elektron ámbar

Original Article

Revista elektron, Vol. 7, No. 1, pp. 7-18 (2023)
https://doi.org/10.37537/rev.elektron.7.1.180.2023

Red adversaria generativa aplicada a la eliminación de ruido y artefactos en sinogramas de tomografía optoacústica

Generative Adversarial Network Applied to the Elimination of Noise and Artifacts in Optoacoustic Tomography Sinograms

Delfina Montilla*[1], Martín G. González*†, Leonardo Rey Vega*†

*Facultad de Ingeniería, Universidad de Buenos Aires
Paseo Colón 850, C1063ACV, Buenos Aires, Argentina
†Consejo Nacional de Investigaciones Científicas y Técnicas, (CONICET)
Godoy Cruz 2290, C1425FQB, Buenos Aires, Argentina
[1]dmontilla@fi.uba.ar

Recibido: 30/05/23; Aceptado: 13/06/23

***Resumen*— El objetivo de este trabajo es el estudio de un método de pre-procesamiento de los datos medidos por un tomógrafo optoacústico bidimensional para reducir o eliminar los artefactos introducidos por la escasa cantidad de detectores en el sistema experimental y el acotado ancho de banda de estos. Para esta tarea, se utilizó una red neuronal profunda generativa adversaria y se comparó su rendimiento con una red neuronal de referencia U-Net. En la mayoría de los casos de testeo realizados, se encontró una leve mejora aplicando la red propuesta al medir la correlación de Pearson y la relación señal a ruido piso entre la imagen reconstruida producto de los datos procesados por el modelo y la imagen de alta resolución de referencia.**

Palabras clave: tomografía optoacústica; aprendizaje profundo; GAN.

***Abstract*— The goal of this work is to study a preprocessing method for the data measured by a two-dimensional optoacoustic tomograph in order to reduce or eliminate artifacts introduced by the low number of detectors in the experimental setup and their limited bandwidth. A generative adversarial deep neural network was used to accomplish this task and its performance was compared with a reference U-Net neural network. In most of the test cases carried out, a slight improvement was found by applying the proposed network when measuring the Pearson correlation and the peak signal noise ratio between the reconstructed image product of the data processed by the model and the high-resolution reference image.**

Keywords: optoacustic tomography; machine learning; GAN.

I. Introducción

Un enfoque muy prometedor para la obtención de imágenes biológicas es la tomografía optoacústica (TOA), también conocida como tomografía fotoacústica o PAT por sus siglas en inglés [1]–[3]. Es una técnica no invasiva, segura y de elevada resolución que puede utilizarse para una variedad de aplicaciones en la investigación clínica y preclínica [4], [5], incluyendo la detección de tumores [6], [7]. Combina la excitación óptica con la detección ultrasónica, lo que ofrece varias ventajas para la obtención de imágenes biológicas [8], como permitir la diferenciación de estructuras específicas en el tejido, dependiendo de la longitud de onda utilizada. La luz incidente sólo necesita ser absorbida por el objeto que se pretende estudiar para generar una señal acústica que pueda ser detectada de manera confiable en lo profundo del tejido. Otra ventaja es que, comparado con la microscopía óptica, proporciona una penetración mucho mayor con una resolución espacial escalable al ser aplicada a tejido biológico [9], [10]. Además, es una técnica de imagen que no se basa en el uso de la radiación ionizante, como la tomografía computada (TC), o de la fluorescencia; sino en la relajación no radiativa de las moléculas. Por lo tanto, sirve para visualizar cualquier molécula siempre y cuando se produzca esta relajación no radiativa. Incluso sería posible el desarrollo de un equipo portátil de TOA, a diferencia del caso de la TC donde existen limitaciones de seguridad por la utilización de radiación ionizante, o la resonancia magnética donde se requiere de superconductores para la generación de los campos magnéticos.

El mayor desafío en lo que concierne a la TOA es la adquisición de datos a velocidad elevada con una matriz de transductores ultrasónicos de elementos múltiples. Si bien los sistemas de adquisición de datos multicanal ($\geq$ 128 canales) están disponibles comercialmente, estos son todavía costosos [11]. La calidad de la imagen OA reconstruida depende en gran medida de la cantidad de datos disponibles, que a su vez es proporcional al número de detectores empleados. En caso de datos limitados (debido a la menor cantidad de detectores causado por restricciones de costo/instrumentación), las imágenes reconstruidas sufren de artefactos y, a menudo, son ruidosas. Asimismo, otra desventaja para adquirir grandes cantidades de datos es un mayor tiempo de escaneo de la muestra bajo estudio [12],

[13]. Además, los detectores utilizados para las mediciones tomográficas tienen un ancho de banda limitado y sólo pueden cubrir un rango de apertura, por lo cual es posible que no cubran todo el objeto, resultando en datos limitados en cantidad y calidad [14]. Por otro lado, el sinograma es la representación gráfica de las señales acústicas en función del tiempo medidas por los detectores de ultrasonido (señales OA). Contiene la información sobre la distribución espacial y la amplitud de las señales OA capturadas por los detectores durante el escaneo.

En este trabajo se estudia el uso de una red adversaria generativa (GAN) [15] para la super-resolución (aumento de la calidad de reconstrucción con un número limitado de datos), la mejora del ancho de banda, y la remoción de artefactos y ruido en señales acústicas provenientes de mediciones de un sistema para TOA bidimensional. Se tiene como antecedente el trabajo de investigación [16], donde se propuso el primer uso de una red neuronal profunda aplicada exclusivamente al pre-procesamiento de las señales OA medidas, en vez de hacerlo sobre la imagen reconstruida. Es interesante destacar que uno de los atributos más importantes de un esquema basado en una red neuronal profunda es la velocidad con la que pueden procesar los datos de entrada. Para redes pequeñas, esto puede ser útil en entornos donde se requiere la obtención de imágenes dinámicas o en tiempo real [17]. Otra motivación adicional para usar modelos de aprendizaje profundo en la reconstrucción de imágenes OA, es la disponibilidad de herramientas como TensorFlow [18] y PyTorch [19], que hacen que el empleo de estos nuevos métodos presente una curva de aprendizaje suave al proveer una documentación completa y tutoriales para los nuevos usuarios. El código correspondiente a este trabajo se encuentra disponible en un repositorio de GitHub, https://github.com/delfimontilla/PATGAN.

II. Métodos

A. Generación de los datos de entrenamiento, validación y testeo

Los componentes principales del sistema experimental TOA incluyen un láser de pulsos cortos para la generación eficiente de señales de banda ancha (BW), un transductor ultrasónico de banda ancha o una matriz de transductores para la detección de señales, un sistema de adquisición de datos para amplificación y digitalización de señales y una computadora para la sincronización del sistema, recolección de datos y reconstrucción de las imágenes [8]. El modelo directo para la generación de imágenes de TOA se expresa mediante la siguiente ecuación:

$$Ax = b \tag{1}$$

donde A es la matriz del sistema que contiene las respuestas al impulso de todos los píxeles en la región de la imagen, x es el vector que representa el aumento de presión inicial y b es el sinograma [20]. En este contexto, las respuestas al impulso representan el comportamiento de los píxeles individuales dentro de la región correspondiente a la imagen cuando se aplica un pulso. Cada píxel tiene su propia respuesta, la cual captura cómo reacciona este a la señal, incluyendo factores como la absorción, la dispersión y otras propiedades físicas. El número de columnas en la matriz (A) es igual al número de píxeles en el dominio de imágenes; y el número de filas es equivalente a la cantidad de píxeles en el dominio del sinograma. En consecuencia, construir la matriz del sistema es una tarea costosa desde el punto de vista computacional cuando se desea una resolución elevada.

Existen varios algoritmos para obtener una imagen a partir del sinograma; se pueden clasificar como métodos analíticos o métodos iterativos basados en modelos. Dentro del primer grupo mencionado, uno de los métodos matemáticamente más simples es el denominado retroproyección lineal (LBP, por sus siglas en inglés). En este enfoque, la reconstrucción aproximada de la imagen x_{bp} se puede obtener a través de la siguiente ecuación:

$$x_{bp} = A_T b \tag{2}$$

donde A_T representa la transpuesta de la matriz que modela el sistema experimental y b es el sinograma en forma vectorial unidimensional [21]. Este método fue elegido para este trabajo ya que tiene bajo tiempo de procesamiento (sin tener en cuenta el tiempo que conlleva generar A_T) y no posee ningún parámetro de ajuste. Aunque es posible utilizar esquemas basados en modelos para lograr una mayor calidad de imagen se decidió utilizar LBP para reforzar que la mejora en la calidad de la imagen reconstruida se debe exclusivamente a la mejora en los datos del sinograma [16].

En este trabajo, los sinogramas se obtuvieron a partir de una base de datos de 59 mil fantomas mamarios computacionales generados a partir del procesamiento de resonancias magnéticas de alta resolución adquiridas de pacientes, en las cuales se clasificó cada píxel según el tipo de material al que correspondía (aire, tejido adiposo, tejido glandular y tejido cutáneo) [22]. En primer lugar, de este conjunto de datos se seleccionó cuidadosamente un subconjunto de 2126 imágenes de forma tal de evitar redundancia y sesgos innecesarios. A su vez fue dividido de forma azarosa en tres grupos: 70 % para el entrenamiento (1500 imágenes), 19% para la validación (400 imágenes) y 11% para el testeo (226 imágenes). Los primeros dos grupos fueron utilizados en la etapa de entrenamiento de las redes neuronales y el último grupo se reservó para testear el modelo resultante. En segundo lugar, se generaron los sinogramas multiplicando los fantomas mamarios, en forma de vector unidimensional, por la matriz del sistema experimental. Utilizando Python, se construyó la matriz del sistema con los mismos parámetros experimentales que en [16]. Como se puede ver en la Fig. 1, se empleó una cuadrícula computacional de $n \times n$ píxeles. Para la generación de datos, se utilizó una grilla de alta dimensión de $nx_{gen} \times nx_{gen}$ píxeles; en cambio, para la reconstrucción de los datos, la grilla era de $nx_{recon} \times nx_{recon}$ píxeles. Se colocaron transductores en el límite del tejido de manera circularmente equidistante en un radio dsa; estos muestrearon observaciones con una frecuencia F. En total, se tomaron Nt muestras temporales. Se supuso que la velocidad del sonido en el medio, el tejido bajo investigación, era uniforme sin absorción ni dispersión e igual a 1500 m/s.

Para la generación de sinogramas de alta calidad se simularon Ns_{hq} detectores de ultrasonido sin limitación de

ancho de banda, resultando en sinogramas de dimensiones $Ns_{hq} \times Nt$. Mientras que para la generación de sinogramas de baja calidad, se colocaron la mitad detectores Ns_{lq} con ancho de banda limitado, se agregó ruido gaussiano con una relación señal-ruido de entre 10 y 70dB y se interpoló de $Ns_{lq} \times Nt$ a $Ns_{hq} \times Nt$ utilizando el método de vecinos cercanos (*nearest neighbour*). Estos sinogramas de menor resolución, ancho de banda limitado y con ruido serán procesados por los modelos de aprendizaje profundo con el objetivo de que se asemejen a los sinogramas de alta calidad anteriormente mencionados. En el caso de la construcción de la matriz para la reconstrucción de los sinogramas, se simularon Ns_{hq} detectores de ultrasonido sin limitación de ancho de banda.

A continuación, se especificarán los valores de los parámetros para generar la matriz del sistema experimental. Primero, la grilla computacional es de 501×501 píxeles con $0,1$ mm/píxel, lo que la convierte en un tamaño de cuadrícula de imágenes de 50 mm por 50 mm. El largo del lado de la grilla de generación de datos, nx_{gen}, es de 401 píxeles; y el lado de un píxel cuadrado es de 50μm. En cambio, para la reconstrucción de los datos, el largo del lado de la grilla nx_{recon} es igual a 201 píxeles, donde el valor del largo del píxel es 100μm. La cantidad de sensores para la generación de sinogramas de baja calidad, Ns_{hq}, es de 128; mientras que para los sinogramas de alta calidad y la matriz de reconstrucción con LBP, $Ns_{lq} = 256$. Los sensores se encuentran a $22,5$ mm del centro de la grilla computacional, la mencionada distancia dsa. La cantidad de muestras temporales, Nt, fueron 512; y la frecuencia de muestreo, llamada F, era de 20 Mhz.

Para la generación de sinograma se realizaron los siguientes pasos:

- Re-escalar el fantoma a las dimensiones de la grilla de la muestra $nx_{gen} \times nx_{gen}$
- Convertirlo en un vector unidimensional
- Generar la matriz A del sistema con los valores indicados para la generación de dato, eligiendo la cantidad de detectores requeridos dependiendo del tipo de sinograma deseado
- Multiplicar A por el fantoma vectorizado
- Convertir el sinograma unidimensional resultante en la matriz correspondiente
  - Para un sinograma de alta calidad $Ns_{hq} \times Nt$
  - Para un sinograma de baja calidad $Ns_{lq} \times Nt$
- Para el caso de un sinograma de baja calidad:
  - Agregar filtro pasabanda
  - Agregar ruido gaussiano con una relación señal-ruido de entre 10 y 70 dB
  - Interpolar a $Ns_{hq} \times Nt$

Para la reconstrucción de las imágenes se realizaron los siguientes pasos:

- Convertir el sinograma de dimensiones $Ns_{hq} \times Nt$ en un vector unidimensional
- Generar la matriz A del sistema con los valores indicados para la reconstrucción de datos
- Trasponer A, obteniendo A_T
- Multiplicar A_T por el sinograma vectorizado
- Convertir la imagen vectorizada resultante en la matriz correspondiente de dimensiones $nx_{recon} \times nx_{recon}$

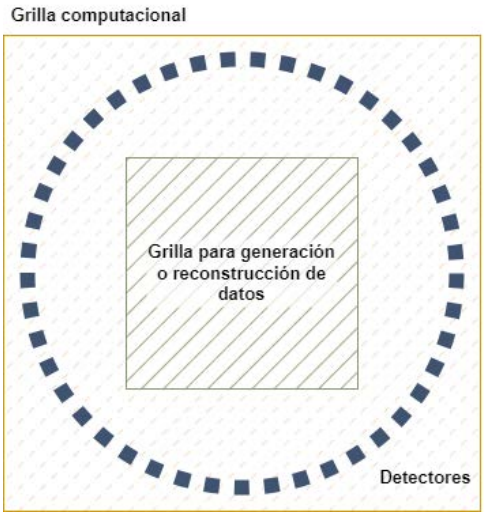

Fig. 1: Representación gráfica de la configuración para la recopilación de datos.

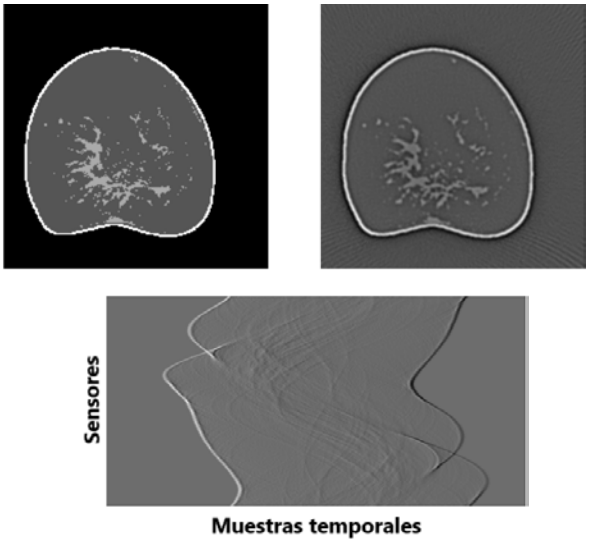

Fig. 2: Ejemplo de un fantoma mamario original (arriba a la izquierda) y la reconstrucción utilizando LBP (arriba a la derecha) del sinograma de alta calidad (abajo).

En último lugar, para la preparación de los datos para el entrenamiento se generaron 105 parches de dimensiones 64×64, utilizando un paso de 32 muestras, para todos los sinogramas de los dos grupos. Este procedimiento, al igual que en [16], se realiza para que la red pueda aprender a corregir detalles locales de los sinogramas.

En la Fig. 2 se puede ver un ejemplo de un fantoma mamario y su reconstrucción utilizando este método partiendo del sinograma de alta calidad. Este tipo de reconstrucción es la imagen de mejor calidad que podrá ser obtenida con LBP y por lo tanto será utilizada como referencia para la comparación de las imágenes obtenidas con los modelos basados en redes neuronales profundas. Con el fin de testear diferentes aspectos del modelo resultante, se utilizaron cuatro imágenes distintas que no formaron parte de los datos de entrenamiento [23] y se muestran en la primera columna de la Fig. 3. La imagen con la letras PAT ayuda a determinar la

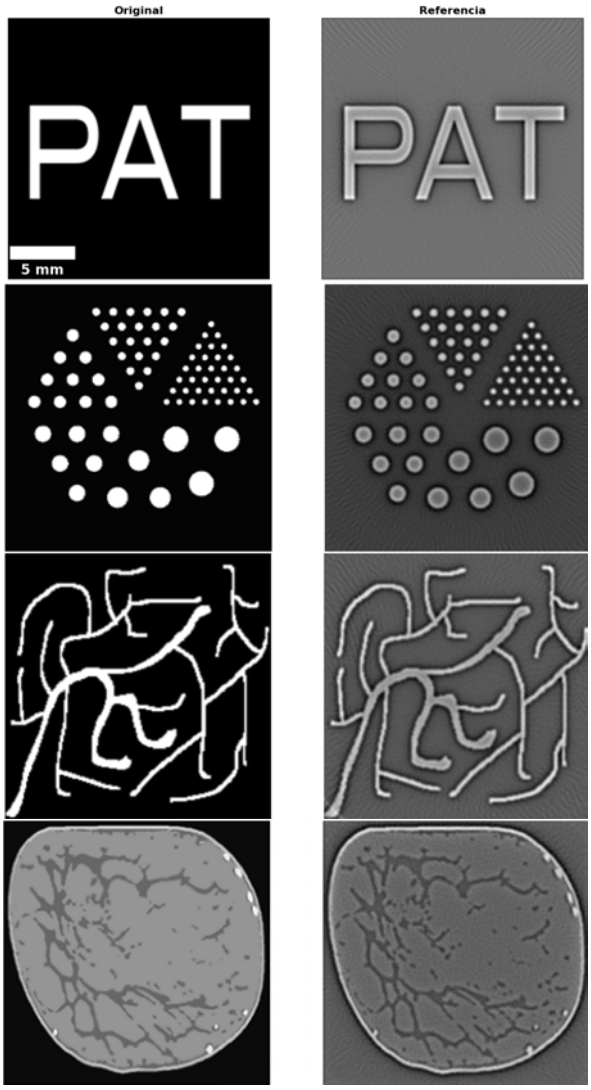

Fig. 3: Imágenes originales (izquierda) y la reconstrucción utilizando LBP de los sinogramas de alta calidad (derecha).

eficacia para recuperar objetos nítidos. La segunda imagen se denomina Derenzo y está compuesta por grupos de objetos circulares con diferentes radios que ayudan a evaluar el poder de reconstrucción de objeto pequeños y grandes. La tercera imagen que se asemeja a vasos sanguíneos se utiliza para analizar el poder de reconstrucción de estructuras amorfas complicadas. Estas tres imágenes mencionadas son binarias, con '1' para el objeto de interés y '0' para el fondo. Por otro lado, la cuarta imagen corresponde a un fantoma mamario y sirve para testear el caso de una imagen OA compleja que presenta un contraste variable y ruido. A continuación, se generaron los sinogramas de alta calidad de estas imágenes utilizando el mismo procedimiento explicado anteriormente. En la segunda columna de la Fig. 3 se presentan sus reconstrucciones usando LBP. Por último, se generaron los sinogramas de baja calidad; para el caso de estas cuatro imágenes, el ruido gaussiano agregado tenía una relación señal-ruido de 60 dB.

B. U-Net: red neuronal de referencia

1) Arquitectura: La U-Net es una red neuronal convolucional cuya estructura es simétrica y tiene forma de "U". La

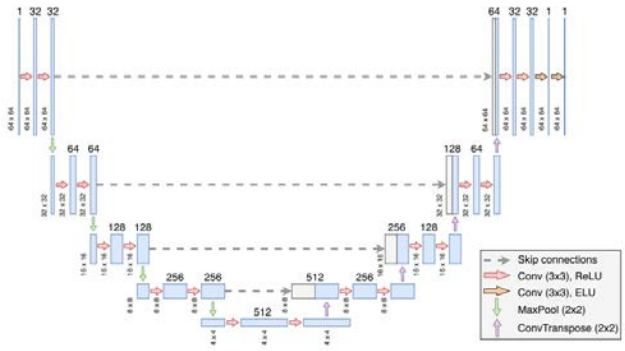

Fig. 4: Estructura de la U-Net implementada por [16]. Cada rectángulo azul corresponde a un mapa de características multi-canal (*multi-channel feature map*) donde en la parte superior se indica el número de canales, y en el borde inferior izquierdo, el ancho y alto de las imágenes.

primera parte de la red, denominada "ruta de contracción", consiste en una sucesión de capas convolucionales, de activación y de agrupación. Mientras que la segunda parte de la red, denominada "ruta de expansión", es una secuencia de capas convolucionales, convolucionales transpuestas y de activación. Asimismo, esta estructura cuenta con conexiones residuales entre las dos "rutas" para mantener la estructura general de la entrada en la salida cumpliendo además funciones de estabilidad durante el entrenamiento, minimizando los efectos del gradiente desvanesciente [24]. La entrada y salida de la U-Net tienen dimensiones idénticas debido a la simetría de las operaciones. La arquitectura original fue presentada por Ronneberger et al. [25] en 2015 y Awasthi et al. [16] eligieron esta red neuronal para llevar a cabo la tarea de superresolución, remoción de ruido y mejora de ancho de banda en sinogramas obtenidos de una sistema para TOA. En la Fig. 4 se ilustra la estructura implementada en [16].

2) Estrategia de entrenamiento: El entrenamiento de un modelo implica determinar valores óptimos para todos los pesos. En el aprendizaje supervisado, un algoritmo construye un modelo examinando muchos ejemplos e intentando encontrar aquellos pesos que minimicen una función de error o pérdida. Este proceso se denomina minimización empírica del riesgo. Entonces, se puede pensar en la función de pérdida como la forma de evaluar la calidad de la predicción realizada por el modelo. Si la predicción del modelo es perfecta, el valor de la función de pérdida es cero. El objetivo de entrenar un modelo es encontrar un conjunto de pesos que tengan una pérdida promedio baja para todos los ejemplos de prueba [26].

Se reprodujo la misma estrategia de entrenamiento que la presentada en [16] donde los hiper-parámetros ya se encuentran optimizados. La entrada al modelo corresponde a los parches de 64×64 píxeles de baja calidad que, luego de ser procesados por la red con 32 filtros de entrada, se comparan con los datos *target* (o sea, los parches de 64×64 píxeles de los sinogramas de alta calidad). La función de pérdida elegida fue la raíz del error cuadrático medio escalada. Este escalado se implementa para minimizar el problema del desvanecimiento del gradiente al aplicar *backpropagation* en el entrenamiento con sinogramas que contienen valores

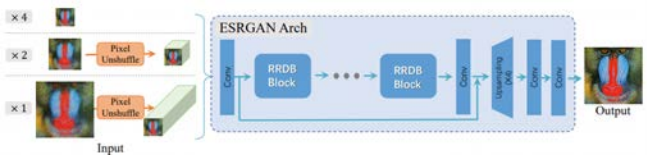

Fig. 5: Generador del modelo Real-ESRGAN [27]

del orden de magnitud alrededor de $1 \cdot 10^{-4}$. En [16] se determinó empíricamente que el factor de multiplicación para esta aplicación es 10000:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \|\hat{x}_i - \phi(x_i)\|^2 \times \tau \tag{3}$$

donde $\phi(x_i)$ es la salida predicha por la red; $\hat{x}$, el parche *target*; y τ, el factor multiplicador. Por otro lado, se utilizó el optimizador de Adam para entrenar la red con una tasa de aprendizaje de $1 \cdot 10^{-3}$. y una tasa de decaimiento de $0,98$ con un tamaño de paso de 2. Por último, el número de *epochs* de entrenamiento y el *batch size* o tamaño de grupo de entrenamiento fueron de 250 y 128, respectivamente.

3) Estrategia de testeo: Los sinogramas de testeo de baja calidad, 256×512 píxeles, se rellenaron (*padding*) utilizando el modo reflejo con el objetivo de llevar el tamaño a 512×512 píxeles. Luego, fueron introducidos en la U-Net entrenada para obtener una versión mejorada. Los sinogramas devueltos por la red se reconstruyeron usando el método LBP. Finalmente, se realizó el mismo procedimiento de testeo con los sinogramas de baja calidad obtenidos a partir de las imágenes de la Fig. 3.

C. Real-Enhanced Super Resolution Generative Adversarial Network - Real-ESRGAN

1) Arquitectura: La Real-ESRGAN es una red neuronal puramente convolucional para realizar super-resolución en imágenes [27]. La arquitectura de esta red fue diseñada para lograr un buen equilibrio entre la mejora de detalles locales y la eliminación de artefactos. En este trabajo se estudia su aplicación en el pre-procesamiento de sinogramas OA de baja calidad. El generador es una red neuronal profunda denominada ESRGAN [28] que está compuesta por capas convolucionales, 16 bloques convolucionales residuales (RRDB, por sus siglas en inglés) y capas de sobremuestreo. En la Fig. 5 se muestra su arquitectura.

En particular, un bloque RRDB (Fig. 6) consiste en tres conjuntos idénticos sucesivos de cinco capas convolucionales intercaladas con capas de activación *Leaky ReLU*:

$$\text{LeakyReLU}(x) = \begin{cases} x, & \text{si } x \geq 0 \\ \alpha x, & \text{si } x < 0 \end{cases}$$

donde α es igual a $0,2$. Las conexiones residuales son para prevenir inestabilidades en el entrenamiento. Asimismo, el escalado residual puede interpretarse como una herramienta para corregir una inicialización incorrecta, evitando así aumentar la magnitud de los valores de las señales de entrada [28]. Con la arquitectura Real-ESRGAN se puede a realizar super-resolución con un factor de escala de $\times 1$, $\times 2$ y $\times 4$. En este trabajo se eligió la opción $\times 2$. Para ese caso, los datos de entrada pasan por un proceso llamado *Pixel Unshuffle*

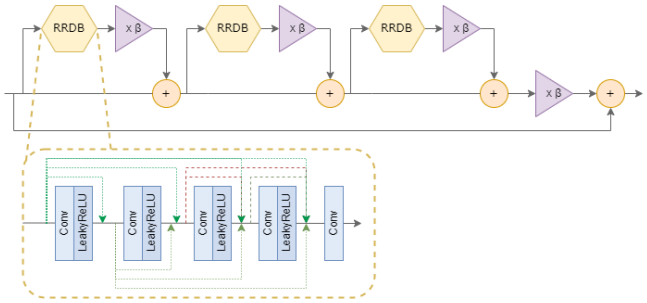

Fig. 6: RRDB (*Residual-in-Residual Dense Blocks*)

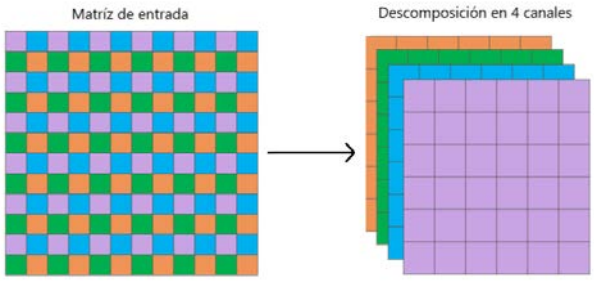

Fig. 7: Pixel Unshuffle

donde la matriz de entrada de un sólo canal se descompone en 4 canales, tal como se puede ver en la Fig. 7. El objetivo es reducir el tamaño espacial para que los cálculos realizados por la red se realicen en un espacio de resolución más chico, y así disminuir la utilización de la memoria de la GPU y el consumo de recursos computacionales.

Por otro lado, el discriminador de la Real-ESRGAN es una red neuronal convolucional U-Net, Fig. 8, pero a diferencia a la U-Net mencionada en la sección anterior, esta red utiliza capas de activación *Leaky ReLU* (con pendiente $0,2$) y normalización espectral [29] excepto la primera y última capa convolucional. La normalización espectral consiste en re-escalar los pesos de la siguiente forma:

$$\mathbf{W}_{SN} = \frac{\mathbf{W}}{\sigma(\mathbf{W})}, \quad \sigma(\mathbf{W}) = \max_{\mathbf{x}\mathbf{x}\neq 0} \frac{\|\mathbf{W}\mathbf{x}\|_2}{\|\mathbf{x}\|_2} \tag{4}$$

donde $\sigma(\mathbf{W})$ es el máximo valor singular de $\mathbf{W}$. Este discriminador en particular fue elegido para que haga foco en degradaciones locales y para estabilizar el entrenamiento [27]. Asimismo, la U-Net genera un valor para cada píxel que indica el nivel de *realismo* y así puede proporcionar información detallada por píxel al generador.

2) Estrategia de entrenamiento: Los datos de entrada de la red generadora fueron los sinogramas sin pre-procesamiento; y, a partir de ellos, el objetivo era estimar sinogramas de elevada calidad. En consecuencia, se entrenó

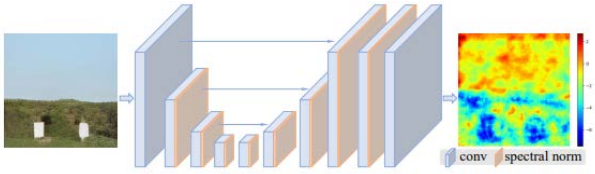

Fig. 8: Discriminador de la Real-ESRGAN: U-Net con normalización espectral [27]

la red Real-ESRGAN optimizando una combinación pesada entre la pérdida $\mathcal{L}_1$ o error absoluto medio, la pérdida perceptual [30] y la pérdida adversaria [15], [26], [31]. En primer lugar, el error absoluto medio se define como el promedio de las diferencias absolutas entre el valor real y el predicho:

$$\mathcal{L}_1 = \frac{\Sigma_{i=1}^{n} |y_i - gt_i|}{n} \tag{5}$$

donde y_i y gt_i son los valores predichos y los reales, correspondientemente, y n refiere a la cantidad total de valores. Mide la magnitud promedio de los errores en un conjunto de predicciones, sin considerar sus direcciones. $\mathcal{L}_1$ es más resistente a los valores atípicos o *outliers* en comparación con el error cuadrático medio. En segundo lugar, la pérdida perceptual analiza las diferencias entre las representaciones intermedias que son extraídas por las redes neuronales convolucionales previamente entrenadas y capturan características visuales importantes. De esta forma, se logra medir la similitud entre las características visuales extraídas del sinograma generado y del de referencia de forma más robusta que la pérdida $\mathcal{L}_1$. Esta función de pérdida está compuesta por dos términos que se suman: la pérdida de reconstrucción de características y la pérdida de reconstrucción de estilo. La primera transfiere conocimiento semántico a la Real-ESRGAN conservando el contenido y la estructura espacial general del sinograma; y se calcula como:

$$\mathcal{L}_p = \omega_p \sum_{k=1}^{n} w_k \cdot \mathcal{L}_c(f_k(x), f_k(gt)) \tag{6}$$

donde ω_p representa el peso de la pérdida de características, n es el número de capas de características de la red pre-entrenada utilizadas, w_k es el peso asignado a la k-ésima capa, f_k es la función de la característica de la k-ésima capa y x y gt son el sinograma de entrada y el de referencia respectivamente. $\mathcal{L}_c$ es una función de pérdida que mide la diferencia entre las características de entrada y de referencia (en este caso, la función elegida es la $\mathcal{L}_1$). La red pre-entrenada utilizada es la VGG19 [30]. Con respecto al otro término, la pérdida de estilo analiza las diferencias en color, textura y patrones comunes y se calcula como:

$$\mathcal{L}_s = \omega_s \sum_{k=1}^{n} w_k \cdot \mathcal{L}_c(\mathrm{Gram}(f_k(x)), \mathrm{Gram}(f_k(gt))) \tag{7}$$

donde ω_s representa el peso de la pérdida de estilo, n es el número de capas de características de VGG utilizadas, w_k es el peso asignado a la k-ésima capa, $\mathcal{L}_c$ es la función de pérdida que mide la diferencia entre las matrices Gram de las características de la k-ésima capa, $\mathrm{Gram}(f_k)$; y x y gt son el sinograma de entrada y el de referencia, respectivamente. La matriz de Gram informa sobre qué características tienden a activarse juntas y se define como

$$G_k^f(x)_{c,c^T} = \frac{1}{C_k H_k W_k} \sum_{h=1}^{H_k} \sum_{w=1}^{W_k} f_k(x)_{h,w,c} f_k(x)_{h,w,c^T} \tag{8}$$

donde C_k, H_k y W_k son las dimensiones de los canales, altura y ancho del mapa de características, respectivamente. Por último, la pérdida adversaria, que es específica de las redes GAN, mide la capacidad del generador para producir datos que sean indistinguibles de los datos reales, es decir, ayuda al generador a producir sinogramas con las características de los sinogramas de elevada calidad originales. El generador trata de maximizar la probabilidad de que el discriminador clasifique una muestra generada como real, mientras que el discriminador trata de minimizar la probabilidad de que clasifique una muestra generada como real. La ecuación de la pérdida adversaria usada en este trabajo se muestra a continuación [15]:

$$\mathcal{L}_{GAN}(G, D) = \mathbb{E}_{gt}\left[\log D(gt)\right] + \mathbb{E}_x\left[\log(1 - D\left(G(x)\right))\right] \tag{9}$$

la cual se debe mimimizar sobre G y maximizar sobre D, siendo G el generador, D el discriminador, $D(gt)$ es la estimación del discriminador de la probabilidad de que el dato de entrada de elevada resolución pertenezca a los datos *target*, y $D(G(x))$ es la estimación del discriminador de la probabilidad de que el dato de entrada de baja resolución que pasó por el generador pertenezca a los datos *target*. La función para estimar las probabilidades depende del tipo de GAN que se esté utilizando. En el código proporcionado en [27], se implementan varios tipos de GAN. Para el tipo *Vanilla GAN* [15], se utiliza entropía cruzada binaria (*BCE With Logits Loss*) [19]):

$$\mathcal{L}(x, k) = -\frac{1}{n} \sum_{i=1}^{n} \left[k_i \cdot \log \sigma(x_i) + (1 - k_i) \cdot \log(1 - \sigma(x_i))\right] \tag{10}$$

donde x es el sinograma de entrada de la red neuronal, k es el valor asignado según el tipo de sinograma que sea (baja o elevada calidad), n es el tamaño del *batch size*, y σ es la función sigmoida.

El entrenamiento de la red GAN se dividió en dos etapas: el pre-entrenamiento del generador y el entrenamiento conjunto del generador y discriminador. Se realiza un pre-entrenamiento del generador ya que se ha demostrado que ayuda a evitar mínimos locales no deseados para el generador y ayuda al discriminador a enfocarse más en las texturas en el siguiente entrenamiento en conjunto, debido a que recibe datos relativamente buenos de un generador pre-entrenado en lugar de datos más aleatorios [28]. En la siguiente etapa, durante el entrenamiento de G y D, el discriminador busca distinguir los sinogramas de elevada calidad de los sinogramas producidos por el generador, mientras que el generador optimiza sus sinogramas de salida para *engañar* al discriminador. Con respecto a la convergencia, a medida que el generador mejora a lo largo del entrenamiento, el rendimiento del discriminador empeora porque este no puede distinguir fácilmente la diferencia entre los sinogramas de elevada resolución originales y los producidos por el generador. Idealmente si la red generadora funcionase perfectamente, el discriminador tendría una precisión del 50%. Otro punto para considerar es que la retroalimentación del discriminador hacia el generador se vuelve menos significativa con el tiempo, lo que dificulta la convergencia de la GAN. Si la GAN continúa entrenando más allá del punto en que el discriminador está dando

valores completamente aleatorios en las pérdidas, entonces el generador se ve afectado y su propia calidad puede deteriorarse gravemente.

Se llevaron a cabo diversos entrenamientos, con variaciones en los hiperparámetros, que se resumen en la Tabla I. Sin embargo, debido a la limitación de tiempo y recursos computacionales, no se realizó una exploración exhaustiva de todas las posibilidades. Para el primer entrenamiento de la Real-ESRGAN, denominado de acá en adelante Real-ESRGAN M1, se utilizó de base la configuración de hiperparámetros descripta en [27]. El pre-entrenamiento se realiza durante un total de $1 \cdot 10^6$ iteraciones, con un optimizador Adam, cuya tasa de aprendizaje se estableció en $2 \cdot 10^{-4}$ con decaimiento de $0,5$ a las $3 \cdot 10^5$ iteraciones. En esta instancia, se utilizó únicamente la pérdida $\mathcal{L}_1$; de esta forma, según [27], el generador pre-entrenado se encuentra orientado a optimizar el valor pico de la relación señal-ruido (PSNR). La siguiente etapa fue el entrenamiento en conjunto por $4 \cdot 10^5$ iteraciones del discriminador con el generador pre-entrenado, utilizando en ambos casos un optimizador Adam con tasa de aprendizaje $1 \cdot 10^{-4}$ y decaimiento de $0,5$ a las $2 \cdot 10^5$ iteraciones. Aquí se utilizó una combinación de pérdida $\mathcal{L}_1$, pérdida perceptual y pérdida adversaria. El segundo entrenamiento se llamará de acá en adelante Real-ESRGAN M2. Para el pre-entrenamiento del generador, se utilizaron los mismos hiperparámetros que Real-ESRGAN M1, pero se optó por combinar la perdida $\mathcal{L}_1$ con la pérdida perceptual. El entrenamiento del discriminador y generador fue realizado conservando la configuración de Real-ESRGAN M1. En el tercer entrenamiento se salteó el pre-entrenamiento del generador, es decir, el modelo Real-ESRGAN M3 constó solamente del entrenamiento en conjunto del generador y discriminador y utilizando los mismos hiperparámetros que los usados en Real-ESRGAN M1. Es importante destacar que los parches de entradas a la red de 64×64 son submuestrados a 32×32 ya que, como se mencionó anteriormente, la Real-ESRGAN usada en este trabajo tiene un factor de escala de $\times 2$.

3) Estrategia de testeo: Al ser el generador una red completamente convolucional, las dimensiones de los sinogramas de entradas no se encuentran fijas y las dimensiones de la salida son proporcionales a las de la entrada. Por esta razón, se diseñaron dos estrategias de testeo para las cuales no se debió realizar ningún cambio en la Real-ESRGAN. Por un lado, los sinogramas de testeo de 256×512 píxeles de baja calidad se submuestrearon a 128×256 píxeles para ser ingresados en la Real-ESRGAN entrenada y así obtener una versión mejorada. Por otro lado, al igual que en el entrenamiento, los 105 parches de cada sinograma de testeo de 64×64 píxeles de baja calidad, se submuestrearon a 32×32 píxeles para ser ingresados en la Real-ESRGAN entrenada y así obtener una versión mejorada. Luego, en este último caso, se rearmaron los parches de los sinogramas para formar sinogramas de tamaño completo 256×512 píxeles. Los sinogramas de salida de ambas estrategias se reconstruyeron siguiendo el procedimiento explicado anteriormente, cuyo resultado fueron imágenes reconstruidas usando LBP.

D. Figuras de mérito

Para evaluar los resultados de los modelos se utilizaron la correlación de Pearson (PC) y la relación señal a ruido pico (PSNR). La correlación de Pearson es una medida de correlación lineal, entre dos imágenes y se define de la siguiente manera:

$$PC(x, y) = \frac{cov(x, y)}{\sigma_x \sigma_y} \tag{11}$$

donde x e y son las imágenes de referencia y reconstruida, respectivamnte, σ denota la desviación estándar y cov es la covarianza. El resultado de la correlación Pearson se encuentra acotado entre -1 (correlación lineal negativa) y 1 (correlación lineal positiva), y un resultado nulo implica que no hay dependencia lineal entre las variables [32]. La PSNR es una medida cuantitativa que evalúa la intensidad de la señal deseada en comparación con el ruido de fondo [33]. En este caso, la señal deseada es la mejor reconstrucción posible de una imagen, y el ruido es el error entre la imagen mencionada y las reconstrucciones producto de los sinogramas de salida de los diferentes modelos. Se puede considerar como una estimación aproximada de la percepción humana de la calidad de la reconstrucción [34]. La PSNR se define como:

$$\mathrm{PSNR}(x, y) = 10 log_{10} \left(\frac{MAX^2}{MSE(x, y)} \right) \tag{12}$$

donde MAX denota el máximo valor que puede tomar un píxel en la imagen y MSE es el error cuadrático medio entre x e y. A diferencia del MSE, un elevado valor de la PSNR (en dB) denota una imagen reconstruida con mejor calidad.

III. Resultados

Se realizó el análisis del rendimiento de los modelos estudiados en este trabajo para mejorar el sinograma sin pre-procesamiento utilizando las cuatro imágenes de testeo de la Fig. 3 y el grupo de fantomas mamarios que se había reservado para el testeo. Con este último grupo, se evaluó y comparó el rendimiento de los modelos utilizando el promedio y la desviación estándar de las métricas anteriormente mencionadas, contrastando las imágenes reconstruidas producto de los sinogramas procesados por las redes neuronales contra la reconstrucción utilizando LBP del sinograma de calidad alta. Esto se realizó con el fin de sólo estar comparando la mejora en los sinogramas sin que influyan las limitaciones del método de reconstrucción en los resultados. Los resultados numéricos se encuentran en la Tabla II. En general, los resultados de todos los modelos se encuentran en la misma franja de valores delimitada por sus desvíos estándar y son ampliamente superiores al resultado obtenido sin pre-procesamiento. Los valores por encima de $0,95$ para la correlación de Pearson indican una fuerte relación positiva entre la imagen de referencia y las imágenes reconstruidas gracias a las distintas redes neuronales. En cuanto a las métricas relacionadas con diferencias numéricas locales, la PSNR de los modelos es más elevada $>$ 12 dB que el caso sin procesar, es decir, debido al procesamiento los valores individuales de los píxeles se acercan más a los

Configuración	Pre-entrenamiento	Entrenamiento
GAN M1	$1 \cdot 10^6$ iteraciones Optimizador Adam Tasa de aprendizaje: $2 \cdot 10^{-4}$ Decaimiento: 0.5 a $3 \cdot 10^5$ iteraciones Pérdida utilizada: $\mathcal{L}_1$	$4 \cdot 10^5$ iteraciones Optimizador Adam Tasa de aprendizaje: $1 \cdot 10^{-4}$ Decaimiento: 0.5 a $2 \cdot 10^5$ iteraciones Pérdidas utilizadas: $\mathcal{L}_1$, pérdida perceptual, pérdida adversaria
GAN M2	$1 \cdot 10^6$ iteraciones Optimizador Adam Tasa de aprendizaje: $2 \cdot 10^{-4}$ Decaimiento: 0.5 a $3 \cdot 10^5$ iteraciones Pérdida utilizada: $\mathcal{L}_1$ y pérdida perceptual	$4 \cdot 10^5$ iteraciones Optimizador Adam Tasa de aprendizaje: $1 \cdot 10^{-4}$ Decaimiento: 0.5 a $2 \cdot 10^5$ iteraciones Pérdidas utilizadas: $\mathcal{L}_1$, pérdida perceptual y pérdida adversaria
GAN M3	- - - - -	$4 \cdot 10^5$ iteraciones Optimizador Adam Tasa de aprendizaje: $1 \cdot 10^{-4}$ Decaimiento: 0.5 a $2 \cdot 10^5$ iteraciones Pérdidas utilizadas: $\mathcal{L}_1$, pérdida perceptual y pérdida adversaria

TABLA I: Configuración de los entrenamientos de Real-ESRGAN

de la mejor imagen posible. En particular, el modelo U-Net muestra el mejor rendimiento en las dos métricas, obteniendo $PC = 0,984$ y $PSNR = 33.6$ dB. Por otro lado, los modelos Real-ESRGAN (M1, M2 y M3) también muestran un rendimiento prometedor. Utilizando tanto parches de imagen (P) como el sinograma completo (C), estos modelos logran un muy buen desempeño. En la Fig. 9, se ejemplifica un resultado cualitativo de todos los modelos junto con la imagen original, la reconstrucción del sinograma de elevada calidad utilizada y la del sinogramas sin pre-procesamiento. A simple vista se nota la mejora producto de la U-Net y de la Real-ESRGAN; los bordes de la muestra de tejido se encuentran definidos más precisamente, al igual que los detalles internos.

En lo que sigue, se analizan los resultados obtenidos para las imágenes de la Fig. 3; como son imágenes individuales, los resultados constan de un valor para cada métrica, sin la desviación estándar. Con respecto a la imagen que se asemeja a vasos sanguíneos, los resultados se muestran en el Tabla III, donde se puede ver que el rendimiento más bajo entre todos corresponde a los resultados obtenidos utilizando datos sin pre-procesamiento, aunque la mejora no es tan substancial como en el caso de fantomas de resonancia mamaria. Además, en la Fig. 10, con la comparación se aprecia esta diferencia entre los resultados de los modelos con la peor reconstrucción como referencia en la primera fila de imágenes a la derecha. De esta forma, se valida el poder de reconstrucción de estructuras complicadas amorfas de todos los modelos, destacando el modelo Real-ESRGAN M2 con la estrategia de testeo de sinograma completo que arroja los mejores resultados en todas las categorías. Acerca del patrón de tejido mamario, los resultados aparecen en el Tabla III. Se observó que todos los modelos Real-ESRGAN superaron el rendimiento de la reconstrucción con datos sin pre-procesamiento en todas las métricas evaluadas. Para el modelo Real-ESRGAN M2, los resultados indican que la utilización del sinograma completo proporciona una mejora en el rendimiento en comparación con el uso de parches de la imagen. En cuanto al modelo Real-ESRGAN M3, tanto con parches del sinograma como con el sinograma completo, mostró un rendimiento aún mejor en comparación con los modelos anteriores. Por su parte, el modelo U-Net muestra un buen rendimiento en la reconstrucción de imágenes en comparación con los datos sin pre-procesamiento, pero no supera el rendimiento de los modelos RealESRGAN evaluados. Esto indica que los modelos RealESRGAN pueden ser más eficaces para la reconstrucción de imágenes en este contexto específico.

En la Fig. 11, se puede apreciar como la U-Net y los distintos modelos entrenados de la Real-ESRGAN mejoran la reconstrucción del sinograma sin pre-procesamiento, validando su uso en imágenes fotoacústicas complejas con ruido y contraste variable.

La Tabla III muestra los resultados numéricos de las métricas para cada modelo en el caso de la imagen Derenzo que evalúa la capacidad del modelo para recuperar objetos circulares de diferentes tamaños.

En lo que respecta a los modelos de Real-ESRGAN, el Real-ESRGAN M1 con parches y el Real-ESRGAN M1 con sinograma completo presentan mejoras significativas en las métricas evaluadas en comparación con la reconstrucción del sinograma sin pre-procesamiento. Prosiguiendo con el análisis, Real-ESRGAN M2 presentan un desempeño mejorado en comparación con el modelo anterior; tanto Real-ESRGAN M2 con parches como el Real-ESRGAN M2 con sinograma completo muestran valores promedio más altos en todas las métricas evaluadas. Por su parte, considerando las dos técnicas de testeo del modelo Real-ESRGAN M3, se consiguieron los valores promedio más altos en todas las métricas evaluadas de todos los modelos, incluida la U-Net. Sin embargo, se debe destacar que aunque los resultados obtenidos por los modelos sean mejores que los producidos por los datos sin pre-procesamiento, la diferencia numérica no es tan grande. En la Fig. 12, se tiene una comparación con todas las reconstrucciones mencionadas.

En cuanto a las reconstrucciones de la imagen con las letras PAT utilizando los diferentes modelos, el modelo Real-ESRGAN M2 con la estrategia de testeo de parches demostró el mejor desempeño. Estos resultados indican que este modelo logra una mejor correlación con la imagen de alta calidad y una mayor relación señal-ruido pico en comparación con los otros modelos evaluados. Todos los modelos de Real-ESRGAN tienen mejores resultados que la U-Net. En la Fig. 13, se puede apreciar la capacidad de los modelos para recuperar objetos con bordes afilados y bien definidos.

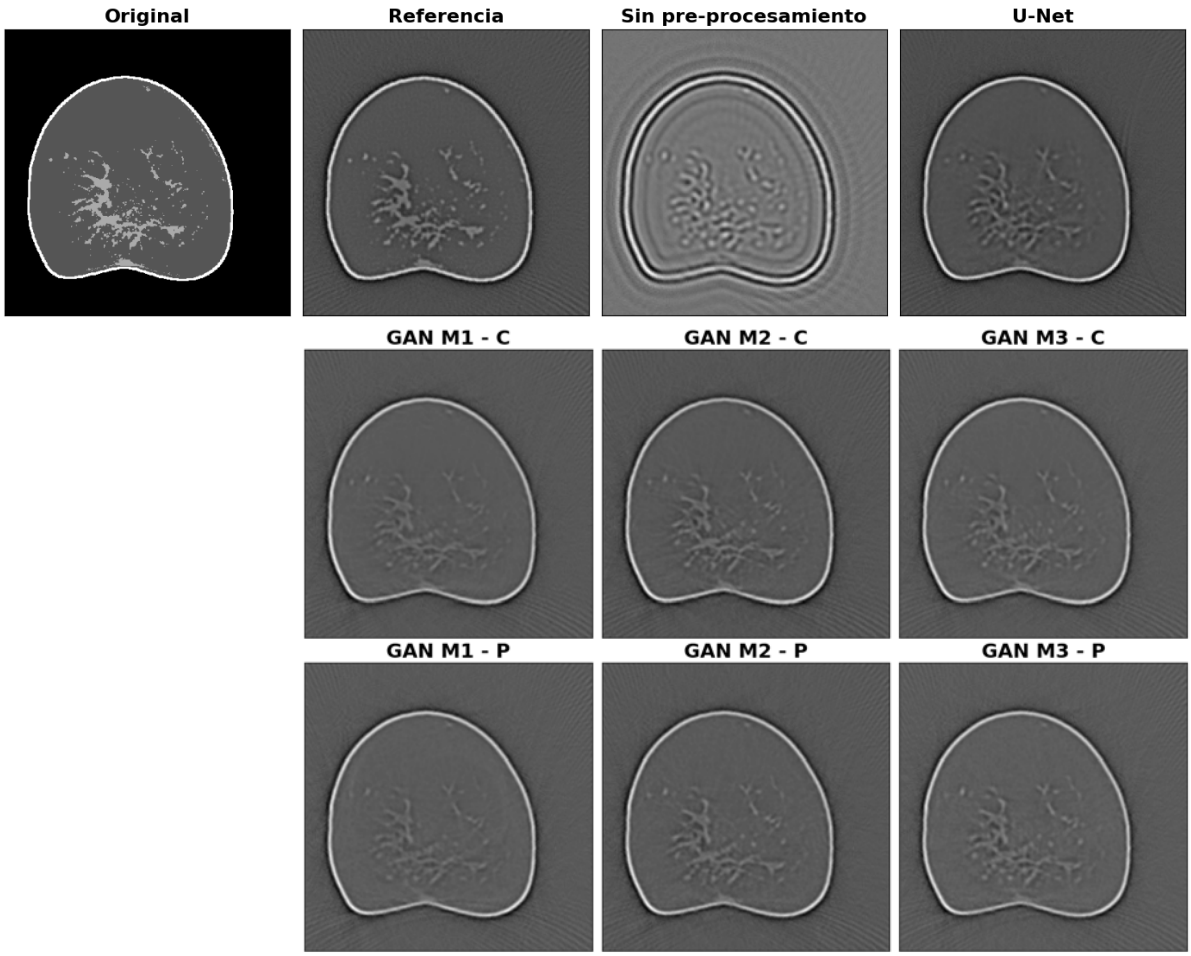

Fig. 9: Imagen original, imagen de referencia producto de la reconstrucción del sinograma de alta calidad y del sinograma sin pre-procesamiento, reconstrucciones de los modelos de la imagen de vasos sanguíneos. P: testeado utilizando los parches del sinograma. C: testeado utilizando el sinograma completo.

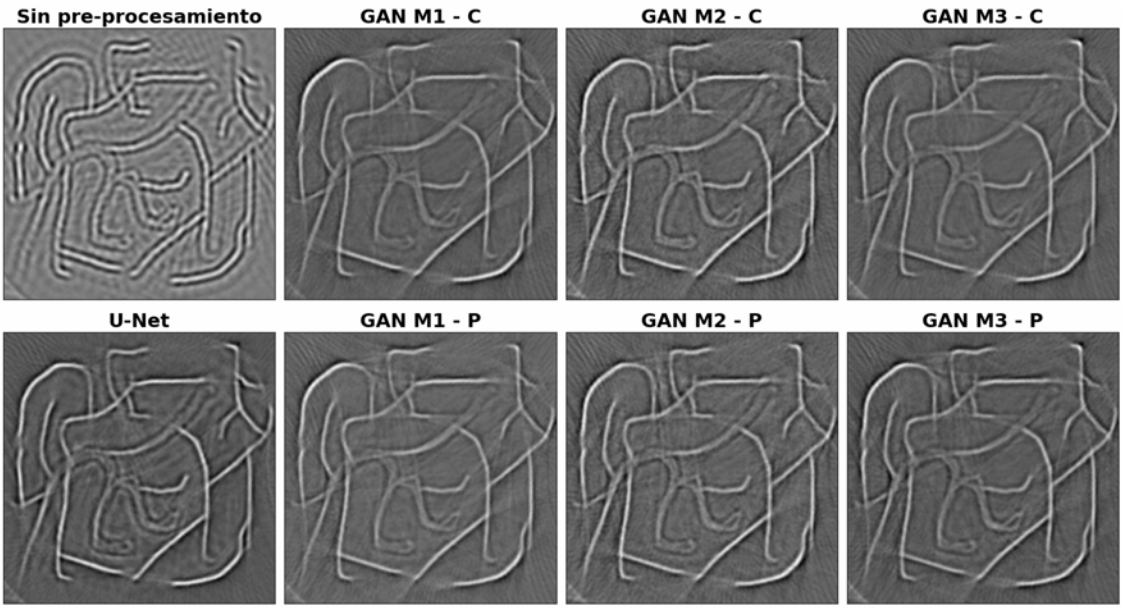

Fig. 10: Imágenes producto de la reconstrucción del sinograma sin pre-procesamiento y los sinogramas procesados por los modelos, correspondiente a la imagen de vasos sanguíneos.

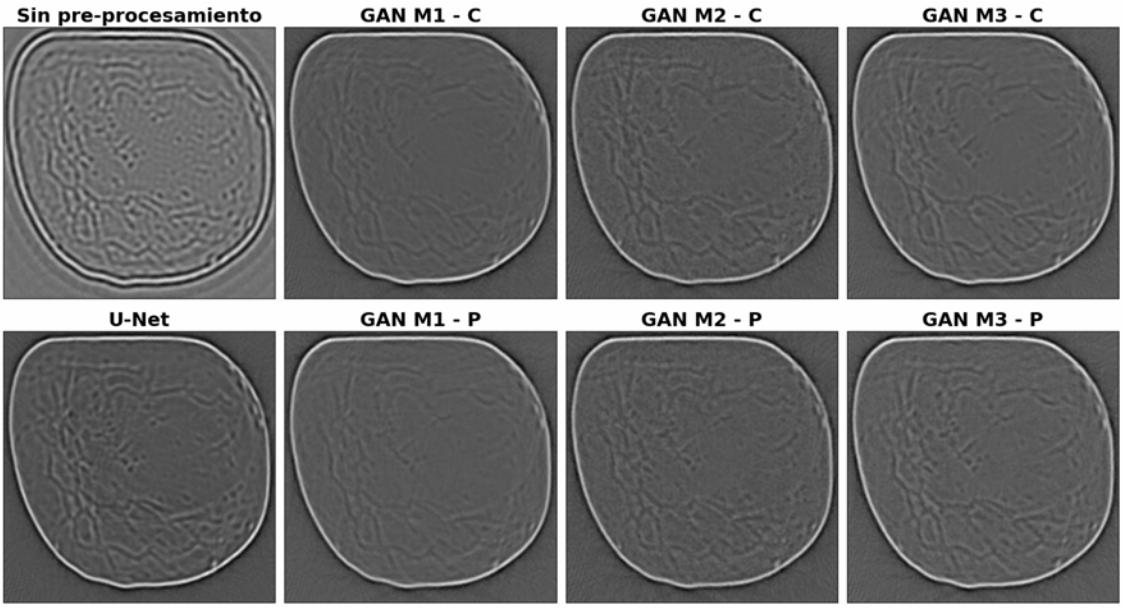

Fig. 11: Imágenes producto de la reconstrucción del sinograma sin pre-procesamiento y los sinogramas procesados por los modelos, correspondiente a la imagen de tejido mamario.

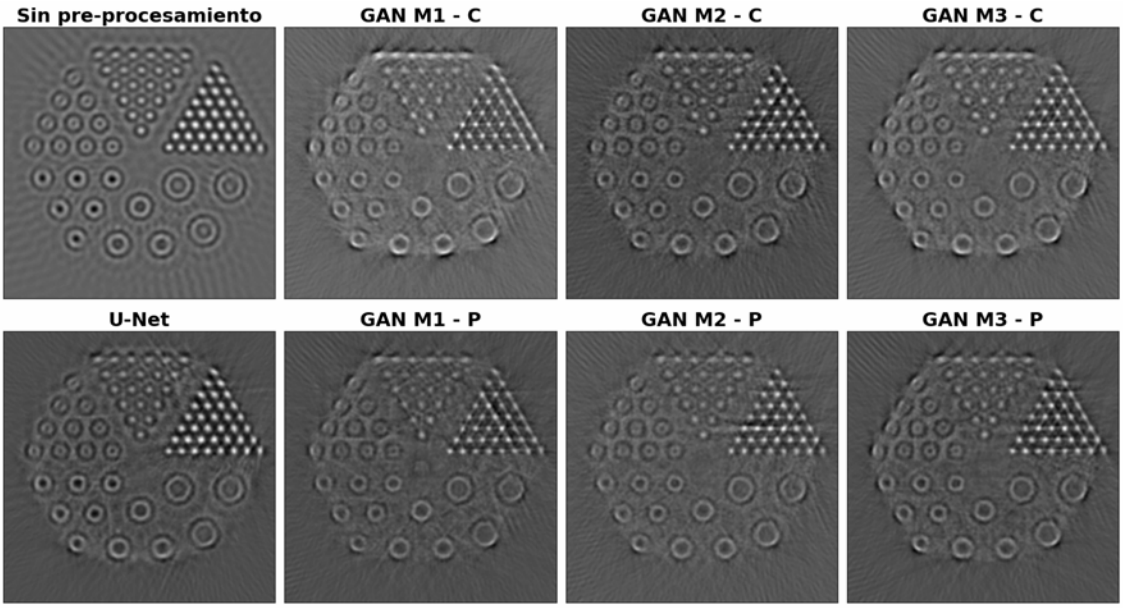

Fig. 12: Imágenes producto de la reconstrucción del sinograma sin pre-procesamiento y los sinogramas procesados por los modelos, correspondiente a la imagen Derenzo.

TABLA II: Resultados de la comparación de las imágenes reconstruidas por los diferentes modelos y la mejor reconstrucción posible con los datos de alta calidad para el conjunto de datos de testeo de fantomas numéricos de resonancias mamarias. P: estrategia de testeo con parches. C: estrategia de testo con el sinograma completo.

	PC $\pm$ std	PSNR $\pm$ std
Sin procesamiento	0.730 $\pm$ 0.013	18.696 $\pm$ 0.676
GAN M1 - P	0.973 $\pm$ 0.007	31.163 $\pm$ 1.197
GAN M1 - C	0.975 $\pm$ 0.005	31.482 $\pm$ 1.158
GAN M2 - P	0.977 $\pm$ 0.006	31.975 $\pm$ 1.240
GAN M2 - C	0.979 $\pm$ 0.005	32.071 $\pm$ 1.268
GAN M3 - P	0.975 $\pm$ 0.006	31.501 $\pm$ 1.108
GAN M3 - C	0.977 $\pm$ 0.005	31.944 $\pm$ 1.178
U-Net	0.984 $\pm$ 0.006	33.559 $\pm$ 1.758

IV. Discusión

El análisis de los resultados de la sección anterior revela que los modelos Real-ESRGAN y U-Net superan significativamente el rendimiento de los datos sin pre-procesamiento en las dos métricas evaluadas. Esto indica que los modelos son capaces de mejorar la calidad de las imágenes reconstruidas en comparación con la reconstrucción obtenida a partir de los sinogramas sin pre-procesamiento. Sin embargo, se encontró un caso de falla de estos modelos: la reconstrucción de la imagen Derenzo. Esto indicaría que los modelos no son capaces de generalizar y reconstruir precisamente objetos circulares pequeños y grandes. El resultado no es sorprendente si se consideran las características de las imágenes de entrenamiento de los modelos; estas eran resonancias

TABLA III: Resultados de la comparación de las imágenes reconstruidas por los diferentes modelos y la mejor reconstrucción posible con los datos de alta calidad para la imagen de vasos sanguíneos, de tejido mamario, Derenzo y con las letras PAT.

	Vasos sanguíneos		Tejido mamario		Derenzo		Letras PAT	
	PC	PSNR	PC	PSNR	PC	PSNR	PC	PSNR
Sin procesamiento	0.650	14.311	0.672	16.937	0.619	17.649	0.488	15.548
GAN M1 - P	0.812	16.832	0.891	22.432	0.643	17.851	0.738	18.350
GAN M1 - C	0.821	16.857	0.898	22.700	0.649	18.008	0.773	18.786
GAN M2 - P	0.826	17.397	0.893	22.493	0.657	18.040	0.786	19.052
GAN M2 - C	0.847	17.975	0.902	22.822	0.685	18.410	0.749	18.541
GAN M3 - P	0.822	17.006	0.901	23.041	0.667	18.165	0.745	18.470
GAN M3 - C	0.832	16.909	0.909	23.334	0.684	18.417	0.782	18.872
U-Net	0.832	17.504	0.891	22.681	0.667	18.236	0.676	17.726

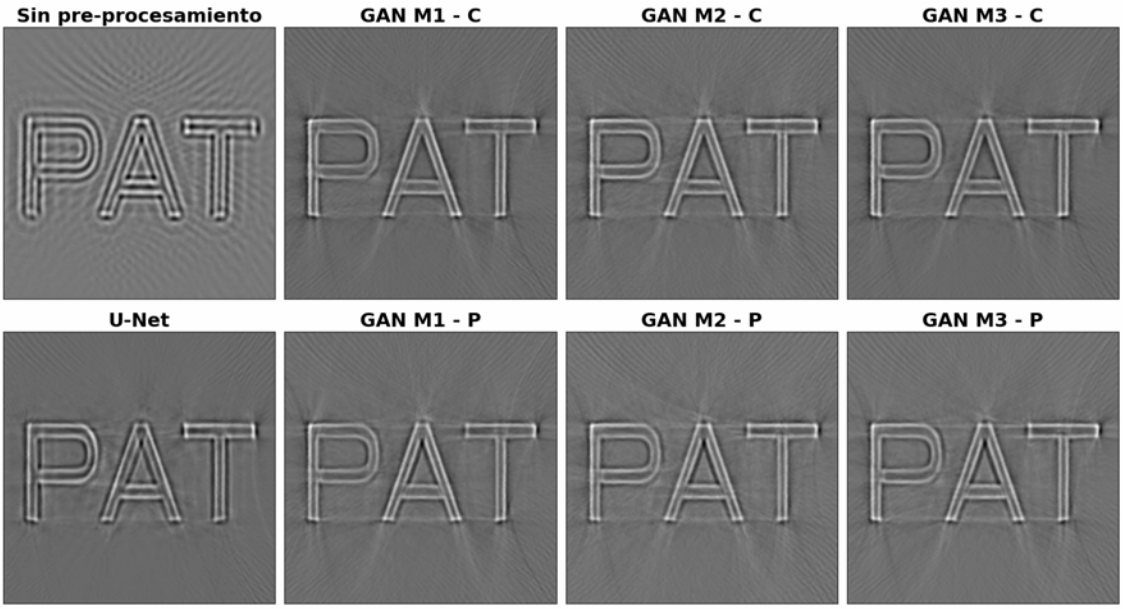

Fig. 13: Imágenes producto de la reconstrucción del sinograma sin pre-procesamiento y los sinogramas procesados por los modelos, correspondiente a la imagen de las letras PAT.

mamarias que no contenían formas circulares tan definidas como las de la imagen Derenzo.

En lo que concierne a las imágenes reconstruidas, en general se observa un desempeño similar comparando los resultados obtenidos por la U-Net y por la Real-ESRGAN. No obstante, cabe destacar que esta última necesita menos datos que la anterior para lograr esos resultados. El sinograma de baja calidad, al momento de ser generado, contiene la información de 128 detectores y 512 muestras temporales; se interpola esta matriz obteniendo un sinograma de 256 detectores y 512 muestras con el fin de ser ingresado a la U-Net y obtener un sinograma con las dimensiones correctas a la salida (256×512). En cambio, antes de ingresar en la Real-ESRGAN, se submuestrea el sinograma degradado; de esta forma, con un sinograma de entrada de 128 detectores y 256 muestras temporales, la Real-ESRGAN genera un sinograma de 256 detectores y 512 muestras temporales. En otras palabras, la Real-ESRGAN sólo necesita el 50% de los datos reales de entrada (la interpolación necesaria en la U-Net estaría agregando datos artificialmente), y sólo el 25% de los datos totales que ingresan en la U-Net.

Los modelos basados en la familia Real-ESRGAN, especialmente el Real-ESRGAN M2 y M3, muestran el mejor rendimiento en términos de las imágenes de prueba y las figuras de mérito usadas en este trabajo. Además, se destaca la utilidad de la estrategia de testeo con los sinogramas completos para obtener mejores resultados en la reconstrucción de imágenes.

En cuanto a los diferentes modelos entrenados de Real-ESRGAN, se ve una leve mejora en el modelo Real-ESRGAN M2 con el generador preentrenado utilizando la pérdida $\mathcal{L}_1$ en conjunto con la pérdida perceptual, en vez de sólo la $\mathcal{L}_1$ como en el caso de Real-ESRGAN M1. Esto se puede deber a que este último modelo no logra capturar la incertidumbre en la recuperación de detalles de alta frecuencia perdidos, como la textura; ya que al minimizar el error absoluto medio esta pérdida se queda con los promedios de los valores de píxeles de soluciones posibles, que generalmente son demasiado suaves, y por lo tanto tienen una calidad de percepción deficiente [31].

V. Conclusiones

En general, se puede concluir que tanto el modelo U-Net como los modelos Real-ESRGAN son efectivos para la reconstrucción de imágenes OA en el contexto de resonancias mamarias. Estos modelos logran mejorar la calidad de las imágenes finales en comparación con las reconstrucciones producto de los sinogramas con ruido y ancho de banda limitado. En particular, los resultados son prometedores, ya que indican que la red neuronal Real-ESRGAN puede

ser considerada como una opción viable para la super-resolución, remoción de ruido y mejora de ancho de banda de los sinogramas provenientes de sistemas para TOA. Estos hallazgos son relevantes para el campo de la biomedicina, ya que demuestran el potencial de los modelos de aprendizaje automático en la mejora de la calidad de las imágenes de resonancias mamarias provenientes de sistemas para TOA, lo que podría ayudar a mejorar la precisión y confiabilidad de los diagnósticos médicos. En futuros trabajos sería interesante explorar el uso de las redes neuronales transformers para realizar esta tarea [35], [36]. Por otro lado, se podría implementar un barrido exhaustivo de hiperparámetros, incluyendo la variación de la cantidad de capas convolucionales de los RRDB, prueba que no se podía realizar en este trabajo por limitaciones de memoria de GPU.

Agradecimientos

Este trabajo fue financiado por la Universidad de Buenos Aires (UBACYT 20020190100032BA), CONICET (PIP 11220200101826CO) and la Agencia I+D+i (PICT 2018-04589, PICT 2020-01336).

Referencias

[1] R. A. Kruger, W. L. Kiser, D. R. Reinecke, G. A. Kruger, and K. D. Miller, "Thermoacoustic molecular imaging of small animals," *Molecular imaging*, vol. 2, no. 2, p. 15353500200303109, 2003.

[2] X. Wang, Y. Xu, M. Xu, S. Yokoo, E. S. Fry, and L. V. Wang, "Photoacoustic tomography of biological tissues with high cross-section resolution: Reconstruction and experiment," *Medical physics*, vol. 29, no. 12, pp. 2799–2805, 2002.

[3] X. Wang, Y. Pang, G. Ku, X. Xie, G. Stoica, and L. V. Wang, "Noninvasive laser-induced photoacoustic tomography for structural and functional in vivo imaging of the brain," *Nature biotechnology*, vol. 21, no. 7, pp. 803–806, 2003.

[4] P. Beard, "Biomedical photoacoustic imaging," *Interface focus*, vol. 1, no. 4, pp. 602–631, 2011.

[5] I. Steinberg, D. M. Huland, O. Vermesh, H. E. Frostig, W. S. Tummers, and S. S. Gambhir, "Photoacoustic clinical imaging," *Photoacoustics*, vol. 14, pp. 77–98, 2019.

[6] M. Mehrmohammadi, S. Joon Yoon, D. Yeager, and S. Y Emelianov, "Photoacoustic imaging for cancer detection and staging," *Current Molecular Imaging (Discontinued)*, vol. 2, no. 1, pp. 89–105, 2013.

[7] P. Hai, Y. Qu, Y. Li, L. Zhu, L. Shmuylovich, L. A. Cornelius, and L. V. Wang, "Label-free high-throughput photoacoustic tomography of suspected circulating melanoma tumor cells in patients in vivo," *Journal of biomedical optics*, vol. 25, no. 3, p. 036002, 2020.

[8] L. V. Wang and J. Yao, "A practical guide to photoacoustic tomography in the life sciences," *Nature methods*, vol. 13, no. 8, pp. 627–638, 2016.

[9] R. A. Kruger, R. B. Lam, D. R. Reinecke, S. P. Del Rio, and R. P. Doyle, "Photoacoustic angiography of the breast," *Medical physics*, vol. 37, no. 11, pp. 6096–6100, 2010.

[10] G. Ku and L. V. Wang, "Deeply penetrating photoacoustic tomography in biological tissues enhanced with an optical contrast agent," *Optics letters*, vol. 30, no. 5, pp. 507–509, 2005.

[11] A. Fatima, K. Kratkiewicz, R. Manwar, M. Zafar, R. Zhang, B. Huang, N. Dadashzadeh, J. Xia, and K. M. Avanaki, "Review of cost reduction methods in photoacoustic computed tomography," *Photoacoustics*, vol. 15, p. 100137, 2019.

[12] N. Awasthi, S. K. Kalva, M. Pramanik, and P. K. Yalavarthy, "Vector extrapolation methods for accelerating iterative reconstruction methods in limited-data photoacoustic tomography," *Journal of biomedical optics*, vol. 23, no. 7, 2018.

[13] N. Awasthi, R. Pardasani, S. K. Kalva, M. Pramanik, and P. K. Yalavarthy, "Sinogram super-resolution and denoising convolutional neural network (srcn) for limited data photoacoustic tomography," *arXiv preprint arXiv:2001.06434*, 2020.

[14] W. Choi, D. Oh, and C. Kim, "Practical photoacoustic tomography: realistic limitations and technical solutions," *Journal of Applied Physics*, vol. 127, no. 23, p. 230903, 2020.

[15] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[16] N. Awasthi, G. Jain, S. K. Kalva, M. Pramanik, and P. K. Yalavarthy, "Deep neural network-based sinogram super-resolution and bandwidth enhancement for limited-data photoacoustic tomography," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 67, no. 12, pp. 2660–2673, 2020.

[17] A. Hauptmann and B. T. Cox, "Deep learning in photoacoustic tomography: current approaches and future directions," *Journal of Biomedical Optics*, vol. 25, no. 11, 2020.

[18] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[19] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[20] C. B. Shaw, J. Prakash, M. Pramanik, and P. K. Yalavarthy, "Least squares qr-based decomposition provides an efficient way of computing optimal regularization parameter in photoacoustic tomography," *Journal of Biomedical Optics*, vol. 18, no. 8, 2013.

[21] J. Prakash, A. S. Raju, C. B. Shaw, M. Pramanik, and P. K. Yalavarthy, "Basis pursuit deconvolution for improving model-based reconstructed images in photoacoustic tomography," *Biomedical optics express*, vol. 5, no. 5, pp. 1363–1377, 2014.

[22] A. Sarno, G. Mettivier, F. di Franco, A. Varallo, K. Bliznakova, A. M. Hernandez, J. M. Boone, and P. Russo, "Dataset of patient-derived digital breast phantoms for in silico studies in breast computed tomography, digital breast tomosynthesis, and digital mammography," *Medical Physics*, vol. 48, no. 5, pp. 2682–2693, 2021.

[23] S. Gutta, M. Bhatt, S. K. Kalva, M. Pramanik, and P. K. Yalavarthy, "Modeling errors compensation with total least squares for limited data photoacoustic tomography," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 25, no. 1, pp. 1–14, 2017.

[24] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[25] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer, 2015, pp. 234–241.

[26] G. Developers. Descending into ml: Training and loss — machine learning. [Online]. Available: https://developers.google.com/machine-learning/crash-course/descending-into-ml/training-and-loss

[27] X. Wang, L. Xie, C. Dong, and Y. Shan, "Real-esrgan: Training real-world blind super-resolution with pure synthetic data," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 1905–1914.

[28] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. Change Loy, "Esrgan: Enhanced super-resolution generative adversarial networks," in *Proceedings of the European conference on computer vision (ECCV) workshops*, 2018.

[29] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," *arXiv preprint arXiv:1802.05957*, 2018.

[30] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*. Springer, 2016, pp. 694–711.

[31] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. P. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network. corr abs/1609.04802 (2016)," *arXiv preprint arXiv:1609.04802*, 2016.

[32] L. Statistics. Pearson product-moment correlation. [Online]. Available: https://statistics.laerd.com/statistical-guides/pearson-correlation-coefficient-statistical-guide.php

[33] M. M. H. Center. Psnr. [Online]. Available: https://nl.mathworks.com/help/images/ref/psnr.html

[34] U. Sara, M. Akter, and M. S. Uddin, "Image quality assessment through fsim, ssim, mse and psnr—a comparative study," *Journal of Computer and Communications*, vol. 7, no. 3, pp. 8–18, 2019.

[35] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[36] C. Yao, S. Jin, M. Liu, and X. Ban, "Dense residual transformer for image denoising," *Electronics*, vol. 11, no. 3, p. 418, 2022.