Ciencia Latina Revista Científica Multidisciplinar
Marzo-Abril, 2023, Volumen 7, Número 2.
https://doi.org/10.37811/cl_rcm.v7i2.5871

Avances en las técnicas de eliminación de marcas de agua visibles basadas en aprendizaje profundo

Karen Alejandra Valerio-Trigueros[1]
kvaleriot2100@alumno.ipn.mx
https://orcid.org/0009-0009-1898-5614
Instituto Politécnico Nacional, CDMX

Luis Angel Olvera-Martinez
lolveram1500@alumno.ipn.mx
https://orcid.org/0000-0002-9289-5037
Instituto Politécnico Nacional, CDMX

Carlos Adolfo Diaz-Rodriguez
cdiazr1302@alumno.ipn.mx
Instituto Politécnico Nacional, CDMX

Manuel Cedillo-Hernandez
mcedilloh@ipn.mx
https://orcid.org/0000-0002-9149-9841
Instituto Politécnico Nacional, CDMX

Enrique Tonatihu Jimenez-Borgonio
ejimenezb2200@alumno.ipn.mx
https://orcid.org/0000-0002-3467-5861
Instituto Politécnico Nacional, CDMX

RESUMEN

En los últimos años los algoritmos de inteligencia artificial han demostrado tener grandes resultados en diferentes áreas de aplicación, tales como robótica, medicina, seguridad informática, finanzas, entre otras. En el contexto de procesamiento digital de imágenes, el uso del aprendizaje profundo está siendo aplicado para remover marcas de agua visibles en imágenes digitales, con la finalidad de eliminar la protección de derechos de autor de los propietarios de las imágenes en cuestión. El presente trabajo realiza una recopilación de los trabajos más recientes que remueven marcas de agua visibles a través de aprendizaje profundo, con la finalidad de analizar las tendencias actuales que permitan diseñar algoritmos de marcado de agua visible más robustos ante este tipo de herramientas de remoción.

Palabras clave: *aprendizaje profundo; marcado de agua visible; seguridad de la información; protección de derechos de autor; redes neuronales.*

[1] Autor Principal

Advances on visible watermark removal techniques based on deep learning

ABSTRACT

In recent years, artificial intelligence algorithms have shown great results in different application areas, such as robotics, medical, informatics security, financial services, among others. In the context of digital image processing, the use of deep learning is being applied to remove visible watermarks from the visual content of digital images, to remove the copyright protection of the owners of the images in question. This paper makes a briefly survey of the most recent works that remove visible watermarks employing deep learning, with the purpose of analyzing the current trends that allow designing more robust visible watermarking algorithms against this type of removal tools.

Keywords**:** *copyright protection; deep learning; information security; neural networks; visible watermarking.*

INTRODUCCIÓN

Las marcas de agua digitales se utilizan para la protección de los derechos de copia y propiedad de los archivos de datos multimedia, posibilitando la identificación de la fuente, autor, propietario, distribuidor o consumidor autorizado, de imágenes digitales, grabaciones de audio o video. (Orúe, 2002) . Es decir, el marcado de agua digital es la técnica de ocultar información en un contenido digital conocido como "host" o "anfitrión" con el objetivo de protegerlo contra la manipulación o uso ilegal. (Vargas, 2016).

Cabe mencionar que esta información añadida tiene diversas cualidades, las cuales son:

- **Capacidad:** Se refiere a la cantidad de información que se puede ocultar. Se mide en bpp (bits por píxel). La capacidad necesaria depende de la aplicación en particular.
- **Imperceptibilidad:** La marca no debe ser visible, salvo en el caso de los logos que constituyen un tipo especial de información embebida para autenticación y protección contra copias.
- **Robustez:** Es la cualidad de persistir pese a ataques intencionales o daños colaterales. (Vargas L. M., 2015)

De acuerdo con el propósito que se le dé, las marcas de agua pueden ser visibles o invisibles:

- **Visibles:** Se puede apreciar la marca de agua a simple vista.
- **Invisibles:** Es aquella que es imperceptible al ojo humano. Para este tipo de marcas se busca que al introducirla no degrade a la imagen que se está marcando, (Álvarez, 2020).

Con el fin de determinar la efectividad de las marcas de agua en la protección de los derechos de autor, diversos investigadores, (Lubin, 2003) , (Pei, 2006), (Tanha, junio de 2012) , (Xu, 2017), comenzaron a realizar intentos cada vez más sofisticados para la detección y remoción de marcas de agua visibles. En la actualidad la tendencia más generalizada en tema de investigación se orienta en el uso de técnicas de aprendizaje profundo (Cheng, 2018), (Jiang, 2020), (Li, 2021), (Liang, 2021), (Liu, 2021), ya que han mostrado tener grandes resultados en esta área. El propósito del presente trabajo es realizar una revisión de las propuestas más actuales en lo que respecta al tema de detección y remoción de marcas de agua visibles utilizando técnicas de deep learning. El presente trabajo está conformado como sigue: en la sección uno se realiza un breve repaso de los conceptos más importantes en el tema, particularmente las redes neuronales más utilizadas de aprendizaje profundo y los conceptos de las métricas que utilizan para medir la eficiencia de los resultados en las propuestas; en la segunda sección

se presenta la revisión de las propuestas más recientes reportadas en la literatura cientifica. Finalmente, en la última sección se presenta el análisis de resultados y discusión de las propuestas abordadas en el trabajo, así como las conclusiones del trabajo.

Aprendizaje profundo

Son aquellas técnicas de aprendizaje automático que hacen uso de arquitecturas de redes neuronales, este utiliza redes más profundas de neuronas en múltiples capas. El aprendizaje profundo tiene como objetivo extraer características de las representaciones de datos que se pueden generar utilizando machine learning (ML). Una red neuronal artificial es un sistema inspirado en los sistemas biológicos, donde cada nodo de la red (una neurona artificial) puede transmitir una señal a otros nodos. Por otro lado, una red neuronal de aprendizaje profundo consta de una capa de entrada, varias capas ocultas y una capa de salida: la palabra 'profundo' refiere a la cantidad de capas a través de las cuales se transforman los datos. Cada una de estas capas aprende a transformar sus datos de entrada (que son la salida de la capa anterior) en una representación un poco más abstracta y compuesta, que a su vez se alimenta como entrada a la capa siguiente. (Regazzoni, 2021). Dentro de las redes neuronales que se emplean dentro de la literatura científica relacionada con remover marcas de agua visibles contenidas en imágenes se encuentran las siguientes:

U-Net: Esta red tiene una estructura codificador - decodificador. En la primera parte la imagen es reducida a través de filtros convolucionales para extraer características de bajo nivel. En la segunda parte, este vector de características sigue el camino contrario, aumentando en cada capa convolucional sus dimensiones hasta obtener el tamaño de la imagen original, donde a la salida se encuentra la señal reconstruida. (Verdeguer Gómez, 2021)

Retinanet: Es una red única y unificada compuesta por una red troncal y dos subredes específicas de tareas. La red troncal es responsable de calcular un mapa de características de convolución sobre una imagen de entrada completa. La primera subred realiza la clasificación en la salida de la red troncal; la segunda subred realiza la regresión del cuadro delimitador de convolución. (Guerrero Citelly, 2018)

VGG19: Esta Red neuronal convolucional está compuesta por 16 capas convolucionales, 3 fully-connected, 5 MaxPool y 1 SoftMax, con un aproximado de 143 millones de parámetros. (Pérez-Aguilar, 2021)

VGG16: Es una red compuesta por 16 capas que fue entrenada con la base de datos ImageNet, suponiendo mejoras en relación con la arquitectura AlexNet, puesto que reemplaza los grandes filtros de los kernels por un conjunto de filtros de tamaño 3 × 3. (Pérez-Aguilar, 2021)

Las métricas que adoptaron diversos autores para evaluar la efectividad de la remoción de la marca de agua visible son las siguientes:

Peak Signal to Noise Ratio (PSNR): Es una métrica para medir el nivel de distorsión que contiene una imagen reconstruida, de este modo, si el valor del PSNR es alto o tiende a infinito se concluye que la distorsión en la señal de imagen no es perceptible para el sistema visual humano. El PSNR calcula la relación entre el valor máximo posible de una señal y el valor del ruido de distorsión que afecta la calidad de su representación, donde dicha relación entre ambas imágenes es cuantificada en decibeles (dB) y puede ser expresada por la ecuación 1:

$$PSNR = 10\ \log_{10}\frac{peakval^2}{MSE} \tag{1}$$

Donde *peakval* (Peak Value) corresponde al valor máximo en escala de grises de la imagen. De este modo, si la imagen tiene una resolución de 8 bits por píxel, el valor de *peakval* es 255. Como se puede observar el PSNR es una representación de error absoluto en dB. (Sara, 2019)

Structure Similarity Index Measure **(SSIM)**: Estima la calidad percibida en imágenes, es decir mide la similitud entre dos imágenes, la original y la reconstruida (Wang, 2004). Para ello el método del índice SSIM se encarga de calcular una métrica referente a la calidad de las imágenes en tres aspectos que son: la luminiscencia l (usada para comparar el brillo entre dos imágenes), el contraste c (utilizada para diferenciar los rangos entre la región más brillante y la más oscura de dos imágenes) y el término estructural o de correlación s (se utiliza para comparar el patrón de luminancia local entre dos imágenes para encontrar la similitud y disimilitud de las imágenes). Dicho SSIM, puede ser expresado a través de esos tres términos como se muestra en la ecuación 2.

$$SSIM(x,y) = [l(x,y)]^{\alpha} \cdot [c(x,y)]^{\beta} \cdot [s(x,y)]^{\gamma} \tag{2}$$

Nuevamente la luminiscencia, el contraste y el termino estructural pueden ser expresadas de manera independiente como se muestra en la ecuación 3.

$$l(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2+\mu_y^2 + C_1}, c(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2+\sigma_y^2 + C_2}, s(x,y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \quad (3)$$

Donde μ_x y μ_y son los promedios locales, σ_x y σ_y son las desviaciones estándar y σ_{xy} es la covarianza cruzada de las imágenes x, y. (Sara, 2019)

METODOLOGÍA

En aras de la brevedad se procederá a hacer un resumen muy puntual de artículos que hablan sobre eliminación de marcas de agua.

(Cheng, 2018) Propone la detección automática de marcas de agua visibles, además exploran su eliminación y crean un conjunto de datos de marcas de agua visibles a gran escala para lograr una detección en condiciones desconocidas. Para lograr la detección, toma un algoritmo basado en aprendizaje profundo, llamado RetinaNet. El modelo toma como entrada una imagen con marca de agua y estima las probabilidades de todos los candidatos con diferentes escalas y proporciones, en todas las ubicaciones de la imagen clasificadas como el área que está estrechamente cubierta por una marca de agua, con el fin de lograr la detección. Una vez que las marcas de agua en las imágenes fueron detectadas, los resultados de la detección se utilizan para la eliminación. La eliminación de marcas de agua visibles se lleva a cabo con redes neuronales profundas, el cual consta de dos componentes: red de eliminación de marca de agua y red de pérdida. Cada parche con marca de agua x se introduce en la red de eliminación de marcas de agua para obtener el parche sin marcas de agua estimado ỹ. Luego, la pérdida $L1$ y la pérdida perceptiva se calculan en función de la clasificación absoluta y los parches estimados. En lugar de transferir una imagen completa, se espera que los píxeles dentro del área detectada se recuperen a una condición sin marcar, mientras que aquellos en el área sin marcar en la imagen con marca de agua permanecerán sin cambios. La pérdida de $L1$ penaliza la distancia de píxel entre la clasificación absoluta y la salida, y se denota como L_{L1} como en la ecuación 4.

$$L_{L1}(x,y) = \|f(x) - y\|_{1,} \quad (4)$$

x se denota como un parche de entrada con marca de agua detectado y recortado de una imagen con marca de agua, y se refiere al parche de verdad sin marca de agua. $f(x)$ es la salida de U-net.

La formulación de la pérdida de percepción se puede denotarse como en la ecuación 5.

$$L_{pl}^{\Phi,j}('\hat{y},'y) = \frac{1}{C_j\ H_j\ W_j}\ \|\Phi_j('\hat{y}) - \Phi_j('y)\|_2^2 \tag{5}$$

En este trabajo, utilizan relu2_2 de VGG-16 para mantener los detalles de la información de entrada. Por lo tanto, la función objetivo de esta red de eliminación es definido por la ecuación 6.

$$L_{whole} = L_{L1} + \alpha L_{pl}^{\Phi,relu2_2} \tag{6}$$

Donde $\alpha \geq 0$ es un peso para regularizar el efecto de la pérdida de $L1$ y la pérdida de percepción. *Cabe* mencionar que el modelo no obtuvo resultados ideales, debido a la estructura de red simple que realizaron y una función de pérdida ineficiente que explotaron. (Jiang, 2020)

(Jiang, 2020) Este artículo propone una arquitectura de red de eliminación de marca de agua visible basada en redes antagónicas generativas condicionales (CGAN) y redes antagónicas generativas de mínimos cuadrados (LSGAN). En este articulo construyeron un modelo de eliminación de marca de agua visible de dos etapas, extrayendo las características de la marca de agua y la imagen, respectivamente. La primera etapa es extraer la marca de agua mediante la identificación de la característica de esta y luego eliminarla. La segunda etapa es la imagen en pintura, que obtiene directamente una imagen sin marca de agua mediante la codificación y decodificación de la imagen de eliminación de la primera etapa. En particular, el generador del modelo propuesto incluye dos partes, la red de extracción y la red de pintura, en las que se utilizan dos discriminadores para identificar las imágenes generadas y las imágenes reales.

La red de extracción se basa en U-Net, que consta de tres capas de codificador; tres capas de decodificador y cuatro capas de convolución dilatada que conectan el codificador y el decodificador. El codificador se utiliza para extraer características significativas de la entrada de imágenes. El decodificador está diseñado para decodificar la salida de convolución dilatada. La red de extracción se centra en las áreas relevantes de la imagen con marca de agua, tomando la imagen I_0 como entrada y extrayendo la marca de agua M_1 como salida. Después de obtener M_1, se resta M_1 de la imagen I_0 para obtener una imagen de eliminación preliminar.

La red de pintura interior se centra en toda la imagen con marca de agua, que toma la imagen de eliminación preliminar y la imagen I_0 como entrada y genera una imagen sin marca de agua. Para hacer

que la marca de agua extraída se acerque más a la marca de agua real y que la imagen generada se acerque más a la imagen real, utilizaron dos discriminadores PatchGAN condicionales; D_M y D_I. En este trabajo, se utilizó la pérdida de reconstrucción y la pérdida adversaria para orientar el modelo propuesto. Específicamente, la pérdida de reconstrucción incluye pérdida ℓ_2 y pérdida de percepción. La pérdida ℓ_2 se utiliza para evaluar el cuadrado medio de la distancia de píxeles entre la imagen generada y la imagen real. Mientras la pérdida percibida lo hace a nivel de característica y se obtiene a través de convolución, que puede capturar la información semántica de la imagen. En el modelo propuesto, la pérdida de ℓ_2 incluye L_M y L_I. En particular, L_M es la distancia ℓ_2 entre la marca de agua real y marca de agua generada. L_I es la distancia ℓ2 entre la imagen real sin marca de agua y la imagen generada. L_M está dada por la ecuación 7.

$$L_M(f) = \| f(I_0) - M_{gt} \|_2 \tag{7}$$

Donde I_0 es una imagen con marca de agua, $f(\cdot)$ denota la salida de la red de extracción y M_{gt} es la marca de agua real. L_I se define por la ecuación 8.

$$L_I(f, g) = \| g\big(I_0 - f(I_0)\big) - I_{gt} \|_2 \tag{8}$$

Donde, $g(\cdot)$ denota la salida de la red de pintura I_{gt} es la imagen real sin marca de agua. Para extraer características de alto nivel y capturar efectivamente la información semántica de la imagen, utilizaron la pérdida de percepción como parte de la pérdida de reconstrucción, definida por la ecuación 9.

$$L_{per}^{\Phi.j}(f, g) = \frac{1}{C_j H_j W_j} \| \Phi_j\Big(g\big(I_0 - f(I_0)\big)\Big) - \Phi_j\big(I_{gt}\big) \|_2 \tag{9}$$

Donde $\Phi_j(\cdot)$ representa el mapa de características de la capa j-ésima de una transformación de convolución entrenada previamente para extraer características avanzadas. $\Phi_j(\cdot)$ corresponde al mapa de características de la capa Relu2-2 de la red VGG19 previamente entrenada en el conjunto de datos de ImageNet. El tamaño del mapa de características es $C_j x H_j x W_j$. La red de extracción y la red de pintura se entrenan en función de LSGAN condicional. La función objetivo condicional LSGAN, está definida por la ecuación 10

$$min_D\, V_{LSGAN}(D) = \frac{1}{2} \pounds_{x \sim Pdata(x)}\big[\big(D\big(x\big|\Phi(y)\big) - 1\big)^2\big] + \frac{1}{2} \pounds_{x \sim Pdata(x)}\big[\big(D\big(G(z)\big|\Phi(y)\big)\big)^2$$

$$min_G V_{LSGAN}(G) = \frac{1}{2} \pounds_{z \sim Pdata(z)}\big[\big(D\big(G(z)\big|\Phi(y)\big) - 1\big)^2\big] \tag{10}$$

Donde D es el discriminador, G es el generador, x son los datos reales, z es la entrada del generador e y es la condición. D_M representa la verdadera probabilidad de la región correspondiente de la imagen de entrada. La red de extracción y red de repintado, se definen en la ecuación 11 y L_f es la pérdida adversaria para la red de extracción y la red de repintado.

$$\min_{D1} L_{D1} = \min_{D1} \frac{1}{2}\mathbb{E}_{Igt\sim P1_{gt},\, I_0\sim p_{I0}}[D_I(I_0, I_{gt}) - 1]^2 \; + \frac{1}{2}\mathbb{E}_{I0\sim P_{I0}}[D_1\left(I_0, g(f(I_0))\right)]^2$$

$$\min_{f,g} L_{f,g} = \min_{f,g} \frac{1}{2}\mathbb{E}_{I0\sim P_{I0}}[D_I(I_0, I_{gt}) - 1]^2 \qquad (11)$$

De manera similar, al final, el discriminador D_I ,no puede distinguir si la imagen sin marca de agua generada es verdadera o falsa, y el generador puede eliminar la marca de agua de manera efectiva.

(Liang, 2021) Proponen una red de eliminación de marcas de agua a través de Self-calibrated Localization and Background Refinement (SLBR), que consiste en una etapa gruesa y una etapa de refinamiento. En la etapa gruesa, adoptaron una arquitectura U-Net con enlaces de salto que conectan funciones de codificador y decodificador, específicamente emplean una estructura de codificador de bloque y decodificador de bloque. La localización y la eliminación de marcas de agua son tratados como dos tareas, que comparten los cinco bloques codificadores y el primer bloque decodificador. Pero tienen tres bloques decodificadores separados, que forman la rama del decodificador de máscara y el decodificador de fondo sucursal por separado. En la rama del decodificador de máscara, está equipado con un módulo de refinamiento de máscara auto calibrado (SMR) diseñado y asignado para indicar la posición de la marca de agua.

El Módulo de refinamiento de máscara auto calibrado (SMR) se caracteriza por una función donde Xm es el mapa de características usado para predecir la máscara de marca de agua $\widehat{M}$. Utilizan la pérdida de entropía cruzada binaria para hacer que $\widehat{M}$ se acerca a la marca de agua real de la máscara M, y se define en la ecuación 12.

$$\mathrm{L}_{mask} = -\sum_{i,j}(M_{i,j} log\widehat{M}_{i,j} + (1 - M_{i,j}) log(1 - \widehat{M}_{i,j})) \qquad (12)$$

Donde $M_{i,j}$ (resp., $\widehat{M}_{i,j}$) es la (i, j)- ésima entrada en M (resp., $\widehat{M}$). Primero aplican esta máscara estimada aproximadamente $\widehat{M}$ al mapa de características Xm para agrupar el vector de características promediado xm. Una vez que se realiza la evaluación anterior se puede aplicar la misma perdida para

supervisar el mapa de afinidad usando $\widehat{M}'$. Se define por la ecuación 13.

$$L'_{mask} = -\sum_{i,j}(M_{i,j}\, log\, \widehat{M'}_{i,j} + (1 - M_{i,j}) log(1 - \widehat{M'}_{i,j})) \quad (13)$$

En la que $\widehat{M}'i, j$ es la (i, j)- ésima entrada en $\widehat{M}$.También se diseñó un módulo de mejora de fondo guiado por máscara (MBE) para guiar el flujo de información, desde la rama de decodificación de la máscara a la rama de decodificación de fondo, en cada uno de esos módulos se concatena la máscara de salida $\widehat{M}'$ del módulo SMR correspondiente con las características del bloque decodificador previo y el salto de conexión. Después se aplica una capa de convolución de 3 × 3 a la característica concatenada para generar una característica residuo, la cual es añadida al fondo de la característica de entrada. Se denota la imagen de fondo generada como $\widehat{I_C}$, que se espera que sea cercana a la imagen libre de marca de agua verdadera, usando la pérdida $L1$, viene dado por la ecuación 14.

$$L^c_{bg-L_1} = \| I - \hat{I}^c \|_1 \quad (14)$$

Ahora para la etapa de refinamiento se utiliza un módulo de fusión de características de nivel cruzado (CFF), se realiza un sobre muestreo de la característica del encoder de alto nivel, al mismo tamaño de diferentes características de encoder de bajo nivel, posterior a esto se concatena la característica del encoder de alto nivel con sobre muestreo, con cada característica del encoder de bajo nivel, y se aplica bloques residuales apilados a cada una de las características del encoder, definido por la ecuación 15.

$$L^r_{bg-L_1} = \| I - \hat{I}^r \|_1 \quad (15)$$

Para garantizar aún más la calidad de la imagen sin marca de agua generada,

emplean la pérdida de percepción, basada en VGG16, preentrenado en ImageNet. La pérdida de percepción viene dada por la ecuación 16.

$$L_{bg-vgg} = \sum_{k\in 1,2,3} \|\Phi^k_{vgg}\left(\hat{I}^r\right) - \Phi^k_{vgg}(I)\|_1 \quad (16)$$

En el que $\Phi kvgg\,(\cdot)$ significa el mapa de activación de la k-ésima capa en VGG16.

Finalmente, recolectan las pérdidas en la etapa gruesa y la etapa de refinamiento, lo que lleva a la pérdida total, definido por la ecuación 17.

$$L_{all} = L^{c}_{bg-L_1} + L^{r}_{bg-L_1} + \lambda_{vgg} L_{bg-vgg} + \lambda_{mask}(L_{mask} + L'_{mas}) \quad (17)$$

El modelo planteado en este articulo pueden localizar la marca de agua y recuperar la imagen sin marca de agua simultáneamente. [Liang,2021]

Por otra parte**, (Li, 2021)** propone una red de detección y eliminación de marcas de agua visibles que combina la convolución parcial y la convolución ordinaria. Incluye 4 redes: una red de detección, una red de reconocimiento, una red de eliminación preliminar basada en convolución parcial y una red de optimización de rama de entrada dual. La propuesta emplea tres tipos de redes: DecNet, SegNet y WRNet. La red de detección de marcas de agua (denominada DecNet) emplea RetinaNet para detectar el área cubierta por el patrón de marca de agua, es definida por la ecuación 18.

$$I^{c}_{W}, I^{b}_{w} = DecNet(I_w) \quad (18)$$

Donde I^{c}_{w},I^{b}_{w} representan el área de alta correlación 'cubierta' y el área de baja correlación 'fondo' respectivamente. La red de segmentación de marcas de agua (denominada SegNet) utiliza una red convolucional completa multitarea (MFCN), para marcar el patrón de marca de agua a nivel de píxel, la cual se basa en una arquitectura VGG-16 totalmente convolucional. Tiene dos ramas de salida; una rama genera la etiqueta de superficie del patrón de marca de agua, mientras que la otra genera la etiqueta de límite del patrón para distinguir con precisión las áreas con marca de agua y sin marca de agua. Las salidas sin procesar de ambas ramas, denominadas gráfico de probabilidad de superficie y gráfico de probabilidad de límite, se umbralizan y se cruzan para producir un gráfico de máscara binaria formado por la ecuación 19.

$$I^{c}_{m} = SgeNet(I^{c}_{w}) \quad (19)$$

Donde I^{c}_{m} es la máscara de marca de agua correspondiente a I^{c}_{w}.

La red de eliminación de marcas de agua (denominada WRNet) incluye una red basada en convolución parcial (denominada PCNet). En esta parte, se emplea convolución parcial para restringir la operación de esta solo a los pixeles útiles. Ésta se define por la ecuación 20.

$$I^{c}_{r,p} = PCNet(I^{c}_{m}, I^{c}_{w}) \quad (20)$$

En caso de presentar un marcado de agua transparente, se emplea una red de sucursales de doble entrada (denominada como DBNet). Aquí, se usa una arquitectura U-Net, que puede conservar la posición, la textura y otras características de bajo nivel en la imagen de salida. DBNet se define por la ecuación 21.

$$I_{r,y}^{c} = DNet\left(I_{m,}^{c} I_{w}^{c}\right) \tag{21}$$

WRNet es la combinación de la red DBNet y PCNet. Esta red puede generar imágenes más realistas y detalladas y se define por la ecuación 22.

$$I_{r}^{c} = WRNet(I_{w}^{c}) \tag{22}$$

Donde I_{r}^{c} es la misma que la previamente mencionada $I_{r,y}^{c}$.Este método puede eliminar eficazmente las marcas de agua de diferentes colores y transparencias. (Li, 2021)

En el trabajo de **(Liu, 2021)**, se propone un generador de dos etapas, llamado Watermark Network (WDNet), donde la primera etapa tiene como objetivo predecir una descomposición aproximada de toda la imagen con marca de agua, y la segunda etapa, localiza el área con marca de agua para refinar los resultados que la remueven. Este método se apoya de un marco cGAN y redes profundas. Se construye un conjunto de datos llamado CLWD, que contiene principalmente marcas de agua de colores. El rendimiento superior de WDNet está estrechamente ligado con este conjunto de datos. Para una buena eliminación de marcas de agua, se puede usar una formula inversa de descomposición tal que pueda descomponer marcas de agua de imágenes que las contengan.

Modelo de descomposición de imagen con marca de agua: Una imagen X con marca de agua se obtiene normalmente superponiendo una marca de agua W a una imagen natural Y en las áreas de interés. La relación se envía entre los píxeles con marca de agua $X(p)$ y los píxeles sin marca de agua píxeles $Y\ (p)$.La relación se se define por la ecuación 23:

$$\mathrm{X(p)} = \alpha\mathrm{(p)W(p)} + \left(1 - \alpha\mathrm{(p)}\right)\mathrm{Y\ (p)} \tag{23}$$

Donde $p = (i, j)$ representa la ubicación del píxel en la imagen, $\alpha(p)$ es una opacidad que varía espacialmente, nombrada máscara alfa, utilizado en el procesamiento de imágenes. De acuerdo con la Ec. 23, si $\alpha(p) = 1$ en todas partes, X se reduce a W; en caso contrario si $\alpha(p) = 0$ en todas partes, X es igual a Y. Se debe obtener una imagen sin marca de agua Y de su contraparte de imagen con marca

de agua X. Considerando la ecuación 23, dados W y α,se puede invertir trivialmente el proceso de sintetizar una imagen con marca de agua a través de la operación por píxel, definida por la ecuación 24.

$$Y(p) = \frac{X(p) - \propto (p) W(p)}{1 - \propto (p)} \tag{24}$$

Teniendo en cuenta que las áreas sin marca de agua se mantienen igual en X y Y. Al construir tal modelo de descomposición dentro de las redes neuronales, considerando únicamente las áreas con marca de agua, ahorra capacidad de aprendizaje y es potencialmente útil para el resultado. Por lo tanto, se introduce una máscara de marca de agua $M(p) \in \{0, 1\}$ para ayudar a la fase de descomposición. $M(p) = 1$ significa el píxel p en áreas con marca de agua. Fusionando $M(p)$ con la ecuación 24, la imagen sin marca de agua se obtiene mediante la ecuación 25.

$$Y_o(\mathrm{p}) = \mathrm{M(p)} \cdot \mathrm{Y\,(p)} + \big(1 - \mathrm{M(p)}\big) \cdot \mathrm{X(p)} \tag{25}$$

Marco de eliminación de marca de agua: Toda la red incorpora un generador y un discriminador; el generador, denominado Watermark-Decomposition Network (abreviado como WDNet), utiliza el modelo de descomposición de marcas de agua, para generar la imagen sin esta y el discriminador predice parches.

Red de descomposición de marca de agua (WDNet): La arquitectura de un generador WDNet, tiene como entrada una imagen con marca de agua X. WDNet tiene como objetivo traducir la imagen con marca de agua a la imagen sin esta correspondiente. La eliminación directa de marcas de agua de una imagen contiene implícitamente dos procedimientos:

1) detección de la región rugosa de la marca de agua.
2) Eliminación de ruido detallada de la región con marca de agua en píxeles.

Teniendo esto en cuenta, se diseña una red de dos etapas donde cada etapa tiene como objetivo lograr uno de los procedimientos anteriores. Por lo tanto, WDNet se compone de dos subredes; una red de descomposición en el frente y una red de refinamiento al final.

La red de **descomposición** predice inicialmente un resultado aproximado de la descomposición de la marca de agua, a partir de la predicción de los parámetros $\alpha, \mathrm{W}, \mathrm{M}$. En consecuencia, primero se adopta una subred, denominada DecompNet, basada en la arquitectura U-Net, para predecir los parámetros de marca de agua $\widehat{\alpha}, \widehat{W}, \widehat{M}$ a partir de la entrada de imagen con marca de agua.

Específicamente, DescompNet contiene N capas convolucionales de muestreo descendente (la i-ésima) capa con 2^{5+i} canales, $i \in \{1, 2, \ldots, N\}$) y N capas convolucionales con sobre muestreo (la j-èsima capa con 2^{10-i} canales, $j \in \{1, 2, \ldots, N\}$) ($N = 4$). Cada mapa de características después de la i-èsima capa de codificación se une a la i-èsima capa de decodificación correspondiente, a través de concatenación profunda con la capa anterior. Se presenta una pequeña red para refinar la salida combinada $\hat{Y}^{pre}$. La red, llamada RefineNet, mira de cerca las partes inferiores de $\hat{Y}^{pre}$ y se enfoca en ajustar algunos píxeles para hacer el resultado final más agradable y suave. Específicamente, la primera entrada para RefineNet es una imagen preliminar sin marca de agua enmascarada Y_M^{pre}, que es igual a $\widehat{M} \cdot \hat{Y}^{pre}$. Otra entrada para RefineNet es un mapa de características F_U con 64 canales producidos por DecompNet. La arquitectura de RefineNet es bastante simple con solo 3 bloques residuales. Cada bloque residual conecta el mapa de características anterior con el actual a través de sumas y salidas, un mapa de características con 180 canales, lo que ayuda a facilitar el entrenamiento y evitar la pérdida de información. A través de la Ec. 25, se obtiene la última imagen libre de marcas de agua $\hat{Y}_o$, que será evaluada por el discriminador.

Discriminador: Emplea el discriminador basado en parches en la fase de entrenamiento. Concatena la imagen con marca de agua y la imagen con marca de agua eliminada como entrada, y las asigna a un mapa de características, que representa las probabilidades de que los parches de entrada sean reales (1 para real; 0 para falso).

Función de pérdida: La función de pérdida para el método propuesto es una suma ponderada de la pérdida de contenido, pérdida contradictoria, que se define por la ecuación 26.

$$L^* = arg \min_G \max_D L_{adv}\ (G, D) + L_{con}\left(\hat{Y}_o, \widehat{M}, \widehat{W}, \widehat{\propto}\right) \quad (26)$$

Durante el entrenamiento, el generador G es entrenado para minimizar el término objetivo adversario contra el discriminador D, que es entrenado para maximizar tal contraste en el término. Por lo tanto, la pérdida de contenido también sirve como una parte importante de L^*. La función de pérdida de contenido se expresa con la ecuación 27.

$$L_{con}\left(\hat{Y}_o, \widehat{M}, \widehat{W}, \propto\right) = \lambda_1 L_1\left(\hat{Y}_o\right) + \lambda_2 L_{per}\left(\hat{Y}_o\right) + \lambda_3 L_1\left(\widehat{M}\right) + \lambda_4 (L_1\left(\widehat{W}\right) + L_1(\widehat{\propto}) \quad (27)$$

Donde $L_1(\hat{\theta})|L_{per}(\hat{\theta})$ se refiere a la L_1. o la pérdida perceptual L_1 entre el objeto generado $\hat{\theta}$ y la clasificación absoluta $\theta \cdot \lambda$ ayuda a equilibrar los diferentes términos de pérdida. Para calcular la pérdida perceptual L1, se usa las salidas de la capa relu2_2 de una red VGG-16 pre-entrenada, para representar las características aprendidas de $\widehat{Y_o}$ y Y , luego calcular su diferencia L_1.

RESULTADOS Y DISCUSIÓN

Para evaluar la eficiencia de los métodos antes mencionados se realizó una tabla de comparación con las métricas, PSNR y SSIM, estos datos fueron proporcionado en cada uno de los artículos antes mencionados y se encuentran en la Tabla1.

Tabla 1 Evaluación de la eliminación de marcas de agua visibles

Métodos	PSNR	SSIM
(Cheng, 2018)	30.86	0.914
(Jiang, 2020)	44.92	0.997
(Liang, 2021)	40.88	0.988
(Li, 2021)	35.075	0.975
(Liu, 2021)	39.81	0.98865

Figura 1: Resultados de visualización de diferentes métodos

Método	Imagen con marca de agua	Imagen sin marca de agua
(Cheng, 2018)		
(Jiang, 2020)		
(Liang, 2021)		
(Li, 2021)		

En resultados mostrados en la Figura1, se puede apreciar que, la propuesta de (Cheng, 2018)a pesar de utilizar una propuesta que consta de dos componentes: (1) detección de marcas de agua, que se formula como una tarea de detección de objetos. (2) eliminación de marca de agua. La estructura de la red resulta ser simple para la tarea designada, que conllevo a estos resultados. Por otro lado, la estructura propuesta por (Jiang, 2020), basada en U-Net y entrenada a través de pérdida ℓ2. Presenta resultado que son capaces de eliminar de forma eficiente las marcas de agua de color con colores similares al fondo, presentando resultados bastante interesantes. La propuesta de una red multitarea de dos etapas de (Liang, 2021), presenta la capacidad de localizar la marca de agua y recuperar la imagen sin marca de agua simultáneamente. Sin embargo, a pesar de su alto rendimiento, se puede notar en la imagen, aún tiene problemas para diferenciación entre la marca de agua y la imagen de fondo. La red de redes de dos ramas consecutivas con convolución parcial para la tarea de eliminación de marca de agua de (Li, 2021). Puede eliminar eficazmente las marcas de agua de diferentes colores y transparencia, además pueden

localizar la marca de agua y recuperar la imagen sin marca de agua simultáneamente. Por otra parte, la propuesta de (Liu, 2021), denominada WDNet, que utiliza el proceso de descomposición de marcas de agua y una red RefineNet para refinar los resultados. Presenta una capacidad de separar marcas de agua de las imágenes de entrada, que pueden ser útiles para crear más datos para el entrenamiento, y mejorar aún más el rendimiento de las pruebas.

CONCLUSIONES

En este trabajo se realizó una breve revisión bibliográfica acerca de métodos de remoción de marcas de agua visibles contenidas en imágenes digitales mediante técnicas de aprendizaje profundo (Deep learning). En términos generales, los trabajos analizados obtienen resultados muy competitivos en términos de calidad visual de las imágenes reconstruidas, que hipotéticamente se podría determinar continuarán incrementando su efectividad en cuestión de remoción conforme los avances en inteligencia artificial vayan siendo más sofisticados, particularmente los modelos generativos antagónicos. Esto por una parte resulta atractivo si el contenido visual de una imagen contiene algún patrón obstructivo que pudiese no permitir la correcta visualización del contenido de la imagen. Sin embargo, resulta de relevancia destacar lo preocupante de este tipo de herramientas en cuanto a protección de derechos de autor de imágenes digitales, dada su efectividad para vulnerar la propiedad intelectual. Será entonces todo un reto científico y tecnológico el poder desarrollar nuevos algoritmos de marcado de agua visible que hagan frente a este tipo de inteligencia artificial, considerándola como un ataque más del amplio repertorio de procesamiento de señales que promueven la remoción de marcas de agua visibles.

LISTA DE REFERENCIAS

Álvarez, M. &. (2020). Restauración de imágenes digitales mediante una marca de agua basada en la transformada Wavelet Discreta.

Cheng, D. L. (2018). Large-scale visible watermark detection and removal with deep convolutional networks. *In Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*.

Guerrero Citelly, J. F. (2018). Traductor de lenguaje de señas portatil por medio de reconocimiento de imágenes.

Jiang, P. H. (2020). Two-stage visible watermark removal.

Li, T. F. (2021). Visible Watermark Removal Based on Dual-input Network. . *ACM International Conference on Intelligent Computing and its Emerging Applications*, (pp. 46-52).

Liang, J. N. (2021). Visible Watermark Removal via Self-calibrated Localization and Background Refinement. *In Proceedings of the 29th ACM International Conference on Multimedia* , (pp. 4426-4434).

Liu, Y. Z. (2021). WDNet: Watermark-Decomposition Network for Visible Watermark Removal. . *In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* , (pp. 3685-3693).

Lubin, J. B. (2003). Robust content-dependent high-fidelity watermark for tracking in digital cinema. In Security and Watermarking of Multimedia Contents .

Orúe, A. B. (2002). Marcas de agua en el mundo real.

Pei, S. C. (2006). A novel image recovery algorithm for visible watermarked images. . *IEEE Transactions on information forensics and security, 1(4), 543-550.*

Pérez-Aguilar, D. R.-R.-P. (2021). Transfer learning en la clasificación binaria de imágenes térmicas Transfer Learning for Binary Classification of Thermal Images. Machine learning (ML), 550, 4.

Regazzoni, F. P. (2021). Protecting artificial intelligence IPs: a survey of watermarking and fingerprinting for machine learning. . *CAAI Transactions on Intelligence Technology, 6(2), 180-191.*

Sara, U. A. (2019). Image quality assessment through FSIM, SSIM, MSE and PSNR—a comparative study. . *Journal of Computer and Communications, 7(3), 8-18.*

Tanha, M. T. (junio de 2012). Una descripción general de los ataques contra las marcas de agua digitales y sus respectivas contramedidas. *En Actas Título: Conferencia Internacional sobre Seguridad Cibernética, G*.

Vargas, L. M. (2015). Marcas de agua múltiples para autentificación y detección de adulteraciones en imágenes digitales médicas.

Vargas, L. M. (2016). Marcas de agua: una contribución a la seguridad de archivos digitales. . *Revista de la Facultad de Ciencias Exactas, Físicas y Naturales, 3(1), 49-54.*

Verdeguer Gómez, J. (2021). Redes neuronales para la clasificación y segmentación de imágenes médicas.

Wang, Z. B. (2004). Image quality assessment: from error visibility to structural similarity. . *IEEE transactions on image processing, 13(4), 600-612.*

Xu, C. L. (2017). An automatic visible watermark removal technique using image inpainting algorithms. *Conference on Systems and Informatics (ICSAI) . IEEE.*