

# Style transfer para imágenes

Botero, Sebastian. Saavedra, Miguel Angel. Salazar, Gustavo.  
Universidad Autónoma de Occidente

*Resumen - Con la popularidad de las redes convolucionales, nuevas aplicaciones como style-transfer han surgido y captado la atención de la comunidad científica. Style-transfer es una técnica que mimetiza el estilo de una imagen a otra [1] a partir de un modelo convolucional VGG-19 pre-entrenado. Dicha técnica ha encontrado diversas aplicaciones como filtros en redes sociales, reconstrucción de imágenes y transformación del estilo de texto de una persona a otra [2]. En este trabajo se presentará un modelo de style-transfer basado en la red VGG-19 y matrices Gram con la intención de generar filtros de estilo para redes sociales de tal forma que el estilo de una imagen sea transferida a otra.*

**Key words:** *Style transfer, VGG-19, Image transformation, Convolutional Neural Networks.*

## I. PROBLEMÁTICA

La transferencia de estilo en imágenes es una problemática la cual busca transferir el “estilo” de una imagen de referencia a una imagen de contenido seleccionada como destino. Esto puede ser visto a nivel de aplicaciones de redes sociales, las cuales tienen la problemática de mantener un flujo constante de filtros nuevos para conservar el interés de sus usuarios, en contenido y novedad de su servicio [3]. Al seleccionar propiamente la imagen de estilo, es posible que la imagen de contenido reciba un conjunto de características definidas como el “estilo” de la imagen de referencia, por lo que podría hacer parecer que la foto fue tomada con distinta iluminación, día, clima, o incluso hacer parecer que la foto tiene el estilo artístico de diversas obras de arte [4].

Lo anterior ha permitido que esta técnica se lleve a distintas aplicaciones en el entorno cotidiano, como lo serían filtros de redes sociales, reconstrucción de imágenes, transformaciones de textos, e incluso generación de nuevas obras artísticas [2]. También se han visto aplicaciones para adaptar diferentes cambios

de colores y estilos en interfaces web o de apps, con la intención de tener una perspectiva diferente a nuevas paletas de colores, de construcción de la interfaz o brindar un enfoque artístico basado en obras reconocidas.

Estas técnicas hasta el momento están limitadas por la diversidad de las escenas/transferencias que puede manejar, o en la exactitud de replicar el estilo de la imagen de estilo. En este trabajo se busca conseguir un acercamiento de solución orientada al campo de fotografía donde la foto seleccionada tenga un cambio enfocado a una pintura y su estilo, utilizando redes neuronales convolucionales profundas. La solución planteada originalmente se construye a partir de matrices Gram para calcular todas las características que definen el estilo a transferir [5], con el fin de transformar fotografías en obras de arte que puedan posteriormente ser usadas en redes sociales.

## II. MARCO TEÓRICO

Esta sección describe el marco teórico del proyecto con la intención de definir conceptos claves para la implementación de style-transfer.

### A. Style transfer

Style-Transfer es una técnica de deep learning que permite aplicar el estilo de una imagen a otra. Esta técnica funciona separando el contenido del estilo de una imagen para así, fusionarlos en una imagen nueva. Inicialmente el style transfer es planteado por L. Gatys et al [6] aprovechando las capas convolucionales de redes ya entrenadas, dejando a un lado las Fully connected. Las capas convolucionales son utilizadas para recopilar información de estilo y contenido a través de correlaciones con matrices gram. Finalmente

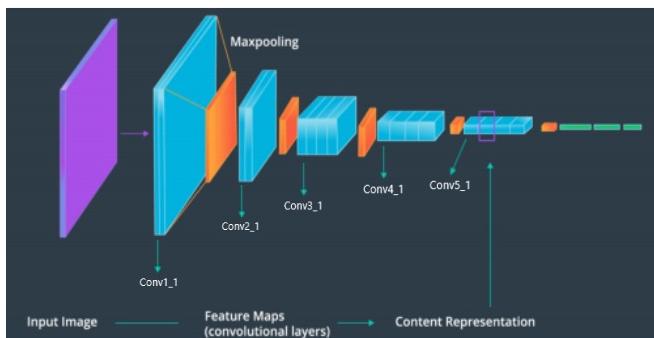
se transforma una imagen contenido o canvas transfiriendo el estilo deseado.

### B. Transfer Learning

El transfer learning es una técnica que se basa en utilizar los pesos de una red ya entrenada para obtener las características que esta entrega al pasarle una imagen [7]. En este trabajo se utilizará la red VGG-19 entrenada en el dataset de Imagenet [8] en donde la salida de sus capas convolucionales serán usadas para transformar la imagen generada, pero sin re-entrenar o cambiar los pesos ya establecidos en el modelo.

### C. VGG19

VGG19 [9], es una red neuronal convolucional propuesta por el grupo Visual Geometry de Oxford [10], esta red es una extensión de la red VGG16, y se considera la sucesora de la red AlexNet. Esta red utiliza varias ideas de sus predecesores pero mejora la precisión ya que contiene en los bloques intermedios exactamente 3 capas convolucionales más que su predecesora. Lo anterior permite aprovechar el poder de procesamiento extra de estas capas y así obtener una mayor precisión. Para style-transfer las características son obtenidas de los primeros mapas de características de cada bloque convolucional de uno de los cinco bloques. Adicionalmente, el contenido es extraído del segundo mapa de características del quinto bloque. La anterior información puede ser vista en la Fig. 1.



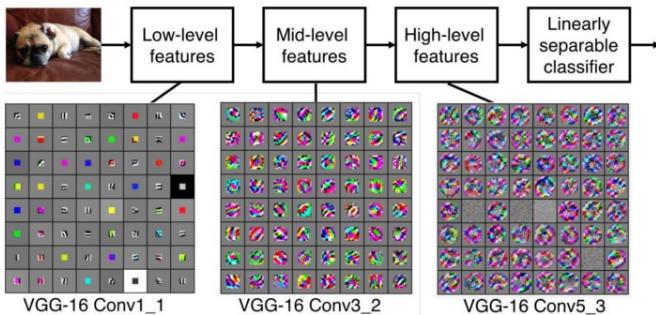
*Fig 1. Arquitectura VGG19 y capas utilizadas en style-transfer [11]*

### D. Imágenes de contenido y de estilo

En el marco de redes convolucionales, no es tan sencillo pensar en estilo y contenido. A medida que una CNN es entrenada, sus mapas de características son capaces de aprender características cada vez más y más complejas, yendo de lo general en sus primeras capas a lo específico en sus capas más profundas. Las capas de Max Pooling tienen la facultad de descartar detallada información espacial de la imagen de entrada, información que es innecesaria para tareas de clasificación. Por lo anterior, a medida que la red va más profundo, los mapas de características se empiezan a preocupar más acerca del contenido de la imagen (estructura) en lugar de textura, o color de pixeles. Las capas profundas de una CNN suelen ser referidas como representaciones de contenido. Por lo tanto, modelos convolucionales pre entrenados ya saben extraer el contenido de una imagen.

Ahora bien, para representar el estilo con un CNN, se debe usar un espacio de características que capture la información de textura y color de la imagen. Este espacio puede ser imaginado como correlaciones espaciales en una capa específica de la red. Entiéndase correlación como una medida de la cercanía o similitud entre dos o más variables [12]. En el contexto de CNN, es posible aplicar el concepto de correlación para medir qué tan fuerte es la relación entre cada característica detectada a las otras características en un mismo mapa convolucional. Por ejemplo, si en un mismo mapa de características hay colores y formas similares, entonces se podría considerar que estos son parte del estilo de la imagen.

La imagen presentada en la Fig. 2 da un ejemplo de cómo a medida que se va más profundo en una red convolucional VGG16, las características de la misma se hacen más complejas (contenido).



#### E. Gram matrix

La matriz Gram permite medir la correlación espacial entre los mapas de características de la red VGG y así, extraer el estilo de una imagen. Esta matriz se extrae a lo largo de diferentes mapas de características de la red ya que a medida que se va más profundo, es posible obtener representaciones de estilo en múltiples escalas.

Para computar la Gram Matrix de la imagen de estilo, suponga un mapa de características de dimensiones  $n_c, n_h, n_w$  donde  $n_c$  son el número de canales, y  $(n_h, n_w)$  son el alto y ancho de la imagen; respectivamente. El primer paso sería vectorizar o aplanar, las dimensiones  $(n_h, n_w)$  del mapa de características como un vector fila y construir una nueva matriz de dimensiones  $(n_c, n_h * n_w)$ . Cada fila de esta nueva matriz corresponde a un canal del mapa de características y cada fila es la información contenida en dicho canal.

Una vez se tiene esta versión desarrollada del mapa de características, se procede a hacer la multiplicación de este mismo por su transpuesta. En esencia, lo que se está haciendo es multiplicar las características de cada mapa para conseguir la matriz Gram. Esta operación trata cada valor en el mapa de características como una muestra individual, sin relación espacial a otros valores. Por lo tanto, la matriz Gram posee información no localizada sobre cada capa en el mapa de características. Esta información es muy importante ya

que es información que se mantendrá incluso si la imagen ha sido modificada en el espacio de imagen.

El resultado final es una Gram Matrix cuadrada de tamaño  $(n_c, n_c)$  cuyos valores miden la similitud entre diferentes capas (canales) de un mismo mapa de características. Es importante resaltar que el tamaño de la matriz Gram está relacionada únicamente al número de canales de un mapa de características y no al tamaño de la imagen [6].

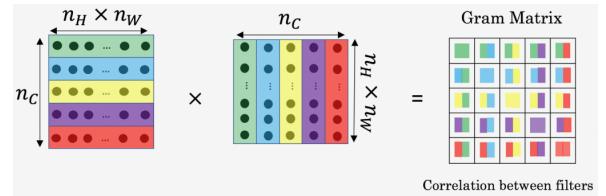


Fig 3. Funcionamiento de una Gram Matrix [14]

#### F. Función de Pérdida para estilo y contenido

El loss de contenido está definido como la diferencia entre las imágenes contenido y destino. La representación de contenido deseada será representada como  $C_c$  y la representación de contenido de la imagen destino se definirá como  $T_c$ . La función de pérdida será definida como el MSE (Error cuadrático medio) de las dos representaciones de contenido. Esta función de pérdida mide qué tan lejos se encuentran ambas representaciones de contenido. El objetivo del algoritmo es minimizar esta función de costo para que ambas imágenes sean lo más similares posible en contenido [6] [15].

La siguiente ecuación establece dicha relación:

$$L_{content} = \frac{1}{2} \sum (T_c - C_c)^2 \quad (1)$$

##### E.1 Función de pérdida de contenido.

La pérdida de estilo está definida como la distancia media al cuadrado entre las matrices gram de la imagen de estilo y la destino. En total se computan 5 pares de matrices Gram de los mapas de características mencionados anteriormente de la red VGG-19. Los

cinco pares son extraídos de la imagen de estilo y destino. Cabe resaltar que únicamente estos mapas de características son utilizados para computar el estilo.

La lista de estilos en la imagen de estilo se define como  $S_{s,i}$ , donde  $i \in [0,1,2,3,4]$  y para la imagen contenido como  $T_{s,i}$ . La constante  $a$  se define como un factor de normalización que tiene en cuenta el número de valores en cada capa convolucional e.g.  $[c*w*h]$  y  $w_i$  es un vector que contiene un peso para cada matriz gram de estilos. Los autores definieron el vector  $w_i$  con un valor constante de 0.2 para todas las capas, sin embargo, se hizo una pequeña modificación y se establecieron estos pesos como  $[1, 0.8, 0.5, 0.3, 0.1]$ . La función de pérdida de estilo se encuentra escrita a continuación.

$$L_{style} = a \sum_i w_i (T_{s,i} - S_{s,i})^2 \quad (2)$$

### E.2 Función de pérdida de estilo.

Finalmente, las dos funciones de pérdida anteriores se unen en un loss total de estilo y contenido como se muestra en la ecuación a continuación.

$$L_{Total} = \alpha (\sum_c (T_c - C_c)^2) + \beta (a \sum_i w_i (T_{s,i} - S_{s,i})^2) \quad (3)$$

### E.3 Función de pérdida de estilo.

Cabe resaltar que las constantes  $\alpha, \beta$  son utilizadas para aplicar un balance entre la pérdida de estilo y contenido, las cuales pueden tener valores muy diferentes.  $\alpha$  está definido como el peso de contenido y  $\beta$  como el peso de estilo. Generalmente  $\alpha, \beta$  son descritos como un ratio  $\alpha/\beta$  donde sus valores fueron definidos por los autores como  $\frac{\alpha}{\beta} = 1 * 10^{-3}, 1 * 10^{-4}, 1 * 10^{-6}$ . Para este caso particular se utilizará un ratio de  $1 * 10^{-6}$ . Menores valores de ratio significa que el estilo tendrá mayor efecto. La imagen a continuación ejemplifica lo mencionado anteriormente, tomada del paper original.



Fig 4. Cambios en la imagen final según el ratio escogido.

## III. METODOLOGÍA

En el paper original [6], style-transfer fue hecho a partir de una red VGG 19 pre entrenada. En este problema se utiliza únicamente las capas convolucionales. Los autores definieron que el estilo de una imagen puede ser extraído de los primeros mapas de características de cada bloque convolucional, es decir, de las capas 'Conv1\_1', 'Conv2\_1', 'Conv3\_1', 'Conv4\_1', 'Conv5\_1'. Los mapas anteriormente mencionados fueron los que dieron mejores resultados a la hora de transferir el estilo de una imagen a otra según los autores. De igual modo, los autores definieron que el contenido sería extraído del mapa de características 'Conv4\_2' de la red VGG19 como se ilustra en la Fig. 1.

Inicialmente se reciben dos imágenes, una de contenido(C), y la otra de estilo(S) usada para estilizar la imagen destino o target (suele ser igual a la contenido). Posteriormente, se realiza preprocessamiento de la imagen para resaltar más las características a estilizar (bordes, colores, distribución de histograma, etc).



Fig 5. Representación de las tres imágenes de entrada(C, S, T).

Se carga el modelo ya entrenado VGG-19 sin su cabeza clasificadora y se inicializa aleatoriamente la imagen que va a ser generada(T) o se clona de la contenido, ya que, según los autores  $C \approx T$ .

Posteriormente se toman las 3 imágenes C, S, T como se ve en la Fig. 5, y se pasan de forma separada a través de la red convolucional, para extraer los mapas de características de estilo y contenido mencionados anteriormente.

De esta lista de mapas de características se utilizan las características de estilo para construir las matrices gram las cuales contendrán información del estilo y será utilizada en la función de costo. Para el contenido, el mapa de características es usado directamente de la salida de las capas convolucionales a la función de costo.

Estos valores son enviados a la función de costo Ec. 3, que está compuesta por dos partes aplicada al contenido y estilo deseado(C y S) junto con la imagen destino(T). La imagen T es pasada repetidamente por la red VGG para extraer el contenido y estilo de la misma y a través de la minimización de la función de pérdida compuesta se empiezan a modificar los píxeles de la imagen destino hasta que la imagen destino tiene tanto el estilo como el contenido deseado. La CNN no está siendo entrenada para clasificar una imagen, sino para cambiar sólo la imagen generada, actualizando su apariencia hasta que su representación coincide con la de la imagen de contenido y estilo. La CNN se utiliza para extraer características y minimizar el Loss de contenido y de estilo.

En este caso, la función de costo para el contenido se definirá como la diferencia media al cuadrado entre las dos representaciones (Ec.1). Esta función de pérdida mide la distancia entre ambas representaciones. Como el algoritmo pretende crear la mejor imagen generada, el objetivo es minimizar este error.

La representación del estilo de la imagen, se basa en buscar correlaciones entre las características en las capas individuales de la red. Las similitudes incluirán los colores y texturas generales que se encuentran en esa capa. Las representaciones de estilo se extraen de ciertas capas convolucionales como las mencionadas con anterioridad y, a medida que la red comienza a profundizar, es posible obtener una representación de estilo multiescala de la imagen de entrada. Las

correlaciones en cada capa de la CNN son calculadas utilizando la matriz de Gram para la imagen de estilo y la imagen generada. Dicha matriz funciona como una medida del ‘estilo’ en el contexto de style-transfer.

De esta manera la imagen generada va tomando forma, mezclando contenido y estilo para entregar después de varias iteraciones un resultado similar al presentado en la Fig. 6.

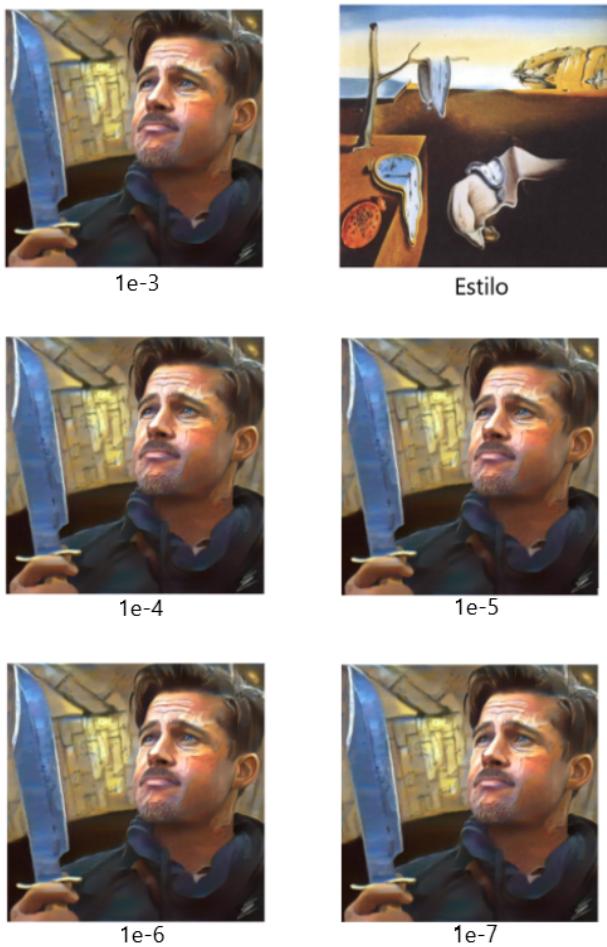


*Fig. 6 Style transfer con imagen Contenido, Style y Target [5].*

#### IV. RESULTADOS

En esta sección se presentan los resultados obtenidos y experimentos realizados para validar el funcionamiento de style-transfer.

Para validar el funcionamiento de style-transfer y el efecto de sus parámetros se varió el ratio  $\frac{\alpha}{\beta}$  con los valores  $[1e^{-3}, 1e^{-4}, 1e^{-5}, 1e^{-6}, 1e^{-7}]$  respectivamente. Cabe resaltar que los pesos de estilo  $w_i = [1, 0.8, 0.5, 0.3, 0.1]$  no se modificarán a lo largo de las pruebas. Este ratio funciona de tal manera de que a menor valor de  $\frac{\alpha}{\beta}$ , mayor es la influencia de la imagen de estilo. En la prueba se utilizó la misma imagen de estilo y contenido (Fig. 10(a)), como target se parte de la imagen contenido. Se itera un total de 1000 veces y los resultados son presentados en la Fig. 7.

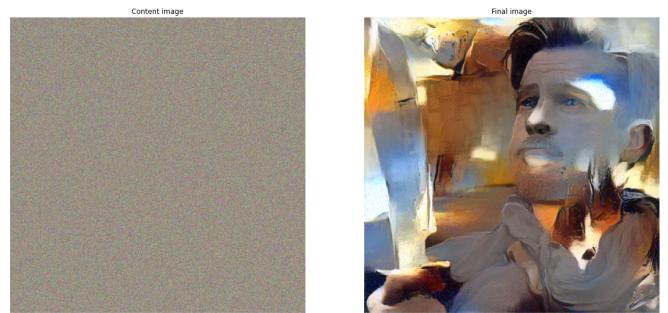


*Fig 7. Imágenes resultantes con la variación del ratio.*

De esta prueba se evidencia un cambio mínimo al variar el ratio  $\frac{\alpha}{\beta}$ , ya que, las imágenes generadas bajo diferentes ratio pero mismo número de iteraciones se ven casi idénticas como puede verse en los resultados mostrados en la Fig 7. Los cambios más significativos pueden ser apreciados en el cuchillo del actor donde se perciben formas diferentes a medida que se varía el valor del ratio. Sin embargo, hay que tener cuidado a la hora de variar este ratio, ya que, un valor muy pequeño podría desaparecer completamente los gradientes al propagarse a través de la red o en su defecto, explotarlos. Lo anterior puede entenderse al analizar la Ec. 3. De esta primera prueba podemos concluir que el ratio  $\frac{\alpha}{\beta}$  tiene un impacto sobre la imagen final variando

la influencia del estilo sobre el destino pero no de manera significativa.

Ahora bien, ya que en la anterior prueba se partió de la imagen contenido, se validará la robustez del algoritmo reconstruyendo una imagen desde cero. Para esto se genera un target(T) de ruido blanco y se modifica el ratio de tal manera que  $\frac{\alpha}{\beta} = 1$  con la intención de pasar contenido y estilo de manera equitativa a la imagen destino. La imagen contenido y estilo utilizadas fueron las mismas que las de la Fig. 7. El resultado obtenido puede ser visto en la Fig. 8.



*Fig 8. Imagen inicial a la izquierda (ruido blanco) y resultado final después de style-transfer.*

El anterior resultado nos permite validar como el modelo es capaz de reconstruir una imagen desde cero con un estilo basado en la pintura “the persistence of memory”. Adicionalmente, se puede ver como la imagen cuenta con gran parte de la estructura definida de la imagen de contenido pero la textura final se encuentra definida en gran medida por la imagen de estilo.

Cabe resaltar que para obtener el anterior resultado se requirieron un total de 10000 iteraciones o 15 minutos de cómputo en google collab. A pesar de que la imagen es muy llamativa visualmente, se vuelve muy costoso computacionalmente generarla y por lo tanto, se recomienda partir de la imagen de contenido como recomienda el autor.

Otro experimento llevado a cabo fue pensado en ¿qué pasa si se cambia la imagen estilo por la contenido? La Fig. 9 presenta el resultado obtenido.



*Fig 9. Invirtiendo estilo y contenido*

En esta se puede apreciar como la textura de las olas se mantiene a lo largo de la imagen pero su paleta de colores pasa a un color pastel. Por otra parte, se puede ver como el fondo de la escena empieza a tener el tapiz de la imagen de “estilo” escogida Fig. 10-b, presentando un borroneo de la imagen haciendo que los bordes estén menos resaltados. Finalmente, la figura de la persona no es resaltada en la pintura, y es únicamente el estilo del tapiz el que adquiere fuerza sobre esta imagen, pero aun así logra percibirse que el modelo intenta dibujar una mano o parte de ella en la zona central, y es muy posible que con muchas más iteraciones se hubiera esbozado parte de la silueta de esta persona sobre la pintura.

Por último, se establecieron como parámetros  $\frac{\alpha}{\beta} = 1e^{-6}$  y un total de 2000 iteraciones para generar imágenes visualmente llamativas para redes sociales. De cada grupo de imágenes la de la izquierda corresponde al contenido(C), la inferior al estilo (S) y la de la derecha al resultado final(T). Para este experimento se partió de la imagen de contenido para generar la imagen destino. Los resultados pueden ser vistos en la Fig. 10.



*b.*



*c.*

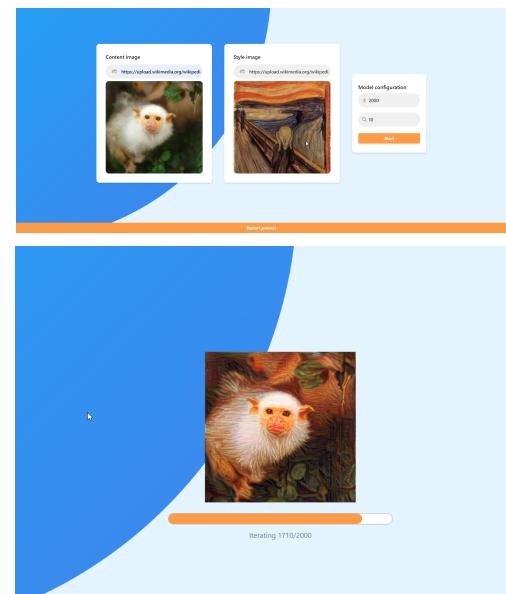


*d.*

*Fig 10. Imágenes resultantes variadas.*

Las anteriores imágenes muestran una variada gama de resultados o filtros obtenidos con diferentes estilos y contenidos. De aquí es posible validar que el algoritmo funciona de manera correcta y basado en el estilo seleccionado, es posible generar imágenes muy llamativas visualmente y con alto potencial de ser implementadas como filtros para redes sociales.

Para esto mismo se realizó una plataforma web donde es posible generar imágenes a partir de una imagen de estilo y una de contenido, mostrando los cambios en tiempo real, esto con el fin de validar los cambios que presenta la imagen y así poder habilitar a los usuarios de redes sociales una forma fácil y rápida de estilizar las fotografías y así validar la solución a la problemática planteada.



*Fig 11. Pruebas del algoritmo en un ambiente web.*

Finalmente, gracias a la plataforma web, se puede validar el desempeño del algoritmo en un caso de uso real, permitiendo medir tiempos de respuesta, concurrencia y posibles oportunidades de mejora que se presenten para mejorar la respuesta del mismo en ambientes de alto desempeño.

## V. CONCLUSIONES

De los experimentos realizados se puede concluir que la implementación de style-transfer es satisfactoria y permite realizar la tarea en cuestión de transferir el estilo de una pintura a una persona de manera robusta y sin importar el estado inicial de la imagen destino.

Con las pruebas realizadas en distintos ambientes, se puede concluir que a mayor cantidad de iteraciones se tiene mejores resultados, pero es muy costoso computacionalmente, por lo que es difícil mantenerlo en proyectos de alta concurrencia de usuarios.

Una de las técnicas que se pueden utilizar a partir del style-transfer es utilizando una imagen inicial como ruido blanco, lo que permite construir imágenes mucho más llamativas que conserven las características principales de la imagen de contenido y estilo al costo de muchas más iteraciones.

El uso de redes convolucionales para la tarea en cuestión demuestra el alto abanico de aplicaciones de estas redes ya que, la mayoría de veces su uso se ve limitado únicamente a clasificación y detección de objetos en imágenes.

## VI. REFERENCIAS

- [1] Y. Jing, Y. Yang, Z. Feng, J. Ye, Y. Yu and M. Song, "Neural Style Transfer: A Review," in IEEE Transactions on Visualization and Computer Graphics, vol. 26, no. 11, pp. 3365-3385, 1 Nov. 2020, doi: 10.1109/TVCG.2019.2921336.
- [2] Fu, Z., Tan, X., Peng, N., Zhao, D., & Yan, R. (2018). Style Transfer in Text: Exploration and Evaluation. Proceedings of the AAAI Conference on Artificial Intelligence, 32(1). Retrieved from <https://ojs.aaai.org/index.php/AAAI/article/view/11330>
- [3] "Engagement with Social Media and Social Media Advertising: The Differentiating Role of Platform Type", Taylor & Francis, 2021. [Online]. Available: <https://www.tandfonline.com/doi/full/10.1080/00913367.2017.1405754>. [Accessed: 07- Apr- 2021].
- [4] F. Luan, S. Paris, E. Shechtman and K. Bala, "Deep Photo Style Transfer", 2021.
- [5] T. Friedrich and S. Menzel, "Standardization of Gram Matrix for Improved 3D Neural Style Transfer," 2019 IEEE Symposium Series on Computational Intelligence (SSCI), Xiamen, China, 2019, pp. 1375-1382, doi: 10.1109/SSCI44817.2019.9002780.
- [6] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. arXiv preprint arXiv:1508.06576, 2015.
- [7] G. A. Salazar Gomez, N. D. Salazar and J. A. Lopez Sotelo, "Application of Transfer Learning for Object Recognition Using Convolutional Neural Networks", 2018 IEEE 1st Colombian Conference on Applications in Computation a Intelligence (ColCACI), Medellin, 2018, pp.1-6.
- [8] Deng, Jia & Dong, Wei & Socher, Richard & Li, Li-Jia & Li, Kai & Li, Fei-Fei. (2009). ImageNet: a Large-Scale Hierarchical Image Database. IEEE Conference on Computer Vision and Pattern Recognition. 248-255. 10.1109/CVPR.2009.5206848.
- [9] Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In ICLR, 2015.
- [10] 2021. [Online]. Available: [https://www.researchgate.net/profile/Li-Jia-Li/publication/221361415\\_ImageNet\\_a\\_Large-Scale\\_Hierarchical](https://www.researchgate.net/profile/Li-Jia-Li/publication/221361415_ImageNet_a_Large-Scale_Hierarchical). [Accessed: 07- Apr- 2021].
- [11]"Style Transfer using Deep Neural Network and PyTorch", Medium, 2021. [Online]. Available: <https://medium.com/udacity-pytorch-challengers/style-transfer-using-deep-nural-network-and-pytorch-3fae1c2dd73e>. [Accessed: 07- Apr- 2021].
- [12] Guo Y., Bai L., Lao S., Wu S., Lew M.S. (2014) A Comparison between Artificial Neural Network and Cascade-Correlation Neural Network in Concept Classification. In: Ooi W.T., Snoek C.G.M., Tan H.K., Ho CK., Huet B., Ngo CW. (eds) Advances in Multimedia Information Processing – PCM 2014. PCM 2014. Lecture Notes in Computer Science, vol 8879. Springer, Cham. [https://doi.org/10.1007/978-3-319-13168-9\\_26](https://doi.org/10.1007/978-3-319-13168-9_26)
- [13] [C. CNN and H. Kashiani, "Convolution layer in CNN", Stack Overflow, 2021. [Online]. Available: <https://stackoverflow.com/questions/48220598/convolution-layer-in-cnn/48280069>. [Accessed: 07- Apr- 2021].
- [14][子. note) and D. Transfer, "Deep Learning & Art: Neural Style Transfer - 子清观(lifelog and study note)", 子清观(lifelog and study note), 2021. [Online]. Available: <http://ziqingguan.net/index.php/2020/05/30/deep-learning-art-neural-style-transfer/>. [Accessed: 07- Apr- 2021].
- [15]H. Yessou, G. Sumbul and B. Demir, "A Comparative Study of Deep Learning Loss Functions

for Multi-Label Remote Sensing Image Classification", arXiv.org, 2021. [Online]. Available: <https://arxiv.org/abs/2009.13935>. [Accessed: 07- Apr- 2021].