

Detección de melanoma mediante segmentación semántica

Mario Alberto Flores Hernández

Universidad Autónoma de Nuevo León
Facultad de Ingeniería Mecánica y Eléctrica

2020

Índice

1 Introducción

- Motivación
- Hipótesis
- Objetivos

2 Antecedentes

- Función de Costo y Gradiente
- Red Neuronal

3 Estado del Arte

- Características
- Literatura Revisada

4 Implementación

- Resolución Admitida

- Pre-procesado

- Entrenamiento

5 Experimentos

- Filtrado de Umbral

- Coeficiente de Datos

- Criterio de Jaccard

6 Conclusión

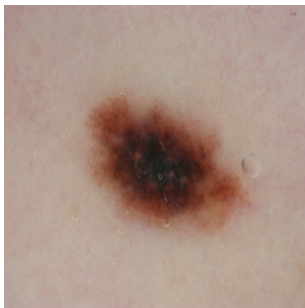
- Discusión

- Trabajo a Futuro

7 Referencias

Introducción

La inteligencia artificial nos permite crear modelos que replican secuencias de transformaciones definidas por los datos entrantes.



(a) Imagen Entrante



(b) Salida Deseada

Figura 1: Ejemplo de un modelo entrada-salida.

Motivacion

Contribuir a la detección temprana del melanoma de piel para reducir la tasa de mortalidad por dicho padecimiento.

Hipótesis

Mediante la implementación de redes neuronales convolucionales e imágenes dermatológicas es posible entrenar modelos que repliquen la técnica de segmentación semántica, clasificando así las regiones dentro de la imagen.

Objetivos

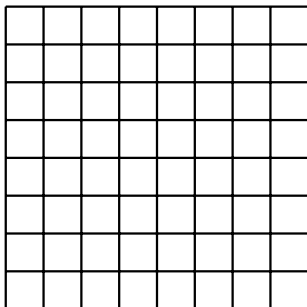
Objetivos Generales

Implementar la tecnología de redes neuronales convolucionales con el fin de obtener un modelo cuya entrada sean imágenes dermatológicas y su salida sea un mapa probabilístico de las regiones dentro de ella.

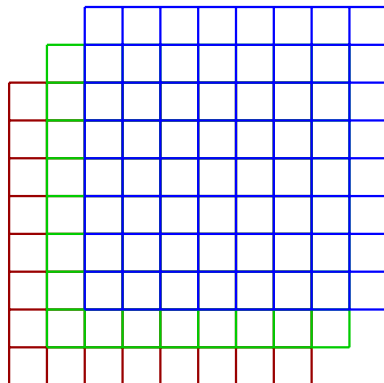
Objetivos Específicos

Determinar la secuencia de pre-procesamiento necesario para adaptar imágenes de diferentes rangos de resolución a la resolución admitida por la arquitectura de la red neuronal convolucional y verificar la confiabilidad de la predicción mediante criterios de evaluación de mapas binarios y probabilísticos.

Antecedentes



(a) Blanco y Negro



(b) Color

Figura 2: Características dimensionales de las imágenes.

Antecedentes

$$l(y, \hat{y}) = - \sum_{j=1}^q (y_j \log \hat{y}_j), \quad (1)$$

$$\frac{\partial}{\partial m} = \frac{2}{N} \sum_{i=1}^N -X_i(y_i - (mX_i + b)), \quad (2)$$

$$\frac{\partial}{\partial b} = \frac{2}{N} \sum_{i=1}^N (y_i - (mX_i + b)), \quad (3)$$

Antecedentes

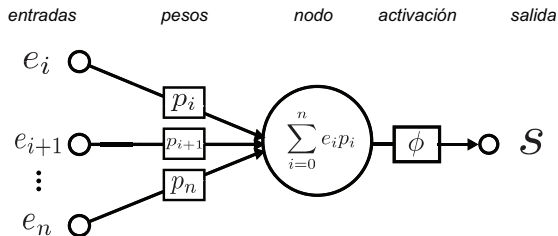


Figura 3: Red Neuronal.

Estado del Arte

Clasificación

El sistema puede clasificar entre distintas categorías.

Segmentación

El sistema puede segmentar las imágenes.

Supervisado

El sistema requiere de datos de datos de entrenamiento.

Pre-entrenamiento

El sistema puede inicializarse con pesos pre-entrenados como alternativa a la inicialización con pesos aleatorios.

Evaluación

El sistema cuenta con una función de evaluación y un algoritmo de optimización.

Salida

Características de los datos obtenidos en la salida del modelo.

Estado del Arte

Cuadro 1: Similitudes y diferencias entre los trabajos revisados; las características implementadas se representan con ✓, mientras que las no implementadas con X.

Trabajo	Modelo	Clasificación	Segmentación	Supervisado	Pre-entrenamiento	Evaluación	Salida
Badrinarayanan <i>et al.</i> [1]	SegNet	✓	✓	✓	✓	✓	mapa de etiquetas
Ronneberger <i>et al.</i> [7]	U-net	✓	✓	✓	X	✓	mapa de etiquetas
Chen <i>et al.</i> [2]	DeepLab	✓	✓	✓	X	✓	mapa de etiquetas
Teichmann <i>et al.</i> [8]	MultiNet	✓	✓	✓	X	✓	mapa de etiquetas
Kroner <i>et al.</i> [5]	VGG16	✓	✓	✓	✓	✓	mapa de calor
Kadampur y Al Riyae [4]	CNN	✓	X	✓	X	✓	mapa de etiquetas
Zhou <i>et al.</i> [9]	ML / SVM	✓	X	✓	X	✓	etiqueta
Luc <i>et al.</i> [6]	CNN/GAN	✓	✓	✓	✓	✓	mapa de etiquetas
Jain <i>et al.</i> [3]	A.B.C.D	X	✓	X	X	X	etiqueta
Propuesta de tesis	FPN	✓	✓	✓	✓	✓	mapa de etiquetas

Implementación

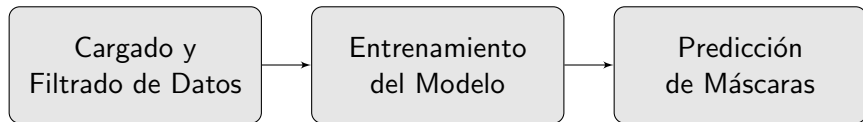


Figura 4: Diagrama de flujo general de la herramienta propuesta.

Resoluciones Admitidas

Las dimensiones de entrada admitidas por la red neuronal convolucional tienen que ser un múltiplo de 32, debido a que

$$\text{múltiplo} = 2^N \quad (4)$$

donde N es el número de capas del codificador de la red neuronal.

Implementación

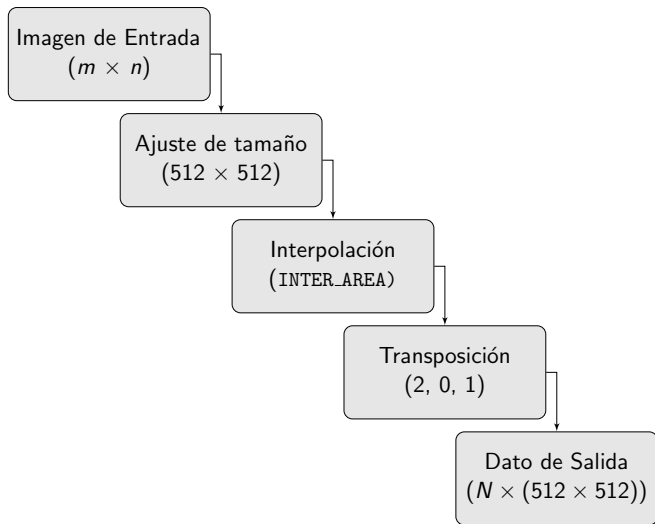


Figura 5: Línea de pre-procesamiento.

Implementación

```
1 # train_model.py
2 max_score = 0
3 for i in range(0,40):
4     print('\n Epoch: {}'.format(i))
5     train_logs = train_epoch.run(train_loader)
6     valid_logs = valid_epoch.run(valid_loader)
7     if max_score < valid_logs['iou_score']:
8         max_score = valid_logs['iou_score']
9         torch.save(model,('Model/'+encoder+'.pth'))
10    print('Highest Score Model Saved: {}'.format(
        max_score))
```

Código 1: Búcle de entrenamiento.

Experimentos

Filtro de Umbral

El objetivo de este experimento es obtener el perfil de la intensidad de los pixeles.

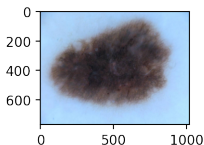
Coeficiente de dados

El objetivo de este experimento es determinar el error durante las iteraciones para comprobar su reducción.

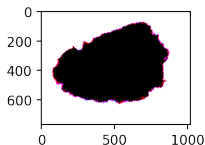
Criterio de Jaccard

El objetivo de este experimento es evaluar la similitud de la máscara real con la máscara predicha.

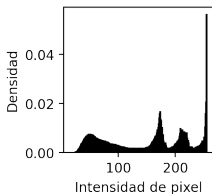
Experimentos



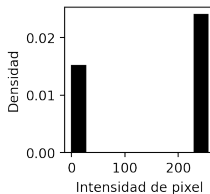
(a) Imagen Entrante



(b) Imagen Filtrada



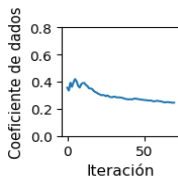
(c) Histograma Original



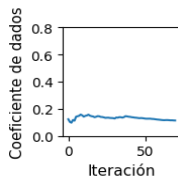
(d) Histograma Resultante

Figura 6: Filtrado de umbral antes y después.

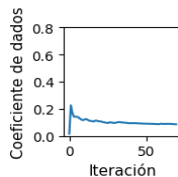
Experimentos



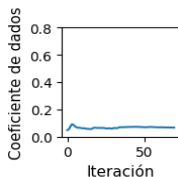
(a) Época 1



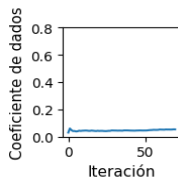
(b) Época 2



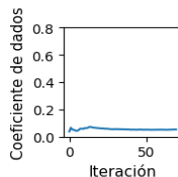
(c) Época 3



(d) Época 4



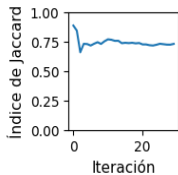
(e) Éποχα 5



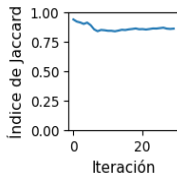
(f) Éποχα 6

Figura 7: Error según el coeficiente de datos.

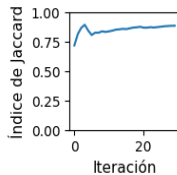
Experimentos



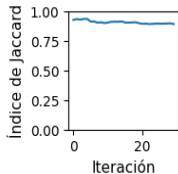
(a) Época 1



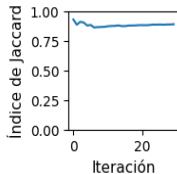
(b) Época 2



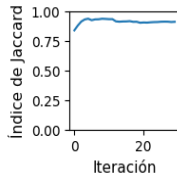
(c) Época 3



(d) Época 4



(e) Época 5



(f) Época 5

Figura 8: Similitud según el criterio de Jaccard.

Conclusión

La implementación de la red neuronal convolucional y el entrenamiento del modelo cumplieron satisfactoriamente con la hipótesis establecida al estimar con un 96 % de precisión las máscaras en el conjunto de datos de pruebas.

Trabajo a Futuro

Implementar redes neuronales convolucionales para la segmentación multicategórica de imágenes médicas.

Referencias I

- [1] BADRINARAYANAN, V., A. KENDALL y R. CIPOLLA (2015), «SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation», *CoRR*, **abs/1511.00561**, 1511.00561 [cs.CV].
- [2] CHEN, L., G. PAPANDREOU, I. KOKKINOS, K. MURPHY y A. L. YUILLE (2016), «DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs», *CoRR*, **abs/1606.00915**, 1606.00915 [cs.CV].
- [3] JAIN, S., V. JAGTAP y N. PISE (2015), «Computer Aided Melanoma Skin Cancer Detection Using Image Processing», *Procedia Computer Science*, **48**, págs. 735–740. doi: 10.1016/j.procs.2015.04.209.
- [4] KADAMPUR, M. A. y S. AL RIYAAE (2020), «Skin cancer detection: Applying a deep learning based model driven architecture in the cloud for classifying dermal cell images», *Informatics in Medicine Unlocked*, **18**, pág. 100282. doi: 10.1016/j.imu.2019.100282.
- [5] KRONER, A., M. SENDEN, K. DRIESSENS y R. GOEBEL (2020), «Contextual encoder–decoder network for visual saliency prediction», *Neural Networks*, **129**, págs. 261–270. doi: 10.1016/j.neunet.2020.05.004.

Referencias II

- [6] LUC, P., C. COUPRIE, S. CHINTALA y J. VERBEEK (2016), «Semantic Segmentation using Adversarial Networks», *CoRR*, **abs/1611.08408**, 1611.08408.
- [7] RONNEBERGER, O., P. FISCHER y T. BROX (2015), «U-Net: Convolutional Networks for Biomedical Image Segmentation», *CoRR*, **abs/1505.04597**, 1505.04597 [cs.CV].
- [8] TEICHMANN, M., M. WEBER, J. M. ZÖLLNER, R. CIPOLLA y R. URTASUN (2016), «MultiNet: Real-time Joint Semantic Reasoning for Autonomous Driving», *CoRR*, **abs/1612.07695**, 1612.07695 [cs.CV].
- [9] ZHOU, J., B. HUANG, Z. YAN y J.-C. G. BÜNZLI (2019), «Emerging role of machine learning in light-matter interaction», *Light: Science & Applications*, **8**(1), págs. 1–7.