



Deconvolución de una imagen satelital por medio del filtro de Wiener usando Python

A satellite image deconvolution through Wiener filter using Python

Duvan Alexander Robles Mondragón¹, Giselle Helena Toro Garay²

Para citar este artículo: Robles, D. y Toro, G. (2016). Deconvolución de una imagen satelital por medio del filtro de Wiener usando Python. *Revista de Topografía Azimut*, (7), 7-14.

Recibido: 2-septiembre-2016 / **Aceptado:** 10-octubre-2016

Resumen

La toma de una imagen satelital siempre está sometida a fuentes de degradación, esto por condiciones físicas o de interferencia; algunas de estas degradaciones son mínimas, por lo cual no se toman en cuenta; sin embargo, otras dificultan los procesos de interpretación visual y tratamiento digital de las imágenes. Para dar solución a ello, existen varios filtros que permiten reconstruir la señal por medio de modelos matemáticos, esto se conoce como deconvolución. Entre los filtros más utilizados se tiene: el filtro inverso constreñido, el de máxima entropía y el de Wiener. Este último, en su versión paramétrica con valores adecuados, puede aplicarse tanto a imágenes con ruido como borrosas; busca encontrar una imagen restaurada que se acerque a la imagen original. El presente artículo evalúa la efectividad de la aplicación del filtro de Wiener en la reconstrucción de la señal, aplicando una degradación conocida sobre una imagen WorldView-2 de Bogotá, Colombia, utilizando el lenguaje de programación Python la cual ofrece simplicidad y baja carga computacional en los cálculos realizados.

Palabras clave: convolución, deconvolución, filtro de Wiener, imagen satelital, Python, reconstrucción de señal, scikit-image.

Abstract

Taking a satellite image, is always subject to degradation sources this physical conditions or interference, some of these degradations are minimal and therefore not taken into account; however other difficult processes visual interpretation and digital image processing, to solve; there are several filters that allow to reconstruct the signal by means of mathematical models, this is known as deconvolution. Among the most commonly used filters it has: the constrained inverse filter, maximum entropy and Wiener. The latter in its parametric version with proper values can be applied to images with noise and blur. This filter seeks to find a restored image that closely matches the original image. This article assesses the effectiveness of applying the Wiener filter in the reconstruction of the signal, applying a known degradation on an image WorldView-2 Bogota Colombia, using the Python tool which offers simplicity and low computational load calculations made.

Keywords: convolution, deconvolution, python, satellite image, scikit-image, signal reconstruction, Wiener filter.

¹ Terra-Ing G&A. Correo electrónico: daroblesm@correo.udistrital.edu.co

² Terra-Ing G&A. Correo electrónico: daroblesm@correo.udistrital.edu.co

INTRODUCCIÓN

El análisis digital de imágenes tiene como objetivo cuantificar las propiedades de los objetos presentes, para generar distintas aplicaciones (Chávez, 2010). El tratamiento digital de las mismas busca el mejoramiento de las condiciones para un mejor análisis, creciente por las necesidades de precisión (Medina, *et al.*, 2014), esto implica la transformación de la imagen de una forma u otra. En general, son posibles dos tipos distintos de procesamiento: el primero, implica una forma de transformación que traerá como resultado una nueva imagen deseable; el segundo, consiste en un resultado que no es una imagen pero puede tomar la forma de una decisión, una abstracción, o una parametrización (Stokham, 1972).

Las imágenes están sometidas a cierta degradación implícita por las condiciones en que son capturadas, entre ellas condiciones atmosféricas o de resolución de los satélites (Chávez, 2010). Existen múltiples métodos de mejoramiento, entre ellos la fusión de imágenes (Choi, *et al.*, 2004; Medina, *et al.*, 2013) y la aplicación de filtros de reconstrucción de la señal (Khoumri, *et al.*, 1998).

La reconstrucción de la señal permite aproximar y mejorar condiciones como el desenfoque sobre una imagen, obteniendo así un producto viable para el análisis posterior. Esta reconstrucción se conoce como deconvolución de la señal, para la cual existen varios métodos de realización como las técnicas algebraicas de restauración y los filtros (Chávez, 2010).

El presente artículo simula una degradación sobre una imagen Worldview-2, formada a partir de un filtro espacial de suavizado, para, posteriormente, aplicar el filtro de Wiener en la reconstrucción de la señal.

Los resultados fueron evaluados con el índice de calidad universal que evidencia la distorsión en tres factores (Wang y Bovik, 2002) entre la imagen original y la deconvolución. El procesamiento fue realizado con el lenguaje de programación Python dado que es orientado a objetos, claro y potente (Westra, 2010), lo que disminuye el tiempo de ejecución y procesamiento de datos incluso en imágenes de gran tamaño.

FILTRO ESPACIAL: SUAVIZADO

El concepto de filtrado en el procesamiento de la señal, tiene sus inicios en el uso de la llamada transformada de Fourier en el dominio de la frecuencia (González, *et al.*, 1996). Para la realización de filtros espaciales se usa un proceso llamado convolución espacial, este es un método matemático utilizado en procesamiento y análisis de señales y se conoce también como filtro de respuesta finita al impulso (Aldalur y Santamaría, 2002). La implementación del filtrado espacial consiste en elegir una máscara de filtro, también conocida como kernel, ventana o plantilla, este es una matriz cuadrada generalmente de dimensión 3x3 o 5x5 (Aldalur y Santamaría, 2002), y moverla punto a punto en una imagen; en cada punto (x, y), la respuesta es dada por una suma de productos de los coeficientes y los correspondientes píxeles de la imagen de la zona abarcada (Mehl y Peinado, 1997).

“Los elementos son los valores digitales de brillo de los píxeles del núcleo y los valores constantes son los pesos, llamados coeficientes de convolución” (Aldalur y Santamaría, 2002). Considerando una matriz de 3x3, la convolución se calcula como:

$$O(x, y) = w(-1, 1) + f(x-1, y-1) + \\ w(-1, 0) + f(x-1, y) + \dots + \\ w(0, 0) + f(x, y) + \dots + w(1, 0) + f(x+1, y) + \dots \\ w(1, 1) + f(x+1, y+1) \quad (1)$$

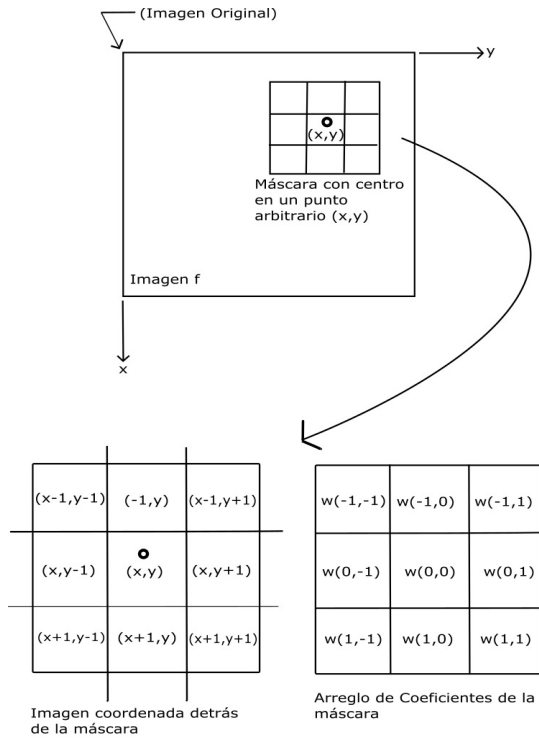


Figura 1. Implementación del filtro espacial

La Figura 1 muestra gráficamente la matriz de la máscara para el ejemplo de tamaño 3x3, que realizará la convolución sobre la imagen original, haciendo un acercamiento de la misma donde se exponen los coeficientes de la máscara a aplicar y el cambio sobre la imagen bajo la convolución, resumiendo así el proceso de filtrado espacial (González, *et al.*, 1996). Estos filtros tienen el efecto de difuminar los bordes de la imagen (González, *et al.*, 1996)

Filtro de Wiener

Este método de filtrado busca encontrar una imagen restaurada que se aproxime en gran medida a la original, intenta particularizar explícitamente este grado de fidelidad con que la estimación se parece a la imagen base (Chávez, 2010).

Es necesario definir un criterio de minimización global

$$e^2 = E\{[f(r) - \hat{f}(r)]^2\} \tag{2}$$

Con esto se espera minimizar el valor de la diferencia al cuadrado entre \hat{f} y f , razón por la cual a este filtro se le conoce como filtraje de mínimos cuadrados, también es conocido como restauración estocástica de imágenes. La ecuación (3) muestra la aplicación numérica del filtro de Wiener.

$$L_w(i,j) = \frac{H^*(i,j)}{|H(i,j)|^2 + S_{vv}(i,j)/S_{ff}(i,j)} \tag{3}$$

Donde S_{vv} y S_{ff} son la densidad espectral de la imagen degradada y original, mientras que H^* la función de modulación; los índices (i,j), enteros, denotan que la ecuación es válida en el dominio discreto (Chávez, 2010).

Índice de calidad universal

Este modelo de índice de calidad identifica cualquier distorsión como una combinación de tres factores: pérdida de correlación, distorsión de luminancia y contraste de distorsión (Medina, *et al.*, 2014; Wang y Bovik, 2002). La ecuación (4) muestra su cálculo.

$$Q = \frac{\sigma_{xy}}{\sigma_x \sigma_y} \cdot \frac{2\bar{x}\bar{y}}{(\bar{x})^2 + (\bar{y})^2} \cdot \frac{2\sigma_x \sigma_y}{\sigma_x + \sigma_y} \tag{4}$$

Python y scykit-image

Python es un lenguaje multiplataforma de alto nivel que es adecuado para una amplia variedad de tareas de programación, es un lenguaje de tipo dinámico fácilmente legible y que soporta orientación de objetos (Westra, 2010). Python se ha sido utilizado ampliamente en sistemas de

programación web (Mesa, 2012), aplicaciones de escritorio, juegos, programación científica (Nogueras, 2008) y bigdata (Nogueras, 2008).

Existen diferentes scripts en el área de procesamiento de imágenes, se encuentran librerías como lo son numpy, para manejo numérico y operaciones complejas, matplotlib la cual permite hacer salidas gráficas, scikit-image (Van Der Walt, *et al.*, 2014) que maneja operaciones sencillas y complejas de tratamiento de imágenes, etc.

METODOLOGÍA

Área de estudio

El área de estudio fue seleccionada por la presencia de objetos geográficos espaciales en los que se hacía posible la comparación entre la degradación y la aplicación del filtro de Wiener, está localizada en la ciudad de Bogotá, Colombia. La imagen original fue capturada por el satélite WorldView-2 el 15 de noviembre del año 2011, se tomó un recorte de 1024 x1024 píxeles.

La Figura 2 muestra el área de estudio escogida en la que se evidencian diferentes coberturas, entre ellas cuerpos de agua, zonas urbanas, áreas descubiertas de cultivos y zonas verdes. Se ofrece adicionalmente un acercamiento a dos zonas dentro de la imagen.

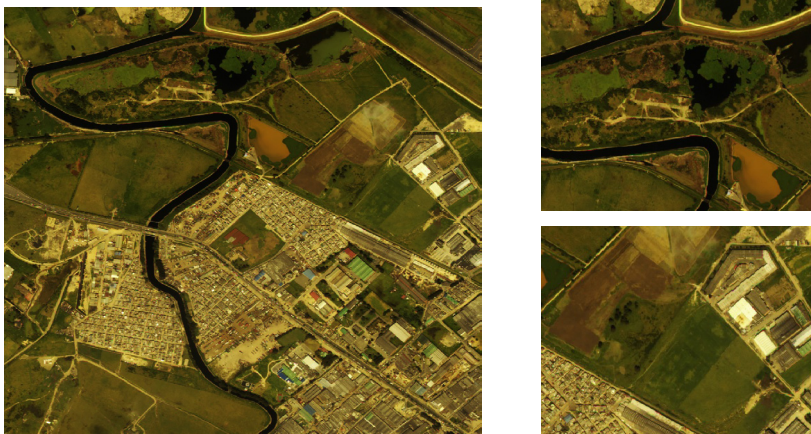


Figura 2. Área de estudio

Aplicación del filtro espacial

Se realizó la aplicación del filtro espacial, para generar el ruido sobre la imagen, utilizando un kernel de tamaño 5x5 de tipo ponderado en el que se concibe el desenfoque de la imagen con el mismo peso para todos los píxeles analizados. Por tal razón, para un mejor desarrollo computacional, todos los valores son 1 y están divididos entre el tamaño de la matriz, m columnas por n filas, para el caso un valor de 25.

Aplicación del filtro de Wiener

El filtro de Wiener se aplicó sobre la imagen resultante del punto anterior con la intención de reconstruir la señal y obtener una imagen lo más cercana posible a la imagen original (Figura 2).

Evaluación de resultados

Para la evaluación de la efectividad del procedimiento realizado, se implementó el índice de calidad universal (ecuación 2) comparando la imagen original, en niveles de gris, con las imágenes resultantes tanto de la aplicación del filtro espacial suavizado como del filtro de restauración de Wiener, se consideró el ideal dada su evaluación de la distorsión en las imágenes.

RESULTADOS Y DISCUSIÓN

Los resultados de la aplicación de los diferentes procesos se desarrollaron y visualizaron sobre la IDLE de Python, esta herramienta permite ejecutar scripts y visualizar cada una de las imágenes como una salida gráfica. Inicialmente fue concatenada la imagen a una combinación RGB (431), posteriormente se de ruido se puede lograr con un filtro no lineal cuya función básica es calcular la mediana valor de nivel de gris en el vecindario en el que está situado el filtro (González, *et al.*, 1996).



Figura 3. Imagen en escala de grises

La Figura 3 muestra la conversión realizada sobre la imagen RGB a niveles de gris para la aplicación del filtro espacial de convolución. Posterior a esto se obtiene la matriz predefinida o kernel a utilizar para la aplicación del filtro espacial; para el caso se usó la contante de normalización 1/nm.

La matriz implementada se puede observar en la (Ecuación 3)

$$\frac{1}{25} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \quad (3)$$

La ecuación se utilizó considerando que el desenfoque iba a tener el mismo peso para todos los pixeles, definiendo así un filtro de caja con un tamaño común en la literatura de 5x5.

```
>>> import numpy as np
>>> psf = np.ones((5, 5)) / 25
>>> psf
array([[ 0.04,  0.04,  0.04,  0.04,  0.04],
       [ 0.04,  0.04,  0.04,  0.04,  0.04],
       [ 0.04,  0.04,  0.04,  0.04,  0.04],
       [ 0.04,  0.04,  0.04,  0.04,  0.04],
       [ 0.04,  0.04,  0.04,  0.04,  0.04]])
```

Figura 4. Kernel de caja utilizado en el procesamiento

La Figura 4 muestra el código desplegado sobre la línea de comandos de Python con los resultados finales de la normalización. La matriz anterior recorrió la imagen en escala de grises, utilizando la convolución, calculando como respuesta una imagen degradada. Esta degradación implicó pérdida de enfoque, difuminación de bordes y poca diferenciación de coberturas.

La Figura 5 muestra dos acercamientos sobre la imagen base, buscando la comparación visual entre esta y la imagen difuminada tras la aplicación de la convolución con el kernel utilizado. En ella se observa la poca diferenciación de bordes y características de objetos y una degradación general de la imagen.

La teoría indica que una manera eficaz de evaluar el filtro de Wiener, cuando es posible, es utilizando un filtro de desenfoque o una fuente de ruido conocida (Chávez, 2010) para comparar los resultados tanto de la imagen base como de la imagen restaurada. La aplicación del filtro de Wiener permitió obtener una imagen restaurada lo más cercana posible a la original.

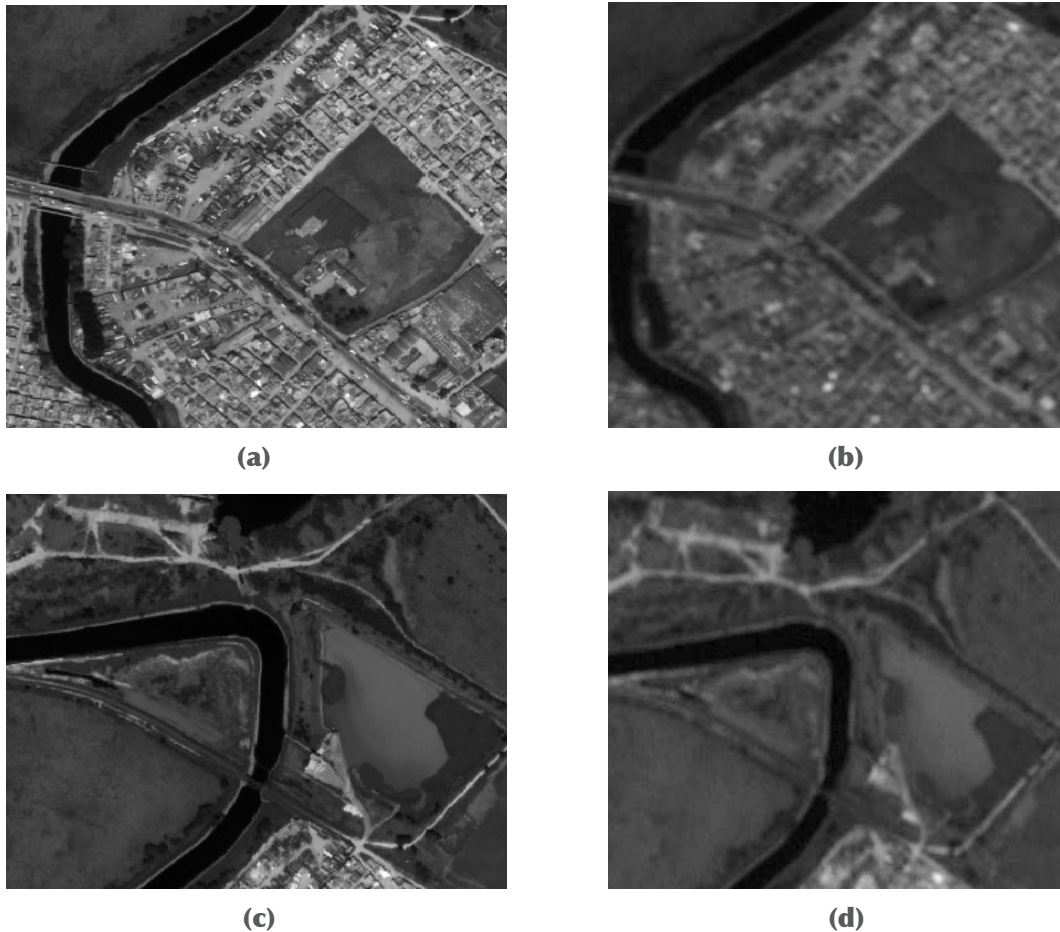


Figura 5. Acercamiento área urbana y (a), (c) Imágenes base escala de grises. (b), (d) Imágenes difuminadas a partir de convolución

La Figura 6 presenta los resultados de dicha aplicación; comparada con la imagen base (6a), la imagen de salida recupera los bordes de objetos, se observó una mejor definición de las características de la imagen; muestra las coberturas de cuerpos de agua, cultivos y zonas urbanas más claramente definidas en cuanto a forma. Se observa, sin embargo, que en los bordes el filtro no se comporta adecuadamente, eliminando el ruido únicamente en los puntos de vecindad.

La evaluación de la efectividad de la utilización del filtro de Wiener para la eliminación de ruido se realizó aplicando el índice de calidad universal, el

cual permite obtener la pérdida de información en tres factores importantes: distorsión de luminancia, contraste de distorsión y pérdida de correlación.

La Tabla 1 presenta los resultados de la evaluación sobre la imagen, este valor de calidad universal fue calculado en Python obteniendo para las dos imágenes resultados cercanos a uno, que resulta ser el ideal del índice, esto significa que la pérdida de información fue poca y, además, que la imagen restaurada se acercó en un valor mayor a la imagen original con un valor de 0.95. Podría interpretarse que conservo 95% de las características de la imagen original.



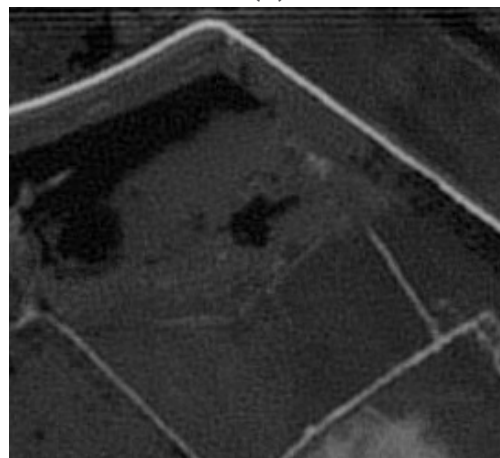
(a)



(b)



(c)



(d)

Figura 6. (a), (b) Imagen base. (c), (d) Imagen restaurada partir del filtro de Wiener

Tabla 1. Evaluación índice de calidad universal

Imagen con filtro espacial	Imagen con deconvolución
(Qu) 0,93094476	0,95262681

CONCLUSIONES

El filtro de Wiener es efectivo para tratar la eliminación de ruido a través de la reconstrucción de la señal; sin embargo, no es óptimo en la reconstrucción de los valores extremos de la imagen.

Además, se comprobó que aun después de reconstruida la señal, la imagen resultante conserva

ruido en algunos puntos, en especial los de mayor reflectancia.

En la imagen WorldView-2, se encontró una reconstrucción con valores altos a partir de la evaluación realizada con el índice de calidad universal con un valor de 0.95; el filtro de Wiener permite observar objetos más pequeños en la imagen reconstruida, pero la interpretación psicomane puede notar esto como piezas únicamente, por esto, el filtro de Wiener no siempre es el mejor, aunque la evaluación de un resultado efectivo, el análisis visual denota la presencia de ruido.

Existe la posibilidad de encontrar otros tipos de filtros a partir de una función de dispersión puntual

o PSF, que se puede obtener a partir de los parámetros de degradación del sensor, o por métodos estadísticos; este nuevo kernel proporciona valores y pesos adecuados para la matriz y podría tener resultados más óptimos que los encontrados en el presente artículo.

La utilización del lenguaje Python permitió desarrollar una amplia cantidad de tareas con un tiempo de ejecución bajo, lo que lo convierte en una herramienta eficiente.

Aun así, de este lenguaje no se encontró evidencia por los autores de librerías ni Scripts, especializados en realizar evaluación de calidad de imágenes espectrales, lo que representa un campo de investigación y programación de tareas que tendrían mejor ejecución que softwares que requieren mayor recurso gráfico y de memoria.

REFERENCIAS

- Aldalur, B. y Santamaría, M. (2002). Realce de imágenes: filtrado espacial. *Rev. Teledetec.*, 17, 31-42.
- Chávez, J. (2010). *Tratamiento digital de imágenes multiespectrales*. México: Universidad Nacional Autónoma de México. Recuperado de <http://www.lulu.com>
- Choi, M., Kim, R. and Kim, M. (2004). The curvelet transform for image fusion. *Int. Soc. Photogramm. Remote Sens, ISPRS*, 35, 59-64.
- González, R., Woods, R., Rodríguez, F. y Rosso, L. (1996). *Tratamiento digital de imágenes*. Addison-Wesley Wilnington eDelaware Delaware.
- Khoumri, M., Blanc-Feraud, L. y Zerubia, J. (1998). Un-supervised deconvolution of satellite images, *Null*, 84.
- Louridas, P. y Ebert, C. (2013). Embedded analytics and statistics for big data. *IEEE Softw*, 6, 33-39.
- Medina, J., Pinilla, C. y Joyanes, L. (2014). Implementing Fast-Haar Wavelet transform on original Ikonos images to perform image fusion: qualitative assessment. *Rev. Fac. Ing. Univ. Antioquia*, 71, 72-84.
- Medina, R., Pinilla, C. and Joyanes, L. (2013, September). Two-dimensional fast Haar wavelet transform for satellite-image fusión. *J. Appl. Remote Sens.*, 7(1), 73698.
- Mehl, H. y Peinado, O. (1997). *Fundamentos del procesamiento digital de imágenes*. Alemania: DSE.
- Mesa, S. (2012). *Desarrollo de un Sistema de Información Geográfica Web para el análisis espacial y temporal de las finanzas del Reino de Castilla en el siglo XVI*. Tesis, Madrid: Universidad Complutense de Madrid.
- Nogueras, G. (2008). *Python como entorno de desarrollo científico*. Recuperado de <http://torroja.dmt.upm.es/media/files/paper.pdf>
- Stockham, T. (1972). Image processing in the context of a visual model. *Proc IEEE*, 60(7), 828-842.
- Van Der Walt, S., Schönberger, J., Nunez-Iglesias, J., Boulogne, F., Warner, J. Yager, N., Gouillart, E. y Yu, T. (2014). Scikit-image: image processing in Python. *PeerJ*, 2, e453.
- Wang, Z. y Bovik, A. (2002). A universal image quality index. *IEEE Signal Process. Lett.*, 9(3), 81-84.
- Westra, E. (2010). *Python geospatial development*. Packt Publishing Ltd.