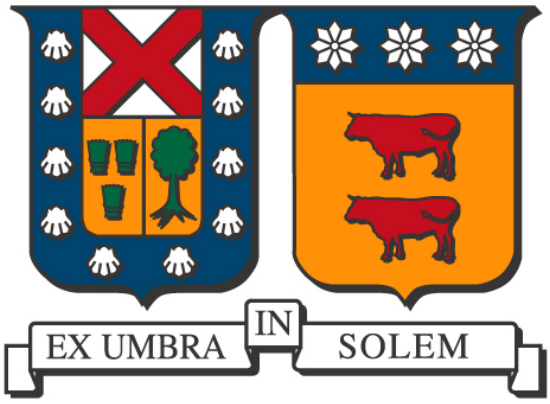
2021-04

EVALUACIÓN DE REDES NEURONALES ARTIFICIALES APLICADAS A SÚPER-RESOLUCIÓN EN VIDEO

GONZÁLEZ YÁÑEZ, JAVIER IGNACIO

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE ELECTRÓNICA
VALPARAÍSO - CHILE



”EVALUACIÓN DE REDES NEURONALES ARTIFICIALES APLICADAS A SÚPER-RESOLUCIÓN EN VIDEO”

JAVIER IGNACIO GONZÁLEZ YÁÑEZ

MEMORIA DE TITULACIÓN PARA OPTAR AL TÍTULO DE INGENIERO CIVIL EN ELECTRÓNICA

Profesor Guía: Gonzalo Carvajal B.
Correferente: Francisca Parra R.

Abril - 2021

Índice de Contenidos

Índice de Figuras

Índice de Tablas

Capítulo 1

Introducción

Este informe reporta los principales resultados asociados al trabajo de memoria de titulación asociado a la exploración de técnicas de procesamiento de imágenes aplicadas a la súper-resolución por medio de redes neuronales convolucionales. Específicamente, durante el desarrollo de esta memoria se estudiaron técnicas propuestas en la literatura reciente para aumentar la resolución de imágenes mediante redes neuronales con aprendizaje supervisado. A partir de una revisión bibliográfica y pruebas preliminares con distintas técnicas reportadas en la literatura (realizado y reportado en la asignatura previa a la memoria de título), se seleccionaron dos estructuras de redes convolucionales para su implementación y evaluación de desempeño en un contexto práctico, evaluando este mediante diversos casos de prueba para determinar la factibilidad de aplicación en distintos contextos.

Este capítulo presenta el contexto y la motivación para el desarrollo de este trabajo, especifica el problema a resolver, los objetivos, alcances y contribuciones de esta memoria de título, y finalmente presenta la organización del resto del informe.

1.1. Motivación y contexto

El trabajo desarrollado en esta memoria de título surge a partir de un requerimiento planteado por estamentos de la Fuerza Aérea de Chile (FACh), en el contexto de explorar técnicas modernas de procesamiento de imágenes para mejorar la resolución de imágenes capturadas con diversos tipos de sensores ópticos aerotransportados (incorporados en satélites, aeronaves, drones controlados remotamente, etc.) mediante post-procesamiento por software de la información obtenida desde el sensor.

El uso de imágenes obtenidas por medio de sensores ópticos aerotransportados es un recurso ampliamente utilizado por distintos estamentos de la FACh. Los datos obtenidos mediante estos sensores se utilizan en distintas aplicaciones civiles y de defensa, donde la percepción remota resulta fundamental para la planificación y toma de decisiones. Por ejemplo, existen organizaciones como el Servicio Aerofotogramétrico (SAF) y el Grupo de Operaciones Espaciales (GOE), ambas administradas por la FACh, encargadas de administrar las imágenes capturadas por el Sistema Satelital para la Observación de la Tierra (SSOT). El SSOT, o más conocido como FASat-Charlie, es un satélite chileno lanzado el año 2011, cuyo objetivo es captar imágenes terrestres para diferentes fines, tanto civiles (explotación de recursos naturales, evaluación de daños por catástrofe, ordenamiento demográfico, etc.), como también para fines militares. La resolución máxima del sistema actual corresponde a 1.45 m. para imágenes en el espectro visible y 5.8 m. para imágenes multiespectrales, obteniendo estas desde su ubicación a más de 600 km. de altura en la atmósfera. En ese caso, la explotación de la data adquirida suele ser post-procesada por sistemas de software para mejorar diferentes aspectos de las escenas y extraer la mayor cantidad de información posible bajo el contexto de aplicaciones específicas.

Otro tipo de sensores que se utilizan son los incorporados como una Carga Útil Operacional (POD) en vehículos aéreos. En este caso, los dispositivos de captura de imágenes operan en escenarios hostiles y altamente dinámicos, lo que naturalmente degrada en gran medida las características de las imágenes adquiridas. Debido a estas restricciones y a la necesidad de que la información adquirida sea entregada en forma oportuna para que el piloto pueda tomar decisiones durante el vuelo, este tipo de sensores favorecen parámetros como la velocidad de adquisición y el campo de visión, en desmedro de la resolución espacial, por lo que las escenas capturadas suelen carecer de detalles. Esto se hace aún más evidente en imágenes capturadas en el espectro infrarrojo (IR), donde la sensibilidad y características propias de estos sensores dificultan aún más capturar los detalles de la escena. Estas restricciones se traducen en la práctica a que, por ejemplo, durante un vuelo el piloto pueda detectar la presencia de un objeto, pero no se logre el reconocimiento o identificación de este. La falta de detalles en la información visual suele compensarse integrando a las imágenes información de otros sensores (por ejemplo, radares), lo cual puede retrasar en algunos casos la realización de una acción.

Cualquiera sea el contexto de uso, mejorar la resolución espacial de las imágenes capturadas permitirá contar con más información para decidir un determinado curso de acción. Como se planteó en el punto anterior, son los detalles obtenidos de la escena los que finalmente podrán completar el ciclo de reconocimiento, y pasar desde la detección de un objeto, aeronave,

vehículo o persona, a la identificación de este, lo que resulta fundamental en actividades de vigilancia del espacio aéreo. Si bien esto se podría realizar directamente utilizando sensores de mayor calidad y resolución que los actualmente instalados, en el caso de las aplicaciones objetivo esto suele ser inviable por temas de accesibilidad y costo. En el caso de los sensores incorporados en sistemas satelitales, no es posible reemplazar sensores que ya se encuentran en órbita. En el caso de PODs incorporados en aeronaves, existe un alto costo asociado tanto al costo directo de los dispositivos que deben cumplir con estrictas especificaciones, como también a los procesos de integración, validación y certificación necesarios para la aprobación de su uso. Por lo tanto, el contar con una herramienta local que permita mejorar la calidad de las imágenes obtenidas con los sensores actualmente disponibles resulta bastante atractivo, ya que permitiría obtener más información sin incurrir en recursos y tiempo que exigiría una renovación de sensores o dispositivos.

En este contexto, desde la FACh se planteó el problema de explorar el uso de técnicas modernas de procesamiento de imágenes basadas en redes neuronales artificiales. El objetivo principal de este trabajo es dar un primer paso en la caracterización de las capacidades y limitaciones de estas técnicas con un enfoque práctico, considerando aspectos funcionales y requerimientos técnicos, entregando documentación y datos que permitan identificar potenciales caminos para seguir explorando estas técnicas y las tecnologías asociadas. Los resultados de este trabajo proveerán una base técnica que permitirá, en un siguiente paso planear, implementar, y evaluar su uso en aplicaciones concretas.

1.2. Planteamiento del problema

El problema a tratar en esta memoria de título apunta a cubrir la necesidad de contar con un estudio que contemple varios aspectos sobre técnicas de súper-resolución en video, como la implementación, evaluación de desempeño y caracterización de costo computacional. En particular, luego de una revisión del estado del arte y pruebas preliminares realizadas como parte de la asignatura de Proyecto de Titulación (ELO307), se revisaron principalmente las técnicas *Frame-Recurrent Video Super-Resolution* [43], *Temporally Coherent GAN for Video Super-Resolution* [10] y *Video Restoration With Enhanced Deformable Convolutional Networks* [53], todas basadas en redes neuronales convolucionales y orientadas a la súper-resolución en secuencias de video, aunque finalmente se determinó trabajar en específico con *Frame-Recurrent Video Super-Resolution* y *Temporally Coherent GAN for Video Super-Resolution*.

Varias soluciones en el estado del arte de la súper-resolución proponen utilizar varias imágenes en baja resolución para calcular la estimación actual [28, 20]. En general, se propone realizar la estimación utilizando como mínimo tres imágenes obtenidas de la secuencia de video, incluyendo la imagen actual, además de imágenes previas e imágenes siguientes de la secuencia. La red FRVSR propone un esquema en el cual se utiliza solamente un par de imágenes en baja resolución, siendo estas la imagen actual y la imagen previa de la secuencia, además de la estimación previamente calculada de alta resolución, por lo que la red neuronal debe procesar menos información por las entradas. La segunda alternativa, TecoGAN, se basa en FRVSR y la arquitectura *Generative Adversarial Network* (GAN), la cual tiene como objetivo el generar imágenes más naturales en comparación a la red FRVSR a costa de requerir más recursos a nivel de memoria y procesamiento durante el entrenamiento. Por lo tanto, se deben analizar los resultados de ambas redes y determinar la factibilidad de usar FRVSR en vez de TecoGAN al ceder calidad imagen pero ahorrando tiempo de entrenamiento. Otro aspecto a considerar es que, a pesar de que ambas soluciones demuestran superar a otras soluciones en el estado del arte de la súper-resolución, las evaluaciones existentes se realizan en términos de resultados finales utilizando una estructura fija luego de un proceso de sintonización de la red, y no hay mayor información sobre los efectos de cambiar la configuración y sintonización local en términos de arquitectura, función de pérdida, parámetros de configuración, entre otros. Esto último resulta relevante para la posible implementación de redes para aplicaciones altamente especializadas (como las propuestas por la FACh), como también para analizar tradeoffs en términos de calidad de la imagen de salida y el uso de recursos computacionales disponibles.

El trabajo en torno a la implementación y evaluación de las redes propuestas considera las siguientes etapas:

- **Replicar y verificar los experimentos reportados en la literatura:** se implementa la arquitectura de las redes FRVSR y TecoGAN, entrenando y evaluando su desempeño bajo condiciones y con datos similares a los reportados en la literatura y repositorios de referencia. Este paso apunta a validar con un enfoque cualitativo las capacidades y limitaciones de las redes seleccionadas.
- **Incorporación de posibles mejoras y pruebas con bases de datos adicionales:** se entrena con un conjunto de datos enfocado en los objetos de interés para las potenciales aplicaciones objetivo. En particular, se apunta a mejorar la presentación de detalles

en videos que contienen vehículos y texto como primera aproximación al problema presentado en detalle anteriormente.

- **Pruebas experimentales:** se varía la resolución de las imágenes de las redes para comparar la diferencia entre la cantidad de información dada para el cálculo de la súper-resolución. Para medir esta diferencia, se calculan las métricas funcionales y de costo computacional, ya sean cuantitativas o cualitativas. Además, se caracterizan los tiempos computacionales asociados a los procesos de entrenamiento e inferencia de las redes seleccionadas utilizando distintas plataformas.
- **Preparación de un demostrador tecnológico para procesamiento en línea:** aplicación funcional que permite, ya sea mediante una interfaz gráfica o archivo de configuración, seleccionar la red neuronal deseada y el video objetivo a procesar. Alternativamente, a medida que se avanza en el procesamiento se calculan las métricas solicitadas por el usuario. Esta aplicación apunta a servir como demostrador para ilustrar las capacidades de las técnicas desarrolladas.

1.3. Alcances y contribuciones

Este informe muestra los resultados obtenidos de los experimentos realizados en torno a la sintonización y evaluación de desempeño de las redes FRVSR y TecoGAN. Para este fin, se prueban diversos parámetros de configuración para realizar el proceso de entrenamiento y comprobar la variación de los resultados mediante una evaluación posterior. En el primer caso de interés se configuran diferentes valores de resolución para las imágenes de entrada a la red con el propósito de observar la influencia de la cantidad de información dada en los detalles generados para la imagen de salida. Este aspecto permite determinar la relación entre la velocidad de procesamiento y calidad de los detalles generados, permitiendo dar una recomendación de configuración según la precisión requerida en cuanto a detalles y la velocidad con que la aplicación requiera la captura de imágenes.

Otro aporte de este trabajo es mostrar la influencia del uso de diferentes conjuntos de datos en el entrenamiento, siendo los utilizados enfocados en flujo de movimiento, vehículos varios y letras, utilizando cada uno por separado o una combinación de estos. Se sabe que especializar el conjunto de datos en el entrenamiento mejora los resultados para los objetos de interés; sin embargo, es necesario realizar estimaciones cuantitativas para tener presente en cuánto mejoran y con esta información discutir la necesidad de incluir más datos a futuro.

Para medir las diferencias de resultados entre las distintas redes a entrenar se propone el uso de métricas clásicas y métricas propias. Las métricas clásicas consisten en métricas de uso general en la literatura para contrastar mejoras con respecto a otros métodos. En este trabajo son utilizadas *Peak Signal-to-Noise Ratio (PSNR)* y *Structural Similarity Index (SSIM)*. Estas dan una primera aproximación a que la calidad de una imagen es mejor en comparación a otra; sin embargo, no sirven en el caso de que ambas imágenes sean similares y no se puede concluir que los detalles de una son mejores que la otra. Por lo tanto, se propone el uso de las métricas propias orientadas a evaluar las ganancias a nivel de aplicación objetivo en lugar de a nivel de píxeles. Las métricas propias se basan en la comparación de resultados al utilizar sistemas para la detección de objetos, clasificación, o alguna otra tarea aplicada a imágenes. Para llevar a cabo esta idea se utiliza la red neuronal *You Only Look Once V3* [41], un sistema de detección de objetos en tiempo real que se encuentra en el estado del arte, comparando el nivel de certeza al detectar un objeto correctamente.

En resumen, la principal contribución de este trabajo es la generación y reporte de evidencia empírica sobre las capacidades y limitaciones del uso de redes neuronales convolucionales en tareas de súper-resolución en imágenes y video, para que la FACh pueda discutir y evaluar futuros desarrollos en esta línea para una potencial incorporación en aplicaciones específicas. Como soporte al reporte escrito, se provee un conjunto de librerías, códigos, y ejemplos que permiten reproducir los resultados acá reportados. Además, se desarrolló un demostrador tecnológico para el procesamiento de video en línea, el cual puede ser usado para tareas demostrativas, educativas, y de difusión. Junto con los códigos, se provee la documentación y ejemplo que permitan su fácil reutilización de las redes desarrolladas utilizando nuevos conjuntos de datos, o bien su modificación para agregar capacidades u optimizaciones orientadas a alguna aplicación específica.

1.4. Estructura del informe de memoria

Este documento se encuentra constituido de la siguiente manera:

- **Capítulo 2 - Marco teórico y estado del arte:** Presentación de los conceptos de súper-resolución, métodos de procesamiento clásico, redes neuronales y arquitecturas de interés, frameworks y hardware para su desarrollo. Adicionalmente se presentan trabajos previos que se han realizado en torno a redes neuronales aplicadas a la súper-resolución.
- **Capítulo 3 - Redes neuronales y sistemas utilizados:** Descripción de las redes neu-

ronales utilizadas en el procesamiento de video y conjuntos de datos utilizados, junto con las métricas y sistemas para evaluar resultados y desempeño.

- **Capítulo 4 - Resultados:** Presentación de los resultados experimentales, analizando los diferentes casos y métricas obtenidas.
- **Capítulo 5 - Conclusiones:** Planteamiento de las conclusiones generales acerca de las distintas configuraciones utilizadas para las redes neuronales y propuestas de trabajo que se pueden desarrollar para la continuidad del proyecto.

Capítulo 2

Marco teórico y estado del arte

En este capítulo se presentan y describen algunos conceptos importantes para entender el trabajo desarrollado, incluyendo los fundamentos de la súper-resolución y su desarrollo a través de los años, redes neuronales convolucionales y arquitecturas de interés en torno a estas, y los sistemas y plataformas utilizadas para desarrollar las redes. Adicionalmente se presentan brevemente trabajos previos desarrollados en torno a la súper-resolución con redes neuronales.

2.1. Súper-resolución

La súper-resolución corresponde al conjunto de técnicas y algoritmos que buscan obtener una o más imágenes en alta resolución a partir de una o varias imágenes de baja resolución, al aumentar la cantidad de píxeles por una cierta escala y cuyos valores son cercanos a los correspondientes de alta resolución. La súper-resolución ha sido un tema de gran interés durante las últimas décadas debido a sus aplicaciones en muchos problemas del mundo real y en diversos campos. Estas aplicaciones van desde la mejora de imágenes satelitales o aéreas hasta el procesamiento de imágenes en la medicina, imágenes obtenidas por ultrasonido [30], análisis de rostros en imágenes [63], análisis de texto lectura de patentes o identificadores [44], sistemas de imágenes infrarrojas [27], mejoramiento de rostros [35], reconocimiento de iris [32], mejoramiento de detalles en huellas dactilares [60], holografía digital [62], y muchas más. La diversidad de aplicaciones ha dado como resultado el desarrollo de múltiples métodos, cada uno proponiendo su propio algoritmo adaptado para el propósito específico de cada aplicación. Entre los métodos desarrollados para la súper-resolución se puede diferenciar entre aquellos cuyo enfoque es un tipo específico de imágenes, tales como rostros [49], paisajes [48] o con propósitos artísticos [23], con el fin de lograr detalles finos apropiados al tipo de objeto. En otros casos se busca generalizar los resultados [14, 17, 58] con el fin de poder procesar cual-

quier tipo de imagen independiente de su contenido, dando énfasis al detallado de bordes y segmentos de línea para lograr este propósito.

Los algoritmos de súper-resolución apuntan a generar detalles más precisos que los dados por la cantidad de píxeles obtenidos desde el sensor, al aumentar estos por unidad de área en una imagen [38]. Para aumentar la cantidad de píxeles existen alternativas basadas en mejorar el hardware asociado al sensor para la captura de imágenes, siendo estas la disminución del tamaño del píxel o incrementar el tamaño de captura del sensor [29]. La reducción del tamaño de píxel en el sensor es una opción viable, pero con esto también se reduce la cantidad de luz capturada, dando como consecuencia un aumento en el ruido de captura, junto con el hecho de que continuar mejorando la tecnología actual supone un esfuerzo mayor, tanto en costo como en las especificaciones del hardware requerido [34]. Además, píxeles de menor tamaño relativo a la apertura del lente, son más sensibles a los efectos de la difracción en comparación a píxeles de mayor tamaño. Por otro lado, el incremento del área del sensor de captura de imágenes tiene como costo reducir su tasa de captura en término de imágenes por segundo, junto con un aumento considerable del costo para adquirir dispositivos de captura de alta resolución. Por lo tanto, las soluciones basadas en algoritmos de post-procesamiento sobre una imagen ya capturada son usualmente preferidos sobre las soluciones basadas en la manipulación directa de los parámetros físicos de los sensores.

A pesar de ser técnicas similares, la súper-resolución no debe confundirse con métodos como la interpolación o restauración de imágenes. La súper-resolución no solo contempla la reconstrucción al aumentar la resolución, sino que también tiene como propósito filtrar distorsiones de la imagen, tales como ruido de la imagen, y corregir espacios borrosos. En el caso de la interpolación, las técnicas no fueron diseñadas para recuperar detalles de alta frecuencia, refiriéndose a estos como espacios en donde los valores de los píxeles adyacentes presentan gran variación en sus valores, lo cual si es un objetivo de la súper-resolución [16]. Las técnicas de restauración de imágenes permiten eliminar zonas borrosas y realzar detalles, pero los tamaños de la imagen de entrada y salida son iguales. En la súper-resolución, además de haber mejoras en la calidad de imagen, la salida presenta un tamaño mayor en comparación a la imagen original, y por lo tanto se compone de una cantidad mayor de píxeles [46].

Las técnicas de súper-resolución reportadas en la literatura reciente pueden clasificarse en dos categorías: técnicas de procesamiento clásico y las basadas en redes neuronales artificiales. Los métodos clásicos se basan en el procesamiento mediante modelos matemáticos explícitos que logran el aumento de resolución y cálculo de la información mediante fór-

mulas directas, análisis de las características extraídas, ecuaciones diferenciales, etc. En la actualidad, la mayoría de las soluciones propuestas se encuentran desarrolladas en torno a redes neuronales convolucionales, las cuales no requieren de un modelamiento explícito, sino que infieren un modelo que procesa las imágenes por medio de una etapa de entrenamiento, la cual se basa en pares de ejemplos de entrada/salida deseada. En general, la literatura reciente muestra que las redes convolucionales suelen entregar mejores resultados en comparación a los métodos clásicos, apreciando su mejor desempeño en diferentes áreas de visión por computador, siendo ejemplos de esto el reconocimiento de rostros [36], seguimiento de objetos [39] y reconocimiento de objetos [40].

2.2. Técnicas clásicas de súper-resolución

Las técnicas clásicas de súper-resolución pueden ser subclasificadas según su enfoque en las siguientes categorías: métodos de predicción, métodos enfocados en bordes, métodos estadísticos y métodos basados en parches. A continuación, se explican brevemente cada uno de estos métodos.

- **Métodos de predicción:** los métodos de predicción se basan en el cálculo mediante alguna fórmula matemática explícitamente definida a tiempo de diseño para generar las imágenes. Un tipo de método de predicción es la interpolación [6], la cual pondera el valor de los píxeles adyacentes en la imagen de baja resolución para calcular el valor del píxel en la imagen de alta resolución, siendo algunos ejemplos de esto la interpolación lineal y la interpolación bicúbica. Estos tienen como propósito generar imágenes con cierta consistencia en los valores de sus píxeles y compatibilizar con la resolución deseada, lo cual lleva a que estos algoritmos carezcan de la capacidad de generar detalles en los bordes y fallan en regiones de alta frecuencia. Otro ejemplo de modelo de predicción se describe en [18], en donde el sistema, iterando para encontrar los valores óptimos de su función propuesta, genera imágenes de baja resolución a partir de las de alta resolución. Luego, utilizando el mismo sistema a la inversa, se generan las estimaciones en alta resolución, lo cual demuestra obtener mejores resultados que la interpolación bicúbica, pero aún presenta limitaciones en la capacidad para generar detalles.

- **Métodos enfocados en bordes:** los bordes en los elementos contenidos en una imagen constituyen un elemento fundamental en la percepción visual. Por este motivo, se desa-

rrollaron métodos capaces de utilizar las características de los bordes para realizar la estimación en alta resolución, tales como el largo y ancho de los bordes [12], o utilizar el gradiente [47]. Este método muestra mejores resultados que los de predicción, siendo las mejoras representadas por la generación de detalles finos, junto con la delimitación correspondiente de objetos.

- **Métodos estadísticos:** los métodos mediante análisis estadístico utilizan varias características que se pueden encontrar en una imagen para realizar la predicción en alta resolución. Algunas de estas características son el análisis de la distribución del gradiente [45], o la variación total de la información para generalizar la reconstrucción [22]. Sin embargo, estos métodos se encuentran limitados por el ruido siempre existente en los sistemas digitales de imágenes.

- **Métodos basados en parches:** los métodos basados en parches consisten en, a partir del correspondiente par de imágenes de alta y baja resolución, seccionar estas para luego de determinar una función que pueda mapear localmente los valores de baja resolución a los correspondientes de alta resolución. Para determinar la función de mapeo se han propuesto variadas soluciones, entre las cuales se pueden mencionar el promedio ponderado [5], regresión de kernel [22], regresión de vectores de soporte [33], entre otros. Inicialmente, para parches sobrepuestos el valor de los píxeles es promediado, además de proponer métodos para tratar el caso de la superposición, tales como la ponderación mediante pesos [57], campo aleatorio de Markov [13], o campos aleatorios condicionales [52].

En general, estos métodos dependen de las características que se puedan extraer de las imágenes mediante un procesamiento previo. Estas características son propuestas por los diferentes autores, según sea el caso y a partir de un análisis riguroso. En algunos casos, puede ser que no se estén utilizando todas las características importantes, o incluso, que su aporte no sea de gran ayuda para estimar la imagen de alta resolución. Las redes neuronales incorporan esta búsqueda, por así decirse, durante el entrenamiento de sus parámetros internos, permitiendo enfocar su desarrollo en otros aspectos para mejorar los resultados. También se menciona que a pesar de que las redes neuronales necesitan de una gran cantidad de datos en comparación a estos métodos clásicos, finalmente son las redes neuronales las que logran destacar debido a la calidad de sus resultados[42].

2.3. Redes neuronales convolucionales

Las redes neuronales convolucionales corresponden a un tipo de red neuronal artificial, la cual se compone de la interconexión de miles de nodos o unidades de procesamiento, con capacidad de auto aprendizaje que le permite mejorar los resultados de si misma, mientras disponga de más datos durante la fase de entrenamiento.

Su estructura consiste de una capa de entrada y de salida, conectadas entre sí por múltiples capas intermedias, como se muestra en la Figura 2.1. La capa de entrada se encarga de recibir la información en el formato correspondiente y ponderarla para las capas posteriores, mientras que la capa de salida entrega el resultado de la red según lo requerido, sea este una imagen, categoría, o segmentación de lo observado. Las capas intermedias se componen en su mayoría de filtros convolucionales, en donde cada uno se encuentra conectado solo a una región de la salida de la capa anterior y aplica la operación de convolución correspondiente sobre esta.

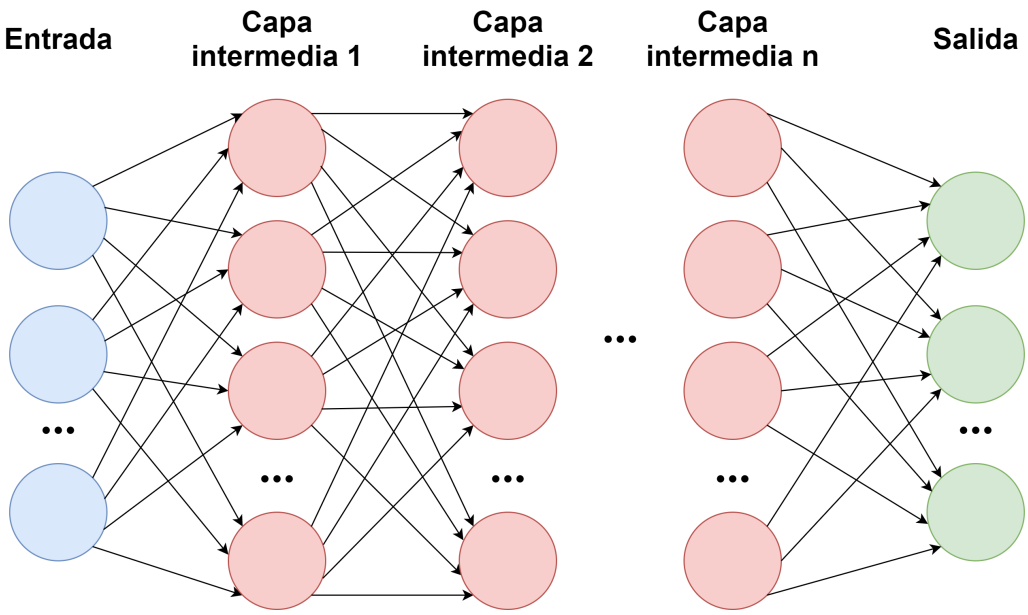


Figura 2.1: Arquitectura de una red neuronal convolucional

El uso de las redes neuronales convolucionales se encuentra bastante desarrollado en el área de clasificación y segmentación de imágenes [24]. Esto se debe a su capacidad de codificar a través de su arquitectura las características intrínsecas de la entrada, lo que hace a las redes neuronales convolucionales más adecuadas para este tipo de aplicaciones.

La característica que distingue a una red neuronal convolucional, como se mencionó anteriormente, es que se encuentra compuesta principalmente por múltiples capas de filtros convolucionales. Los parámetros de los filtros son optimizados durante la etapa de entrenamiento, en la cual la red neuronal aprende a reconocer patrones a partir de los datos provistos. Esto se logra al comparar la imagen de salida de la red neuronal con la imagen objetivo, calculando la diferencia entre ambas mediante la función de pérdida. A partir de los valores indicados por la función de pérdida, los cuales son ponderados por el parámetro conocido como tasa de aprendizaje y cuyo fin es controlar la velocidad a la que el modelo aprende a reconocer patrones, es que se actualizan los parámetros internos de la red neuronal mediante *backward propagation* [7], cálculo secuencial de los pesos desde la salida hasta la entrada a partir del error obtenido. El flujo de este proceso de entrenamiento se ilustra simplificada en la Figura 2.2, siendo esta un ejemplo simple de súper-resolución, en la cual la imagen generada, correspondiente a la salida de la red neuronal utilizando los datos de entrenamiento, es comparada con la imagen original en la función de pérdida, para luego actualizar los valores de los parámetros en cada capa, y así continuamente hasta completar un cierto número de iteraciones. Al culminar el entrenamiento de la red neuronal luego de un número de iteraciones, se espera que los parámetros se encuentran ajustados a lo que se puede referir como sus valores óptimos, o por lo menos cercanos a estos. Es importante mencionar que el correcto entrenamiento de la red depende de la correcta selección de múltiples parámetros e hiperparámetros, para lo cual no hay una forma sistemática de selección y dependen en gran parte de técnicas heurísticas y la experiencia del diseñador, además de un tuneo fino mediante prueba y error. Finalmente, una vez entrenada la red neuronal, se lleva a cabo el proceso llamado inferencia, en donde la red neuronal se mantiene fija y se le dan las entradas para solamente calcular la salida correspondiente, ya sea para evaluar su desempeño ante diferentes datos o simplemente utilizar la salida para la aplicación correspondiente.

Es posible aumentar la cantidad de capas internas en la red o incluso aumentar la cantidad de neuronas con el fin de procesar imágenes de mayor tamaño, pero esto trae consigo dos problemas que se explican a continuación. El primero corresponde a que el tiempo y recursos requeridos para entrenar una red neuronal aumenta considerablemente con la cantidad de parámetros internos de esta. El segundo problema, el cual ya afecta directamente los resultados, es el llamado *overfitting* [51], fenómeno que ocurre cuando los parámetros de la red neuronal se ajustan demasiado a los datos de entrenamiento, reduciendo el error durante esta etapa a valores muy bajos, pero ante la presencia de nuevos datos el error es considerable. De esto se tiene que para optimizar la red neuronal es necesario mantener la cantidad de parámetros requeridos óptimos, en cuanto a que sean lo más bajo sin afectar negativamente a los resultados.

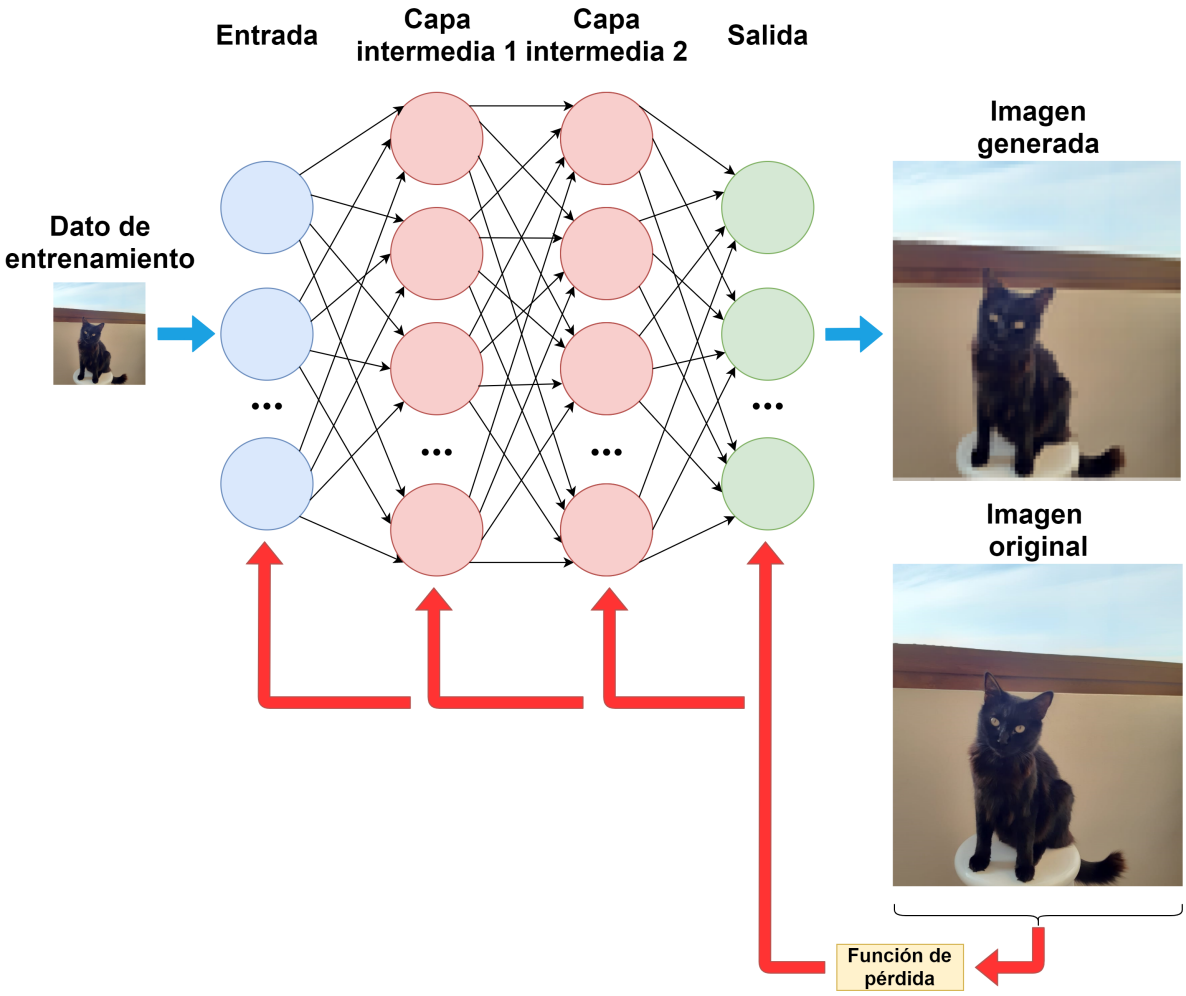


Figura 2.2: Entrenamiento de una red neuronal convolucional

Para asegurarse de que esto ocurra, es necesario calcular y comparar las métricas pertinentes, tanto para los datos utilizados en el entrenamiento, como para los datos diferentes a los del entrenamiento, refiriéndose a estos como datos de evaluación, manteniendo cierta proporción entre estos dos conjuntos de datos.

En cuanto al uso de redes neuronales en la súper-resolución, su desempeño se puede apreciar en los múltiples desafíos en torno a la de procesamiento de imágenes, siendo *NTIRE* uno de los más destacados. *NTIRE* consiste en un desafío anual que se lleva a cabo desde 2016 para diferentes categorías de procesamiento de imágenes. En su versión de 2019 se presentan métodos del estado del arte tanto para la súper-resolución aplicada a imágenes [4] como videos [31]. Cabe destacar que cada solución presentada en todas las versiones de *NTIRE* se encuentran desarrolladas en torno a redes neuronales convolucionales, debido a su elevado desempeño en comparación a métodos de visión por computador o de aprendizaje de máqui-

nas.

2.3.1. Generative Adversarial Network (GAN)

Generative Adversarial Network (GAN) [15] es una arquitectura propuesta por Ian Goodfellow, la cual utiliza dos redes neuronales que compiten entre sí, con el objetivo de generar datos que se consideren reales, o naturales. A lo que se refiere una imagen natural, es a la característica propia de una GAN de imitar la distribución de cualquier tipo de dato con el cual fue entrenada, permitiéndole así generar nuevos datos manteniendo esta distribución. Su uso se encuentra especialmente en la generación de audio, imágenes y video.

En cuanto a su arquitectura, las dos redes neuronales que la componen corresponden a la red generadora, o generador, y la red discriminadora, o discriminador. El generador se encarga de generar los datos, como por ejemplo imágenes, mientras que el discriminador debe diferenciar lo que serían los datos originales de los generados. La idea tras de esto es que durante la etapa de entrenamiento el generador mejora sus resultados utilizando la información del discriminador para poder "engañar" a este último, a la vez que el discriminador aprende a diferenciar entre ambos tipos de datos. Usualmente la salida del discriminador es la correspondencia a una cierta categoría o clase. Este proceso de entrenamiento puede observarse en la Figura 2.3, en donde las imágenes producidas por el generador a partir de los datos de entrenamiento son clasificadas por el discriminador para determinar que tan reales o falsas son, en comparación a las imágenes reales. Al terminar de clasificar ambos conjuntos de imágenes, reales y generadas, la función de pérdida actualiza los parámetros de tanto el generador como del discriminador. Todo este proceso se repite hasta completar el número predeterminado de iteraciones. Al culminar esta etapa, el realizar las inferencias solo requiere de la red generadora, ya que finalmente es esta la que produce los datos y la red discriminadora solo cumple su función de ayudar a mejorar los resultados del generador durante el entrenamiento.

Durante el entrenamiento se deben tener algunas consideraciones para el correcto procesamiento. En cada iteración es necesario mantener los parámetros del generador constantes al entrenar el discriminador, y viceversa. Esto permite a cada red neuronal mantener cierta consistencia del gradiente a partir del cual se ajustan sus parámetros. Además, se debe tener presente que si el discriminador converge más rápido a la solución que el generador, este último puede presentar problemas al utilizar el gradiente. En cambio, si el generador converge más rápido, se pueden llegar a tener problemas en el discriminador, llevándolo a dar falsos negativos en sus resultados. En ambos casos mencionados solo basta con sintonizar

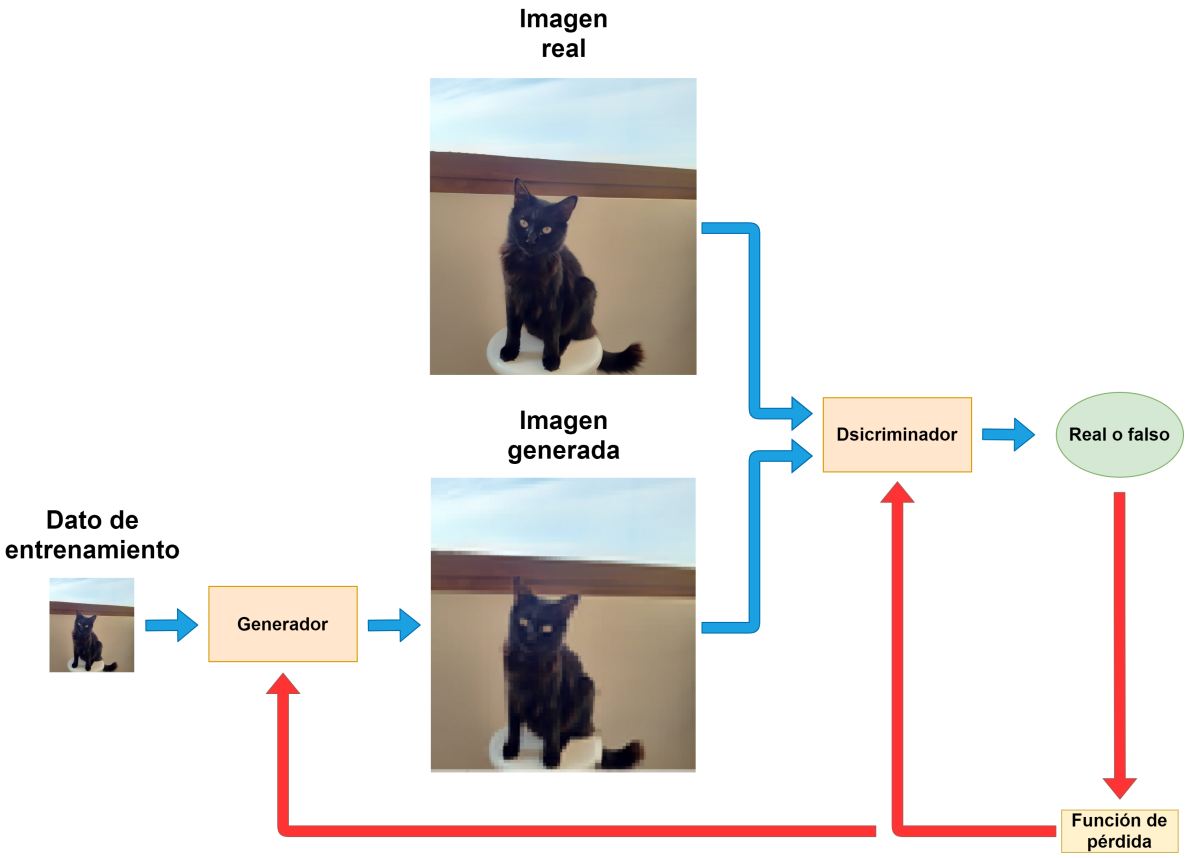


Figura 2.3: Entrenamiento de una red neuronal GAN

apropiadamente la tasa de aprendizaje, comúnmente siendo esta baja para el generador y el discriminador, y la proporción entre la tasa de aprendizaje de cada red según la arquitectura de estas.

El uso y desempeño de la GAN puede encontrarse en múltiples aplicaciones. En un comienzo, su uso fue propuesto para generar imágenes de números que visualmente fueran semejantes a si fueran escritos a mano [15], en donde se toman como datos de referencia el conjunto de datos MNIST [11]. En [2] se demuestra la capacidad de las GAN de generar fotografías sintéticas, obteniendo resultados visualmente indistinguibles (en términos cualitativos considerando la percepción el ojo humano) de una fotografía real. Otro ejemplo se presenta en [21], en donde se logran generar fotografías de rostros, destacando debido a los buenos resultados obtenidos. En este último ejemplo, los datos de entrenamiento utilizados corresponden a celebridades, lo cual resulta en que los rostros generados presenten una cierta semejanza a estas, pero aún así no siendo iguales.

2.4. Evaluaciones preliminares

Previo al trabajo de memoria, y en el contexto de otras asignaturas y actividades de investigación, se realizaron evaluaciones experimentales con redes neuronales convolucionales en el campo de la súper-resolución [61]. A continuación se describe un resumen de lo realizado en estos trabajo previos, presentando brevemente algunos resultados de interés para el trabajo de memoria. El trabajo mencionado consistió en realizar una búsqueda de diferentes modelos en el estado del arte de la súper-resolución, que además utilizaran la arquitectura GAN, para luego comparar su desempeño, diferencias entre los resultados y posibles mejoras logradas al ser usadas en sistemas de reconocimiento facial. Es necesario mencionar que para esta aplicación en particular no solo importa lo bien que se vean los detalles de un rostro, sino que el rostro mostrado posea las características intrínsecas propias de la persona a la que representa. Entonces, los resultados obtenidos permiten mostrar el desempeño de las redes en cuanto a la generación de detalles relevantes que no son visibles al ojo humano, siendo las redes neuronales utilizadas SRGAN [25], SRPGAN [56], ESRGAN [54] y ProSR [55].

En cuanto a las pruebas realizadas, se tratan diferentes escenarios, cada uno con su propio conjunto de datos, con el objetivo de mostrar la dificultad de generar rostros y poder clasificarlos correctamente al variar la cantidad de personas y las imágenes por cada una. En la Figura 2.4 se muestran los resultados para el subconjunto generado a partir del conjunto de datos *CASIA-Webface* [59], el cual consta de 500 mil imágenes para 10 mil celebridades diferentes. El subconjunto obtenido corresponde al escenario con mayor variedad de imágenes e identidades en el trabajo realizado, con un total de 9.526 y 3.283 imágenes para el conjunto de entrenamiento y evaluación respectivamente, para un total de 200 diferentes personas. En conjunto de entrenamiento es utilizado durante el proceso de entrenamiento de las redes neuronales, mientras que para calcular y analizar resultados se hace uso de los datos de entrenamiento y evaluación, esto último con el objetivo de comprobar la diferencia de resultados para datos que son familiares para la red neuronal con otros datos nuevos o diferentes.

De los resultados obtenidos ya es posible apreciar mejoras visuales en comparación a la interpolación bicúbica. Para cuantificar estas mejoras se utilizan dos redes neuronales, *VGG19* y *Simple Recognition*, ambas entrenadas con el subconjunto de entrenamiento de alta resolución. Los resultados presentados en la Tabla 2.1 corresponden a la precisión obtenida por las redes neuronales VGG19 y Simple Recognition al clasificar las imágenes generadas por cada red neuronal de súper-resolución, y la alta resolución representa la precisión máxima posible por cada una. De las diferentes redes neuronales utilizadas se concluye que los mejores re-

sultados corresponden a la red neuronal ProSR, logrando los mejores resultados visuales y de reconocimiento facial, superando al resto de los métodos y logrando valores cercanos a los de alta resolución para Simple Recognition, finalmente demostrando el gran desempeño y aporte de las redes neuronales convolucionales para este tipo de aplicaciones.

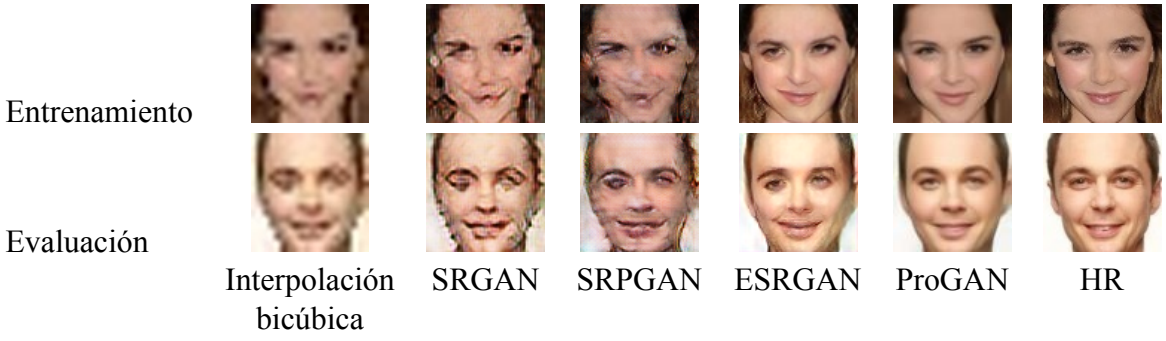


Figura 2.4: Resultado de imágenes para CASIA-Webface

	Simple Recognition		VGG 19	
	Entrenamiento	Evaluación	Entrenamiento	Evaluación
Alta resolución	99.97 %	66.52 %	98.71 %	54.04 %
Interpolación bicúbica	83.74 %	50.78 %	2.62 %	2.01 %
SRGAN	63.25 %	40.05 %	8.84 %	5.33 %
SRPGAN	63.60 %	40.76 %	4.23 %	4.57 %
ESRGAN	77.14 %	45.14 %	34.03 %	22.78 %
ProSR	97.37 %	58.76 %	46.29 %	29.21 %

Tabla 2.1: Resultados de precisión de reconocimiento facial para CASIA-Webface

2.5. Plataformas de cómputo

Para facilitar el desarrollo del aprendizaje de máquinas se han desarrollado en los últimos años múltiples frameworks, estructuras conceptuales y tecnológicas que cuenta con módulos y herramientas para facilitar la organización y desarrollo de software. Para el caso las redes neuronales convolucionales, existen una gran variedad de alternativas de framework para su implementación, siendo algunos ejemplos Pytorch [37], TensorFlow [1], Deeplearning4j [50], Keras [9], MXNet [8], Caffe [19], entre muchos otros, ofreciendo cada uno facilidades tanto para diseñar, entrenar y validar las redes neuronales. Adicionalmente, estos dependen de bibliotecas optimizadas por GPU, tales como cuDNN, NCCL y DALI, para obtener un mejor

desempeño en cuanto su capacidad para acelerar el cálculo de las operaciones.

La GPU corresponde a un chip encargado de procesar tareas de alto requerimiento computacional, sean estas de tipo gráficas o matemáticas, permitiendo a la *Central Processing Unit* (CPU) enfocarse en otras actividades. En cuanto a la diferencia entre la CPU y la GPU, la CPU se compone de un solo núcleo o un número bajo de estos, los cuales se encargan de realizar el procesamiento en sí y están limitados a la realización secuencial de una única tarea cada uno. Las GPUs, en cambio, cuentan con un gran número de núcleos, permitiendo así paralelizar de manera más eficiente el cómputo y obteniendo el resultado de los cálculos en un tiempo mucho menor [26]. Al utilizar redes neuronales convolucionales, en cuanto al código en sí, este se ejecuta principalmente en la CPU, pero es posible asignar ciertas tareas a la GPU mediante ciertas funciones, como se mencionó previamente, mediante la incorporación de bibliotecas, optimizando el cálculo de la función de pérdida, la actualización de los pesos de la red neuronal o la inferencia de resultados a partir de un conjunto de datos dado.

Capítulo 3

Redes neuronales y sistemas utilizados

En este capítulo se describen algunas características relevantes del entorno de trabajo, incluyendo las redes neuronales convolucionales utilizadas con sus respectivas configuraciones, los conjuntos de datos utilizados, herramientas y algoritmos, junto con las especificaciones de los sistemas utilizados en todo el proceso. Todo el código utilizando en la implementación de este trabajo se encuentra en `https://github.com/javiergonzalezya13/Super-Resolution`

3.1. Frame-Recurrent Video Super-Resolution

A continuación se explican los detalles técnicos de Frame Recurrent Video Super-Resolution (FRVSR) [43], en cuanto a la arquitectura de la red y su implementación.

3.1.1. Arquitectura

La arquitectura general de FRVSR puede observarse en la Figura 3.1, la cual se encuentra conformada por múltiples bloques, cuya funcionalidad se explica a continuación. Las entradas utilizadas por la red neuronal corresponden a tres imágenes, siendo estas las imágenes en baja resolución actual y previa obtenidas de la correspondiente secuencia de video, representadas por I_t^{LR} e I_{t-1}^{LR}, y la imagen previamente estimada I_{t-1}^{est}, calculando a partir de la información aportada por estas la estimación actual I_t^{est}. Los módulos en rojo corresponden a aquellos cuyos parámetros internos son entrenables, lo cual significa que estos cambian sus valores durante el entrenamiento para mejorar los resultados obtenidos por la función de pérdida. En cambio, aquellos en amarillo poseen parámetros internos no entrenables, lo cual significa que

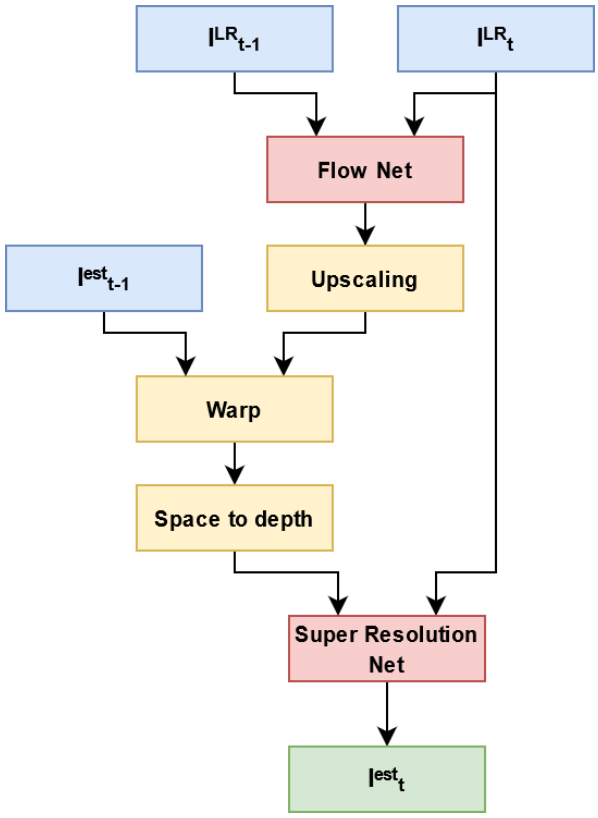


Figura 3.1: Resumen de la arquitectura de Frame-Recurrent Video Super-Resolution

sus valores se mantienen constantes durante todo el proceso de entrenamiento y no afectan el resultado final de la red neuronal. Estos parámetros se encuentran determinados al momento de implementar el mismo bloque, siendo un ejemplo de esto alguna operación algebraica o la ponderación de valores.

La función del bloque *Flow Net*, cuya configuración a nivel de capas se muestra en detalle en la Figura 3.2, es generar un mapa del flujo de movimiento F^{LR} como primer paso para integrar la información temporal en la red neuronal, y por ende, en la estimación de alta resolución. Esto se logra al comparar el par de imágenes de baja resolución, I_t^{LR} e I_{t-1}^{LR}, mejorando así las estimaciones de flujo de movimiento. A continuación, *Upscaling* se encarga de transformar el mapa de flujo obtenido, llevándolo a uno de alta resolución mediante interpolación bilinear, y representado por F^{HR}. Este escalamiento se debe a que el mapa de flujo de movimiento es utilizado en conjunto con la estimación anterior de FRVSR para realizar una primera aproximación a la estimación actual.

El bloque *Warp* hace uso de la información aportada por F^{HR} e I_{t-1}^{est} para obtener, como se mencionó anteriormente, la primera aproximación de la estimación actual, $\tilde{I}_{t-1}^{est}$. A conti-

nuación, *Space to depth* ordena la información de $\tilde{I}_{t-1}^{est}$ en varias imágenes para compatibilizar con las dimensiones de entrada a *Super Resolution Net*. Finalmente, a partir de $\tilde{I}_{t-1}^{est}$ e I_t^{LR}, *Super Resolution Net*, cuya configuración se muestra en la Figura 3.3, calcula la estimación actual I_t^{est}.

3.1.2. Función de pérdida

La función de pérdida se encuentra definida por la ecuación 3.1, compuesta por los términos L_{sr} y L_{flow}. L_{sr} corresponde a la distancia euclidiana entre la imagen estimada I_t^{est} y la imagen objetivo I_t^{HR}, influyendo en la optimización de los parámetros tanto de *Flow Net* como en los de *Super Resolution Net*. En cuanto a L_{flow}, aporta en la consistencia del flujo de movimiento, afectando solo los parámetros de *Flow Net* durante el entrenamiento. La operación realizada por L_{flow} es una comparación entre I_t^{LR} con la aproximación de baja resolución $\hat{I}_{t-1}^{LR}$, en donde esta última se obtiene de la función F, que corresponde al cálculo del mapa de flujo de movimiento mediante Flow Net, y posteriormente obtener la primera aproximación a la imagen de alta resolución mediante la función W, que representa la utilización del bloque Warp.

$$L = L_{sr} + L_{flow} \tag{3.1}$$

$$L_{sr} = ||I_t^{est} - I_t^{HR}||_2^2 \tag{3.2}$$

$$L_{flow} = ||I_t^{LR} - W(I_{t-1}^{LR}, F(I_{t-1}^{LR}, I_t^{LR})||_2 \tag{3.3}$$

3.2. Temporally Coherent GAN for Video Super-Resolution

A continuación se explican los detalles técnicos de Temporally Coherent GAN for Video Super-Resolution (TecoGAN) [10], en cuanto a la arquitectura de la red, implementación, entre otros.

3.2.1. Arquitectura

TecoGAN utiliza como base la arquitectura de FRVSR para implementar el generador de una arquitectura tipo GAN, añadiéndole a su correspondiente discriminador una componente temporal. Además, propone incluir en el generador un bloque paralelo al resto de la red, el cual aumenta directamente la resolución de la imagen de entrada mediante interpolación bicúbica, por lo que el resto de la red se enfocaría en generar los detalles. Entonces, a la salida del generador se suman ambos resultados y así se obtiene la imagen de salida en alta resolución.

En cuanto a la red discriminadora, la arquitectura utilizada es relativamente simple, la cual corresponde a la mostrada en la Figura 3.4. El discriminador utiliza como entrada tres conjuntos de tres imágenes, siendo estas tres imágenes cuadros consecutivos de video, u obtenidas a partir del procesamiento de estas. En cuanto a los conjuntos en sí, el primero de estos es el resultado de aplicar interpolación bilinear a las imágenes de baja resolución, aplicando este método de aumento de resolución para compatibilizar con las dimensiones de los demás conjuntos. El segundo conjunto se encuentra conformado por las estimaciones obtenidas desde la red generadora o las imágenes originales de alta resolución, según sea la etapa de entrenamiento. Además, a partir de este se obtienen aproximaciones mediante el bloque Warp, para conformar así el tercer y último conjunto.

3.2.2. Función de pérdida

La función de pérdida del generador $L_{G,F}$, mostrada en la ecuación 3.4, se calcula mediante la ponderación de varias métricas, con la finalidad de poder comparar de manera más completa la imagen generada I_t^{est} y la objetivo I_t^{HR}. La componente con mayor influencia corresponde al error cuadrático medio entre ambas imágenes. A continuación se representa el aporte del discriminador para lograr que las imágenes generadas sean lo más naturales posibles. Los términos ϕ_{VGG} y ϕ_D integran la información de características intrínsecas calculadas por las redes VGG19 y el discriminador respectivamente.

En cuanto al flujo de movimiento en sí, este es representado por L_{warp} en la ecuación 3.5, al igual que ocurre con FRVSR. Otro aporte de TecoGAN para lograr consistencia en el flujo de movimiento, es planteando la siguiente idea: al entrenar la red con vídeo con las imágenes I_t^{LR}, I_{t-1}^{LR} e I_{t-1}^{est}, se obtiene para el instante t la aproximación I_t^{est}. En caso de que el vídeo fuera procesado en sentido contrario, se obtiene la estimación $\hat{I}_t^{est}$ a partir de las imágenes I_t^{LR}, I_{t+1}^{LR} e I_{t+1}^{est}. Como ambas estimaciones, I_t^{est} e $\hat{I}_t^{est}$, deben lograr generar la misma ima-

gen, entre sí deberían ser idénticas, y por lo tanto, si el movimiento es en un sentido o en otro, no debería afectar finalmente la imagen de salida. La diferencia entre ambas estimaciones, I_t^{est} e $\hat{I}_t^{est}$, se añade finalmente al calcular su error cuadrático medio, y queda representado por el término L_{PP}, el cual queda expresado por la ecuación 3.6 para cada par de imágenes de la secuencia utilizada. Finalmente, los términos λ corresponden al peso que se le da a cada término en la ecuación, lo cual significa que un mayor valor de λ aumenta la contribución de los términos asociados.

$$\begin{aligned} L_{G,F} = ||I_t^{est} - I_t^{HR}||_2 - \lambda_a \log(D(I_{t-1}^{est}, I_t^{est}, I_{t+1}^{est})) \\ + \sum \lambda_l^i ||\phi_D(I_{t-1}^{est}, I_t^{est}, I_{t+1}^{est}) - \phi_D(I_{t-1}^{HR}, I_t^{HR}, I_{t+1}^{HR})||_2 \\ + \sum \lambda_p^i ||\phi_{VGG}(I_t^{est}) - \phi_{VGG}(I_t^{HR})||_2 + \lambda_p L_{PP} \\ + \lambda_w L_{warp} \end{aligned} \tag{3.4}$$

$$L_{warp} = ||I_t^{LR} - W(I_{t-1}^{LR}, F(I_{t-1}^{LR}, I_t^{LR})||_2 \tag{3.5}$$

$$L_{PP} = \sum_{i=1}^{n-1} ||I_t^{est} - \hat{I}_t^{est}||_2 \tag{3.6}$$

3.3. Conjunto de datos

El conjunto de datos utilizado para el proceso de entrenamiento e inferencia de ambas redes neuronales corresponde a una recolección de videos obtenida desde *www.vimeo.com*, compuesta por un total de 60 videos con una resolución de 720, 1080 o 4k píxeles. Los videos se encuentran disponibles para descargar mediante *youtube-dl*, el cual es un programa de libre acceso para descargar videos desde *YouTube*, *Vimeo* u otras plataformas, y la lista de videos utilizada que se encuentra en el repositorio. Estos se componen de tres subconjuntos, cada uno con una finalidad distinta. El entrenamiento de las redes neuronales se realiza principalmente con el subconjunto general, compuesto por 40 videos de simulaciones de flujo de partículas, objetos misceláneos en movimiento, perspectivas a corta y larga distancia, como se observa en los ejemplos de la Figura 3.5, y cuyo objetivo es lograr que se puedan generar objetos con

un movimiento más fluido. Los otros subconjuntos apuntan a mejorar aún más los detalles de ciertos objetos en particular, siendo estos el subconjunto de vehículos, conformado por 10 videos con algunas muestras de este presentadas en la Figura 3.6, y el subconjunto de caracteres, compuesto por 10 videos de caracteres en diversos escenarios y mostrando algunos ejemplos de estos en la Figura 3.7. Para cada subconjunto de datos descrito, el entrenamiento de las redes neuronales considera 75 % de los datos de cada uno, mientras que el 25 % restante es utilizado para evaluación.

La implementación realizada permite al usuario la selección de múltiples carpetas con los videos de interés. Los datos que son utilizados en el proceso de entrenamiento de la red neuronal deben ser previamente procesados, ya sea para modificar las dimensiones, eliminar secuencias que se consideren inútiles, aplicar alguna clase de filtro, entre otros. Para no tener que realizar este proceso cada vez que se vaya a entrenar la red neuronal, usualmente se generan nuevos archivos con los datos ya procesados, por lo que solo es necesario cargarlos en memoria cuando vayan a ser utilizados. En la práctica, al procesar unos cuantos videos del subconjunto general, los archivos generados llegaron a ocupar en el disco hasta 30 giga bytes de espacio. Si se procesaran todos los videos, los archivos generados a partir de estos ocuparían cientos de giga bytes, además de la carga que se daría en la memoria RAM al cargar los datos para su uso, siendo que la RAM se encuentra aún más limitada en capacidad en comparación al disco. Para tratar este uso poco eficiente de los recursos, en la implementación realizada se omite el tener que crear archivos adicionales al generar los datos de entrenamiento constantemente al comienzo de cada iteración, y aunque esto reduce la velocidad del entrenamiento, se permite que los datos sean tan variados como se estime conveniente sin sobrecargar el sistema.

En cuanto al procesamiento de los datos de entrenamiento en sí, durante cada iteración de la etapa de entrenamiento, ya sea para FRVSR o TecoGAN, se realiza una selección aleatoria del total de videos previamente seleccionados. De estos se extraen 40 secuencias de imágenes, a las cuales se les realiza un ajuste de tamaño al de la resolución especificada como alta resolución. Para obtener los pares correspondientes a baja resolución, se aplica sobre las imágenes de alta resolución un filtro gaussiano con parámetro $\sigma = 1{,}5$, seguido finalmente de una selección del valor de uno cada cuatro píxeles, para cada dimensión de la imagen, obteniendo así las imágenes en baja resolución, siendo estas de un cuarto de tamaño en cuanto a las dimensiones espaciales de una imagen de alta resolución.

3.4. Sistema de reconocimiento You Only Look Once

You Only Look Once (YOLO) [41] es un sistema de detección de objetos en tiempo real que se encuentra en el estado del arte, desarrollado en base a redes neuronales convolucionales, el cual calcula el nivel de certeza al detectar un objeto. En cuanto al proceso de detección en sí, en resumen, la imagen a reconocer es dividida en secciones a partir de una grilla de tamaño predeterminado. Esta imagen seccionada sirve luego como entrada a la red neuronal, la cual calcula y da como resultado las coordenadas de los rectángulos contenedores de los objetos encontrados, cada uno con su respectiva predicción de las clases detectadas. Finalmente, a partir de las predicciones realizadas, se determina qué clase es la que posee un mayor nivel de certeza en cada rectángulo, y si esta supera cierto umbral se determina que corresponde efectivamente a tal objeto. La Figura 3.8 muestra un ejemplo de una imagen ya procesada por YOLO, a la cual se le añadieron los rectángulos contenedores, mostrando encima de cada rectángulo el objeto correspondiente reconocido con la mayor probabilidad de que sea tal.

Para este trabajo se hace uso de YOLO para comprobar que los resultados de una red neuronal de súper-resolución son mejores con respecto a otra a través de la comparación de precisión, con lo cual es posible demostrar el aporte de los sistemas de súper-resolución a este tipo de aplicaciones de detección y clasificación, y cuyo esquema de incorporación está representado por la Figura 3.9. En el repositorio de YOLO se encuentra un modelo previamente entrenado, capaz de reconocer objetos generalizados, siendo algunos ejemplos autos, perros y bicicletas, aunque sin poder indicar subcategorías para cada clase. Para las pruebas realizadas se utiliza este modelo pre-entrenado, al cual no se le realiza un tuneo fino de sus parámetros, en donde se define por configuración que el umbral para considerar el objeto detectado, en caso de haber uno o varios, es de un mínimo de 60.00 %.

3.5. Hardware y software utilizados

Debido a que las redes neuronales convolucionales continúan desarrollándose cada vez a mayor escala, es necesario que el hardware que acompañe su avance también cumpla con ciertos requisitos. Como se menciona previamente, el hardware principal para manipular y utilizar las redes neuronales corresponde a la GPU. Ya sea para la fase de entrenamiento o inferencia, los resultados obtenidos no varían en calidad al usar diferentes GPUs, sino más bien en los tiempos de procesamiento. Las especificaciones de la GPU utilizada y del sistema que la acompaña para realizar el proceso de entrenamiento y evaluación corresponde a la GPU

Nvidia Geforce GTX 1060 cuenta con 6GB de RAM y 1280 CUDA cores, incorporada en un Notebook Gamer Lenovo Legion Y720, con procesador Intel Core i7 de séptima generación Quad Core 2.80Ghz, sistema operativo Ubuntu Linux.

En cuanto al software, para poder hacer uso de la implementación desarrollada a lo largo de todo el trabajo, es necesario que se cumplan ciertas especificaciones de bibliotecas para su correcto funcionamiento. Estas se encuentran en el archivo *Pipfile* del repositorio, para usarse al momento de ejecutar *Pipenv*, aplicación de entorno virtual para Python.

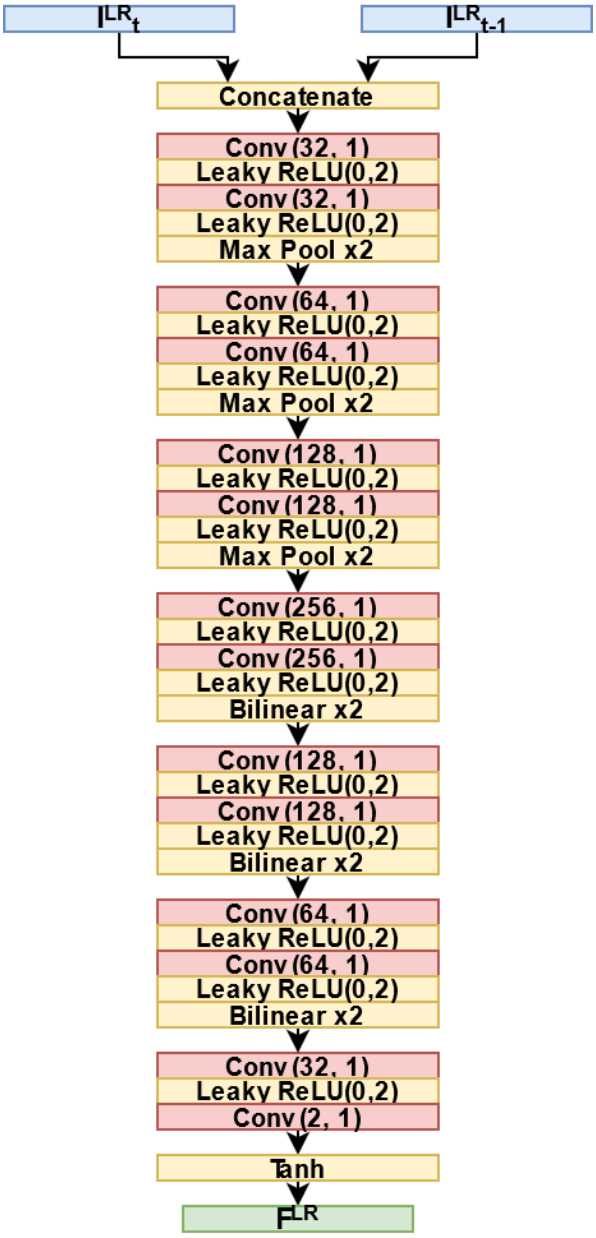


Figura 3.2: Arquitectura de Flow Net

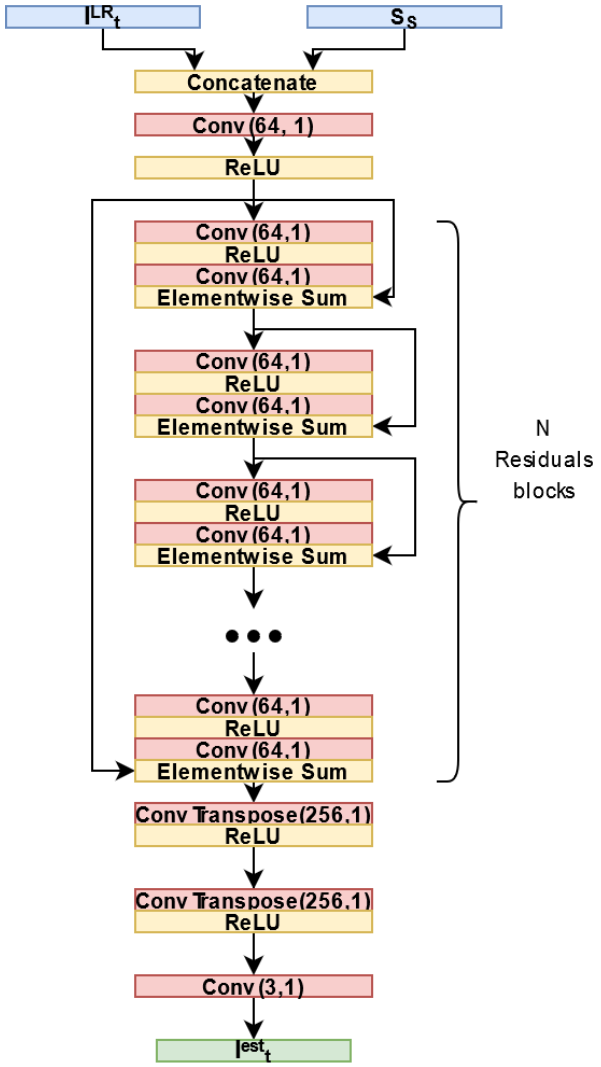


Figura 3.3: Arquitectura de Super Resolution Net

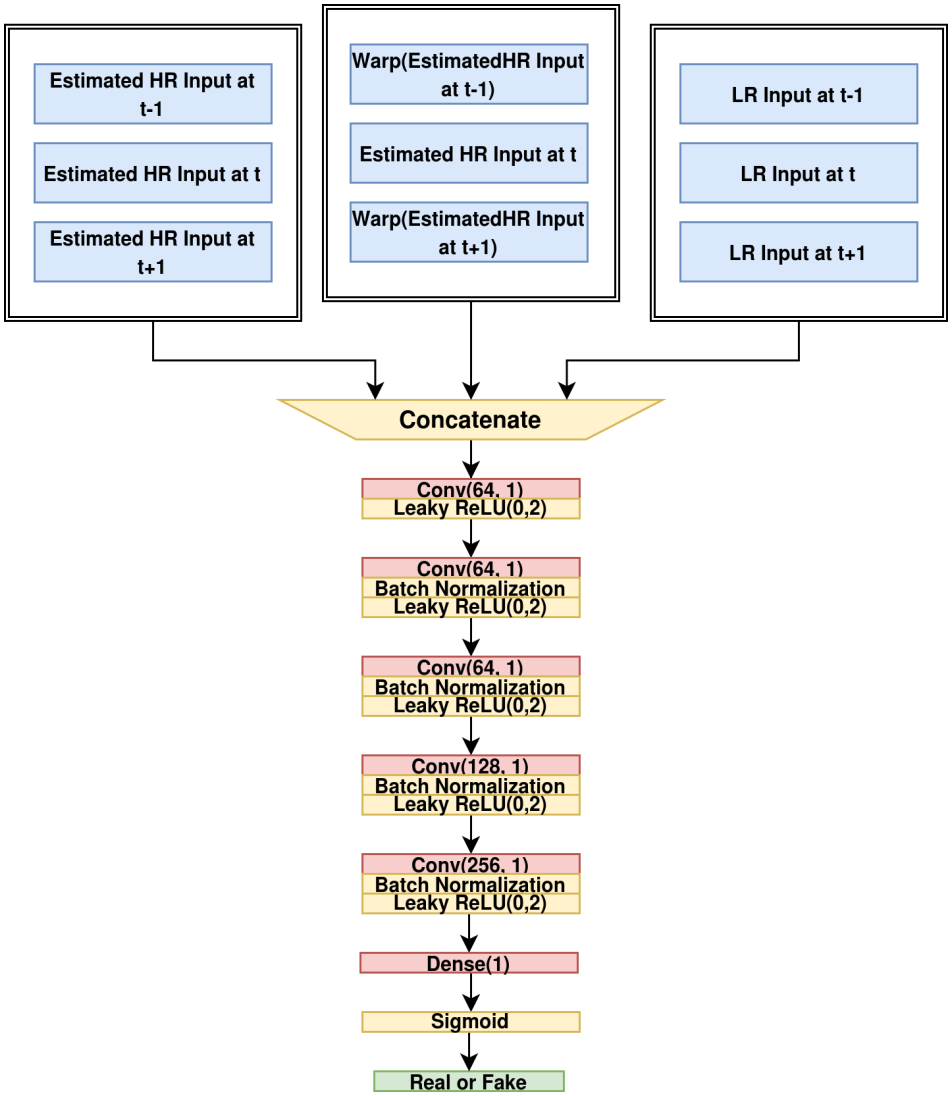


Figura 3.4: Arquitectura del discriminador para Temporally Coherent GANs for Video Super-Resolution

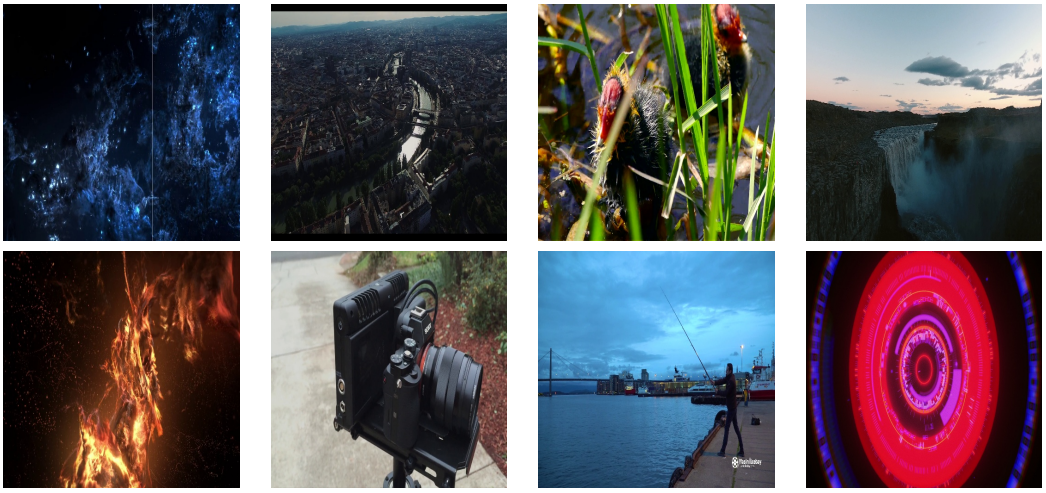


Figura 3.5: Imágenes de muestra del subconjunto general

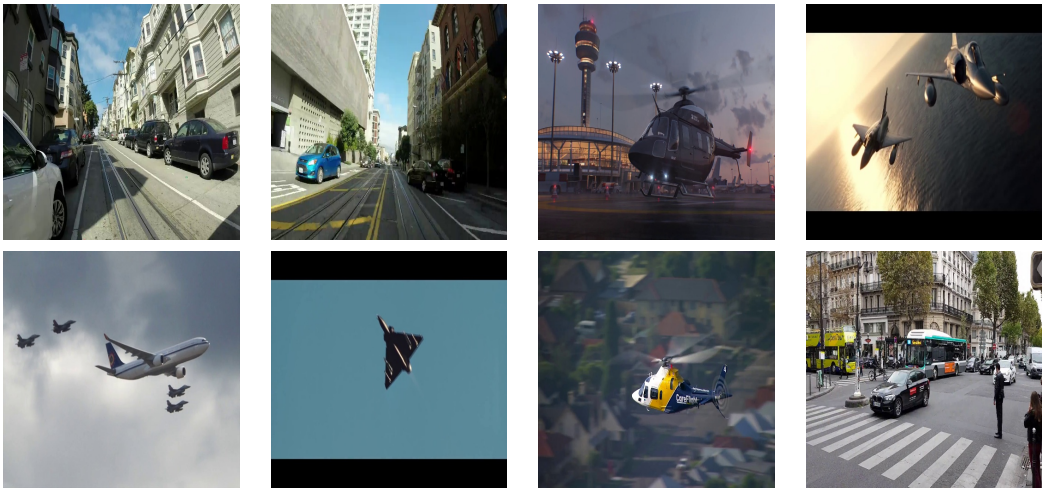


Figura 3.6: Imágenes de muestra del subconjunto de vehículos

Figura 3.7: Imágenes de muestra del subconjunto de caracteres

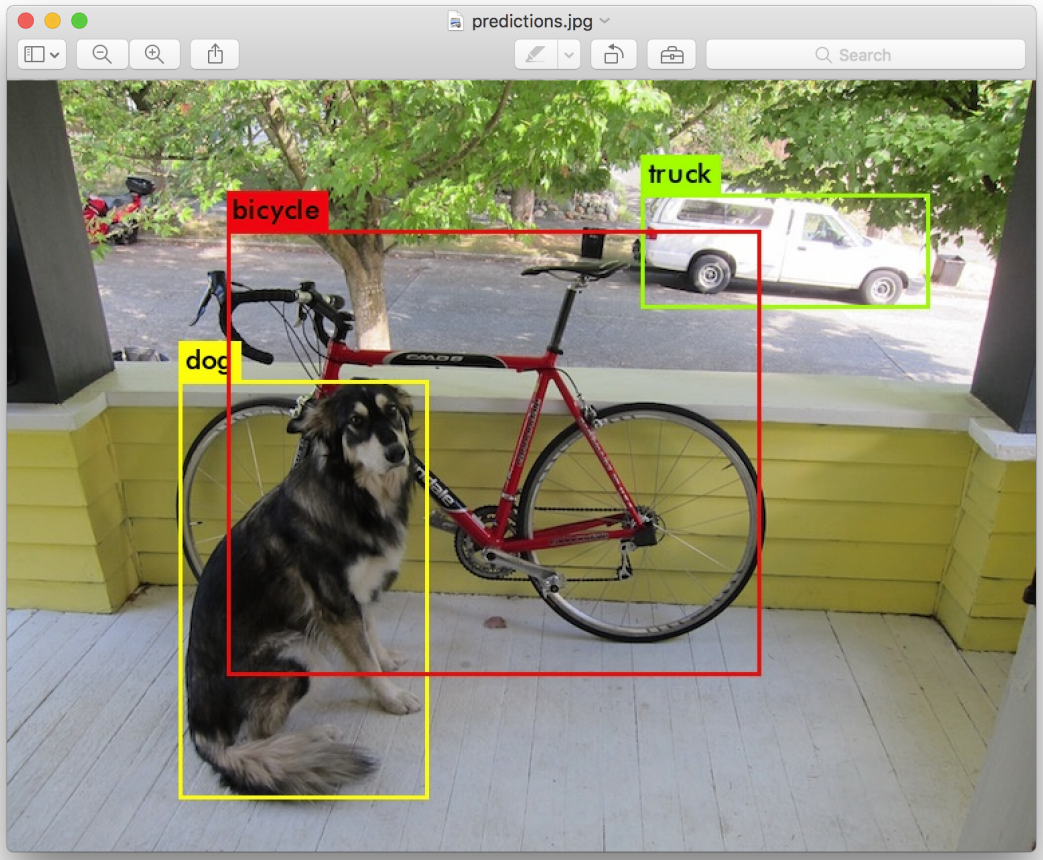


Figura 3.8: Ejemplo de imagen procesada por YOLO

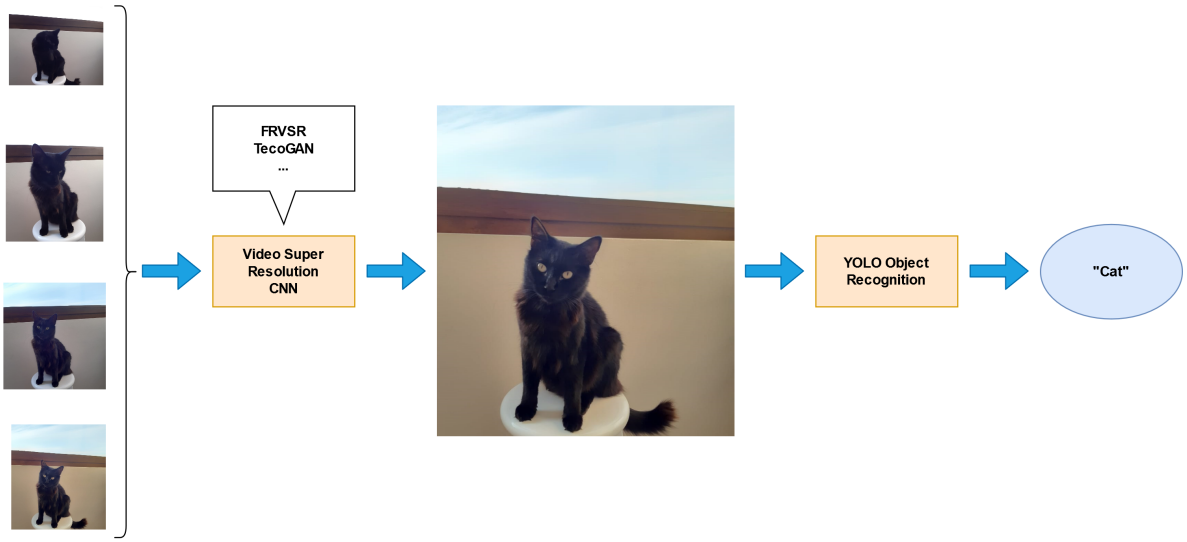


Figura 3.9: Sistema YOLO para súper-resolución

Capítulo 4

Resultados experimentales

En este capítulo se explican y presentan los resultados obtenidos ante los diferentes experimentos desarrollados, además del procedimiento seguido y algunos detalles técnicos en cada uno. Principalmente se tratan tres casos de interés, siendo en primera instancia la influencia del conjunto de datos en el proceso de entrenamiento de las redes neuronales, especializando los datos a ciertos objetos de interés. Luego se realiza una comparación de desempeño al variar la resolución de entrada a las redes neuronales. En estos dos casos estudio se utilizan principalmente las métricas de *Peak Signal-to-Ratio* (PSNR) y *Similarity Structural Index Measure* (SSIM), siendo además que se calculan los resultados para el método de interpolación bicúbica como ajuste de resolución, usando este método como referencia para mencionar diferencias y posibles mejoras al analizar los resultados obtenidos.

4.1. Entrenamiento especializado

Esta sección considera los experimentos realizados al especializar las redes neuronales para mejorar ciertos objetos de interés, esto con el fin de demostrar los efectos a nivel de detalles en las imágenes generadas por las redes neuronales al ser entrenadas con diferentes conjuntos de datos. Para este caso de estudio los datos corresponden a los descritos anteriormente (datos generales, caracteres, y vehículos), cuyo uso se explica más adelante al describir el procedimiento seguido. La realización de este experimento solo considera la red neuronal FRVSR y no TecoGAN, ya que entrenar esta última toma un tiempo considerable en comparación a FRVSR, además de que basta con FRVSR para demostrar la necesidad de utilizar o no un entrenamiento especializado ya que el generador de TecoGAN está basado en FRVSR, y por lo tanto se obtendrían resultados similares.

En cuanto al procedimiento utilizado para el entrenamiento y evaluación, en primera instancia se entrena FRVSR con una configuración de resolución de imagen de entrada de 64x64 píxeles, y por lo tanto de 256x256 píxeles de salida, utilizando el dataset genérico en un comienzo para el 95 % del total de iteraciones. Para finalizar el resto del entrenamiento, en el primer caso se mantienen los datos generales, refiriéndose a la red neuronal obtenida como FRVSR general. En los demás casos se añaden a los datos generales los conjuntos de caracteres y vehículos, cada uno por separado, obteniendo para cada uno las redes FRVSR de caracteres y FRVSR de vehículos respectivamente, teniendo finalmente tres subredes. Luego del entrenamiento, se evalúan todas las subredes con cada subconjunto de datos, tanto para los datos de evaluación y entrenamiento, obteniendo para cada red neuronal los valores promedio de PSNR y SSIM. Además, en el caso de los datos generales y vehículos, se obtienen muestras de las redes neuronales para aplicar la métrica YOLO y analizar el resultado de esta. El entrenamiento especializado considera la reconstrucción a partir de una imagen de 128x128 píxeles, por lo cual esta última es seccionada en cuatro imágenes para poder ser procesada por las redes neuronales.

En el caso de los datos generales se busca medir la capacidad de la red neuronal para generar detalles y texturas varias. A partir de los resultados obtenidos, tabla 4.1, se aprecia que cada red basada en FRVSR supera los resultados de la interpolación bicúbica, tanto en PSNR como SSIM. Al analizar más en profundidad los resultados calculados, en el mejor de los casos, el PSNR de los datos de entrenamiento presenta una mejora del 3.79 % , mientras que en la evaluación la mejora es de un 4.18 %, ambos obtenidos a partir de la red FRVSR general. En el caso del SSIM, tanto para entrenamiento como evaluación, es FRVSR de caracteres la que supera a los demás métodos, obteniendo mejoras del 4.29 % y 2.44 % respectivamente. Los resultados obtenidos no difieren demasiado entre cada red neuronal, lo cual se encuentra dentro de lo previsto, debido a que cada una utiliza datos similares en su mayoría durante el proceso de entrenamiento. La Figura 4.1 presenta muestras generadas por cada método, además de la imagen original de alta resolución. En estas se puede apreciar visualmente las mejoras que logran las redes neuronales en comparación a la interpolación bicúbica, además de que las diferencias entre las imágenes de redes neuronales y alta resolución, por lo menos en este cuadro, son casi imperceptibles.

La Figura 4.2 corresponde a las muestras obtenidas a partir de cada método para una las imágenes generales, y cuyas métricas se presentan en la Tabla 4.2. En esta se reconoce a una persona en la imagen de alta resolución con una certeza de 97.57 %, mientras que la interpolación bicúbica logra que se reconozca esta con tan solo un 63.75 %. En cuanto a las subredes,

	Peak Signal-to-Noise Ratio		Structural Similarity Index	
	Entrenamiento	Evaluación	Entrenamiento	Evaluación
Int. bicúbica	26.39	26.07	0.9118	0.9130
FRVSR general	27.39	27.16	0.9499	0.9349
FRVSR caracteres	27.38	27.11	0.9510	0.9357
FRVSR vehículos	27.09	27.14	0.9485	0.9353

Tabla 4.1: Métricas de PSNR y SSIM promedio para datos generales

estas obtienen valores sobre el 98.00 %, lo que da a entender que, al menos en este caso, los detalles generados por las redes neuronales permiten generar características intrínsecas que definen mejor que la imagen original a una persona como tal según YOLO.

	PSNR	SSIM	Predicción YOLO (persona)
Int. bicúbica	29.74	0.8966	63.75
FRVSR general	33.44	0.9516	98.76
FRVSR caracteres	32.66	0.9430	98.67
FRVSR vehículos	32.64	0.9431	98.38
Alta resolución	-	-	97.57

Tabla 4.2: Métricas de PSNR, SSIM y predicción YOLO para la Figura 4.2

En el caso de los datos de caracteres, tabla 4.3, los mejores resultados se obtienen a partir de la red FRVSR de caracteres. Al evaluar el PSNR, la red FRVSR de caracteres efectivamente es la que obtiene los mejores resultados, logrando mejoras de 15.66 % para datos de entrenamiento y 6.54 % para datos de evaluación. En cuanto al SSIM, al analizar este se observa que FRVSR caracteres destaca sobre los demás métodos, obteniendo mejoras de hasta un 4.90 % y 4.97 % para los datos de entrenamiento y evaluación respectivamente. Esto ocurre debido a que el SSIM se basa el análisis de la forma de lo observado, característica que lo hace fundamental en este escenario, ya que permite comparar de forma efectiva los detalles de los caracteres generados. En las muestras de la Figura 4.3, aunque las tres redes neuronales permiten distinguir a primera vista la palabra que aparece, se puede notar que los detalles de las letras son más claros para la imagen obtenida desde la red FRVSR caracteres. En el caso de la interpolación bicúbica, la imagen obtenida no permite afirmar la presencia de caracteres de algún tipo, lo que demuestra en este caso de interés los beneficios de usar redes neuronales.

Finalmente, se presentan los obtenidos para los datos de vehículos. A partir de las métricas calculadas, tabla 4.4, se tiene que FRVSR vehículos obtiene los mejores resultados, presen-

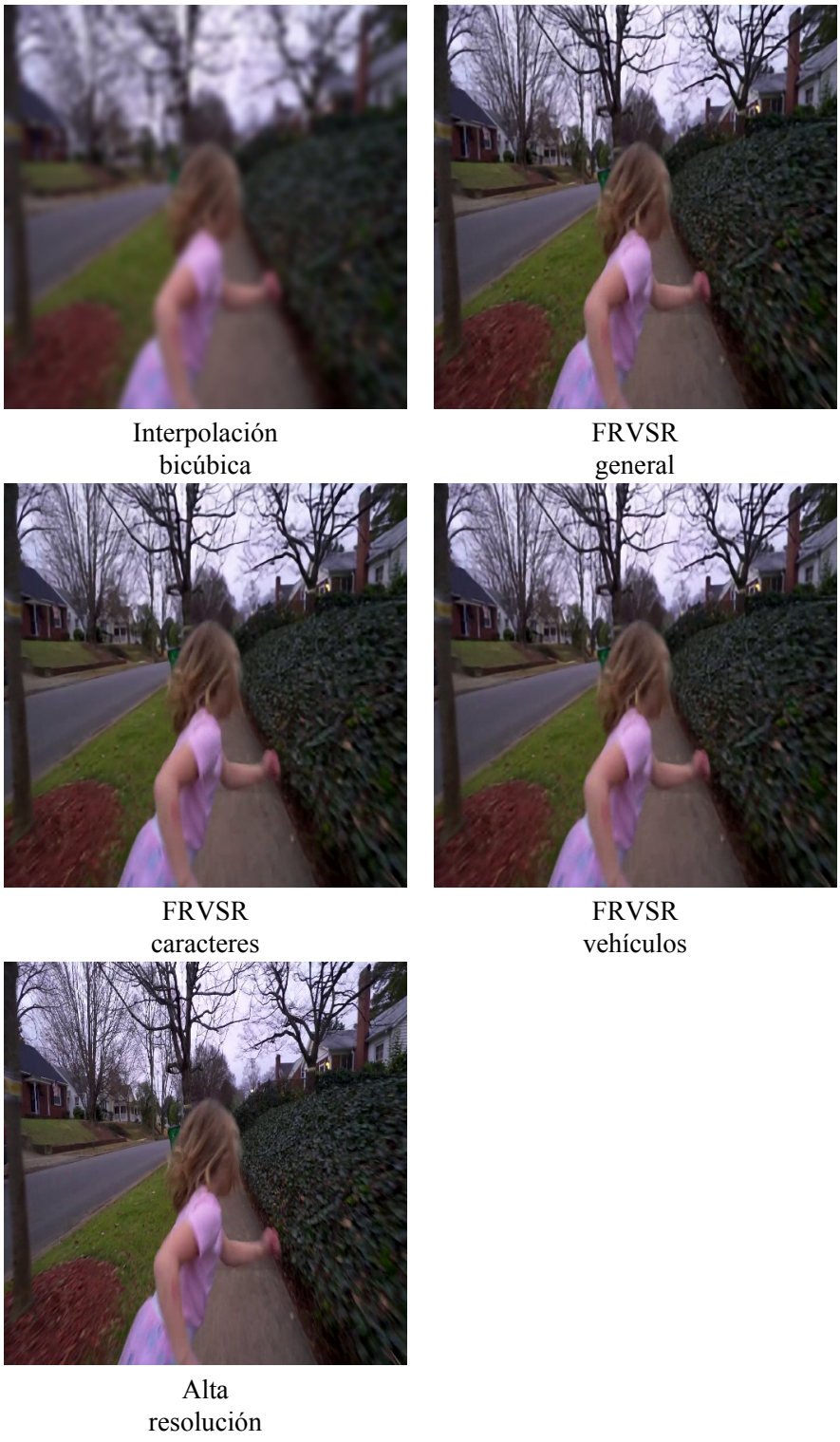
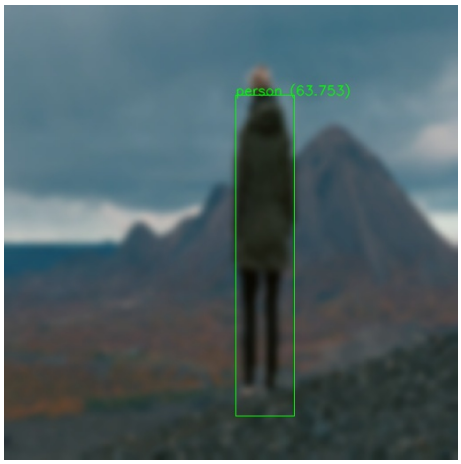
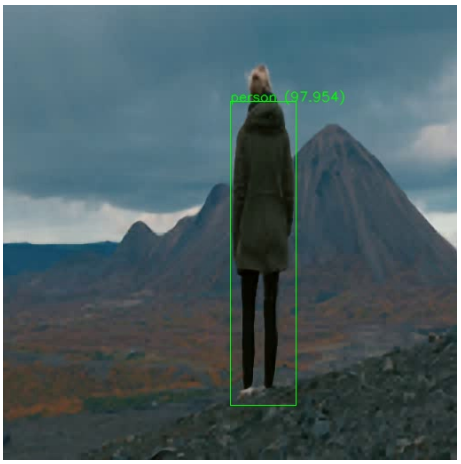


Figura 4.1: Resultado de imágenes generales para FRVSR especializados.

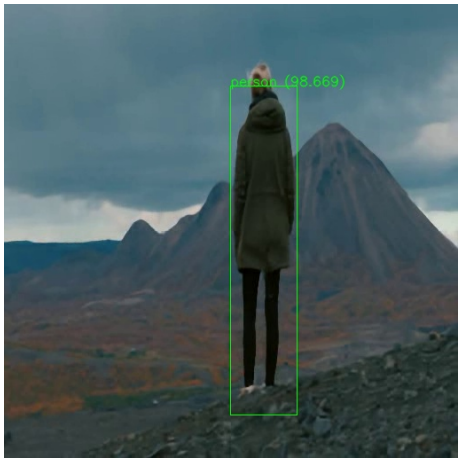
tando mejoras de hasta 10.4 % y 9.7 % en el PSNR para datos de entrenamiento y evaluación respectivamente, cuyos valores no difieren demasiado de las demás FRVSR. En cuanto al



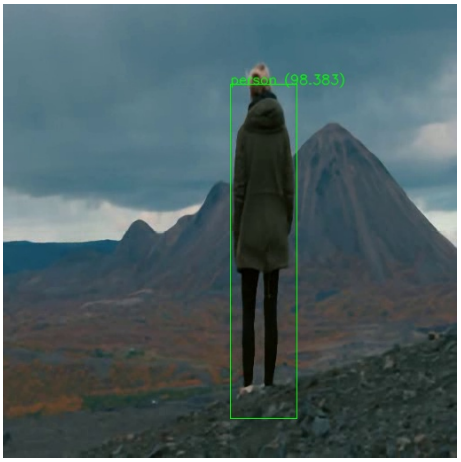
Interpolación
bicúbica



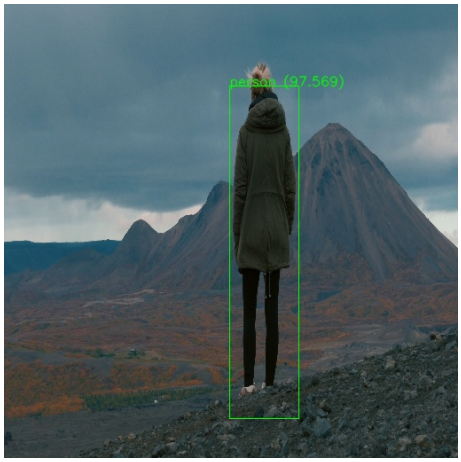
FRVSR
general



FRVSR
letras



FRVSR
vehículos



Alta
resolución

Figura 4.2: Resultado de imágenes generales de para FRVSR especializado aplicando YOLO

	Peak Signal-to-Noise Ratio		Structural Similarity Index	
	Entrenamiento	Evaluación	Entrenamiento	Evaluación
Int. bicúbica	25.48	25.51	0.9382	0.9273
FRVSR general	27.89	26.81	0.9602	0.9410
FRVSR caracteres	29.47	27.18	0.9842	0.9734
FRVSR vehículos	28.08	27.15	0.9607	0.9436

Tabla 4.3: Métricas de PSNR y SSIM promedio para datos de caracteres

SSIM, se presentan mejoras del 4.24 % en el entrenamiento y 2.69 % en la evaluación, en donde se tiene una diferencia suficiente en comparación a los demás métodos para decir que FRVSR de vehículos efectivamente mejora los resultados. Esto demuestra que la especialización de vehículos es necesaria, sobre todo si se utiliza para observar este tipo de objetos a largas distancias, donde los detalles generados adecuadamente toman mayor importancia

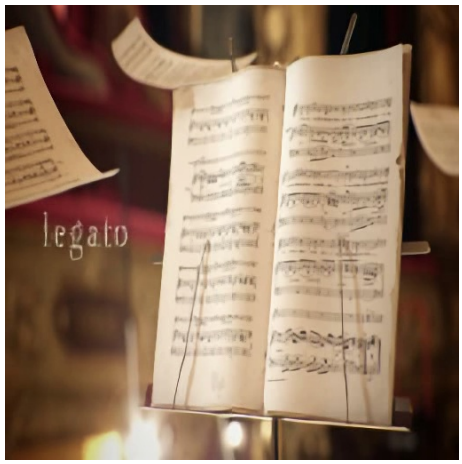
	Peak Signal-to-Noise Ratio		Structural Similarity Index	
	Entrenamiento	Evaluación	Entrenamiento	Evaluación
Int. bicúbica	25.66	25.46	0.9333	0.9351
FRVSR general	27.96	26.89	0.9637	0.9468
FRVSR caracteres	28.03	27.74	0.9644	0.9501
FRVSR vehículos	28.33	27.93	0.9729	0.9603

Tabla 4.4: Métricas de PSNR y SSIM promedio para datos de vehículos

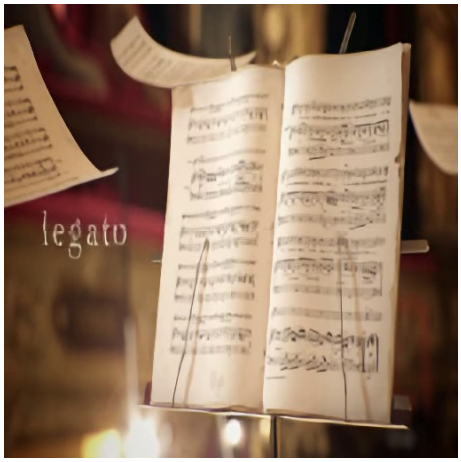
La Figura 4.5 presenta las muestras obtenidas para las imágenes de vehículos, y cuyas métricas calculadas se presentan en la Tabla 4.5. El objeto a reconocer corresponde a un tren, en donde la interpolación bicúbica no permite reconocer este debido a que no supera el umbral, mientras que las redes neuronales se encuentran sobre el 89.00 %. A pesar de en el mejor caso se obtiene un 97.50 % de certeza por parte de FRVSR caracteres y supera a la alta resolución, esta última logra ajustar de manera más precisa el rectángulo contenedor de lo que corresponde al tren en sí.

A partir de los resultados obtenidos para el entrenamiento especializado, se tiene que efectivamente se logran mejores resultados al entrenar con datos de objetos específicos. En el caso de FRVSR caracteres, es esta red la que supera a las demás en la mayoría de los distintos escenarios presentados. A pesar de esto, en los ejemplos mostrados se puede apreciar que la red FRVSR general permite, por lo menos visualmente, generar imágenes bastante similares a las de las redes FRVSR caracteres y FRVSR vehículos, siendo la diferencia con estas los detalles finos obtenidos por cada red. Por lo tanto, la red neuronal FRVSR puede generar resultados

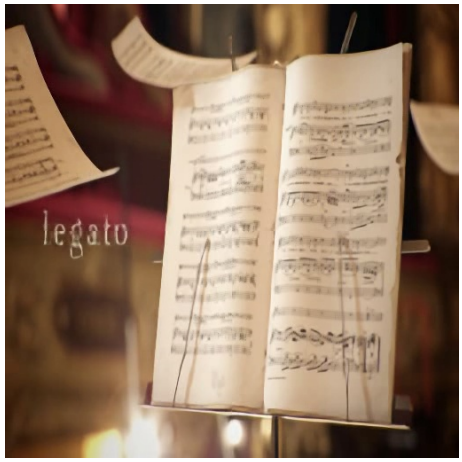
Interpolación
bicúbica



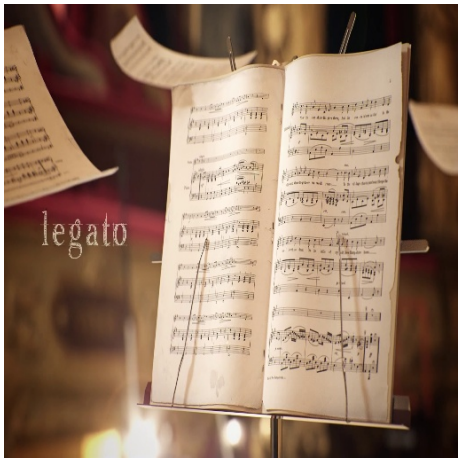
FRVSR
general



FRVSR
caracteres



FRVSR
vehículos



Alta
resolución

Figura 4.3: Resultado de imágenes de caracteres para FRVSR especializados

Interpolación
bicúbica

FRVSR
general

FRVSR
letras

FRVSR
vehículos

Alta
resolución

Figura 4.4: Resultado de imágenes de vehículos para FRVSR especializado

	PSNR	SSIM	Predicción YOLO (tren)
Int. bicúbica	18.42	0.5418	< 60.00
FRVSR general	23.46	0.8483	89.21
FRVSR caracteres	22.85	0.8337	97.50
FRVSR vehículos	23.71	0.8595	96.63
Alta resolución	-	-	82.29

Tabla 4.5: Métricas de PSNR y SSIM para la Figura 4.5

aceptables al generalizar los datos y no es necesario entrenarla nuevamente al querer utilizarla para nuevos objetos de interés. Sin embargo, si fuera el caso de querer mejorar aún más los detalles de estos nuevos objetos, entonces se sugiere añadir los datos pertinentes y entrenar por una cierta cantidad de iteraciones.

4.2. Resolución

En esta sección se consideran las pruebas de variación de resolución. Estas consisten en utilizar diferentes configuraciones de resolución para las imágenes de entrada a las redes neuronales, tanto para FRVSR como TecoGAN, para comprobar los efectos de la información utilizada por la red neuronal en la generación de detalles de la estimación de salida. En este caso se utilizan tres diferentes configuraciones de resolución para la salida de las redes, siendo estas de 128x128, 256x256 y 512x512 pixeles, tanto para FRVSR como TecoGAN, dando así un total de seis subredes a entrenar. Los datos utilizados corresponden al total de los tres subconjuntos (general, caracteres y vehículos), y manteniendo la proporción de un 75 % de cada uno para el entrenamiento y el 25 % como dato de evaluación.

Al ser cada subred neuronal de diferente resolución, es necesario entrenar cada una de forma independiente, para lo cual se utiliza en cada caso el total de las iteraciones, sin añadir o eliminar datos de entrenamiento durante todo el proceso. Para la evaluación de resultados se tiene como entrada a las redes neuronales imágenes de resolución de 128x128 píxeles, seccionando según corresponda para la configuración de cada una. Para cada conjunto de entrenamiento y evaluación, se calcula el promedio de PSNR y SSIM a partir de los resultados de las subredes, utilizando nuevamente la interpolación bicúbica como punto de comparación. Adicionalmente, se presenta el tiempo promedio de procesamiento de cada subred.

En la Tabla 4.6 se presentan los resultados obtenidos para cada red neuronal y la resolución utilizada, utilizando el total de los datos. En general se observa que las redes neuronales nuevamente presentan mejores resultados que la interpolación bicúbica. Analizando más en detalle, los mejores resultados se obtienen a partir de la subred TecoGAN 128, en donde el PSNR presenta mejoras de hasta 17.12 % y 6.74 % para los datos de entrenamiento y evaluación respectivamente. En el caso del SSIM, los datos de entrenamiento mejoraron hasta un 5.35 %, mientras que en los datos de evaluación se logra hasta un 4.17 %. Además, se puede observar que en cada configuración de resolución utilizada, TecoGAN logra obtener mejores resultados que la respectiva FRVSR. En cuanto al tiempo de de inferencia de las subredes, el cual corresponde al tiempo desde que se le da a una red neuronal una cierta entrada hasta que se calcula la salida, este no presenta mayores variaciones entre cada una, obteniendo valores de aproximadamente 29[ms]. Esto se debe a que, aunque el tiempo que toma la red neuronal en procesar la información disminuye proporcionalmente a la cantidad de información de entrada, el tener que procesar varias imágenes por la subdivisión correspondiente lleva a que el tiempo de procesamiento final, al compararlo entre las subredes, no presenta diferencias

significativas. Sin embargo, es necesario mencionar que el tiempo que toma entrenar FRVSR es mucho menor que al entrenar TecoGAN, en donde la diferencia entre ambas redes llega a ser de hasta semanas.

	Peak Signal-to-Noise Ratio		Structural Similarity Index		
	Entrenamiento	Evaluación	Entrenamiento	Evaluación	Tiempo de inferencia
Int. bicúbica	26.28	26.55	0.9218	0.9232	-
FRVSR 32	27.66	26.67	0.9543	0.9503	29[ms]
FRVSR 64	27.96	26.97	0.9639	0.9531	28[ms]
FRVSR 128	29.37	27.54	0.9655	0.9541	28[ms]
TecoGAN 32	28.76	27.45	0.9632	0.9538	30[ms]
TecoGAN 64	30.22	27.96	0.9702	0.9610	28[ms]
TecoGAN 128	30.78	28.34	0.9711	0.9617	30[ms]

Tabla 4.6: Métricas de PSNR y SSIM calculadas para FRVSR y TecoGAN según la resolución

A continuación se presentan algunos ejemplos obtenidos a partir de las subredes al ser procesados posteriormente mediante YOLO. El primer caso, el cual se muestra en la Figura 4.6 y cuyas métricas calculadas se presentan en la Tabla 4.7, corresponde a la imagen de una persona. En la imagen original de alta resolución se tiene que la probabilidad de corresponder a una persona es de un 98.84 %. Al utilizar interpolación bicúbica la probabilidad no supera el umbral de 60.00 %, a diferencia de las imágenes obtenidas al hacer uso de redes neuronales, las cuales logran obtener todas sobre el 87.00 %, y en el mejor caso un 98.42 % correspondiente a la red FRVSR 64. Es necesario mencionar que TecoGAN 128 tiene los mejores valores de PSNR y SSIM, siendo estos de 31.70 y 0.9740 respectivamente, y cuya probabilidad obtenida por YOLO es de un 97.96 %, mientras que FRVSR 64 obtiene un PSNR de 30.88 y SSIM de 0.9740, y aún así la probabilidad de esta última subred es mejor que la obtenida por TecoGAN 128. Por lo tanto, obtener los mejores valores de PSNR o SSIM no implica necesariamente que las características intrínsecas generadas para un cierto objeto sean las más adecuadas.

En el segundo caso, para el cual se presentan las muestras en la Figura 4.7 y las métricas calculadas en la Tabla 4.8, se tiene que el objeto de interés corresponde a un bote navegando, el cual se encuentra una distancia mayor de la cámara en comparación a la persona del caso anterior. A partir de los resultados se tiene que la imagen de alta resolución logra una predicción del 98.84 %. En cuanto a los métodos utilizados para ajustar la resolución, nuevamente solo es la interpolación bicúbica la que no logra superar el umbral predeterminado, mientras que las redes neuronales logran obtener sobre 90.00 %, a excepción de FRVSR con un 81.84 %. En este caso, los mejores valores para las métricas, tanto para el PSNR y SSIM

	PSNR	SSIM	Predicción YOLO (persona)
Int. bicúbica	24.93	0.9024	< 60.00
FRVSR 32	29.48	0.9594	87.86
FRVSR 64	30.04	0.9636	98.42
FRVSR 128	30.33	0.9649	97.55
TecoGAN 32	30.88	0.9698	89.22
TecoGAN 64	31.43	0.9730	98.12
TecoGAN 128	31.70	0.9740	97.96
Alta resolución	-	-	98.84

Tabla 4.7: Métricas de PSNR, SSIM y predicción YOLO para la Figura 4.6

como en la probabilidad según YOLO de corresponder a un bote, son obtenidas por la subred TecoGAN 128, cuya probabilidad es de un 98.40 %.

	PSNR	SSIM	Predicción YOLO (bote)
Int. bicúbica	27.20	0.9034	< 60.00
FRVSR 32	31.19	0.9566	81.84
FRVSR 64	32.55	0.9640	92.30
FRVSR 128	32.73	0.9660	97.33
TecoGAN 32	32.59	0.9688	90.95
TecoGAN 64	33.94	0.9738	96.90
TecoGAN 128	34.12	0.9752	98.40
Alta resolución	-	-	98.70

Tabla 4.8: Métricas de PSNR, SSIM y predicción YOLO para la Figura 4.7

En el último caso, cuyas muestras corresponden a las de la Figura 4.8 y resultados de métricas a las de la Tabla 4.9, el objeto de interés, siendo este un aeroplano, se encuentra a una distancia mayor que en los ejemplos anteriores, y siendo para la imagen de alta resolución una certeza de 95.63 % de que el objeto corresponde a un aeroplano. En este caso la interpolación bicúbica obtiene el mayor PSNR, siendo este de 40.58, mientras que la probabilidad obtenida se encuentra por debajo del 60.00 %. En cuanto a las redes neuronales, se puede observar que FRVSR 128 y TecoGAN 128 tampoco alcanzan el umbral, siendo además que TecoGAN 128 obtuvo el mejor SSIM, con un valor de 0.9951. El resto de las redes neuronales si logran que se pueda reconocer el aeroplano al sobrepasar el umbral, en donde la mejor probabilidad está dada por TecoGAN 32, con un 85.96 %.

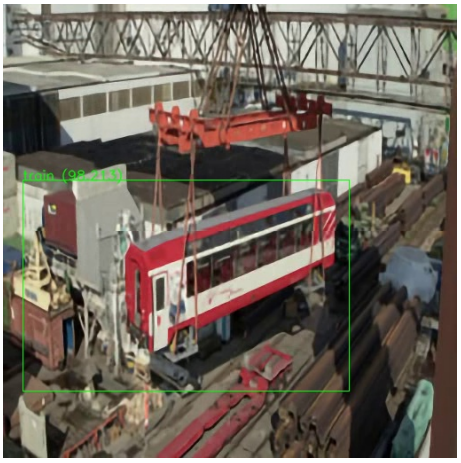
A partir de los resultados obtenidos cuando se varía la resolución de entrada, es posi-

	PSNR	SSIM	Predicción YOLO (aeroplano)
Int. bicúbica	40.58	0.9899	< 60.00
FRVSR 32	36.13	0.9917	76.99
FRVSR 64	36.82	0.9929	72.77
FRVSR 128	38.12	0.9931	< 60.00
TecoGAN 32	37.55	0.9947	85.96
TecoGAN 64	38.27	0.9948	82.11
TecoGAN 128	39.29	0.9951	< 60.00
Alta resolución	-	-	95.63

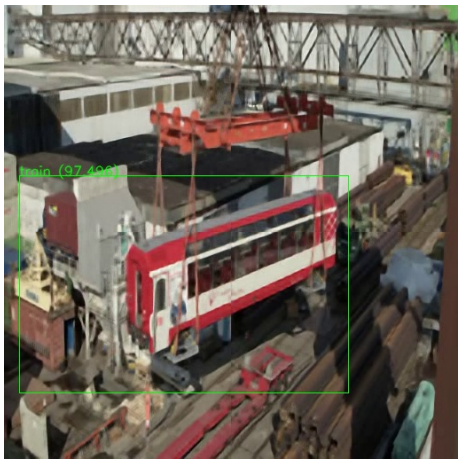
Tabla 4.9: Métricas de PSNR, SSIM y predicción YOLO para la Figura 4.8

ble afirmar que en cuanto al PSNR y SSIM, las imágenes presentan mejores resultados de métricas al aumentar la resolución, siendo la red TecoGAN 128 la que permite obtener los mejores resultados. Esto se debe a que en los bordes de una imagen, la información dada por la vecindad de píxeles es menor, problema al que a las redes neuronales de menor resolución de entrada se les presenta en mayor medida por la subdivisión de la imagen de entrada. Sin embargo, al evaluar mediante la red neuronal YOLO, se demuestra que el aumentar el PSNR o SSIM no significa necesariamente que se codifiquen correctamente las características intrínsecas. Aún así, las redes neuronales de súper-resolución demostraron que efectivamente permiten mejorar sistemas de reconocimiento.

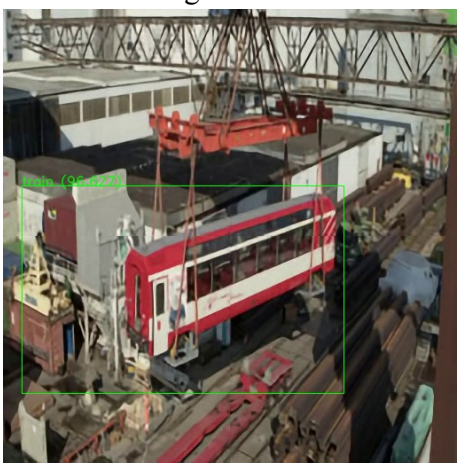
Interpolación
bicúbica



FRVSR
general



FRVSR
letras



FRVSR
vehículos

Alta
resolución

Figura 4.5: Resultado de imágenes de vehículos de para FRVSR especializado aplicando YOLO

Figura 4.6: Resultado de imágenes de FRVSR y TecoGAN ante YOLO

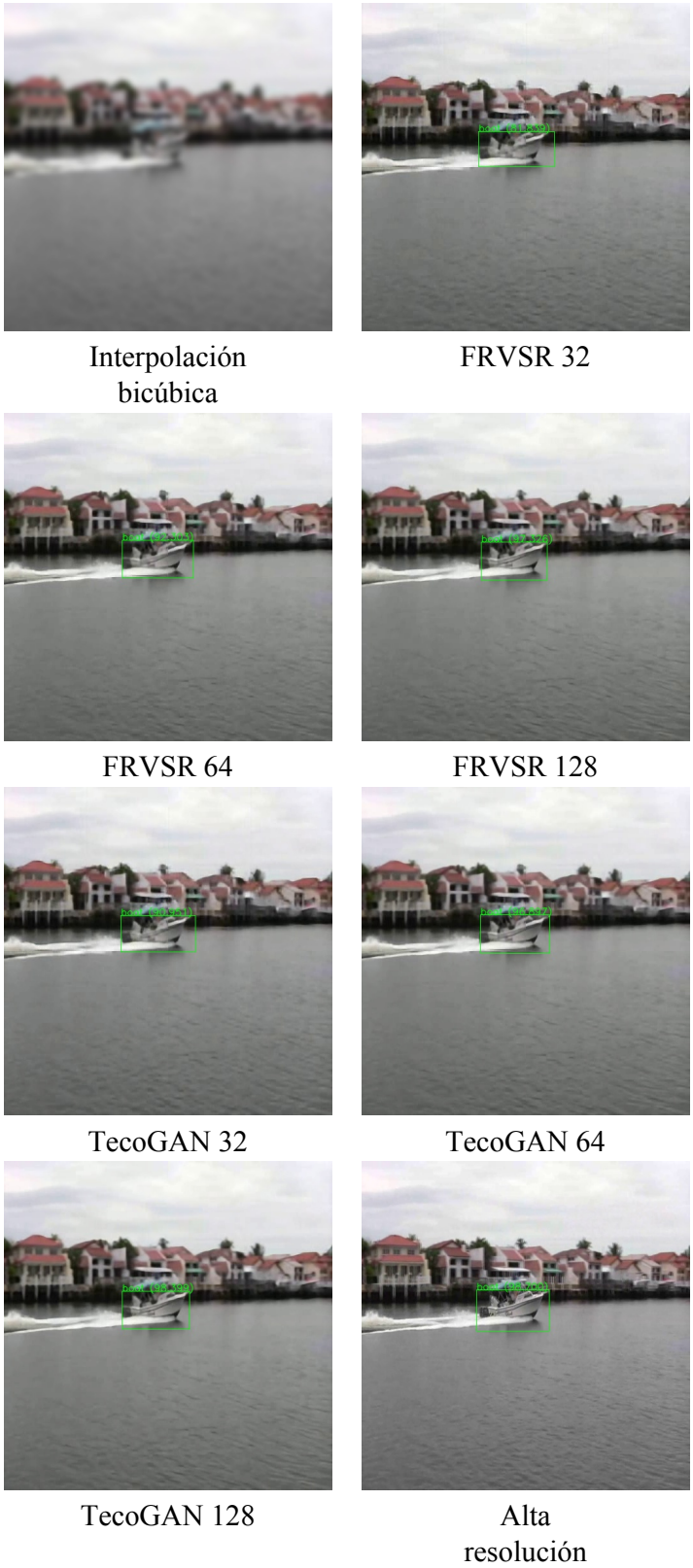


Figura 4.7: Resultado de imágenes para FRVSR y TecoGAN ante YOLO

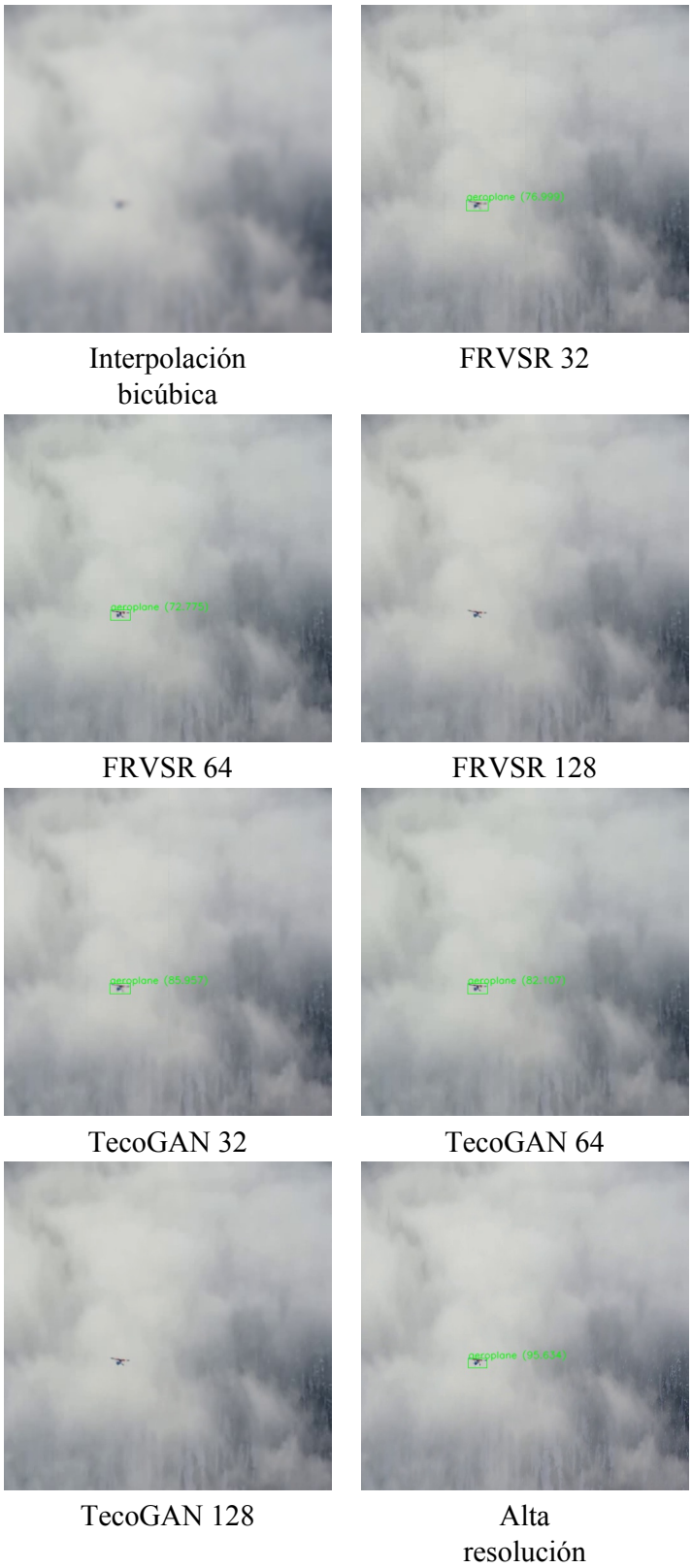


Figura 4.8: Resultado de imágenes para FRVSR y TecoGAN ante YOLO

Capítulo 5

Conclusión

5.1. Conclusiones

A partir del trabajo realizado se logró evaluar el desempeño de redes neuronales aplicadas a video, FRVSR y TecoGAN, para los casos de estudio, caracterizando las métricas de interés y analizando estas según correspondiera.

El entrenamiento especializado muestra que utilizar datos adicionales de objetos de interés ayuda en el entrenamiento de las redes neuronales, mejorando los datos generados tanto en color, PSNR, como en forma, SSIM, al comparar con la interpolación bicúbica. En cuanto a una aplicación más práctica de los datos generados por la redes neuronales, en los ejemplos presentados utilizando YOLO se tiene que la interpolación bicúbica no logra sobrepasar el umbral de 60 % de certeza para los objetos de interés, o apenas logra superar el umbral, mientras que los métodos por redes neuronales, en la mayoría de los casos, obtienen valores que superan el 90 %.

Refiriéndose a lo que sería la percepción visual humana, teniendo en cuenta lo descrito en el planteamiento del problema, en lo que es la necesidad de los pilotos de tomar decisiones rápidas y acordes a la situación, los resultados de la interpolación bicúbica permiten inferir ciertos aspectos de la escena a partir de las siluetas que se observan. Sin embargo, a medida que estas últimas disminuyen en tamaño o se encuentran más alejadas de la cámara, la capacidad de distinguir o inferir información también disminuye. En el caso de las redes, los detalles generados permiten tener de forma más clara el contexto de la escena y de sus características.

De las pruebas realizadas para distintas configuraciones de resolución se puede decir que la mejor reconstrucción, en cuanto a forma y color, se obtiene a medida que se aumenta la

resolución, lo cual se debe a que en resoluciones menores aumenta la pérdida de información en los bordes por la subdivisión de la entrada. Para abordar este problema se propone experimentar con divisiones de mayor tamaño para la imagen de entrada. Esto permitiría superponer las imágenes adyacentes de menor tamaño para luego ponderar su información en los bordes. Siguiendo esta idea, también se puede seccionar la imagen de entrada de otra forma o utilizar otro método para ponderar la información.

Ante cada configuración de resolución de la red neuronal FRVSR, los resultados de PSNR y SSIM son superados por la respectiva red neuronal TecoGAN con la misma resolución, como se esperaba desde un principio a partir del escrito que propone originalmente la red TecoGAN. En cuanto a la aplicación de reconocimiento de objetos mediante YOLO, en los ejemplos mostrados la interpolación bicúbica no supera el umbral del 60 %, superada nuevamente por la mayoría de las redes neuronales, alcanzando en los ejemplos certeza certeza de . En general los resultados de las redes neuronales superan a la interpolación bicúbica, sin embargo, los resultados entre las mismas redes no permiten concluir que una resolución en particular permita reconstruir de mejor manera las características intrínsecas de los objetos en escena.

En general, la incorporación de YOLO como métrica, permitió demostrar el potencial de las redes neuronales de súper-resolución para asistir a sistemas cuyos resultados dependen de la calidad de imagen. Es necesario mencionar que las características intrínsecas que utiliza YOLO para clasificar son desconocidas para FRVSR y TecoGAN. Esto quiere decir que si se llega a utilizar la información de YOLO durante el entrenamiento de la red de súper-resolución, de forma similar a la del discriminador en la arquitectura GAN, la clasificación realizada por YOLO de las imágenes generadas obtendría mejores resultados, en cuanto a lo que serían los valores de certeza para los objetos correctamente identificados. De forma similar, las redes de súper-resolución podrían mejorar los resultados para una aplicación específica al incorporar la información del sistema utilizado por la misma aplicación en el proceso de entrenamiento.

Finalmente, cabe mencionar la ventaja de utilizar redes neuronales convolucionales sobre los métodos clásicos o de aprendizaje de máquinas. Esto se debe a que las redes neuronales presentan la ventaja de no requerir un análisis profundo para determinar qué características de las imágenes son las que son relevantes en la reconstrucción de detalles.

5.2. Trabajo futuro

A pesar de que en este trabajo de memoria de titulación se cubrieron casos de interés relevantes, aún falta revisar otros aspectos que son de ayuda para determinar, a partir de los requisitos de la aplicación que se tenga como objetivo, la mejor configuración de las redes neuronales presentadas. En primera instancia, se puede mencionar la búsqueda de métricas adicionales que sirvan para cuantificar la calidad de las imágenes por las redes neuronales, para lo cual se sugiere utilizar otros sistemas similares a YOLO, en cuanto a lo que serian sistemas de reconocimiento de objetos, reconocimiento facial u otros, teniendo en cuenta el propósito final de las redes. Otro aspecto de interés es el análisis de imágenes de un solo canal, ya que algunos sensores sólo permiten obtener imágenes en la escala de grises. Se ha mostrado en algunos casos que la pérdida de la información del color permite obtener mejores resultados que al contar con esta [3].

También hace falta realizar pruebas más exhaustivas con los sistemas en los cuáles funcionan la redes neuronales. Desde el punto de vista del software, se puede ver otras alternativas de framework, considerando que cada framework está diseñado según su propósito, algunos permiten la computación en paralelo, poseen modelos pre-entrenados, etc. En cuanto al hardware, las características de la GPU influyen directamente en la velocidad de procesamiento, por lo que falta hacer una comparación de los tiempos de inferencia entre diferentes GPU, sobre todo entre aquellas diseñadas para alta potencia y aquellas para mantener eficiencia energética.

Finalmente, se debe considerar que las redes neuronales progresan continuamente, y por lo tanto, se debe realizar la exploración de otros métodos que se encuentren el estado del arte para la súper-resolución aplicada a video. Cabe mencionar que la dificultad para implementar una cierta red neuronal va de la mano con la documentación de esta. Otra alternativa sería investigar la factibilidad de incorporar la información de redes neuronales para el reconocimiento de objetos, o para la aplicación acorde, mediante la arquitectura GAN, en donde esta red pasaría a ser paralela al discriminador, y así, mejorar la información de la imagen producida por el generador durante el entrenamiento.

Bibliografía

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283, 2016.

[2] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

[3] Hieu Minh Bui, Margaret Lech, Eva Cheng, Katrina Neville, and Ian S Burnett. Using grayscale images for object recognition with convolutional-recursive neural network. In *2016 IEEE Sixth International Conference on Communications and Electronics (ICCE)*, pages 321–325. IEEE, 2016.

[4] Jianrui Cai, Shuhang Gu, Radu Timofte, and Lei Zhang. Ntire 2019 challenge on real image super-resolution: Methods and results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[5] Hong Chang, Dit-Yan Yeung, and Yimin Xiong. Super-resolution through neighbor embedding. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 1, pages I–I. IEEE, 2004.

[6] Subhasis Chaudhuri. *Super-resolution imaging*, volume 632. Springer Science & Business Media, 2001.

[7] Yves Chauvin and David E Rumelhart. *Backpropagation: theory, architectures, and applications*. Psychology press, 1995.

[8] Tianqi Chen, Mu Li, Yutian Li, Min Lin, Naiyan Wang, Minjie Wang, Tianjun Xiao, Bing Xu, Chiyuan Zhang, and Zheng Zhang. Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems. *arXiv preprint arXiv:1512.01274*, 2015.

[9] Francois Chollet et al. Keras, 2015.

[10] Mengyu Chu, You Xie, Laura Leal-Taixé, and Nils Thuerey. Temporally coherent gans for video super-resolution (tecogan). *arXiv preprint arXiv:1811.09393*, 2018.

[11] Li Deng. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[12] Raanan Fattal. Image upsampling via imposed edge statistics. In *ACM SIGGRAPH 2007 papers*, pages 95–es. 2007.

[13] William T Freeman, Thouis R Jones, and Egon C Pasztor. Example-based super-resolution. *IEEE Computer graphics and Applications*, 22(2):56–65, 2002.

[14] Daniel Glasner, Shai Bagon, and Michal Irani. Super-resolution from a single image. In *2009 IEEE 12th international conference on computer vision*, pages 349–356. IEEE, 2009.

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[16] Tomomasa Gotoh and Masatoshi Okutomi. Direct super-resolution and registration using raw cfa images. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 2, pages II–II. IEEE, 2004.

[17] Yoav HaCohen, Raanan Fattal, and Dani Lischinski. Image upsampling via texture hallucination. In *2010 IEEE International Conference on Computational Photography (ICCP)*, pages 1–8. IEEE, 2010.

[18] Michal Irani and Shmuel Peleg. Improving resolution by image registration. *CVGIP: Graphical models and image processing*, 53(3):231–239, 1991.

[19] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678, 2014.

[20] Armin Kappeler, Seunghwan Yoo, Qiqin Dai, and Aggelos K Katsaggelos. Video super-resolution with convolutional neural networks. *IEEE Transactions on Computational Imaging*, 2(2):109–122, 2016.

[21] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[22] Kwang In Kim and Younghee Kwon. Single-image super-resolution using sparse regression and natural image prior. *IEEE transactions on pattern analysis and machine intelligence*, 32(6):1127–1133, 2010.

[23] Johannes Kopf and Dani Lischinski. Depixelizing pixel art. In *ACM SIGGRAPH 2011 papers*, pages 1–8. 2011.

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[25] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017.

[26] Feng Li, Yunming Ye, Zhaoyang Tian, and Xiaofeng Zhang. Cpu versus gpu: which can perform matrix computation faster—performance comparison for basic linear algebra subprograms. *Neural Computing and Applications*, 31(8):4353–4365, 2019.

[27] Zhongming Li, Kyle Aleshire, Masaru Kuno, and Gregory V Hartland. Super-resolution far-field infrared imaging by photothermal heterodyne imaging. *The Journal of Physical Chemistry B*, 121(37):8838–8846, 2017.

[28] Ding Liu, Zhaowen Wang, Yuchen Fan, Xianming Liu, Zhangyang Wang, Shiyu Chang, and Thomas Huang. Robust video super-resolution with learned temporal dynamics. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2507–2515, 2017.

[29] Xiang Ma, Junping Zhang, and Chun Qi. An example-based two-step face hallucination method through coefficient learning. In *International Conference Image Analysis and Recognition*, pages 471–480. Springer, 2009.

[30] Renaud Morin, Adrian Basarab, and Denis Kouamé. Alternating direction method of multipliers framework for super-resolution in ultrasound imaging. In *2012 9th IEEE International Symposium on Biomedical Imaging (ISBI)*, pages 1595–1598. IEEE, 2012.

[31] Seungjun Nah, Radu Timofte, Shuhang Gu, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, and Kyoung Mu Lee. Ntire 2019 challenge on video super-resolution: Methods and results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[32] Kien Nguyen, Sridha Sridharan, Simon Denman, and Clinton Fookes. Feature-domain super-resolution framework for gabor-based face and iris recognition. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2642–2649. IEEE, 2012.

[33] Karl S Ni and Truong Q Nguyen. Image superresolution using support vector regression. *IEEE Transactions on Image Processing*, 16(6):1596–1610, 2007.

[34] Sung Cheol Park, Min Kyu Park, and Moon Gi Kang. Super-resolution image reconstruction: a technical overview. *IEEE signal processing magazine*, 20(3):21–36, 2003.

[35] Sung Won Park and Marios Savvides. Breaking the limitation of manifold analysis for super-resolution of facial images. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, volume 1, pages I–573. IEEE, 2007.

[36] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, et al. Deep face recognition. In *bmvc*, volume 1, page 6, 2015.

[37] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[38] Matan Protter, Michael Elad, Hiroyuki Takeda, and Peyman Milanfar. Generalizing the nonlocal-means to super-resolution reconstruction. *IEEE Transactions on image processing*, 18(1):36–51, 2008.

[39] Yuankai Qi, Shengping Zhang, Lei Qin, Hongxun Yao, Qingming Huang, Jongwoo Lim, and Ming-Hsuan Yang. Hedged deep tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4303–4311, 2016.

[40] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[41] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[42] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[43] Mehdi SM Sajjadi, Raviteja Vemulapalli, and Matthew Brown. Frame-recurrent video super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6626–6634, 2018.

[44] Hilario Seibel, Siome Goldenstein, and Anderson Rocha. Eyes on the target: Super-resolution and license-plate recognition in low-quality surveillance videos. *IEEE access*, 5:20020–20035, 2017.

[45] Qi Shan, Zhaorong Li, Jiaya Jia, and Chi-Keung Tang. Fast image/video upsampling. *ACM Transactions on Graphics (TOG)*, 27(5):1–7, 2008.

[46] Kevin Su, Qi Tian, Qing Xue, Nicu Sebe, and Jingsheng Ma. Neighborhood issue in single-frame image super-resolution. In *2005 IEEE international conference on multimedia and expo*, pages 4–pp. IEEE, 2005.

[47] Jian Sun, Zongben Xu, and Heung-Yeung Shum. Image super-resolution using gradient profile prior. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[48] Libin Sun and James Hays. Super-resolution from internet-scale scene matching. In *2012 IEEE International Conference on Computational Photography (ICCP)*, pages 1–12. IEEE, 2012.

[49] Marshall F Tappen and Ce Liu. A bayesian approach to alignment-based image hallucination. In *European conference on computer vision*, pages 236–249. Springer, 2012.

[50] Eclipse Deeplearning4j Development Team. ND4J: Fast, Scientific and Numerical Computing for the JVM. 2016.

[51] Igor V Tetko, David J Livingstone, and Alexander I Luik. Neural network studies. 1. comparison of overfitting and overtraining. *Journal of chemical information and computer sciences*, 35(5):826–833, 1995.

[52] Qiang Wang, Xiaoou Tang, and Harry Shum. Patch based blind image super resolution. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 1, pages 709–716. IEEE, 2005.

[53] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[54] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018.

[55] Yifan Wang, Federico Perazzi, Brian McWilliams, Alexander Sorkine-Hornung, Olga Sorkine-Hornung, and Christopher Schroers. A fully progressive approach to single-image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 864–873, 2018.

[56] Bingzhe Wu, Haodong Duan, Zhichao Liu, and Guangyu Sun. Srpgan: Perceptual generative adversarial network for single image super resolution. *arXiv preprint arXiv:1712.05927*, 2017.

[57] Jianchao Yang, Zhe Lin, and Scott Cohen. Fast image super-resolution based on in-place example regression. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1059–1066, 2013.

[58] Jianchao Yang, John Wright, Thomas S Huang, and Yi Ma. Image super-resolution via sparse representation. *IEEE transactions on image processing*, 19(11):2861–2873, 2010.

[59] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li. Learning face representation from scratch. *arXiv preprint arXiv:1411.7923*, 2014.

[60] Zhi Yuan, Jiong Wu, Sei-ichiro Kamata, Alireza Ahrary, and Peimin Yan. Fingerprint image enhancement by super resolution with early stopping. In *2009 IEEE International Conference on Intelligent Computing and Intelligent Systems*, volume 4, pages 527–531. IEEE, 2009.

[61] J. I. González Yáñez. Super resolution in face recognition. 2019.

[62] Shuqun Zhang. Application of super-resolution image reconstruction to digital holography. *EURASIP Journal on Advances in Signal Processing*, 2006(1):090358, 2006.

[63] Wilman WW Zou and Pong C Yuen. Very low resolution face recognition problem. *IEEE Transactions on image processing*, 21(1):327–340, 2011.