Proyecto 02 - Retina VesselNet

Giovanni Gamaliel López Padilla

Resumen

Índice

1. Introducción

La caracterización de Vessel juega un importante rol en los diagnósticos médicos. Es por ello, que las tareas como la caracterización de su anchor, color, reflectividad, tortuosidad y ramas abnormales son necesarias. El conocimiento de la localización de las lineas de Vessel es de ayuda para la vista de una retinopatía diabética, ya que reduce el número de falsos positivos en la detección de microaneurismas.[1–3] Cuando el número de ramas o de imágenes es grande, la tarea manual de caracterizar el deliniado de las lineas de Vessel se vuelve un trabajo tedioso y complicado. Con la ayuda de algoritmos de segmentación se ha logrado subsanar la tarea de localizar las lineas de Vessel en una gran cantidad de imágenes. El estudio de esta tarea aumento al incio del milenio, planteando algoritmos basados en detección de bordes[4] hasta hoy en día que el estado del arte se encuentra en el uso de aprendizaje profundo[5] para clasificar y obtener diagnósticos de las lineas de Vessel.

2. Materiales y métodos

2.1. Red neuronal Convolucional

Las redes neuronales convolucionales tienen sus bases en el Neocognitron introducido por Kunihiko Fukushima.[6] Este modelo fue mejorado por Yann LeCun[7] al emplear un método de aprendizaje basado en la propagación hacia atrás para entrenar el sistema correctamente. En el año 2012, Dan Ciresan refino e implemento el modelo para unidades de procesamiento gráfico (GPU) consiguiendo mejores resultados.[8]

Las redes neuronales convolucionales consisten en múltiples capas de filtros convolucionales de una o más dimensiones. Por lo general, después del proceso de una capa de filtro se añade una función para realizar un mapeo causal no-lineal. En la figura 1 se muestra la arquitectura de una red convolucional.

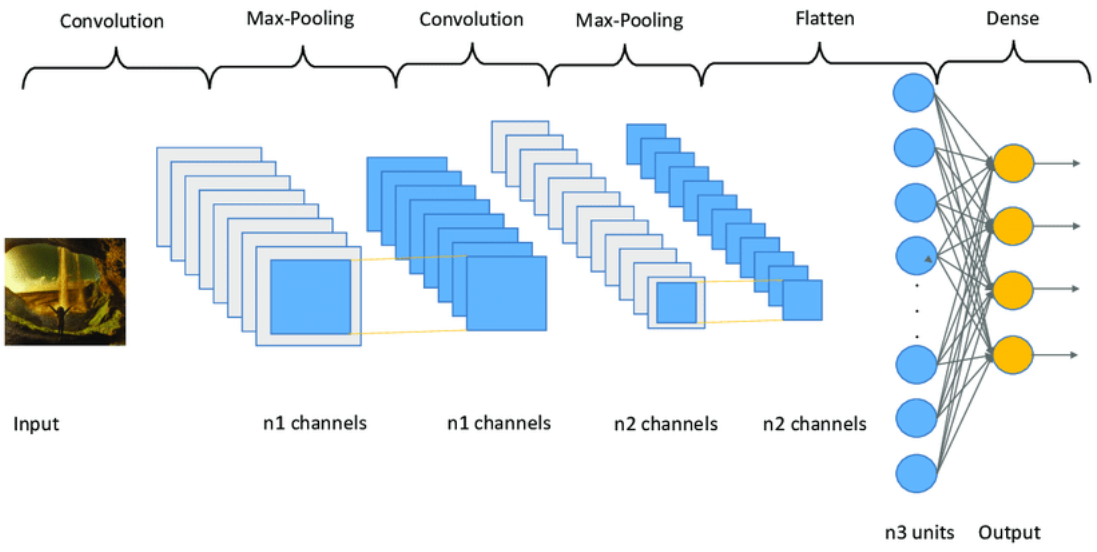

Figura 1: Representación visual de una red convolucional.[9]

2.1.1. Capa Max pooling

La capa de Max pooling realiza una operacion que calcula el máximo valor de una submatriz. Este valor máximo es guardado en una matriz con menor dimensión que la original. En la figura 2 se muestra un ejemplo de como se reduce una matriz de dimensión $4x4$ a una de $2x2$.

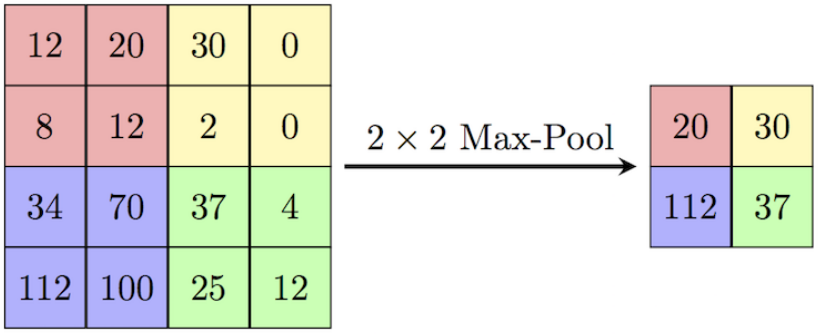

Figura 2: Representación vidual de una capa con la operacion max pooling.

2.2. Modelo U-Net

En la figura 3 se ilustra la arquitectura de una red del tipo U-Net. Este modelo consiste de dos partes, un encoder (lado izquierdo) y un decoder (lado derecho). El encoder tiene una arquitectura típica de una red convolucional donde en cada paso repite una convolución de $3x3$ seguidas de una función rectificador (ReLU) y un max pooling de 2 para una reducción de dimensión. En la etapa del decoder se aplica un aumento de dimensión seguida de una convolución de 2x2 y concadena el resultado con el mapa reducido. En seguida se aplica una dos convoluciones de $3x3$ y se aplica la función ReLU. Al final del proceso se aplica una convolución de $1x1$ que es unada para mapear cada componente a un número de canales. En total, la red tiene 23 capas convolucionales.

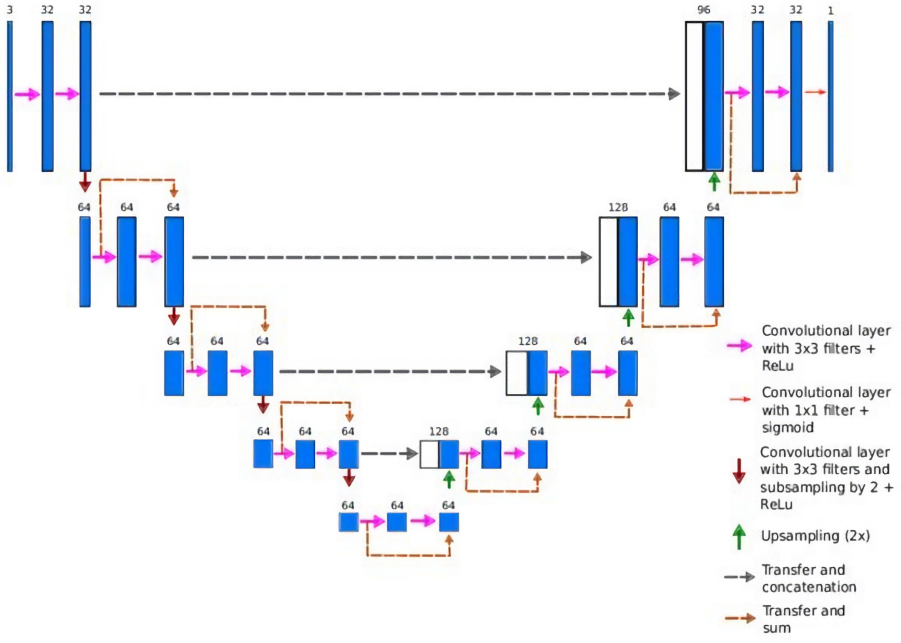

Figura 3: Arquitectura de una red del tipo U-Net.

2.3. Arquitectura del modelo

2.3.1. MobileNet v2

2.3.1.1 Depthwise Separable Convolutions

Depthwise Separable Convolutions es una herramienta para construir de manera eficiente arquitecturas para redes neuronales.[10–12] La idea de la herramienta es remplazar una capa convolucional completa con una versión factorizada en dos capas. La primer capa es llamada *depthwise convolution*, la cual realiza un filtro lihero aplicando una sola convolución en un canal. La segunda cada es una convolución de $1x2$ llamada *pointwise convolution*. Esta capa es la responsable de crear nuevos elementos usando combinaciones lineales de los canales de entrada.

2.3.1.2 Linear Bottlenecks

Consideremos una red neuronal con n capas, donde cada capa contiene la dimensión $h_i \times w_i \times d_i$. Para cada conjunto de imágenes dada, tenemos un conjunto de capas de activación que forman un *manifold of interest*. Este conjunto de capas pueden ser reducidas a un espacio de baja dimensionalidad. Este espacio debe de contener las siguientes propiedades:

- Si el espacio contiene un volumen después de una transformación realizada por la función ReLU, entonces este corresponde a una transformación lineal.
- La función RelU preserva la información del espacio, solo si el espacio esta contenido en un espacio de baja dimensionalidad del espacio de entrada.

Estas propiedades nos dan una pista empirica para la optimización de las arquitecturas de redes. Asumiendo que el espacio de interés corresponde a un espacio de baja dimensional, entonces se puede capturar capas *bottleneck* en un bloque de capas convolucionales. En la figura 4 se muestra el ejemplo de una capa bottleneck.

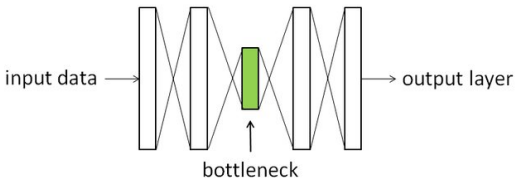

Figura 4: Ejemplo de una capa bottleneck

2.3.1.3 Inverted residuals

Los bloques conpuestos de capas bottleneck se asemejan a un bloque de capas residuales donde cada bloque contiene su entrada seguida de una serie de capas bottleneck seguidas de euna expansión.[13] Como una capa de expansión puede verse como una serie de capas bottleneck acompañadas de una transformación no lienal, se usaron directamente las capas bottleneck. En la figura 5 puede verse una representación de esta decisión.

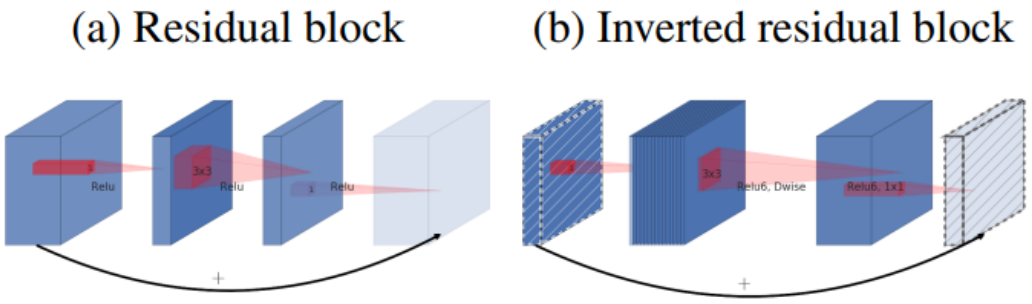

Figura 5: Representación visual de las capas residuales y las capas bottleneck

2.3.1.4 Arquitectura

La arquitectura de MobileNetV2 contiene una capa convolucional de 32 filtros seguida de 19 capas residuales del tipo bottleneck. Se uso la función ReLU6 como una función no lineal por su robustes en una baja precisión numerica.[10] En la tabla 1 se encuentra descrita la arquitectura de MobileNetV2.

Entrada	Operador	t	c	n	s
$224^2 \times 3$	conv2d	-	32	1	2
$112^2 \times 32$	bottleneck	1	16	1	1
$112^2 \times 16$	bottleneck	6	24	2	2
$56^2 \times 24$	bottleneck	6	32	3	2
$28^2 \times 32$	bottleneck	6	64	4	2
$14^2 \times 64$	bottleneck	6	96	3	1
$14^2 \times 96$	bottleneck	6	160	3	2
$7^2 \times 160$	bottleneck	6	320	1	1
$7^2 \times 320$	conv2d 1×1	-	1280	1	1
$7^2 \times 1280$	avgpool 7×7	-	-	1	-
$1 \times 1 \times 1280$	convd2d 1×1	-	k	-	

Tabla 1: Cada linea describe a cada secuencia de uno o más capas identicas. Todas las capas en la misma secuencia tienen el mismo número de canales de salida (c). La primer capa de cada secuencia tiene s pasos y las demás un solo paso. Cada capa convolucional usa un kernel de $3x3$. El factor de expansión t siempre es aplicado como se muestra en la tabla 2.

Entrada	Operador	Salida
$h \times w \times k$	1×1 cond2d, ReLU6	$h \times w \times tk$
$h \times w \times tk$	3×3 dwise s=s, ReLU6	$\frac{h}{s} \times \frac{w}{s} \times tk$
$\frac{h}{s} \times \frac{w}{s} \times tk$	linear 1×1 cond2d	$h \times w \times k'$

Tabla 2: Capa residual del tipo bottleneck que transforma de k a k' canales con s pasos y un factor de expansión de t.

2.3.2. Pix2Pix

2.4. Tipos de entrenamiento

2.4.1. Conexion directa

2.4.2. Fine tunning

2.4.3. Full tunning

2.5. Vessel Dataset

2.6. Filtro de alto contraste

2.7. RGB a escala de grises

3. Resultados

3.1. Conexión directa

3.2. Fine tunning

3.3. Full tunning

4. Conclusiones

5. Referencias

[1] Spencer T, Olson JA, McHardy KC, Sharp PF, Forrester JV. An Image-Processing Strategy for the Segmentation and Quantification of Microaneurysms in Fluorescein Angiograms of the Ocular Fundus. Computers and Biomedical Research. 1996 aug;29(4):284–302. Available from: https://doi.org/10.1006%2Fcbmr.1996.0021.

[2] Frame AJ, Undrill PE, Cree MJ, Olson JA, McHardy KC, Sharp PF, et al. A comparison of computer based classification methods applied to the detection of microaneurysms in ophthalmic fluorescein angiograms. Computers in Biology and Medicine. 1998;28(3):225–238. Available from: https://www.sciencedirect.com/science/article/pii/S0010482598000110.

[3] Larsen M, Godt J, Larsen N, Lund-Andersen H, Sjølie AK, Agardh E, et al. Automated Detection of Fundus Photographic Red Lesions in Diabetic Retinopathy. Invest Ophthalmol Vis Sci Visual Science. 2003 feb;44(2):761. Available from: https://doi.org/10.1167%2Fiovs.02-0418.

[4] Staal J, Abramoff MD, Niemeijer M, Viergever MA, van Ginneken B. Ridge-based vessel segmentation in color images of the retina. IEEE Transactions on Medical Imaging. 2004;23(4):501–509.

[5] Elsharif AAEF, Abu-Naser SS. Retina Diseases Diagnosis Using Deep Learning. International Journal of Academic Engineering Research (IJAER). 2022;6(2):11–37.

[6] Fukushima K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biol Cybernetics. 1980 apr;36(4):193–202. Available from: https://doi.org/10.1007/bf00344251.

[7] Lecun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proc IEEE. 1998;86(11):2278–2324. Available from: https://doi.org/10.1109%2F5.726791.

[8] Cireşan DC, Meier U, Masci J, Gambardella LM, Schmidhuber J. Flexible, High Performance Convolutional Neural Networks for Image Classification. In: Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two. IJCAI'11. AAAI Press; 2011. p. 1237–1242.

[9] García-Ordás MT, Benítez-Andrades JA, García-Rodríguez I, Benavides C, Alaiz-Moretón H. Detecting Respiratory Pathologies Using Convolutional Neural Networks and Variational Autoencoders for Unbalancing Data. Sensors. 2020 feb;20(4):1214. Available from: https://doi.org/10.3390%2Fs20041214.

[10] Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, et al.. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. arXiv; 2017. Available from: https://arxiv.org/abs/1704.04861.

[11] Chollet F. Xception: Deep Learning with Depthwise Separable Convolutions. arXiv; 2016. Available from: https://arxiv.org/abs/1610.02357.

[12] Zhang X, Zhou X, Lin M, Sun J. ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices. arXiv; 2017. Available from: https://arxiv.org/abs/1707.01083.

[13] He K, Zhang X, Ren S, Sun J. Deep Residual Learning for Image Recognition; 2015.