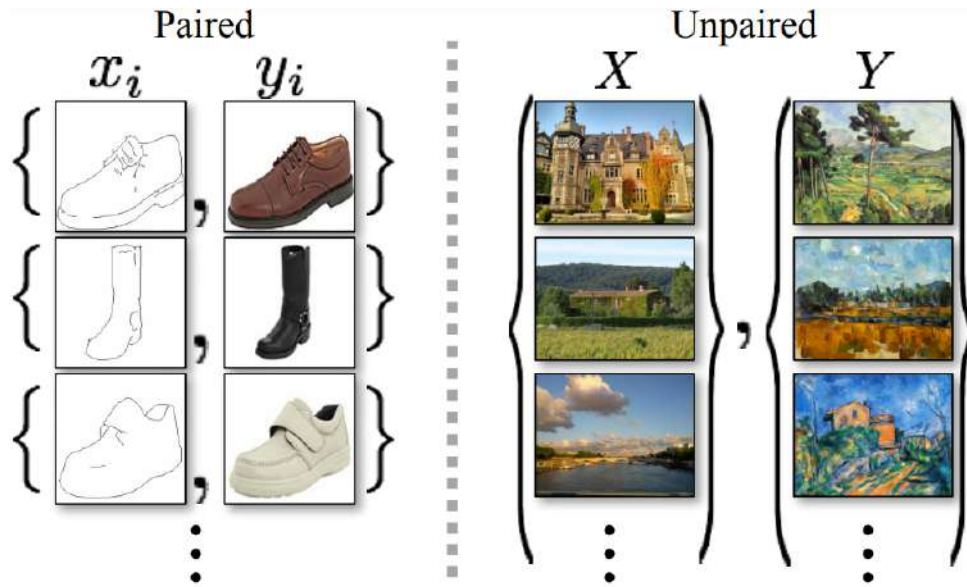
El modelo GAN puede ser inestable y tomar una cantidad considerable de tiempo para entrenar. El entrenamiento de una red neuronal se detiene cuando converge, esto es cuando su función de pérdida se estabiliza [18]. El descenso de gradiente no siempre da el generador que se necesita y cuando el generador se atasca no se puede seguir entrenando y esperar que la GAN converja.

### 2.1.4. CycleGAN

La arquitectura CycleGAN no solo ha mejorado la resolución sino también la calidad de las imágenes generadas por el marco GAN. De acuerdo con [17], CycleGAN aprende a traducir entre dominios de imágenes sin ejemplos emparejados. Una traducción de imagen a imagen consiste en transformar las imágenes en diferentes estilos pero conservando el contenido. La figura 2.7 detalla la diferencia entre una traducción de imagen a imagen emparejada y otra sin emparejar.

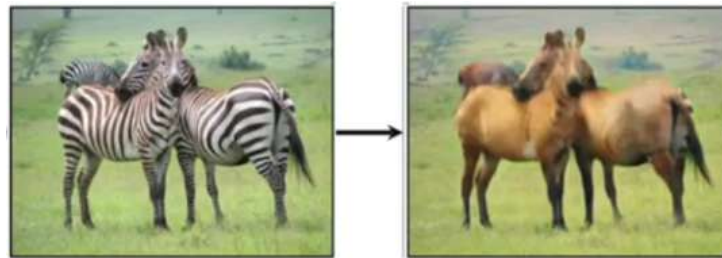
Por lo tanto, se trata de que el modelo encuentre el contenido común (elementos comunes) y los elementos estilísticos generales de dos pilas de imágenes diferentes tal que se





**Figura 2.7:** Datos de entrenamiento emparejados (izquierda) consiste en que cada dato tiene una respectiva etiqueta u objetivo. En cambio, los datos de entrenamiento sin emparejar (derecha) consiste en dos pilas de imágenes donde cada pila posee un estilo diferente [17]

conservar el contenido pero se transfieran los elementos únicos de cada pila. Por ejemplo, los elementos comunes de la figura 2.8 son: la forma general del caballo o la cebra e incluso el fondo, y los elementos únicos: las rayas de la cebra a diferencia de un solo color en los caballos.



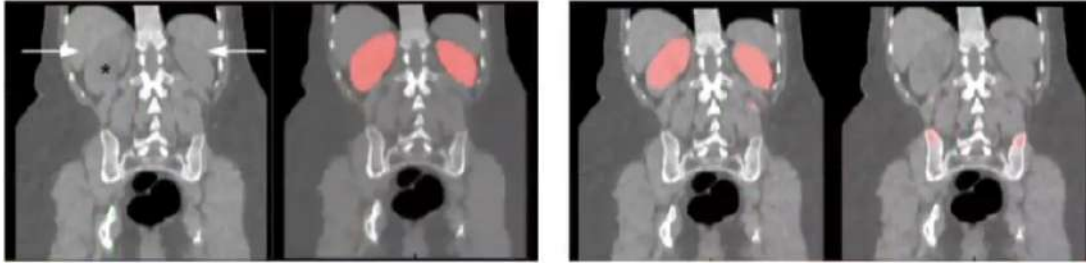
**Figura 2.8:** Ejemplo de imágenes sin emparejar con contenido común y estilo distinto [21]

El enfoque de traducción de imagen a imagen se basa en el marco Pix2Pix [27], que utiliza una GAN condicional [26] para aprender un mapeo de imágenes de entrada a salida. Se han aplicado ideas similares a diversas tareas con ejemplos de entrenamiento emparejados, como la generación de fotografías a partir de bocetos [28] o de diseños de atributos y semánticos [29].

El modelo CycleGAN se ha utilizado ampliamente en el análisis de imágenes médicas [14], [30], como la aplicación descrita en la figura 2.9.

El aprendizaje de esta arquitectura se lleva a cabo mediante la información proporcionada por tres funciones de pérdida en cada uno de los generadores: una pérdida de mínimos cuadrados adversaria, una pérdida de consistencia de ciclo y una pérdida de identidad; a





**Figura 2.9:** Ejemplo de un aumento en datos de tomografía computarizada. De izquierda a derecha: una tomografía computarizada, una segmentación hecha por un experto, segmentaciones hechas por CycleGAN y segmentaciones hechas por un aumento estándar [21]

diferencia de cada discriminador donde solo se usa una función de pérdida de mínimos cuadrados adversaria.

### **Función de pérdida de mínimos cuadrados adversaria del discriminador**

Para imágenes reales  $x$  el discriminador desea predecir la imagen como real por lo que se busca minimizar la distancia a la etiqueta 1. Para imágenes falsas el discriminador desea predecir la imagen como falsa por lo que se busca minimizar su distancia a la etiqueta 0. Tal como se expresa por la siguiente función de pérdida:

$$E_x[(D(x) - 1)^2] + E_z[(D(G(z)))^2] \quad (2.3)$$

### **Función de pérdida de mínimos cuadrados adversaria del generador**

El generador queriendo que sus salidas falsas  $z$  se vean lo más realistas posibles, esta vez busca minimizar la distancia a la etiqueta 1. Tal como se expresa por la siguiente pérdida:

$$E_z[(D(G(z)) - 1)^2] \quad (2.4)$$

### **Funciones de pérdida consistencia de ciclo y de identidad**

La pérdida de consistencia de ciclo ayuda a estabilizar el entrenamiento de las GAN [14] y también a prevenir el colapso de modo [17]. Mientras la pérdida de identidad ayuda a la preservación del color en las salidas, haciendo que la asignación de identidad del generador tenga más sentido. Ambas pérdidas se implementan con la llamada distancia de píxel o diferencia de píxel entre dos imágenes que idealmente serían idénticas, se expresa con:

$$\sum_i |(píxeles \text{ de imagen } A) - (píxeles \text{ de imagen } B)| \quad (2.5)$$

### 2.1.5. Transferencia de estilo neuronal

La idea de este modelo es similar al CycleGAN (sección 2.1.4), excepto que solo puede aprender la traducción de una imagen a otra imagen y no en conjuntos. La traducción funciona manipulando las representaciones secuenciales en una CNN para separar y recombinar el contenido y el estilo de imágenes arbitrarias, de modo que el estilo de una imagen se pueda transferir a otra conservando su contenido original, proporcionando un algoritmo neuronal para la creación de imágenes artísticas de alta calidad perceptiva [31]. Un ejemplo se ilustra en la figura 2.10.



**Figura 2.10:** Transferencia del estilo de la pintura *The Starry Night* por Vincent Van Gogh a una fotografía de Neckarfront en Tübingen, Alemania [31]

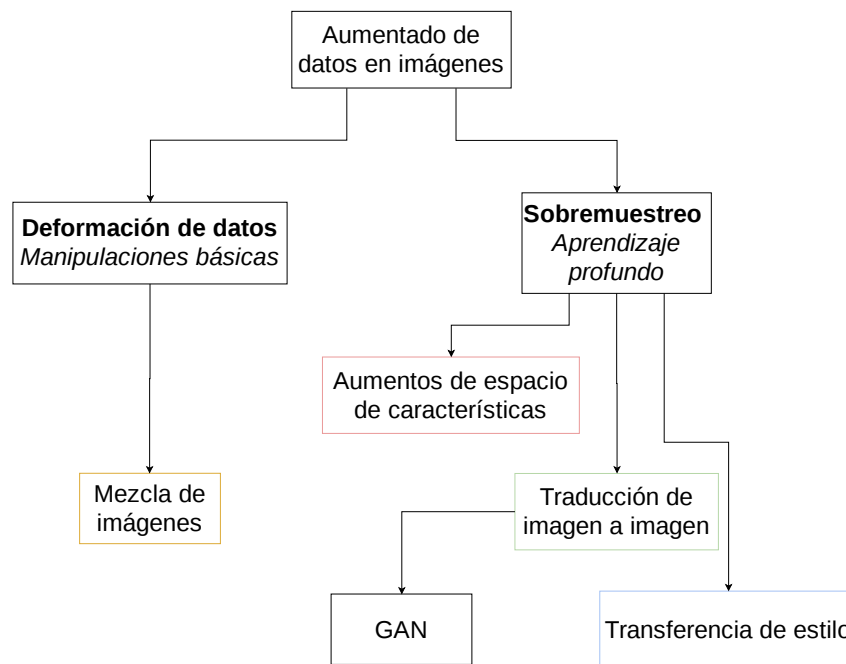
### 2.1.6. Resumen de aumentado de datos en literatura

Los métodos de aumentado de datos en imágenes intentan generar imágenes falsas pero con un realismo aceptable. En secciones anteriores se enumeraron cuatro algoritmos para la síntesis de imágenes y se agruparon en dos categorías diferentes: manipulaciones básicas y aprendizaje profundo, como se resume en la figura 2.11.

Cada algoritmo de aumentado de datos fue analizado en términos de los posibles usos, ventajas y desventajas. Esta comparación se resume en la tabla 2.1.

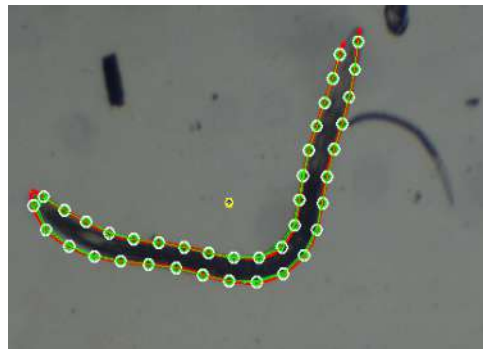
## 2.2. Modelo de forma

El modelo de forma presentado en [10] es utilizado en este trabajo para capturar la forma y los movimientos corporales de organismos vermes. El modelo está basado en subdominios permitidos de siluetas vermes expresados en hitos de frontera que ajustan la silueta de nematodo presente en una imagen digital, como los hitos mostrados en la figura 2.12. Esta información se utiliza para generar secuencias de siluetas vermes



**Figura 2.11:** Clasificación de aumentados de datos en imágenes

mediante un método denominado *serpenteo*, que consigue movimientos continuos a través de caminatas aleatorias entre subdominios vermiformes.



**Figura 2.12:** Silueta de un nematodo segmentado por 40 hitos de frontera anotados en color verde y su respectiva curva de representación en color rojo [10]

Un subdominio de formas permitidas es un conjunto de individuos del dominio de formas vermes. Estos subdominios se aproximan por medio de hipervolúmenes delimitados por  $n$ -símplices, donde los nodos que engendran dicho conjunto son formas vermes afines a su silueta. Estos nodos se obtienen usando el algoritmo de  $k$ -medias y cada nodo es un centroide sobre el dominio de forma. La figura 2.13 muestra el proceso de obtención de los subdominios de forma.

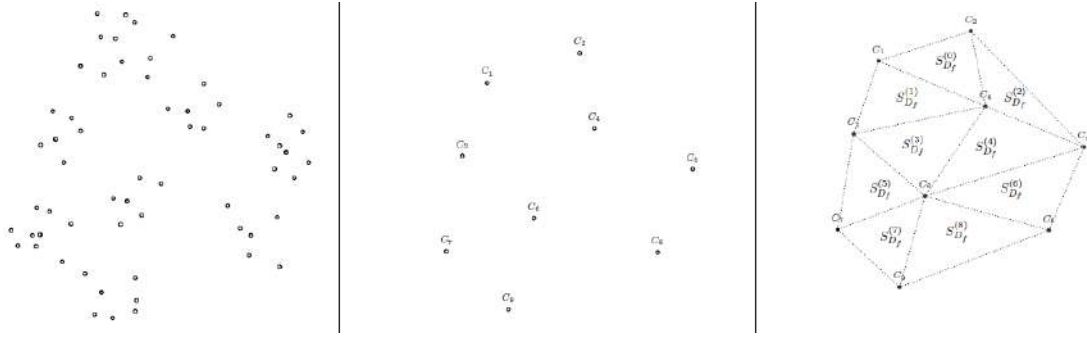
Luego se simulan movimientos vermes por medio de una caminata aleatoria a través de pasos aleatorios que deforman gradualmente la silueta original, logrando el efecto de deformaciones vermes continuas, tal como se muestra en la figura 2.14.

También se utiliza el método de *forrajeo* para simular la deformación de organismos

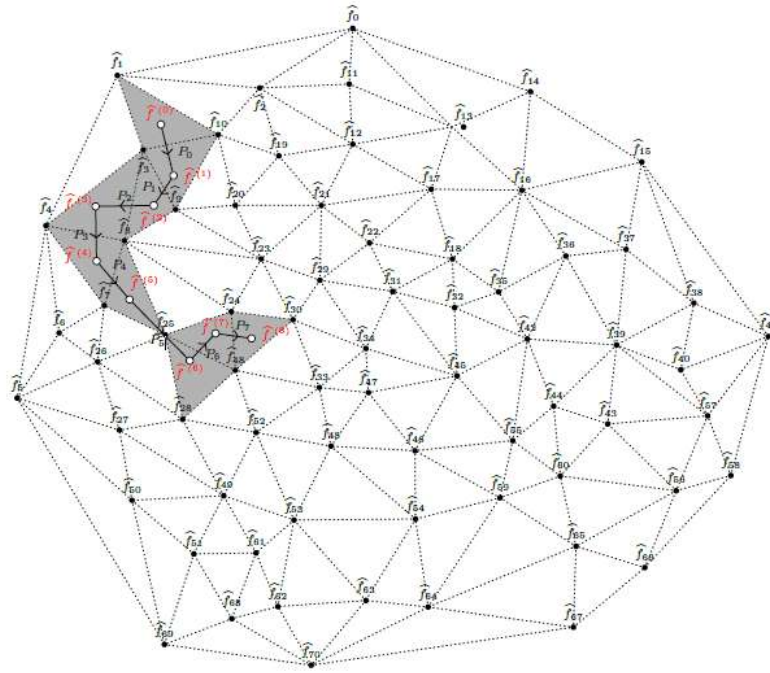
**Tabla 2.1:** Resumen, ventajas y desventajas de distintos métodos de aumentado de datos

Aumentado	Usos	Ventajas	Desventajas
Mezcla de imágenes [14], [24], [32]	Emparejamiento de imágenes mediante recortes o volteos.	Fácil de implementar. Representaciones más sólidas de características de bajo nivel.	Muy bajo realismo.
Aumentos de espacio de características [14]	Adición de ruido, interpolación o extrapolación.	Uso de arquitecturas modernas para modificar el espacio de características como los autocodificadores variacionales (VAE).	Dificultad para interpretar los datos vectoriales. Codificadores difíciles de entrenar y requieren mucho tiempo. Bajo realismo.
CycleGAN [14]	Transferencia de estilos únicos y conservación de elementos comunes. Aplicaciones médicas.	Alto realismo. Traducción entre conjuntos de imágenes sin emparejar. Alta resolución de imágenes.	Inestabilidad. Muchos datos para entrenar.
Transferencia de estilo neuronal	Transferencia de estilos únicos y conservación de elementos comunes. Aplicaciones artísticas.	Alta calidad perceptiva en trabajos artísticos.	Bajo realismo. Traducción de imagen a imagen. Esfuerzo necesario para seleccionar los estilos.

vermiformes en su desplazamiento. Por ejemplo, provocando ondulaciones sinusoidales a través de contracciones del cuerpo, cola y cabeza, y alargamiento.



**Figura 2.13:** Proceso de obtención de los subdominios de forma. De izquierda a derecha: conglomerado de puntos que representa una forma verme cada uno, centroides calculados de los agrupamientos, subdominios de forma construidos que representan  $n$ -símplexes en el espacio de datos [10]



**Figura 2.14:** Caminatas aleatorias entre subdominios vecinos [10]

## 2.3. Procesamiento de imágenes

Las técnicas de procesamiento de imágenes empleadas en este trabajo son implementadas mediante funciones, bibliotecas de análisis de imagen y scripts cortos de algún lenguaje de programación [33]. Estos scripts realizan tareas complicadas mediante operadores de imagen y funciones, como se describe en la tabla 2.2.

**Tabla 2.2:** Operaciones para el procesamiento de imagen

Tipo de operación		Descripción
Geométricos: aquellos que mueven píxeles en lugar de cambiar la intensidad.	Cambio de tamaño	Tiene el propósito de cambiar el tamaño de una imagen, lo cual significa cambiar las dimensiones de la misma, ya sea solo el ancho, el alto o ambos.
	Volteo	Volteado de imagen a través del eje $x$ (vertical), el eje $y$ (horizontal) y a través de ambos ejes.
	Rotación	Rotación de una imagen en un ángulo $\theta$ alrededor de un centro $\mathbf{v}$ .
Intensidad: estos crean una nueva imagen modificando las intensidades de píxel de una imagen original.	Inversión	Inversión de cada bit de una matriz, lo que se conoce como el cálculo del negativo de la imagen.
	Adición de ruido gaussiano	Mediante una función se provoca variaciones en la intensidad de la imagen generadas por una distribución gaussiana normal.

## 2.4. Métricas

La evaluación cuantitativa de modelos generativos es una tarea desafiante especialmente cuando no se tienen imágenes de verdad fundamental (*ground truth*) para probar y validar [34]. Es por eso que las métricas disponibles son tanto juicios visuales cualitativos como cuantitativos de la fidelidad (realismo) y diversidad de las imágenes. Estas métricas permiten comparar y calificar el desempeño de los modelos y dentro de las posibles alternativas se describen las siguientes.

### 2.4.1. Realismo perceptivo

#### Amazon Mechanical Turk

El Amazon Mechanical Turk (AMT) es un servicio comercial utilizado para estudios de percepción “real frente a falsos” [35]. Esto consiste en un experimento en el que a varios participantes se les muestra un par de imágenes y ellos deben decidir si son reales o generadas por un programa informático. El diseño del algoritmo en AMT para la metodología de presentación de los datos puede ser, por ejemplo, mostrar imágenes individuales de resolución 256 x 256 durante un segundo cada una, y después de cada par, se da a los

participantes un tiempo limitado para responder [36]. El costo monetario de usar AMT se compone de dos componentes: la cantidad que paga a los trabajadores, más una tarifa que paga a AMT. Algunos artículos médicos miden la calidad de los resultados de GAN mediante pruebas visuales de Turing; como en la clasificación de lesiones cutáneas y la detección de cáncer de hígado [37].

### Índice Similitud Estructural

El índice de similitud estructural (SSIM) es un método para determinar la similitud entre dos imágenes. En [38] el índice se define por medio de estimaciones locales  $ssim$  entre dos ventanas  $x$  e  $y$  de tamaño  $N \times N$  píxeles calculados alrededor de los píxeles correspondientes en las imágenes  $A$  y  $B$  según:

$$ssim(x,y) = \frac{(\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + C_2)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (2.6)$$

donde  $\mu_x$  es el promedio en la ventana  $x$ ;  $\mu_y$  es el promedio en la ventana  $y$ ;  $\sigma_x^2$  es la varianza en la ventana  $x$ ;  $\sigma_y^2$  es la varianza en la ventana  $y$ ;  $\sigma_{xy}$  es la covarianza de ventanas  $x$  e  $y$ ; las constantes  $c_1 = (k_1L)^2$  y  $c_2 = (k_2L)^2$ ,  $L = 255$  determina el rango dinámico de luminancia;  $k_1 = 0,01$ ,  $k_2 = 0,03$  son constantes definidas experimentalmente.

El índice SSIM global para las imágenes  $A$  y  $B$  se calcula como la media aritmética de las estimaciones locales de acuerdo con la fórmula:

$$SSIM(A,B) = \frac{1}{M} \sum_{xy} ssim(x,y) \quad (2.7)$$

donde  $A$  y  $B$  son dos imágenes de igual tamaño,  $M$  es el número de ventanas. Cada estimación local  $ssim(x,y)$  se calcula en una vecindad (generalmente  $8 \times 8$ ) de píxeles con las mismas coordenadas  $(i,j)$  para ambas imágenes  $A$  y  $B$ . Los valores del índice SSIM se encuentran en el rango de  $-1$  a  $+1$ . El valor de  $+1$  se logra cuando las imágenes comparadas son completamente idénticas. En [39] se ha observado que la medida SSIM puede ser engañosa.

#### 2.4.2. Interpretabilidad semántica

El uso de clasificadores pre-entrenados para evaluar el realismo de los datos sintetizados ha sido utilizado por [36]. En este estudio se mide el grado de realismo de imágenes coloreadas falsas, al ser sometidas a la interpretación de un clasificador VGG pre-entrenado para predecir las clases de ImageNet a partir de fotos en colores reales. Si el clasificador funciona bien, eso significa que las coloraciones son lo suficientemente precisas para ser informativas sobre la clase de objeto. Como resultado, el rendimiento del clasificador cae del 68,3 % al



52,7 % al eliminar los colores de la entrada y después de volver a colorear con su método, el rendimiento mejora al 56,0 %.

### 2.4.3. Consistencia de ciclo

La pérdida de consistencia de ciclo es utilizada por el modelo CycleGAN (sección 2.1.4) para minimizar la diferencia entre la imagen “real  $A$ ” y la “falsa  $A$  retraducida”. Es útil mostrar el nivel de consistencia de ciclo que alcanzó el modelo. Para ello [34] mide la similitud entre las imágenes de entrada y las imágenes de salida después de un ciclo de traducciones con la métrica SSIM (sección 2.4.1).

### 2.4.4. Matriz de confusión

La matriz de confusión como una métrica de evaluación de clasificadores permite medir la efectividad de un modelo donde se tengan dos o más clases para clasificar [40]. La figura 2.15 muestra las partes que componen una matriz de confusión: la diagonal principal indica los aciertos obtenidos y por fuera los fallos y con respecto a qué clases se han cometido los errores. En las columnas se indica a qué clase atribuye la red una entrada.

		Predicción	
Actual		Positivo	Negativo
	Positivo	<b>Verdaderos Positivos (VP)</b> <i>Resultado en el que el modelo predice correctamente la clase positiva</i>	<b>Falsos Negativos (FN)</b> <i>Resultado en el que el modelo predice incorrectamente la clase negativa</i>
	Negativo	<b>Falsos Positivos (FP)</b> <i>Resultado en el que el modelo predice incorrectamente la clase positiva</i>	<b>Verdaderos Negativos (VN)</b> <i>Resultado en el que el modelo predice correctamente la clase negativa</i>

**Figura 2.15:** Ejemplo de matriz de confusión

Dos métricas de rendimiento derivadas de la matriz de confusión son la exactitud y la precisión. Esta métrica de exactitud corresponde al número de predicciones correctas sobre la totalidad de las predicciones hechas por el modelo y la precisión es el número de predicciones positivas correctas sobre el número total de predicciones positivas. Estas métricas se expresan con la siguientes ecuaciones:

$$Exactitud = \frac{VP + VN}{VP + VN + FP + FN} \quad (2.8)$$

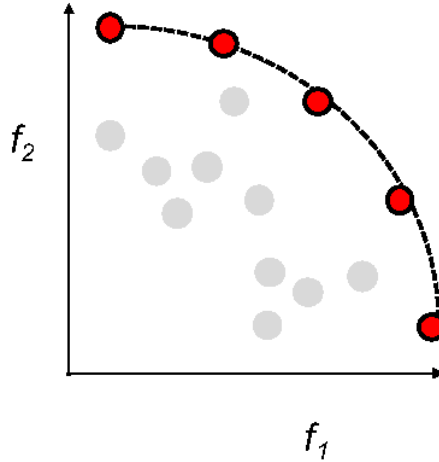
$$Precisión = \frac{VP}{VP + FP} \quad (2.9)$$



### 2.4.5. Optimalidad Pareto

La optimalidad Pareto es un concepto utilizado cuando en un problema se requieren optimizar múltiples objetivos simultáneamente. En [41] la optimización multiobjetivo produce no una solución única, sino un conjunto de ellas llamadas *soluciones no-dominadas* (dominancia), a las que se les llama conjunto de óptimos de Pareto y, al representarlas en el espacio de los valores de las funciones objetivo, conforman el frente de Pareto.

Los conceptos de dominancia y frente de Pareto se ilustran en la figura 2.16.

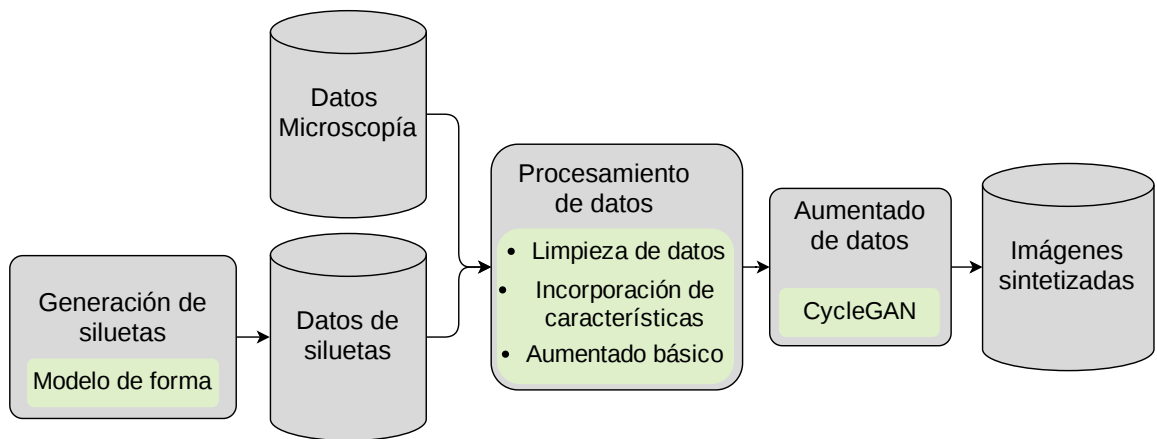


**Figura 2.16:** Frente de Pareto conformado por los puntos rojos. Toda el área dominada está bajo la línea negra que corresponde a los puntos grises [42]

## Capítulo 3

# Sistema de aumentado por síntesis de imagen

La figura 3.1 muestra la estrategia utilizada para las etapas propuestas en la síntesis de imágenes de microscopía óptica digital como sistema de aumentado, que se detallarán en las siguientes secciones de este capítulo. En el caso de la etapa de generación de siluetas, se basa en un modelo de forma. Para la etapa de procesamiento de datos fueron propuestas tres estrategias: una limpieza de datos al conjunto de imágenes de microscopía, una incorporación de características y aumentado básico al conjunto de siluetas de nematodos. La etapa de aumentado de datos se basa en la creación de nuevas instancias sintéticas usando la arquitectura de aprendizaje profundo CycleGAN.



**Figura 3.1:** Diagrama de la *estrategia de síntesis de imágenes* como sistema de aumentado de datos

Las siguientes secciones especifican las operaciones realizadas en cada una de las etapas en un proceso de 3 experimentos de síntesis de imágenes.

### 3.1. Generación de siluetas

El módulo generación de siluetas tiene el objetivo de capturar la forma intrínseca vermiforme de los nematodos a través de una base de datos de hitos de frontera que describen el contorno de nematodos en imágenes de microscopía utilizando principios del modelo de forma introducido en la sección 2.2. El proceso consiste en construir subdominios de formas permitidas con el algoritmo  $k$ -medias. Luego se buscan los  $n$  vecinos más cercanos y se generan coordenadas baricéntricas al azar para obtener la silueta, como el ejemplo de silueta generada de la figura 3.2.



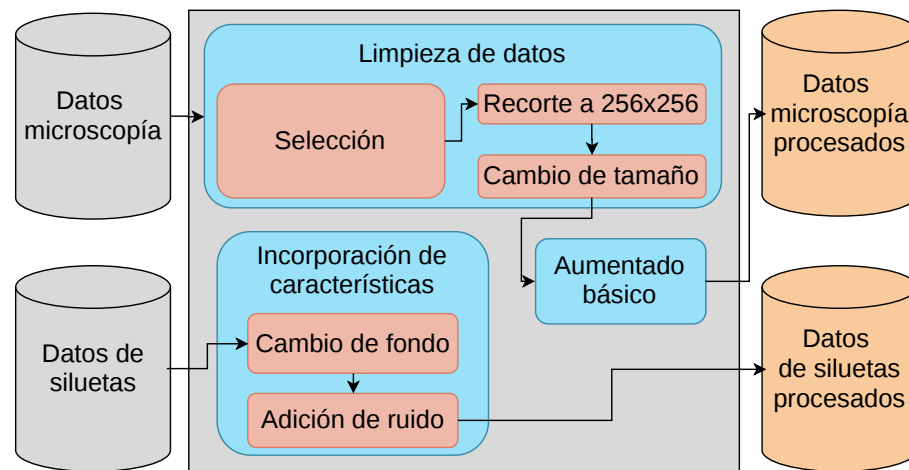
**Figura 3.2:** Ejemplo de silueta generada por el modelo de forma. La silueta consiste en una máscara blanca con forma de nematodo y el resto de la imagen corresponde a fondo negro

### 3.2. Procesamiento de datos

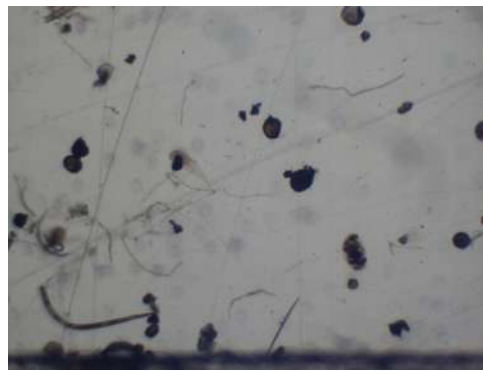
Con el fin de alimentar las redes neuronales profundas con imágenes adecuadas para su propósito, se sigue un proceso de selección, recorte, cambio de tamaño, aumentado básico, cambio de fondo y adición de ruido en el orden mostrado por la figura 3.3 e introducidos en la sección 2.3.

#### 3.2.1. Limpieza de datos

No todos los datos reales disponibles de resolución  $1024 \times 768$  son adecuados. Hay imágenes de microscopía con abundancia de ruido: presencia de detritos, residuos y sedimentos provenientes de los procesos respectivos [10]. También en cada imagen de microscopía aparece más de un nematodo, algunos de estos incluso sin vida. Un ejemplo de datos crudos ruidosos se muestra en la figura 3.4. Además, se desea introducir al modelo CycleGAN imágenes con dimensiones pequeñas para un menor tiempo de procesamiento. Por tanto, se va a experimentar con el nivel de ruido de las imágenes hasta obtener dos bases de datos: imágenes de microscopía ruidosas e imágenes de microscopía limpias.



**Figura 3.3:** Diagrama de flujo de las etapas del procesamiento de datos



**Figura 3.4:** Ejemplo de dato crudo de microscopía ruidoso

### 3.2.2. Incorporación de características de fondo y ruido

Los datos de siluetas generados (sección 3.1) presentan dos variaciones de color, silueta blanca y fondo negro, que podría representar poca información para establecer una relación de contenido y estilo con la imagen de microscopía. Es por ello que se experimenta con el nivel de naturalidad y ruido hasta obtener tres bases de datos: siluetas y fondos crudos, siluetas negativas con fondos naturales ruidosos y siluetas negativas ruidosas con fondos naturales limpios. Un ejemplo de dato de silueta modificado se presenta en la figura 3.5.



**Figura 3.5:** Ejemplo de silueta modificada con negativo y fondo natural

### 3.2.3. Combinatoria de bases de datos

Las bases de datos planteadas en las secciones 3.2.1 y 3.2.2 se combinan para formar tres nuevas bases de datos que se presentan en la tabla 3.1.

**Tabla 3.1:** Bases de datos conformadas por imágenes de siluetas y microscopía

	Clase	
	$S$	$M$
Base de datos 1	Silueta cruda y fondo crudo	Microscopía ruidosa
Base de datos 2	Silueta negativa con fondo natural ruidoso	Microscopía ruidosa
Base de datos 3	Silueta negativa ruidosa con fondo natural limpio	Microscopía limpia

### 3.2.4. Aumentado básico

El aumentado básico se utiliza para completar y equilibrar la base de datos 3 planteada en la sección 3.2.3 y consiste en aplicar operaciones de volteos verticales en 19 %, volteos horizontales en 68,3 % y volteos en ambas direcciones en 12,1 %.

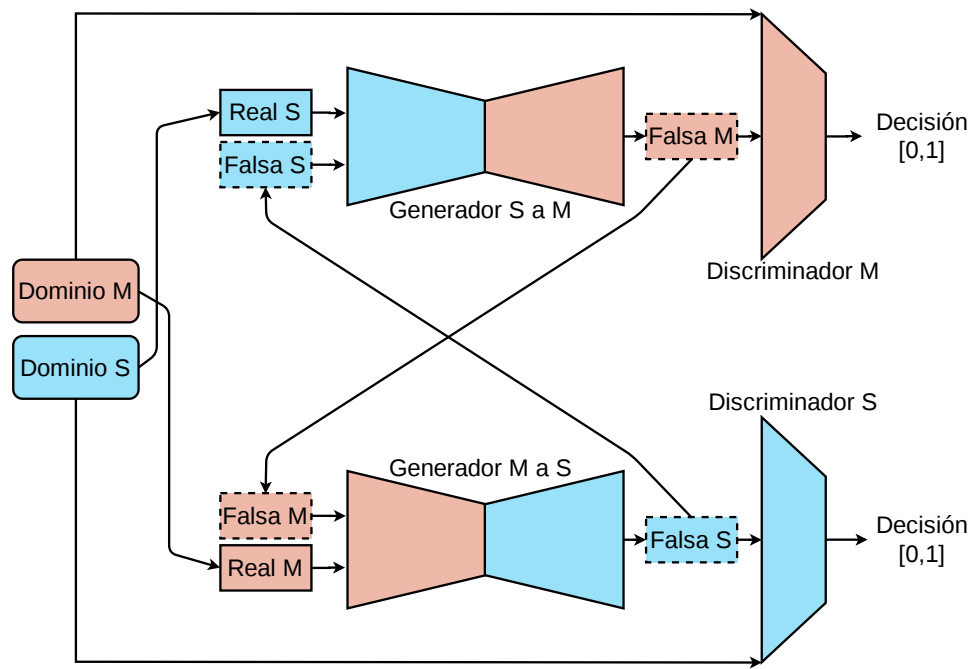
### 3.3. Aumentado de datos

En la sección 2.1 se presentaron posibles métodos de aumentado de datos de los cuales se realiza una selección según los aspectos valorados en la tabla 3.2.

**Tabla 3.2:** Selección de las alternativas propuestas para el aumentado de datos

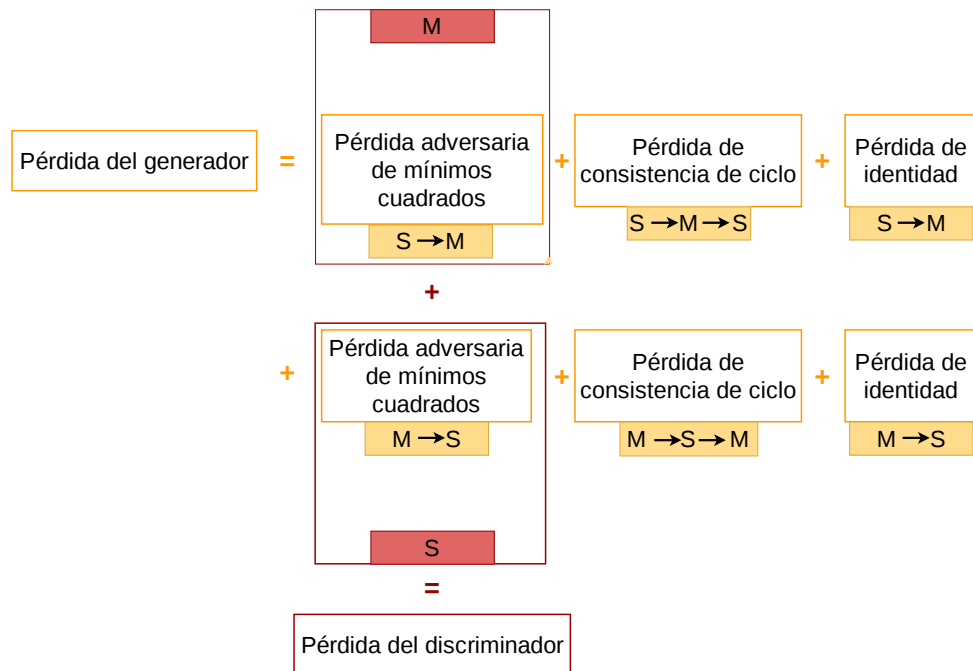
Aumentado	Criterio
Mezcla de imágenes	Debido a que sintetiza imágenes con muy bajo realismo se descarta
Aumentos de espacio de características	Debido a que sintetiza imágenes con bajo realismo se descarta
CycleGAN	Debido a que sintetiza imágenes con alto realismo y realiza una traducción entre conjuntos de imágenes se utiliza
Transferencia de estilo neuronal	Debido a que sintetiza imágenes con bajo realismo y realiza una traducción de imagen a imagen se descarta

La síntesis de imágenes se lleva a cabo mediante la arquitectura CycleGAN, que aborda el problema de la traducción de imagen a imagen sin emparejar (sección 2.1.4). La figura 3.6 muestra los módulos involucrados en esta arquitectura: un dominio de imágenes de microscopía con nematodos ( $M$ ), un dominio de imágenes de siluetas de nematodos ( $S$ ) y dos GAN. Las GAN equivalen a dos redes neuronales generativas usadas para la generación de imágenes y dos discriminativas usadas para clasificar la imagen como falsa o real [17]. El objetivo del módulo generativo es sintetizar imágenes cada vez más realistas durante el entrenamiento a tal punto que le sea imposible al módulo discriminador distinguir la imagen falsa de la real.



**Figura 3.6:** Diagrama de bloques de la arquitectura CycleGAN (inspirado en [34])

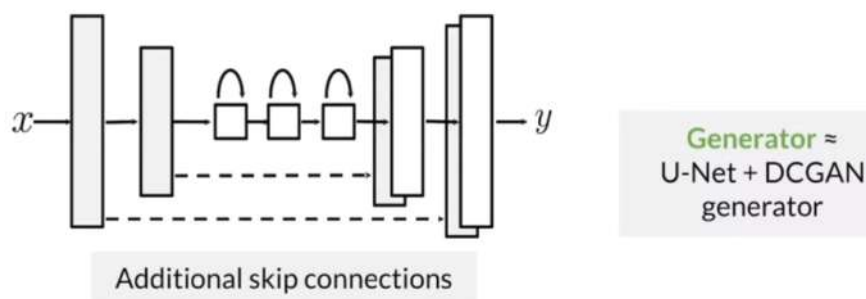
La figura 3.7 resume todas las funciones de pérdida que evalúan los resultados tanto de las redes neuronales generativas como de las discriminativas.



**Figura 3.7:** Esquema de las funciones de pérdida (inspirado en [21])

### 3.3.1. Red generativa

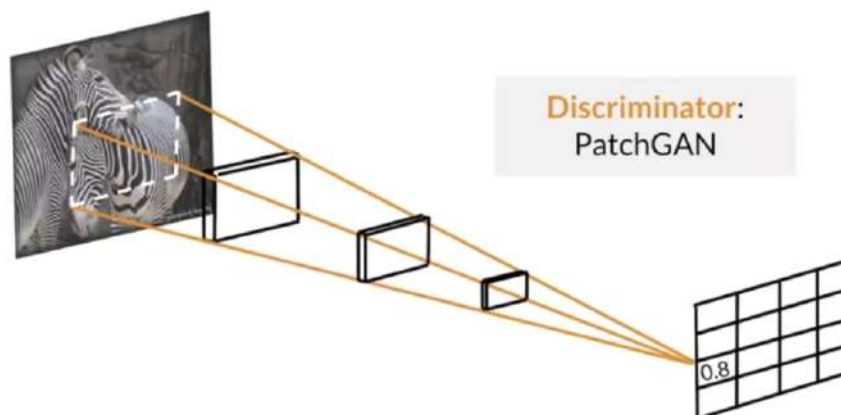
La red generativa está basada en la arquitectura U-Net y DCGAN. La figura 3.8 muestra un diagrama general U-Net, en el cual, hay una sección codificadora que realiza un muestreo descendente y una sección decodificadora para un muestreo ascendente. Ambas se componen de capas convolucionales con la normalización por lotes y funciones de activación ReLU. Además, U-Net tiene una sección de cuello de botella que se expande con más capas convoluciones como sucede con un generador DCGAN. Conexiones de salto adicionales son agregadas tanto a las secciones de codificación-decodificación para poder tener acceso a características muy comprimidas, como a la sección de cuello de botella para realizar transformaciones de imagen al permitir que el modelo aprenda funciones de identidad [21].



**Figura 3.8:** Arquitectura de la red generadora [21]

### 3.3.2. Red discriminadora

La red discriminadora se basa en la arquitectura PatchGAN que funciona capturando parches de  $70 \times 70$  en la imagen de salida, dando con ello una matriz de valores en lugar de un solo valor como puede apreciarse en la figura 3.9. PatchGAN tiene el objetivo de clasificar si los parches de imagen superpuestos son reales o falsos. Esta arquitectura discriminadora a nivel de parche trabaja en imágenes de una manera totalmente convolucional.

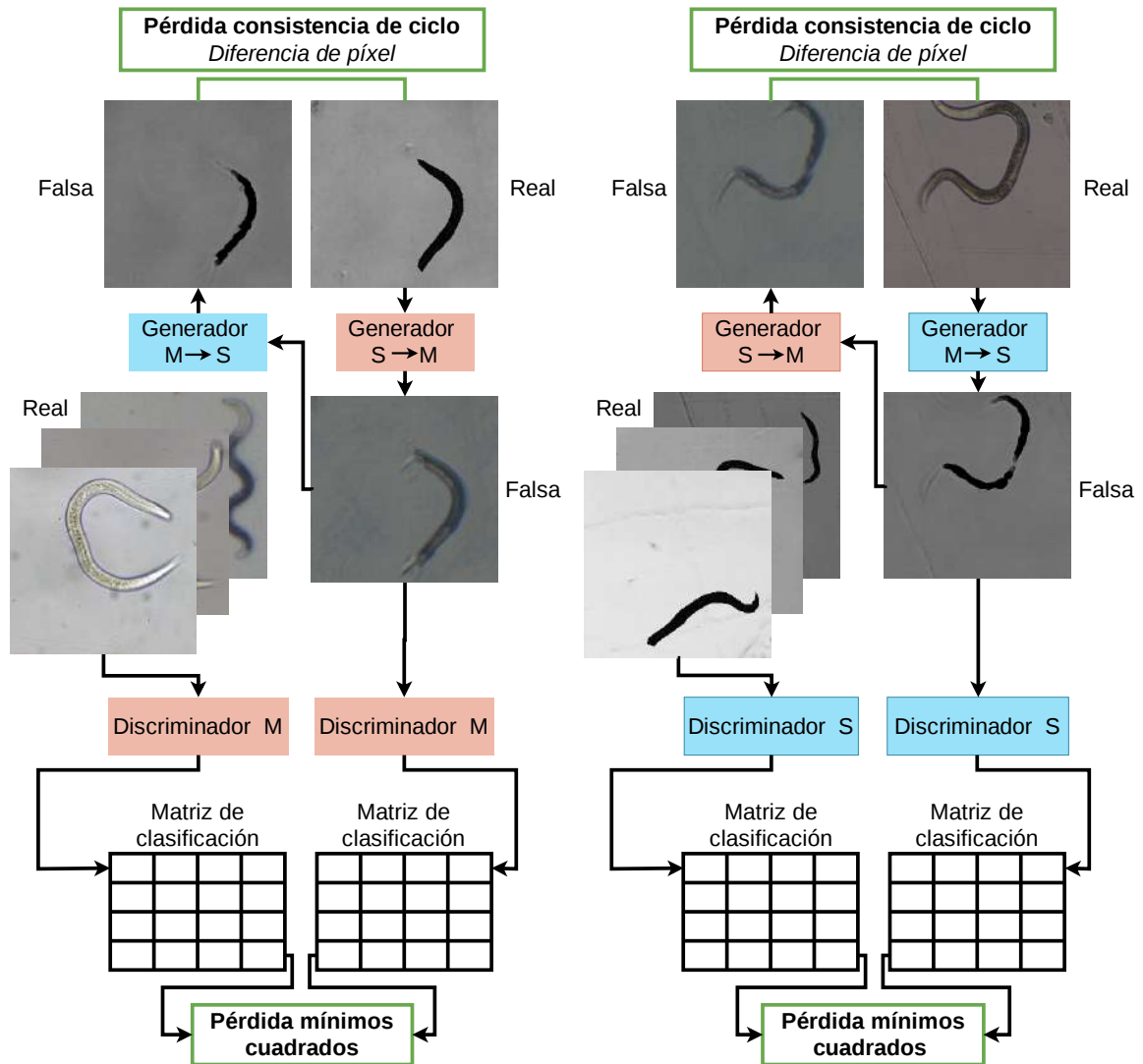


**Figura 3.9:** Parches de la red discriminadora [21]

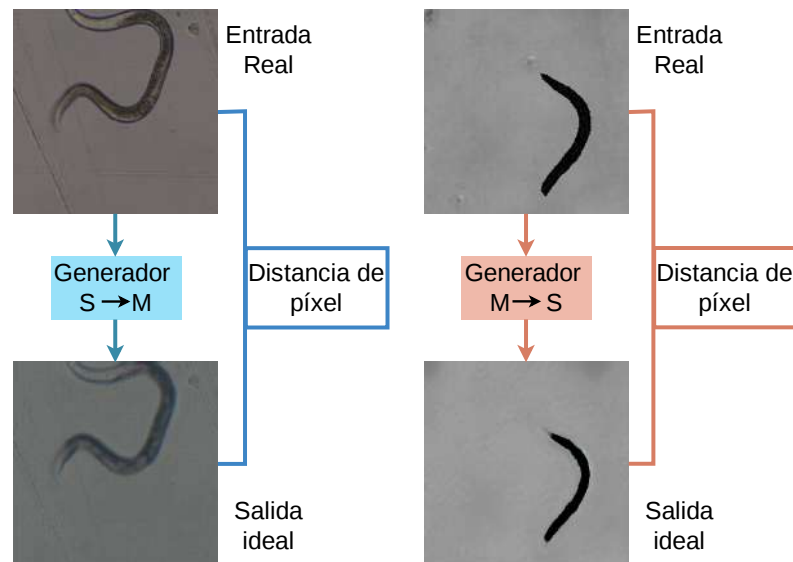


### 3.3.3. Entrenamiento y alternativas de evaluación

En las figuras 3.10 y 3.11 se realiza la adaptación de la arquitectura CycleGAN a este proyecto. Los datos de microscopía reales corresponden al dominio de destino  $M$  y los datos de siluetas corresponden al dominio de origen  $S$  del diagrama de la figura 3.6; es decir, se busca transferir el estilo de las imágenes de microscopía a las siluetas. Téngase en cuenta que esta transferencia también sucede en la dirección opuesta.



**Figura 3.10:** Adaptación del modelo CycleGAN a este proyecto. De izquierda a derecha: mapeo en la dirección  $S$  a  $M$  y mapeo en la dirección opuesta (inspirado en [21])



**Figura 3.11:** Diagrama de la función de pérdida de identidad. De izquierda a derecha: mapeo en la dirección  $S$  a  $M$  y mapeo en la dirección opuesta (inspirado en [21])

Las métricas de evaluación de los resultados se seleccionan de entre las posibles alternativas según los aspectos valorados en la tabla 3.3.

**Tabla 3.3:** Selección de las alternativas propuestas para la evaluación de resultados

Métrica	Criterio
AMT	Debido a los costos asociados se descarta para este proyecto
SSIM	Debido a que se cuenta con imágenes similares durante las traducciones del modelo generativo se utiliza
Interpretabilidad semántica	Debido a que el clasificador VGG necesita muchas imágenes para ser reentrenado se descarta
Matriz de confusión	Debido a que se va a contar con modelos que alcanzan un desempeño específico se utiliza
Optimalidad Pareto	Debido a que el desempeño del modelo discriminativo es variable respecto a un umbral de decisión entonces se utiliza

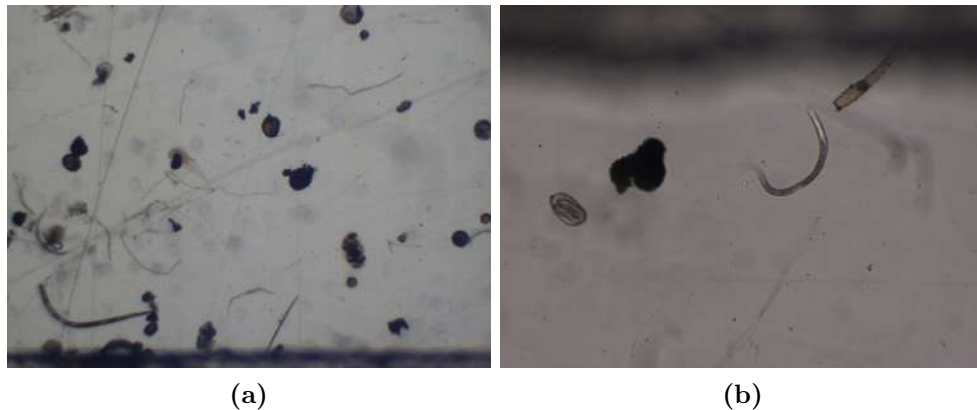
# Capítulo 4

## Resultados y análisis

El objetivo del proceso experimental realizado es determinar el desempeño del aumentado de datos a través de la precisión y exhaustividad del sistema sintetizador de imágenes propuesto. Además, se busca evaluar si la incorporación de características de fondo y ruido mejora o no el resultado obtenido en la generación de imágenes realistas.

### 4.1. Datos de entrenamiento y prueba

La cantidad de datos de entrenamiento y prueba se divide en dos conjuntos: clase  $M$  (microscopía) y clase  $S$  (siluetas) (sección 3.2.3). Los datos de la clase  $M$ , de resolución  $1024 \times 768$ , son en total 7208 que corresponden a secuencias de video del microscopio óptico Nikon Eclipse E200 con cámara Basler A630f [10] de los cuales se seleccionan aquellos con nematodos identificables y con ruido tolerable como se ilustra en la figura 4.1, quedando 1334 imágenes para entrenamiento y 140 para pruebas. Los datos de la clase  $S$ , de resolución  $256 \times 256$ , son en total 1187, dividiéndose en 1067 para entrenamiento y 120 para pruebas.

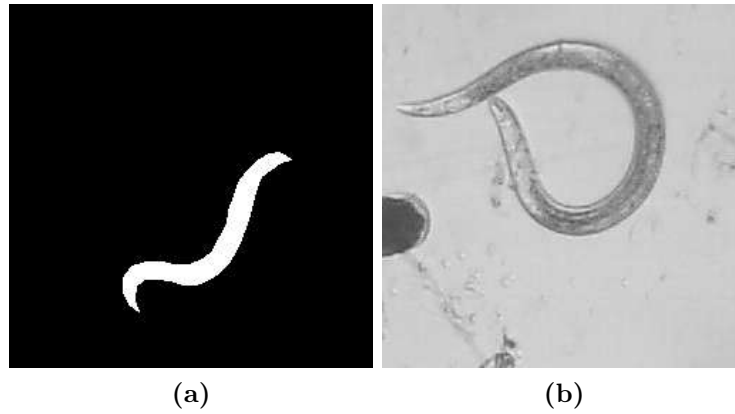


**Figura 4.1:** Selección de datos de microscopía para el conjunto de entrenamiento. (a) Dato descartado debido a la abundancia de ruido (b) Dato seleccionado con ruido tolerable

El entrenamiento y evaluación del modelo CycleGAN se implementa en PyTorch y se desarrolla en los servidores de Google Colab [43]. Se divide en tres experimentos explicados a continuación.

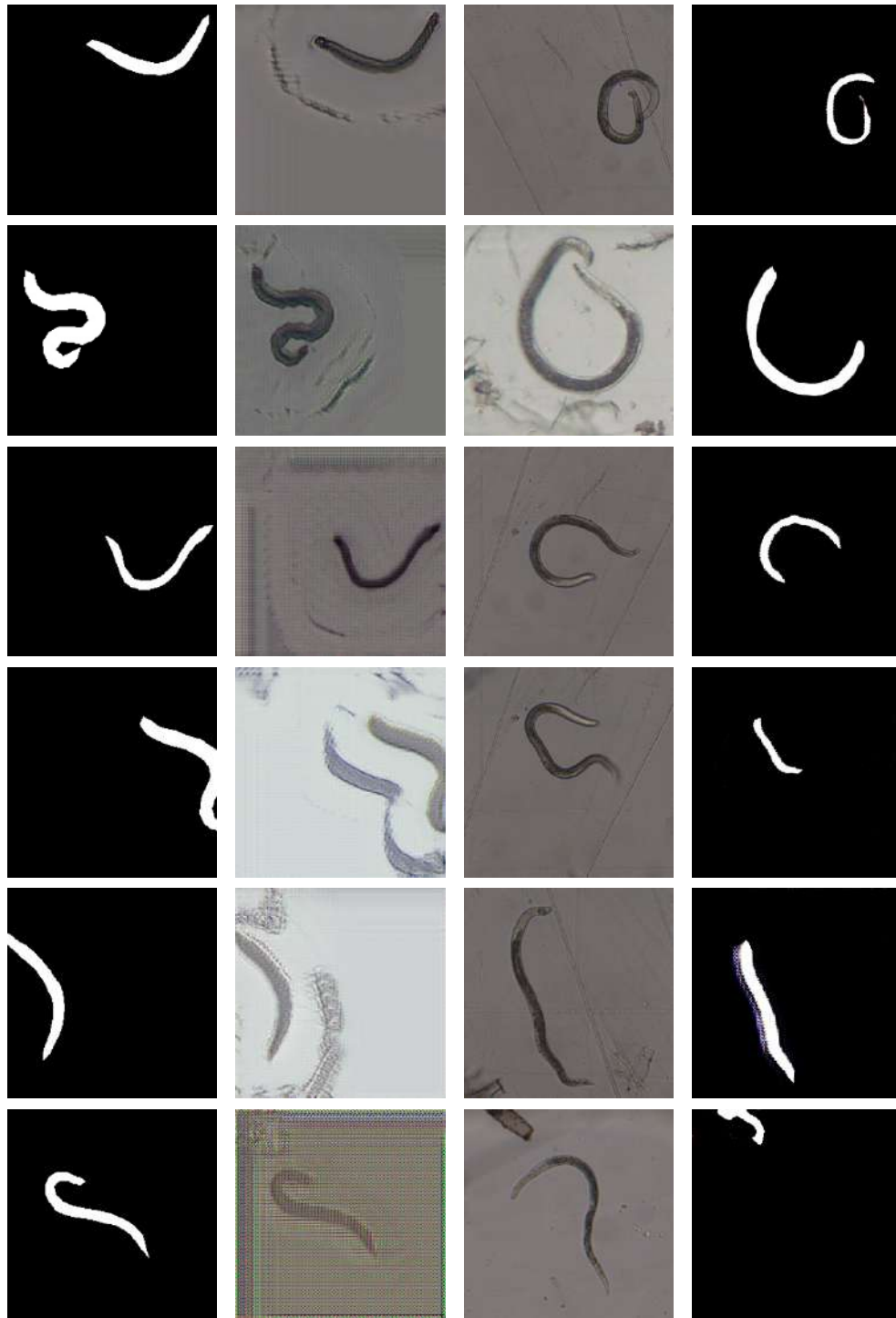
#### 4.1.1. Experimento 1

El entrenamiento del modelo se lleva a cabo con la base de datos 1 (sección 3.2.3), donde la clase  $M$  se obtiene recortando en forma manual aquellas imágenes con la presencia de uno o más nematodos a una resolución de  $256 \times 256$ . La clase  $S$  se mantiene sin modificaciones (sección 3.1). Un ejemplo de ambas clases se ilustra en la figura 4.2.



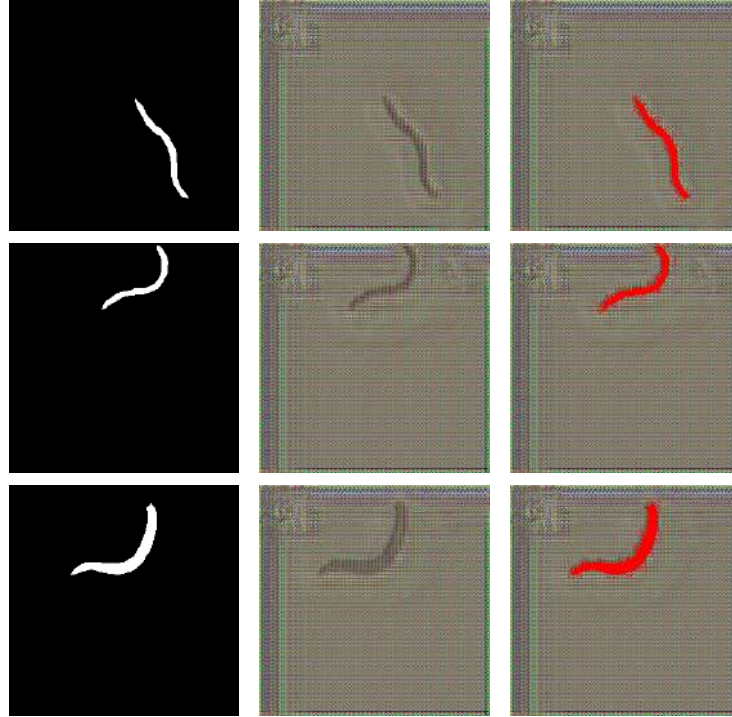
**Figura 4.2:** Ejemplos de la base de datos 1. (a) clase  $S$  (b) clase  $M$

La figura 4.3 muestra las imágenes sintetizadas durante el proceso de aprendizaje de las redes generativas y discriminativas para seis distintas épocas del proceso de entrenamiento. Dicho proceso de entrenamiento requirió 8 días 22 horas y 27 minutos.



**Figura 4.3:** Imágenes sintetizadas durante el entrenamiento del modelo del experimento 1. De izquierda a derecha: dato de silueta, dato de microscopía sintetizado, dato de microscopía real, dato de silueta sintetizado. De arriba a abajo: época 10, época 20, época 50, época 100, época 150 y época 200 (final)

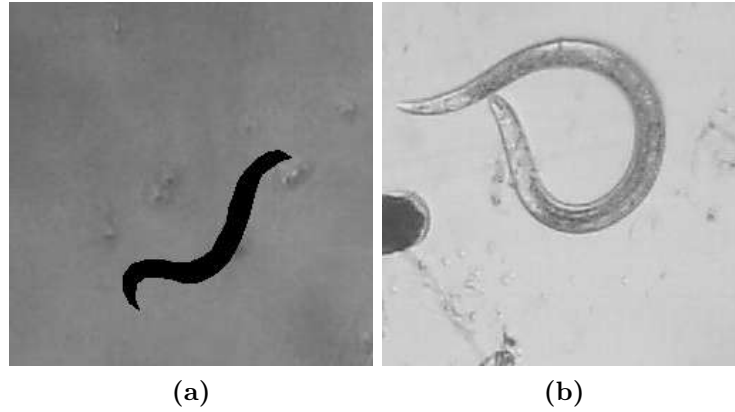
La uniformidad en características y escasez de realismo que se observa puede deberse a un problema de colapso de modo (sección 2.1.3). En este caso, es esperable que la evaluación del modelo entrenado con el conjunto de datos de prueba dé un resultado con bajo realismo, como se aprecia en el ejemplo de la figura 4.4 de tres imágenes sintetizadas. No obstante, el mapeo está conservando perfectamente las dimensiones de la silueta.



**Figura 4.4:** Imágenes sintetizadas evaluando el modelo entrenado del experimento 1 con un conjunto de datos de prueba. De izquierda a derecha: dato de silueta, dato de microscopía sintetizado, dato de silueta superpuesto en sintetizada. De arriba a abajo: resultado 1, resultado 2 y resultado 3

### 4.1.2. Experimento 2

El entrenamiento del modelo se lleva a cabo con la base de datos 2 (sección 3.2.3), donde la clase  $M$  se obtiene recortando en forma manual aquellas imágenes con la presencia de uno o más nematodos a una resolución de  $256 \times 256$ . La clase  $S$  se obtiene cambiando el fondo negro por un fondo real aleatorio ruidoso de microscopía y se aplica un negativo a la máscara blanca del nematodo (sección 3.2.2). Un ejemplo de ambas clases se ilustra en la figura 4.5.



**Figura 4.5:** Ejemplos de la base de datos 2. (a) clase  $S$  (b) clase  $M$

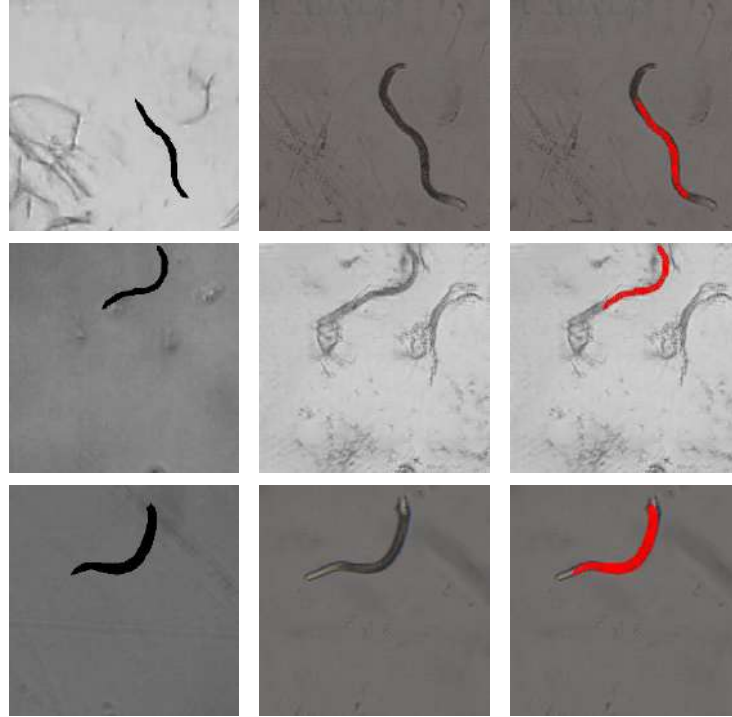
La figura 4.6 muestra las imágenes sintetizadas durante el proceso de entrenamiento para seis distintas épocas. Dicho proceso de entrenamiento requirió 5 días 2 horas y 34 minutos.



**Figura 4.6:** Imágenes sintetizadas durante el entrenamiento del modelo del experimento 2. De izquierda a derecha: dato de silueta, dato de microscopía sintetizado, dato de microscopía real, dato de silueta sintetizado. De arriba a abajo: época 10, época 20, época 50, época 100, época 150 y época 200 (final)



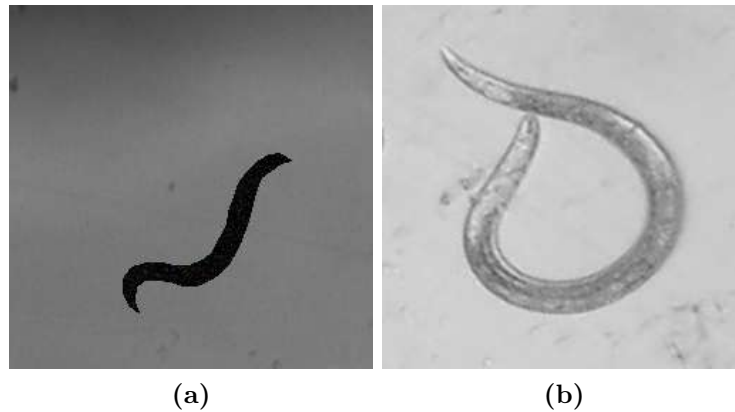
El realismo de imagen obtenido ya en la época 50, es un indicador de que la red generativa ha convergido y se puede prescindir de las 200 épocas. Luego, la evaluación del modelo con datos de prueba genera los tres ejemplos de la figura 4.7 con imágenes sintetizadas. Se observa que la red generativa suele confundir la forma del nematodo con artefactos ruidosos y además se aprecia que no está conservando las dimensiones de la silueta.



**Figura 4.7:** Imágenes sintetizadas evaluando el modelo entrenado del experimento 2 con un conjunto de datos de prueba. De izquierda a derecha: dato de silueta, dato de microscopía sintetizado, dato de silueta superpuesto en sintetizada. De arriba a abajo: resultado 1, resultado 2 y resultado 3

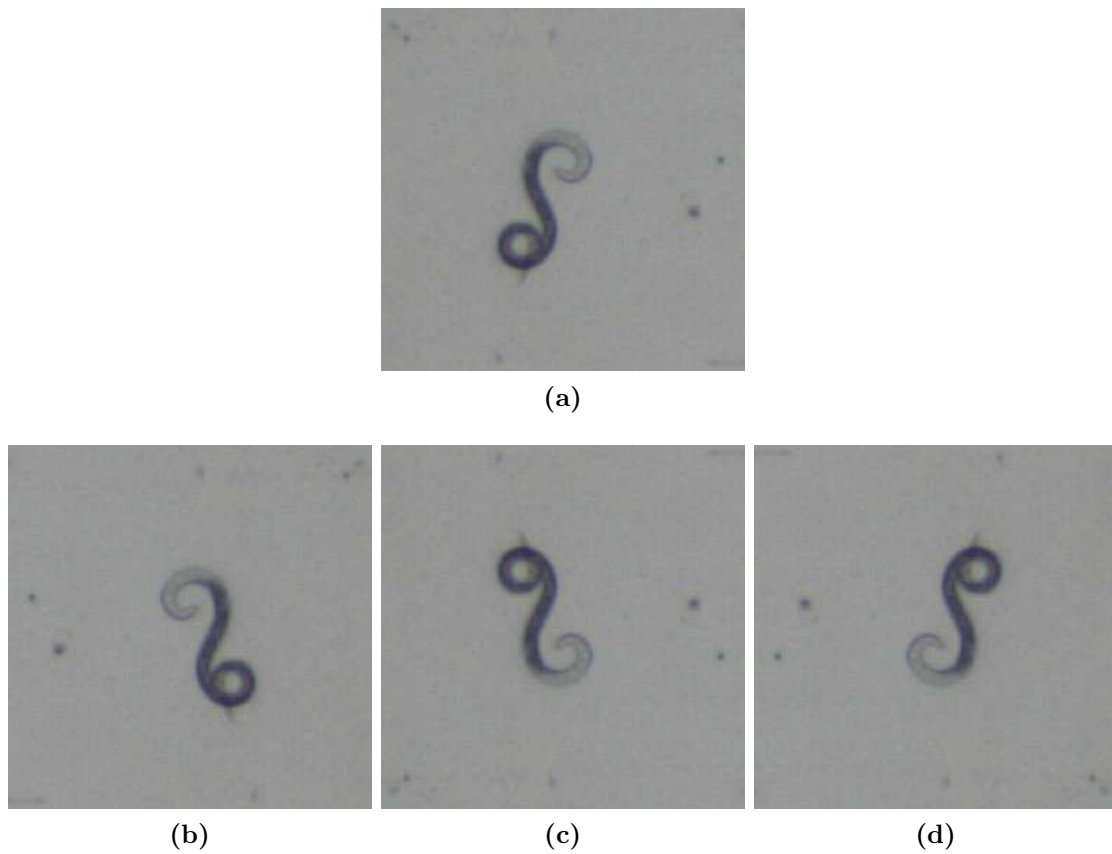
### 4.1.3. Experimento 3

El entrenamiento del modelo se lleva a cabo con la base de datos 3 (sección 3.2.3), donde la clase  $M$  se obtiene recortando en forma manual aquellas imágenes con la presencia de un nematodo a una resolución de  $256 \times 256$ . Al mismo tiempo se amplía o reduce el tamaño de la imagen y se rota ligeramente para esquivar ruido y otros nematodos. La clase  $S$  se obtiene cambiando el fondo negro por un fondo real con menos artefactos ruidosos y se adiciona ruido gaussiano al negativo de la máscara del nematodo (sección 3.2.2). El ruido gaussiano tiene la finalidad de ayudar a las redes neuronales a la identificación de la silueta de nematodo. Un ejemplo de ambas clases se ilustra en la figura 4.8.



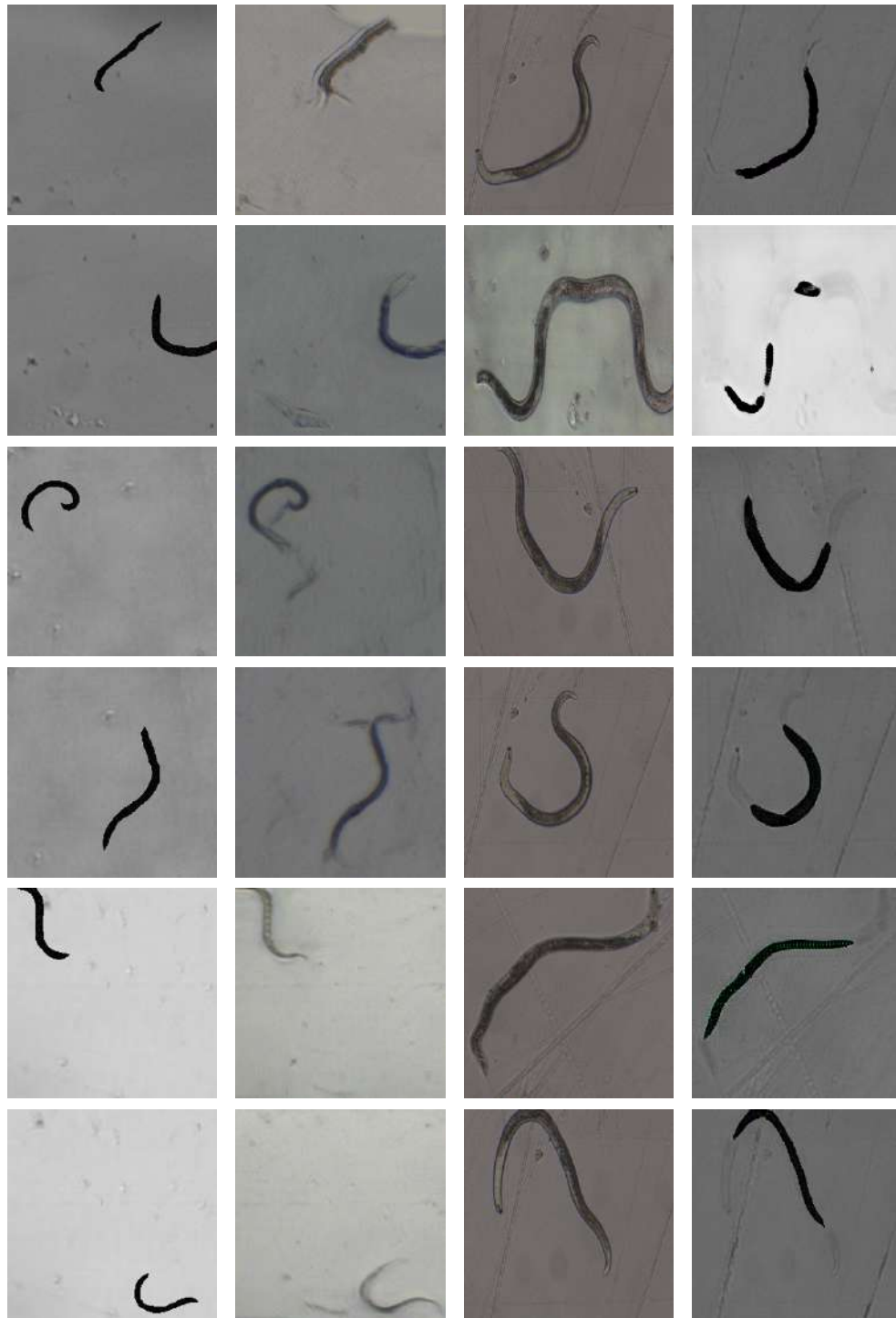
**Figura 4.8:** Ejemplos de la base de datos 3. (a) clase  $S$  (b) clase  $M$

En los experimentos 1 y 2 (secciones 4.1.1 y 4.1.2) la clase  $M$  se divide en 90,5 % para entrenamiento y 9,5 % para pruebas. Sin embargo, el experimento 3 a razón de la limpieza de datos resulta con una clase  $M$  dividida en 87,5 % para entrenamiento y 12,5 % para pruebas, por lo cual, solo al 22,5 % del subconjunto de entrenamiento se aplican operaciones de volteo a través del eje  $x$ , el eje  $y$  y luego a través de ambos ejes (sección 3.2.4) para completar los datos a 90,5 % de entrenamiento. La figura 4.9 muestra la aplicación del aumentado básico.



**Figura 4.9:** Ejemplos de los tipos de aumentado básico. (a) Dato original (b) Volteo horizontal (c) Volteo vertical (d) Volteo en ambas direcciones

La figura 4.10 muestra algunos ejemplos de imágenes sintetizadas durante el proceso de entrenamiento para seis distintas épocas. Dicho proceso de entrenamiento requirió 7 días 2 horas y 1 minuto. El realismo alcanzado en la época 50 es un indicador de que la red ha convergido y se puede prescindir de las 200 épocas.



**Figura 4.10:** Imágenes sintetizadas durante el entrenamiento del modelo del experimento 3. De izquierda a derecha: dato de silueta, dato de microscopía sintetizado, dato de microscopía real, dato de silueta sintetizado. De arriba a abajo: época 10, época 20, época 50, época 100, época 150 y época 200 (final)

La figura 4.11 muestra tres ejemplos de imágenes sintetizadas con el conjunto de datos de prueba. El realismo obtenido de las imágenes es muy parecido al experimento 2 (sección 4.1.2). Sin embargo, puede apreciarse que la forma del nematodo se confunde menos con artefactos ruidosos aunque sin conservar las dimensiones de la silueta, se nota un mayor sobre-ensanchamiento del cuerpo con deformidad en la cola y cabeza.



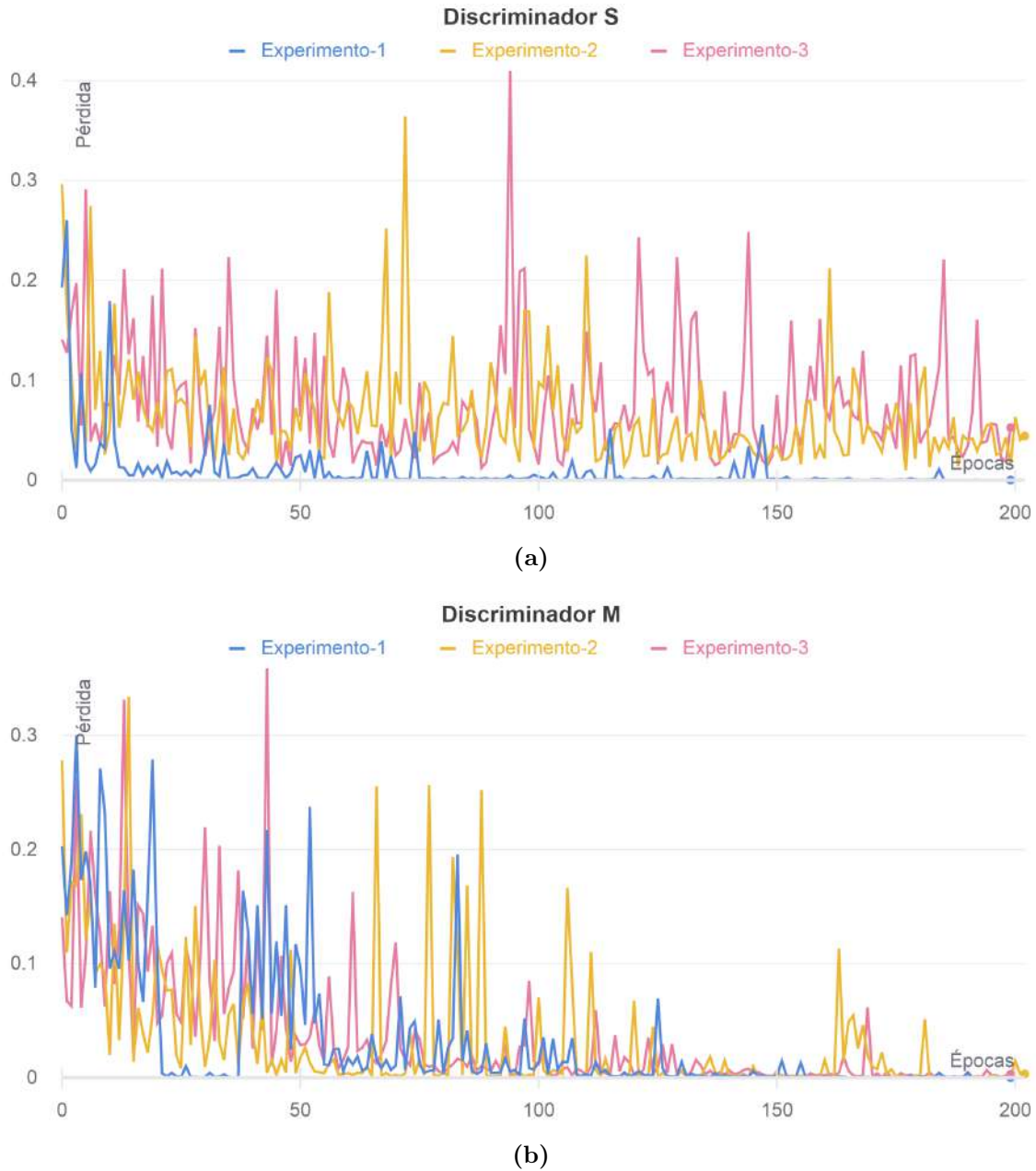
**Figura 4.11:** Imágenes sintetizadas evaluando el modelo entrenado del experimento 3 con un conjunto de datos de prueba. De izquierda a derecha: dato de silueta, dato de microscopía sintetizado, dato de silueta superpuesto en sintetizada. De arriba a abajo: resultado 1, resultado 2 y resultado 3

## 4.2. Análisis comparativo

En esta sección se realiza un análisis conjunto de todos los experimentos. Los gráficos del proceso de entrenamiento se despliegan en Weights & Biases [44].

Durante el proceso de entrenamiento se obtienen todas las métricas de la arquitectura CycleGAN obtenidas con el conjunto de datos de entrenamiento. La figura 4.12 muestra las métricas de mínimos cuadrados adversaria para la red discriminativa. Por el comportamiento descendente de los datos de pérdida para cada época, se puede distinguir que la red tarda las 200 épocas para lograr aprender a reconocer entre una imagen real y una falsa en la dirección de mapeo  $S$  a  $M$  (figura 4.12b), siendo el modelo del experimento 2 quien presenta más oscilaciones (está más confundido que los demás). Este descenso llega hasta cero, es decir, se ha alcanzado una alta minimización de la distancia entre los valores predichos y las etiquetas 0 y 1 (2.3). La dirección opuesta (figura 4.12a) solo alcanza

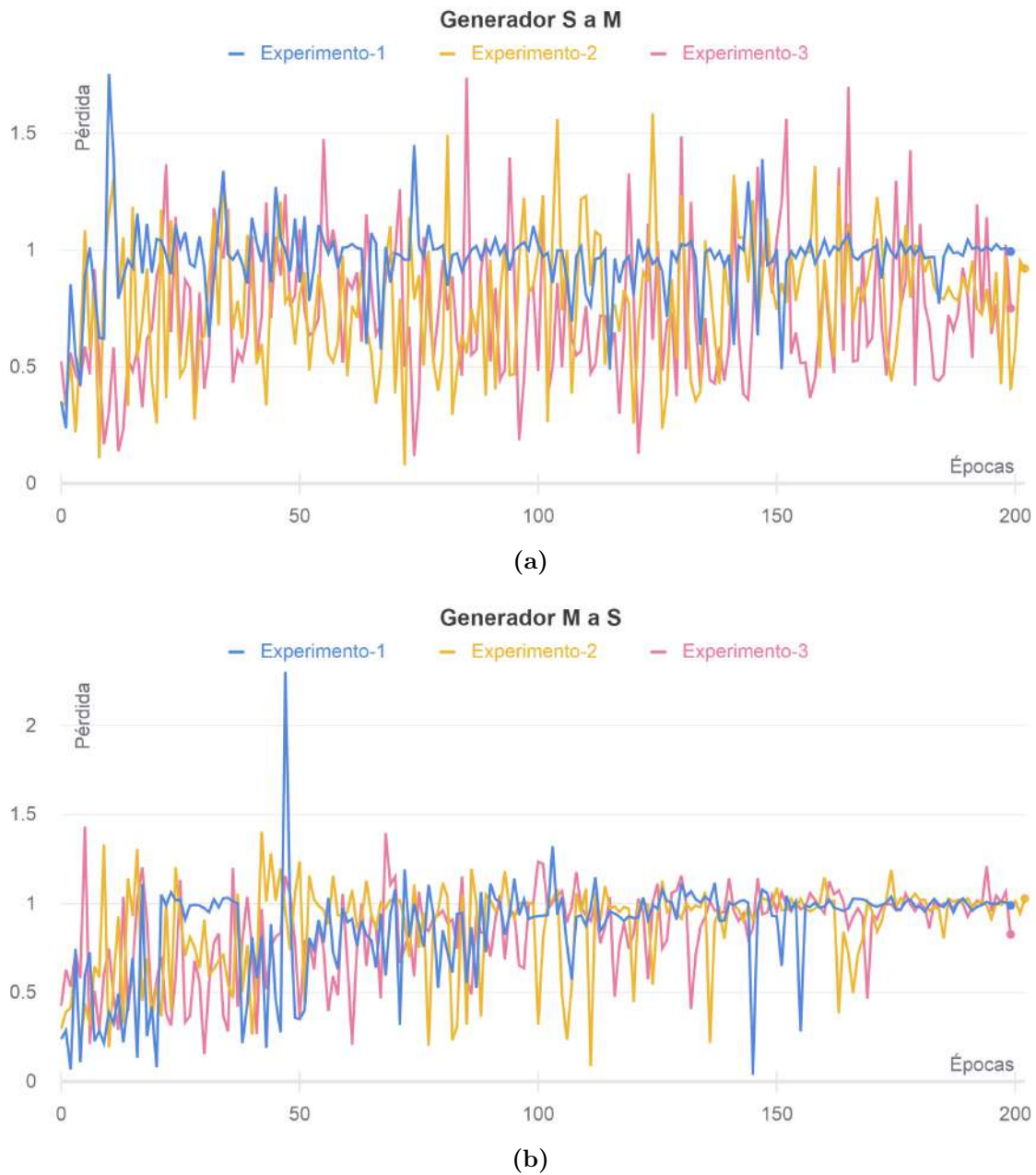
una alta minimización en el modelo del experimento 1, pero no es tan determinante en el objetivo de este proyecto ya que solo interesa mapear en la dirección  $S$  a  $M$ .



**Figura 4.12:** Pérdida del discriminador en función de las épocas para los experimentos 1 (sección 4.1.1), 2 (sección 4.1.2) y 3 (sección 4.1.3). (a) Discriminador para mapeo de  $M$  a  $S$  (b) Discriminador para mapeo de  $S$  a  $M$ .

El discriminador no debió haber minimizado tanto su pérdida. Esto significa que la métrica de mínimos cuadrados adversaria del modelo generativo, en la dirección de mapeo  $S$  a  $M$ , no logra aprender a generar imágenes lo suficientemente realistas; lo que se traduce en una muy baja minimización de la distancia de valores predichos y la etiqueta 1 (2.4). Lo anterior se muestra en la figura 4.13a, la pérdida de este generador está oscilando alrededor de 1 cuando debería caer a cero. Teniendo en cuenta este juego entre el discriminador y

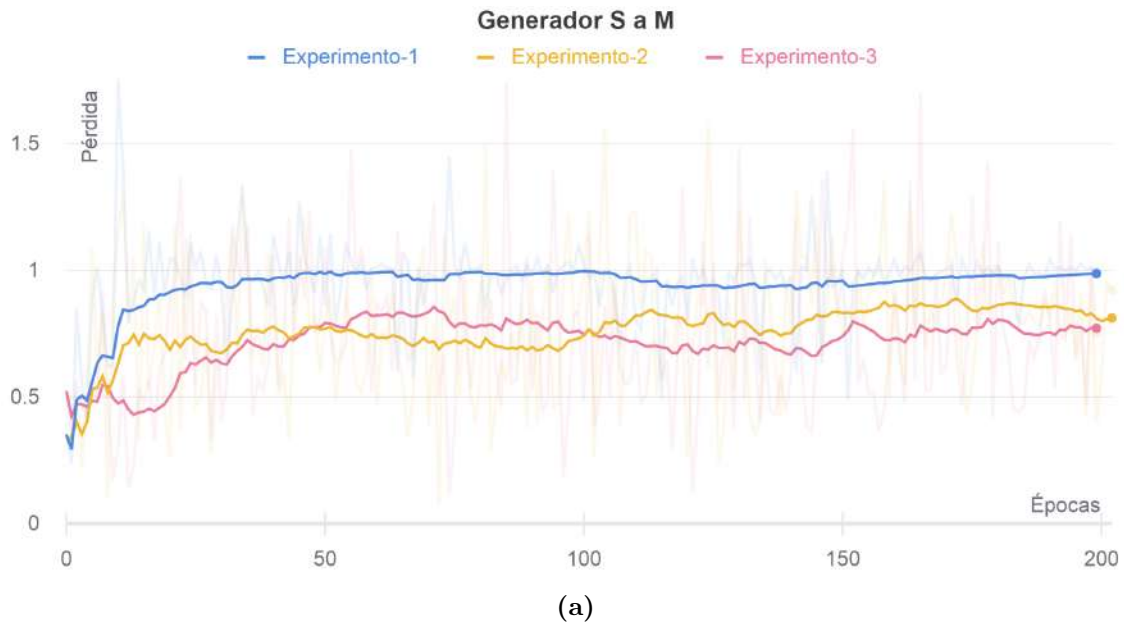
el generador, se comprende por qué el generador en la dirección opuesta (figura 4.13b) presenta pérdidas menores, aunque, significativamente altas.



**Figura 4.13:** Pérdida del generador en función de las épocas de los experimentos 1 (sección 4.1.1), 2 (sección 4.1.2) y 3 (sección 4.1.3). (a) Generador para mapeo de  $S$  a  $M$  (b) Generador para mapeo de  $M$  a  $S$ .

En la figura 4.14 se ha pasado un filtro a las curvas de la figura 4.13a para obtener su media y visualizar mejor el desempeño entre los experimentos 2 y 3. El modelo del experimento 3 logra que baje más esa pérdida del generador, por lo que tiene un mejor desempeño.

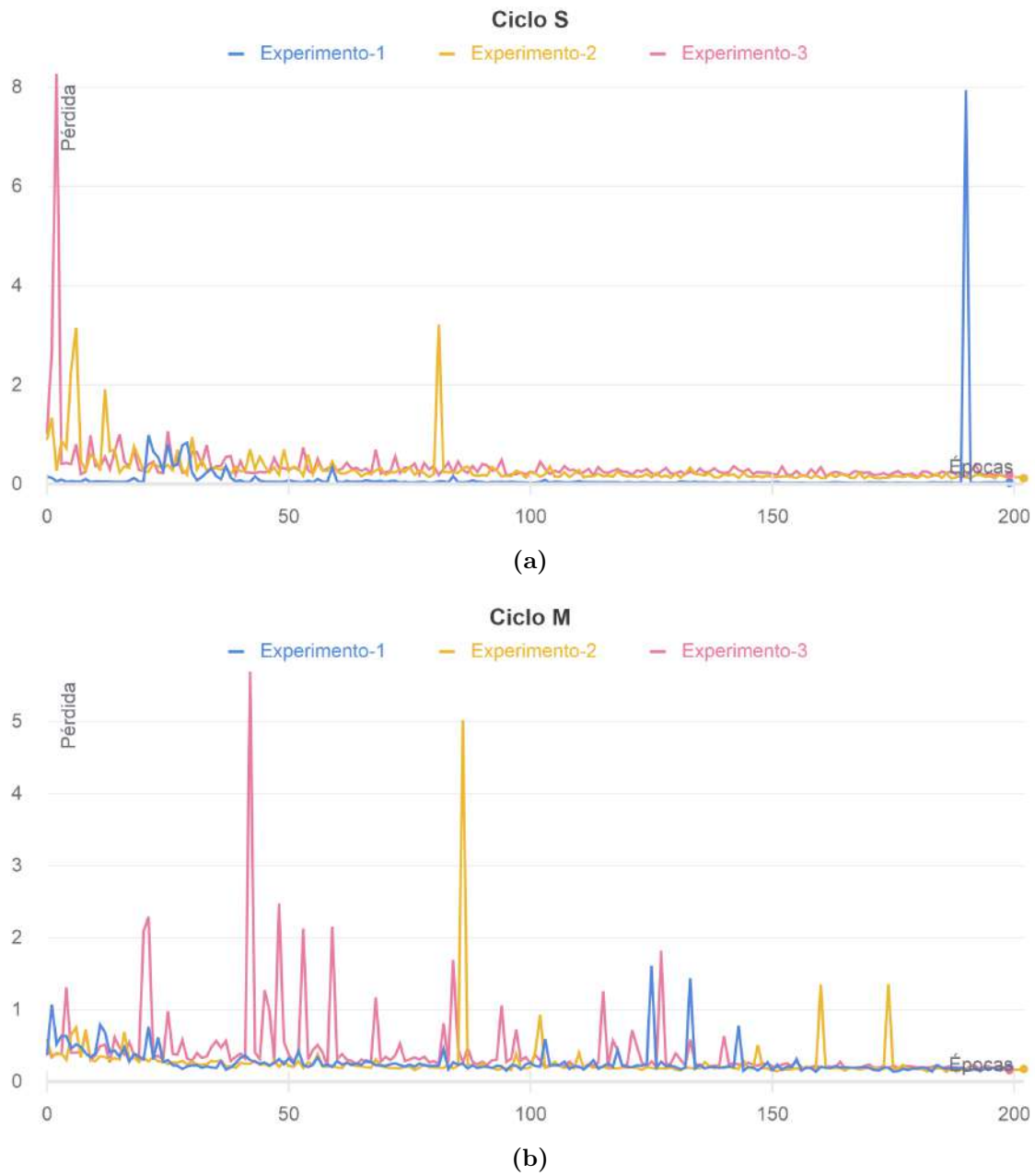




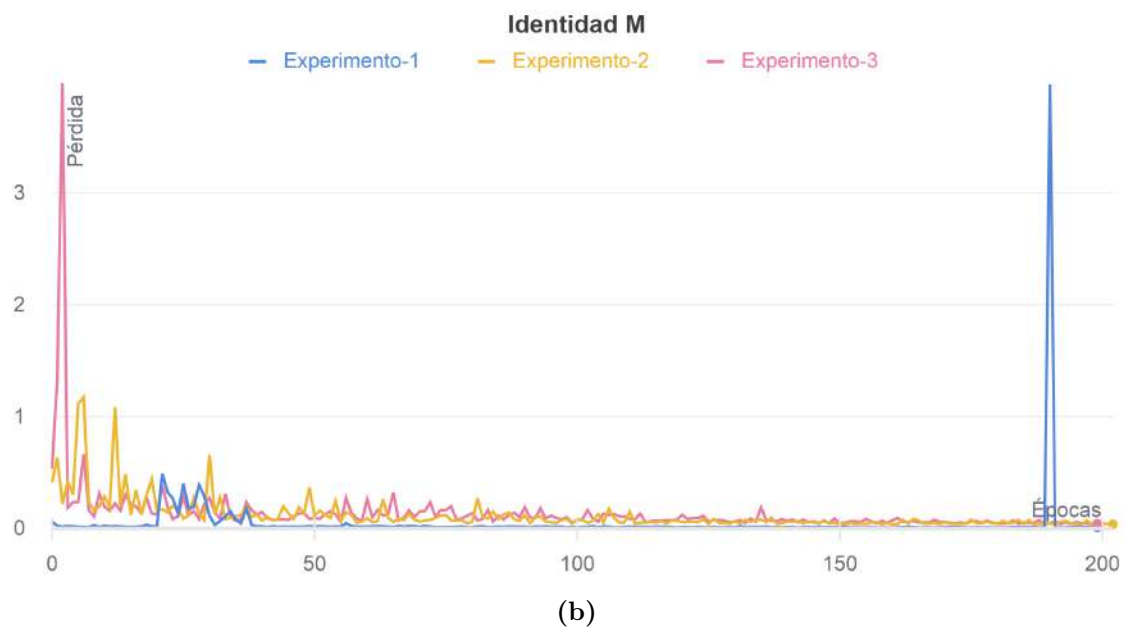
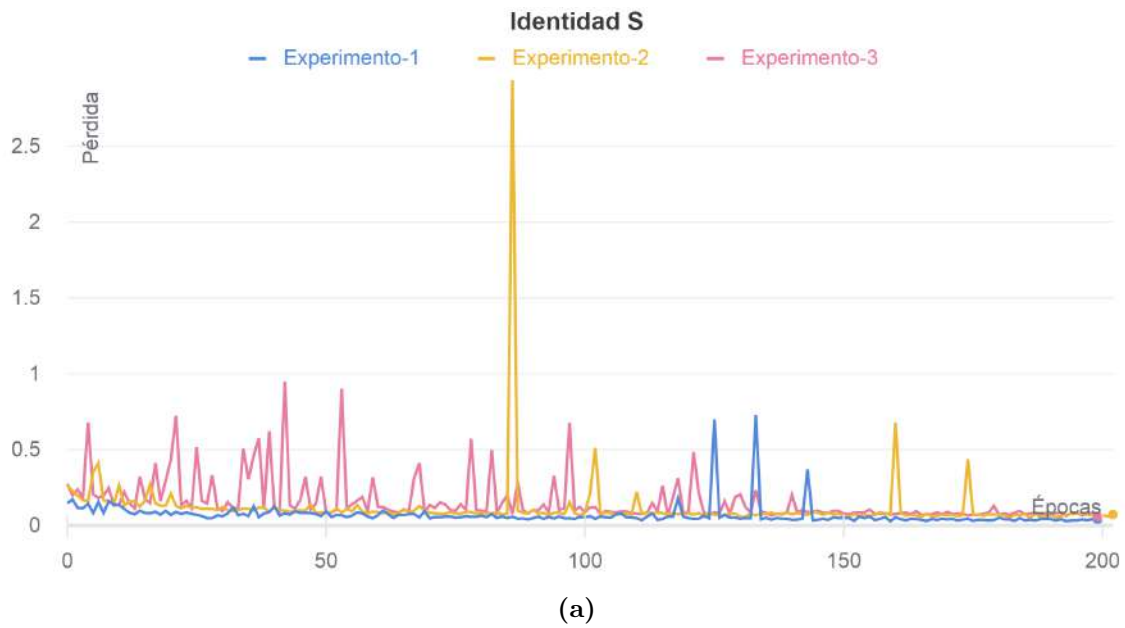
**Figura 4.14:** Pérdida del generador filtrada en función de las épocas de los experimentos 1 (sección 4.1.1), 2 (sección 4.1.2) y 3 (sección 4.1.3) para mapeo de  $S$  a  $M$

Por otra parte, la pérdida de consistencia de ciclo y la pérdida de identidad en las figuras 4.15a y 4.16b en el mapeo de  $S$  a  $M$  respectivamente, presentan un comportamiento muy parecido a lo largo de todo el entrenamiento. Esto se debe a que ambas realizan una diferencia de píxel entre imágenes que idealmente serían idénticas. El experimento 1 prácticamente no necesitó estas pérdidas, pero los experimentos 2 y 3 sí muestran una disminución progresiva y muy cercana a cero, lo que indica que el generador opuesto reconstruye bien las imágenes de siluetas; véase que el generador de  $M$  a  $S$  (figura 4.13b) alcanzó un mejor desempeño con respecto al generador de  $S$  a  $M$  (figura 4.13a).





**Figura 4.15:** Pérdida de consistencia de ciclo en función de las épocas para los experimentos 1 (sección 4.1.1), 2 (sección 4.1.2) y 3 (sección 4.1.3). (a) Consistencia de ciclo para mapeo de  $S$  a  $M$  (b) Consistencia de ciclo para mapeo de  $M$  a  $S$



**Figura 4.16:** Pérdida de identidad en función de las épocas de los experimentos 1 (sección 4.1.1), 2 (sección 4.1.2) y 3 (sección 4.1.3). (a) Identidad  $S$  (b) Identidad  $M$ .

### 4.2.1. Evaluación de discriminabilidad

Una vez realizado el entrenamiento completo del modelo es posible evaluar la capacidad discriminativa que aprendió el modelo discriminador para diferenciar entre una clase u otra, al alimentarla con imágenes que no haya visto antes.

La métrica utilizada para validar la efectividad del modelo discriminativo en el mapeo de  $S$  a  $M$ , es mediante una matriz de confusión (sección 2.4.4), que se implementa tomando en consideración el umbral de decisión aprendido por el discriminador para decidir si la imagen ingresada corresponde a una de microscopía real (valores mayores al umbral) o de microscopía sintetizada (valores menores al umbral). Este umbral será un valor en el rango de 0 a 1.

Entonces se busca el umbral de decisión que demuestra el mejor desempeño del modelo para clasificar las imágenes. Para ello, se alimentó el modelo discriminativo con 50 imágenes sintetizadas y 50 reales, obteniéndose varios umbrales de decisión posibles. Luego para cada umbral se calcula un par precisión y exhaustividad, los cuales se ordenan en las tablas 4.1 y 4.2, y corresponden a las configuraciones utilizadas para encontrar un frente de Pareto (sección 2.4.5).

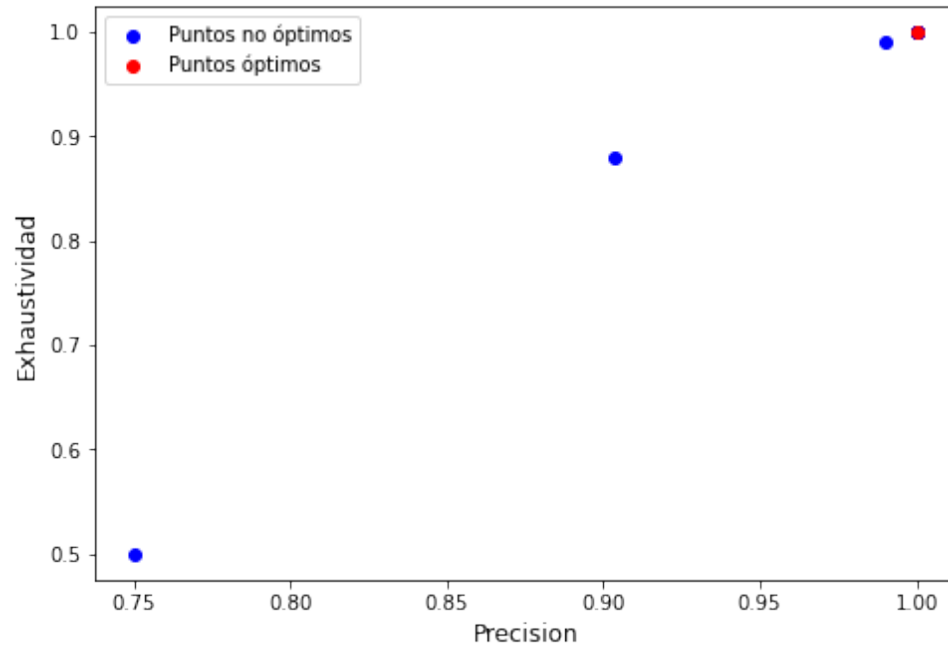
**Tabla 4.1:** Umbrales de decisión del modelo discriminativo con sus respectivos pares de precisión y exhaustividad del experimento 1 (sección 4.1.1)

Experimento 1					
Umbral	Precisión	Exhaustivi- dad	Umbral	Precisión	Exhaustivi- dad
0	0,75	0,5	<b>0,52631579</b>	<b>1</b>	<b>1</b>
0,05263158	1	1	0,57894737	1	1
0,10526316	1	1	0,63157895	1	1
0,15789474	1	1	0,68421053	1	1
0,21052632	1	1	0,73684211	1	1
0,26315789	1	1	0,78947368	1	1
0,31578947	1	1	0,84210526	1	1
0,36842105	1	1	0,89473684	1	1
0,42105263	1	1	0,94736842	0,99019608	0,99
0,47368421	1	1	1	0,90322581	0,88

**Tabla 4.2:** Umbrales de decisión del modelo discriminativo con sus respectivos pares de precisión y exhaustividad de los experimentos 2 (sección 4.1.2) (izquierda) y 3 (sección 4.1.3) (derecha)

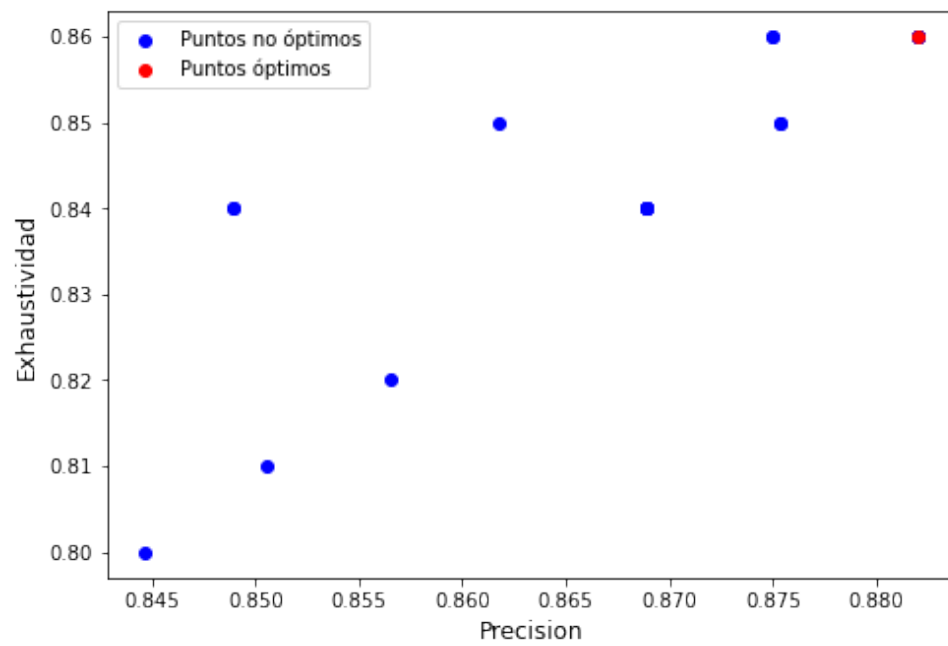
Experimento 2			Experimento 3		
Umbral	Precisión	Exhaustividad	Umbral	Precisión	Exhaustividad
0,14	0,844669	0,8	0,14	0,951626	0,95
0,1432	0,85052	0,81	0,1437	0,951626	0,95
0,1463	0,856506	0,82	0,1474	0,951626	0,95
0,1495	0,868924	0,84	0,1511	0,970188	0,97
0,1526	0,868924	0,84	0,1547	0,970188	0,97
0,1558	0,868924	0,84	0,1584	0,98	0,98
0,1589	0,868924	0,84	<b>0,1621</b>	<b>0,990196</b>	<b>0,99</b>
0,1621	0,868924	0,84	0,1658	0,990196	0,99
0,1653	0,875375	0,85	0,1695	0,990196	0,99
0,1684	0,875375	0,85	0,1732	0,990196	0,99
<b>0,1716</b>	<b>0,882003</b>	<b>0,86</b>	0,1768	0,990196	0,99
0,1747	0,882003	0,86	0,1805	0,990196	0,99
0,1779	0,882003	0,86	0,1842	0,990196	0,99
0,1811	0,882003	0,86	0,1879	0,990196	0,99
0,1842	0,882003	0,86	0,1916	0,990196	0,99
0,1874	0,875	0,86	0,1953	0,990196	0,99
0,1905	0,875	0,86	0,1989	0,990196	0,99
0,1937	0,86172	0,85	0,2026	0,980769	0,98
0,1968	0,848933	0,84	0,2063	0,971698	0,97
0,2	0,848933	0,84	0,21	0,962963	0,96

En las figuras 4.17, 4.18 y 4.19 se tienen las veinte configuraciones de precisión/exhaustividad utilizadas para encontrar los puntos que denotan el frente de Pareto. En todos los casos para cada experimento solo se obtuvo un punto en el frente de Pareto, lo que significa que solo una configuración de precisión/exhaustividad representa el mejor desempeño del modelo a un determinado umbral, a saber: 0,5 para el experimento 1, 0,1716 para el experimento 2 y 0,1621 para el experimento 3.



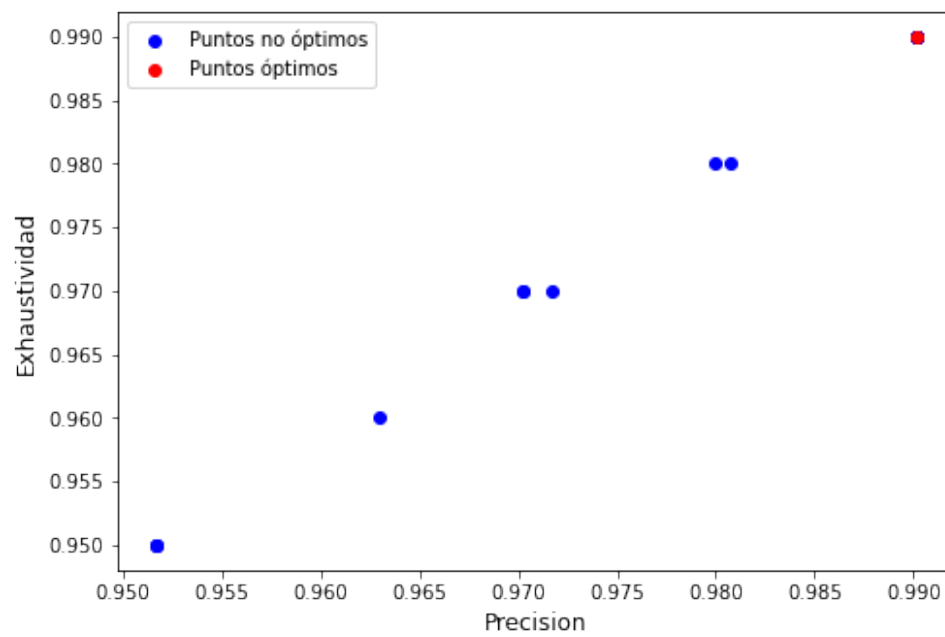
(a)

**Figura 4.17:** Espacio precisión/exhaustividad de varios umbrales del experimento 1 (sección 4.1.1) y el frente de Pareto conformado por los puntos rojos.



(a)

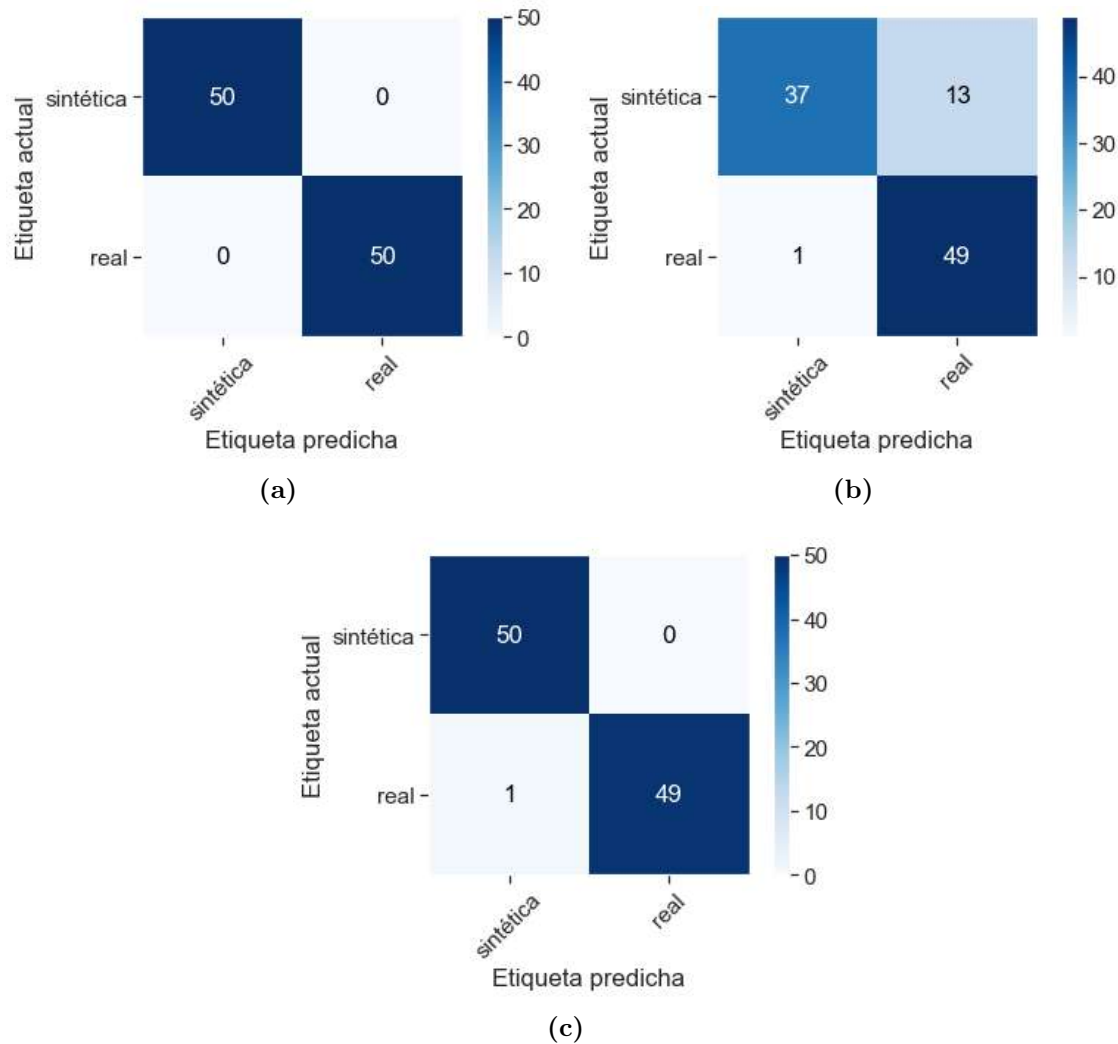
**Figura 4.18:** Espacio precisión/exhaustividad de varios umbrales del experimento 2 (sección 4.1.2) y el frente de Pareto conformado por los puntos rojos.



(a)

**Figura 4.19:** Espacio precisión/exhaustividad de varios umbrales del experimento 3 (sección 4.1.3) y el frente de Pareto conformado por los puntos rojos.

Se obtienen las matrices de confusión mostradas en la figura 4.20. En el modelo discriminativo de los experimentos 1 y 3 existe un umbral de decisión en el que detecta fácilmente si una imagen es sintética o real. Por el contrario en el experimento 2, aunque las 50 imágenes reales también son predichas como reales, de las 50 sintéticas 37 son correctamente predichas como sintéticas y 13 logran confundir al discriminador prediciéndolas como reales. En una GAN se espera que este desempeño alcance alrededor del 50 %.



**Figura 4.20:** Matrices de confusión del modelo discriminativo. (a) Del Experimento 1 (sección 4.1.1) para el umbral mayor (b) Experimento 2 (sección 4.1.2) y (c) Experimento 3 (sección 4.1.3).

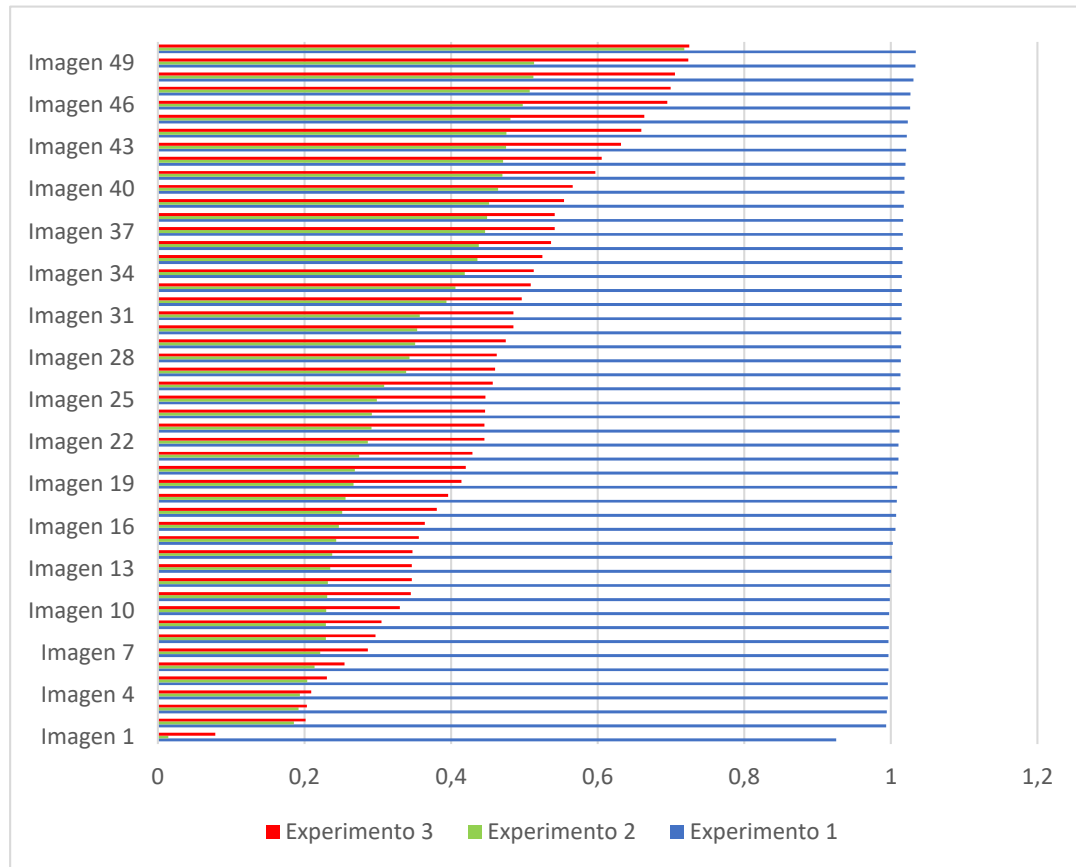
Particularmente, el amplio rango de umbrales especificados en la tabla 4.1 con un desempeño perfecto junto con la matriz de confusión de la figura 4.20a reflejan el bajo realismo alcanzado por el modelo generativo del experimento 1 y el desarrollo de un modelo discriminador robusto que clasifica cualquier imagen exitosamente.

La exactitud y precisión alcanzada por los modelos se presentan en la tabla 4.3. Efectivamente, se confirma que el experimento 2 es el menos preciso, lo que lo hace el experimento más cercano al 50 % de desempeño.

**Tabla 4.3:** Métricas de exactitud y precisión de los modelos discriminativos

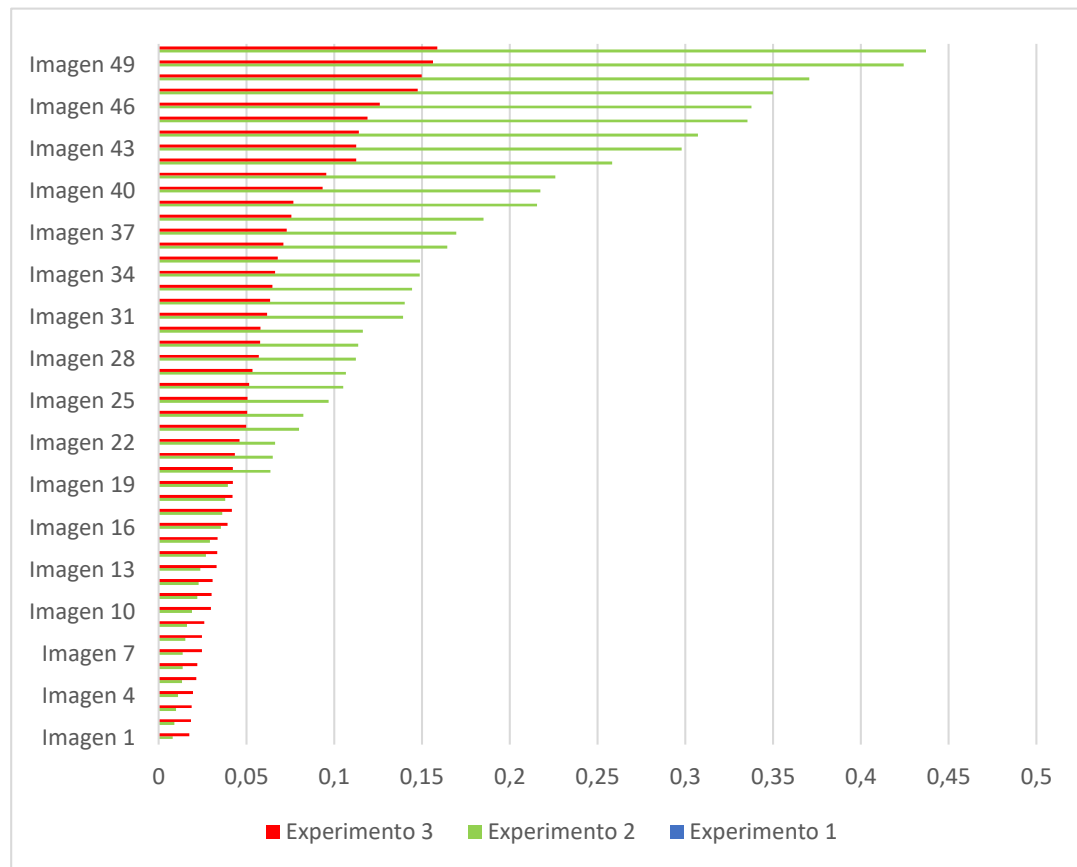
	Experimento 1	Experimento 2	Experimento 3
Exactitud	1	0,86	0,99
Precisión	1	0,97	0,98

Los valores de la salida del discriminador para cada una de las 50 imágenes evaluadas en el caso de las reales, se muestra en la figura 4.21. En este particular, hay una notoria diferencia entre los experimentos 1 y 3; lo que justifica que la matriz de confusión del experimento 3 haya confundido un dato real con uno sintético. Y en el otro caso de las imágenes sintéticas se ilustra en la figura 4.22. El experimento 2 muestra el comportamiento deseable de una GAN en donde, tanto en el caso de datos reales como sintéticos, se mantiene una cuarta parte de las imágenes con valores predichos cercanos o iguales a 0,5 (discriminador inseguro).



**Figura 4.21:** Salida del discriminador para la clasificación de 50 imágenes de microscopía reales





**Figura 4.22:** Salida del discriminador para la clasificación de 50 imágenes de microscopía sintetizadas

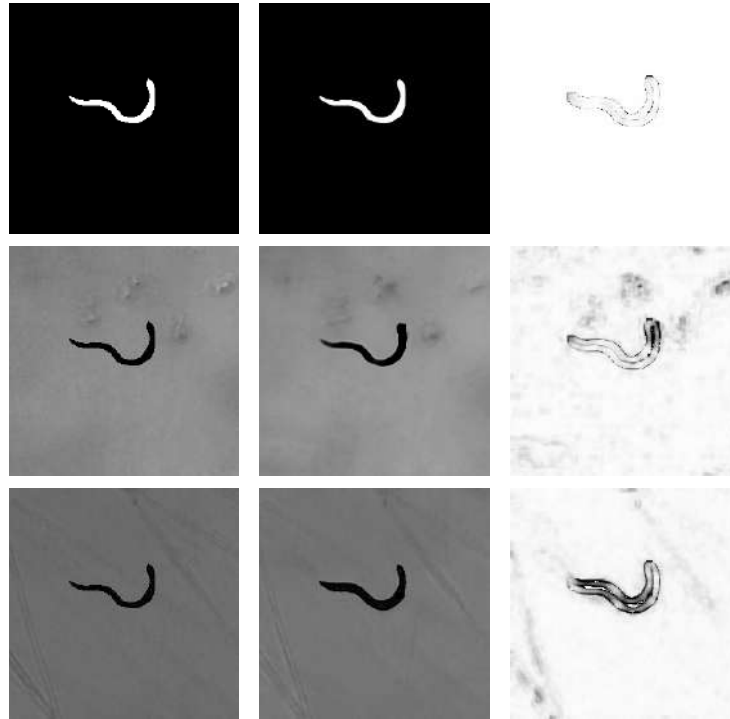
### 4.2.2. Evaluación de traducciones

Durante su entrenamiento, el modelo CycleGAN está utilizando la pérdida de consistencia de ciclo (figura 4.15) como la diferencia entre *real S* y *falsa S* (figura 3.6) que pretende minimizar. Para evaluar cuantitativamente el nivel de consistencia de ciclo que alcanzó el modelo, se mide la similitud entre imágenes de entrada del conjunto de datos de prueba e imágenes de salida después de un ciclo de traducciones. Se utiliza la métrica SSIM y los resultados se muestran en la tabla 4.4. De acuerdo con los valores obtenidos, el modelo del experimento 1 se desempeña mejor en la traducción *S* a *M* que los experimentos 2 y 3.

**Tabla 4.4:** Métrica SSIM para la evaluación del nivel de consistencia de ciclo alcanzado por el modelo

	Experimento y dirección de traducción		
	1 $S \rightarrow M$	2 $S \rightarrow M$	3 $S \rightarrow M$
SSIM	0,99707	0,9539	0,94551

En la figura 4.23 se presentan las imágenes usadas en esta evaluación junto con su respectiva diferencia.



**Figura 4.23:** Imágenes de evaluación de consistencia de ciclo tomadas del conjunto de datos de prueba. De izquierda a derecha: dato *real S*, dato *falso S*, diferencia entre *real S* y *falso S*. De arriba a abajo: experimento 1, experimento 2 y experimento 3

# Capítulo 5

## Conclusiones

El presente trabajo propone una estrategia de síntesis de imágenes de microscopía con presencia de nematodos. Consiste en un aumentado de datos que desarrolla un modelo neuronal convolucional de aprendizaje profundo: CycleGAN. El modelo CycleGAN traduce entre conjuntos de imágenes sin emparejar. Este proceso permite la generación de imágenes realistas de microscopía partiendo de una silueta de nematodo que es la anotación deseada y es determinado por las funciones de pérdida de dos GAN: adversaria de mínimos cuadrados, consistencia de ciclo e identidad.

Otras contribuciones de este trabajo es mostrar una metodología para preparar el set de datos a través de tres experimentos que modifican características y niveles de ruido de manera distinta. Asimismo se exponen criterios de selección del modelo neuronal convolucional para la transferencia de estilo y se plasma su implementación. Finalmente se utilizan métricas cualitativas y cuantitativas para evaluar el desempeño del modelo.

El aumentado de datos por medio del modelo CycleGAN no funciona con datos de siluetas que tienen fondos completamente negros y máscaras de la silueta del nematodo completamente blancas. Sin embargo, la incorporación de características de fondo y ruido a las siluetas, provoca que el modelo sí funcione, de tal manera que, además de las modificaciones anteriores, al disminuir la presencia de artefactos ruidos en las imágenes de microscopía, se obtiene el mejor desempeño del modelo de los tres experimentos realizados.

Un generador logra su desempeño ideal cuando el discriminador llega a un 50 % de exactitud y 50 % de precisión (no está nunca seguro si lo que “ve” es o no sintético) lo que da espacio de mejora para el diseño del generador. El modelo discriminativo obtenido más cercano al desempeño ideal presenta una métrica de precisión de 97 % y exactitud de 86 %, cuando se incorporan solo características de fondo a las siluetas, es decir, el generador no logra confundir al discriminador. Lo anterior se evidencia en las imágenes sintetizadas ya que, la mayoría de imágenes tienen la cola y cabeza fuera del área delimitada por la máscara de las siluetas de nematodos y algunos presentan deformidad.

El modelo generativo logra sintetizar imágenes altamente realistas desde la percepción del ojo humano, pero no guarda las mismas dimensiones delimitadas por las máscaras

de siluetas de nematodos, por tanto, no fue posible crear una base de datos anotada. Al ser las siluetas más delgadas que los nematodos sintetizados no hay una correspondencia entre las imágenes, así que no hay etiquetas.

Si bien es cierto el modelo generativo de la CycleGAN no logra confundir lo suficiente al discriminador, este trabajo presenta un primer acercamiento a la síntesis de imágenes de microscopía de nematodos, ofreciendo una línea base de comparación para trabajos futuros.

Como trabajo futuro del proyecto se debe mejorar el conjunto de datos de siluetas para que el generador logre confundir más al discriminador: puede añadirse naturalismo al contenido de la silueta. También se debe experimentar con otras arquitecturas similares para comparar la calidad de los resultados; puede aplicarse una traducción de imagen a imagen emparejada ya que se cuenta con una base de datos de nematodos segmentados.

# Bibliografía

- [1] R. Flores-Agreda, G. Muñoz, C. P. De Salvo y A. Alcaraz-Rivero, «Análisis de políticas agropecuarias en Centroamérica y República Dominicana,» Banco Interamericano de Desarrollo, inf. téc., 2020 (vid. pág. 1).
- [2] C. R. CAC, S. SICA y C. R. IICA, «Política agropecuaria de la región SICA: 2019-2030,» 2019 (vid. pág. 1).
- [3] C. R. IICA, «La agricultura de Costa Rica: Situación al 2010, su evolución y prospectiva,» *ICAA: Política de Estado para el Sector Agroalimentario y el Desarrollo Rural costarricense*, 2010 (vid. pág. 1).
- [4] M. Á. Taylor-López, «Estrategia para incorporar información previa de forma en segmentación semántica de estructuras vermiformes,» Documento de tesis para optar por grado académico de Maestría en Electrónica, 2020 (vid. págs. 1-3).
- [5] S. Sánchez-Moreno y M. Talavera, «Los nematodos como indicadores ambientales en agroecosistemas,» *Ecosistemas*, vol. 22, n.º 1, págs. 50-55, abr. de 2013. DOI: [10.7818/ECOS.2013.22-1.09](https://doi.org/10.7818/ECOS.2013.22-1.09) (vid. págs. 1, 2).
- [6] G. Rivera-Coto, *Conceptos introductorios a la fitopatología*, 1.ª ed. Universidad estatal a distancia. San Jose Costa Rica, 2007 (vid. pág. 1).
- [7] R. WingChing-Jones, L. Salazar-Figueroa, L. Flores-Sánchez y A. Rojas-Bourrillón, «Reconocimiento de nemátodos en pastos tropicales en las comunidades de Sucre y San Vicente, cantón de San Carlos,» *Agronomía Costarricense*, vol. 32, n.º 2, 2008 (vid. pág. 1).
- [8] J. Jiménez-Chavarría, «SegNema: Nematode segmentation strategy in digital microscopy images using deep learning and shape models,» Tesis de maestría, Escuela de Ingeniería en Computación, ITCR, 2019 (vid. págs. 1-3).
- [9] A. Esquivel, *Morfología de los nematodos. Curso de identificación*, 2015 (vid. pág. 1).
- [10] J. Arroyo-Hernández, «Modelo evolutivo de forma para el ajuste de organismos vermiformes en imágenes digitales,» Tesis doct., Escuela de Ingeniería en Computación, ITCR, 2021 (vid. págs. 2, 12, 13, 15, 21, 29).
- [11] A. Greenblum, R. Sznitman, P. Fua, P. E. Arratia y J. Sznitman, «Caenorhabditis elegans segmentation using texture-based models for motility phenotyping,» *IEEE transactions on bio-medical engineering*, 2014 (vid. pág. 2).

- [12] C. Marcellino, J. Gut, K. C. Lim, R. Singh, J. McKerrow y J. Sakanari, «WormAssay: A Novel Computer Application for Whole-Plate Motion-based Screening of Macroscopic Parasites,» *PLoS neglected tropical diseases*, 2012 (vid. pág. 2).
- [13] M. Zhan, M. M. Crane, E. V. Entchev, A. Caballero, D. A. Fernandes de Abreu, Q. Ch'ng y H. Lu, «Automated processing of imaging data through multi-tiered classification of biological structures illustrated using *ceanorhabditis elegans*,» *PLoS computational biology*, 2015 (vid. pág. 2).
- [14] C. Shorten y T. M. Khoshgoftaar, «A survey on Image Data Augmentation for Deep Learning,» *Big Data*, vol. 6, n.º 60, 2019 (vid. págs. 2, 5-8, 10, 11, 14).
- [15] X. Lu, Y. Wang, S. Fung y X. Qing, «I-Nema: A Biological Image Dataset for Nematode Recognition,» *arXiv preprint arXiv:2103.08335*, 2021 (vid. pág. 3).
- [16] L. Chen, M. Strauch, M. Daub, X. Jiang, M. Jansen, H.-G. Luigs, S. Schultz-Kuhlmann, S. Krüssel y D. Merhof, «A CNN framework based on line annotations for detecting nematodes in microscopic images,» en *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, IEEE, 2020, págs. 508-512 (vid. pág. 3).
- [17] J.-Y. Zhu, T. Park, P. Isola y A. A. Efros, *Unpaired image-to-image translation using cycle-consistent adversarial networks*, 2017 (vid. págs. 3, 9-11, 24).
- [18] J. Berná-García, «Generación de imágenes anotadas para el entrenamiento de modelos automáticos de análisis de anomalías,» Documento de tesis para optar por Grado Ingeniería Informática, 2020 (vid. págs. 5, 7-9).
- [19] Q. Wen, L. Sun, X. Song, J. Gao, X. Wang y H. Xu, «Time Series Data Augmentation for Deep Learning: A Survey,» *CoRR*, vol. abs/2002.12478, 2020. arXiv: 2002.12478. dirección: <https://arxiv.org/abs/2002.12478> (vid. pág. 5).
- [20] J. L. Leevy, T. M. Khoshgoftaar, R. A. Bauder y N. Seliya, «A survey on addressing high-class imbalance in big data,» *Big Data*, vol. 5, n.º 42, págs. 1-30, 2018 (vid. pág. 5).
- [21] S. Zhou, E. Zelikman y E. Zhou, *Apply Generative Adversarial Networks (GANs)*, Available at <http://https://www.coursera.org/learn/apply-generative-adversarial-networks-gans> (2021/06/11) (vid. págs. 6, 7, 9-11, 25-28).
- [22] Y. Lecun, L. Bottou, Y. Bengio y P. Haffner, «Gradient-based learning applied to document recognition,» *IEEE*, vol. 86, n.º 11, págs. 2278-2324, 1998. DOI: 10.1109/5.726791 (vid. pág. 5).
- [23] G. E. Hinton, A. Krizhevsky e I. Sutskever, «Imagenet classification with deep convolutional neural networks,» *Advances in neural information processing systems*, vol. 25, págs. 1106-1114, 2012 (vid. pág. 5).
- [24] H. Inoue, «Data Augmentation by Pairing Samples for Images Classification,» *CoRR*, vol. abs/1801.02929, 2018. arXiv: 1801.02929. dirección: <http://arxiv.org/abs/1801.02929> (vid. págs. 6, 14).
- [25] T. DeVries y G. W. Taylor, «Dataset augmentation in feature space,» *arXiv preprint arXiv:1702.05538*, 2017 (vid. pág. 7).

- [26] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville e Y. Bengio, «Generative adversarial networks,» *arXiv preprint arXiv:1406.2661*, 2014 (vid. págs. 8, 10).
- [27] P. Isola, J.-Y. Zhu, T. Zhou y A. A. Efros, «Image-to-Image Translation with Conditional Adversarial Networks,» *CoRR*, vol. abs/1611.07004, 2016. arXiv: 1611.07004. dirección: <http://arxiv.org/abs/1611.07004> (vid. pág. 10).
- [28] P. Sangkloy, J. Lu, C. Fang, F. Yu y J. Hays, «Scribbler: Controlling Deep Image Synthesis with Sketch and Color,» *CoRR*, vol. abs/1612.00835, 2016. arXiv: 1612.00835. dirección: <http://arxiv.org/abs/1612.00835> (vid. pág. 10).
- [29] L. Karacan, Z. Akata, A. Erdem y E. Erdem, «Learning to Generate Images of Outdoor Scenes from Attributes and Semantic Layouts,» *CoRR*, vol. abs/1612.00215, 2016. arXiv: 1612.00215. dirección: <http://arxiv.org/abs/1612.00215> (vid. pág. 10).
- [30] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford y X. Chen, «Improved Techniques for Training GANs,» *CoRR*, vol. abs/1606.03498, 2016. arXiv: 1606.03498. dirección: <http://arxiv.org/abs/1606.03498> (vid. pág. 10).
- [31] L. A. Gatys, A. S. Ecker y M. Bethge, «A Neural Algorithm of Artistic Style,» *CoRR*, vol. abs/1508.06576, 2015. arXiv: 1508.06576. dirección: <http://arxiv.org/abs/1508.06576> (vid. pág. 12).
- [32] D. Liang, F. Yang, T. Zhang y P. Yang, «Understanding Mixup Training Methods,» *IEEE Access*, vol. 6, págs. 58 774-58 783, 2018. DOI: 10.1109/ACCESS.2018.2872698 (vid. pág. 14).
- [33] J. M. Kinser, *Image Operators: Image Processing in Python*. CRC Press, 2018 (vid. pág. 15).
- [34] O. Tmenova, R. Martin y L. Duong, «CycleGAN for style transfer in X-ray angiography,» *International journal of computer assisted radiology and surgery*, vol. 14, n.º 10, págs. 1785-1794, 2019 (vid. págs. 16, 18, 25).
- [35] Y. Yang, Y. Li, C. Fermüller e Y. Aloimonos, «Neural Self Talk: Image Understanding via Continuous Questioning and Answering,» *CoRR*, vol. abs/1512.03460, 2015. arXiv: 1512.03460. dirección: <http://arxiv.org/abs/1512.03460> (vid. pág. 16).
- [36] R. Zhang, P. Isola y A. A. Efros, «Colorful Image Colorization,» *CoRR*, vol. abs/1603.08511, 2016. arXiv: 1603.08511. dirección: <http://arxiv.org/abs/1603.08511> (vid. pág. 17).
- [37] M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger y H. Greenspan, «GAN-based Synthetic Medical Image Augmentation for increased CNN Performance in Liver Lesion Classification,» *CoRR*, vol. abs/1803.01229, 2018. arXiv: 1803.01229. dirección: <http://arxiv.org/abs/1803.01229> (vid. pág. 17).

- [38] V. Starovoytov, E. Eldarova y K. T. Iskakov, «Comparative analysis of the SSIM index and the pearson coefficient as a criterion for image similarity,» *Eurasian Journal of Mathematical and Computer Applications*, vol. 8, n.º 1, págs. 76-90, 2020 (vid. pág. 17).
- [39] H. Nishar, N. Chavanke y N. Singhal, «Histopathological Stain Transfer Using Style Transfer Network with Adversarial Loss,» en *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2020, págs. 330-340 (vid. pág. 17).
- [40] R. Muñoz-Bocanegra, «Aprendizaje profundo en dispositivo portable,» Documento de tesis para optar al título de ingeniero mecatrónico, 2019 (vid. pág. 18).
- [41] J. Acosta, À. Nebot y J. M. Fuertes, «Método multiobjetivo de aprendizaje para razonamiento inductivo difuso,» 2006 (vid. pág. 19).
- [42] V. Nanda, S. V. Belure y O. M. Shir, «Searching for the Pareto frontier in multi-objective protein design,» *Biophysical Reviews*, vol. 9, págs. 339-344, 2017 (vid. pág. 19).
- [43] E. Bisong, «Google Colaboratory,» en *Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners*. Berkeley, CA: Apress, 2019, págs. 59-64, ISBN: 978-1-4842-4470-8. DOI: [10.1007/978-1-4842-4470-8\\_7](https://doi.org/10.1007/978-1-4842-4470-8_7). dirección: [https://doi.org/10.1007/978-1-4842-4470-8\\_7](https://doi.org/10.1007/978-1-4842-4470-8_7) (vid. pág. 30).
- [44] L. Biewald, *Experiment Tracking with Weights and Biases*, Software available from wandb.com, 2020. dirección: <https://www.wandb.com/> (vid. pág. 39).