

Proyecto - Ajuste de parábola a imagen de fondo de retina

Ramon Orlando Ruiz Olais

06/03/2024

1 Introducción

Los diagnósticos de problemas de vena temporal superior en imágenes de fondo de retina son cualitativos, para realizar seguimientos de los diagnósticos de una forma más precisa, habría que proponer una forma de realizar diagnósticos de manera cuantitativa. Una posible propuesta para conseguir este objetivo sería parametrizar la vena temporal superior de un modo eficaz, para esto se propone realizar un ajuste de una ecuación cuadrática $y = Ax^2 + Bx + C$, por la similitud de la forma de la vena con las parábolas.

2 Pre-procesamiento

Se utilizó el algoritmo para determinar los mejores parámetros del filtro gaussiano aplicado a la base de datos DRIVE, que contiene imágenes de fondo de retina, no se corrió para todos los posibles parámetros por cuestión de tiempo, se tomaron entonces los valores de $T = 13$ y $L = 9$, (obtenidos de la literatura) y se variaron los valores de σ desde 1 hasta 3 en pasos de 0.1 y se obtuvo un $\sigma = 2$.

Las imágenes primero se convirtieron a formato pgm y después se utilizaron los parámetros mencionados para aplicar filtro gaussiano a las imágenes de DRIVE, con estas imágenes se utilizó el método RATS de umbralización, pues del proyecto pasado se calculó como el mejor de los métodos aplicados para estas imágenes.



Figure 1: Imagen original

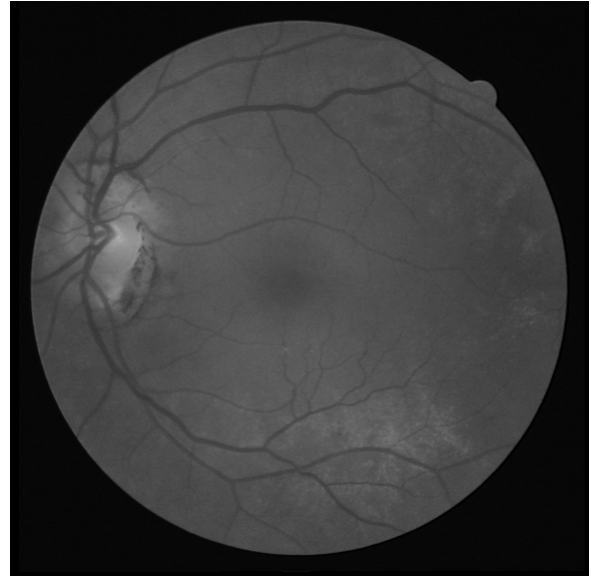


Figure 2: Imagen en formato PGM



Figure 3: Filtro gaussiano

Y una vez umbralizado se calculó la componente conexas más grande y se obtuvo el esqueleto de la vena utilizando el método de Zhang-Suen para tener un objeto con grosor de 1 pixel y poder seleccionar más fácilmente puntos de referencia para ajustar la parábola.

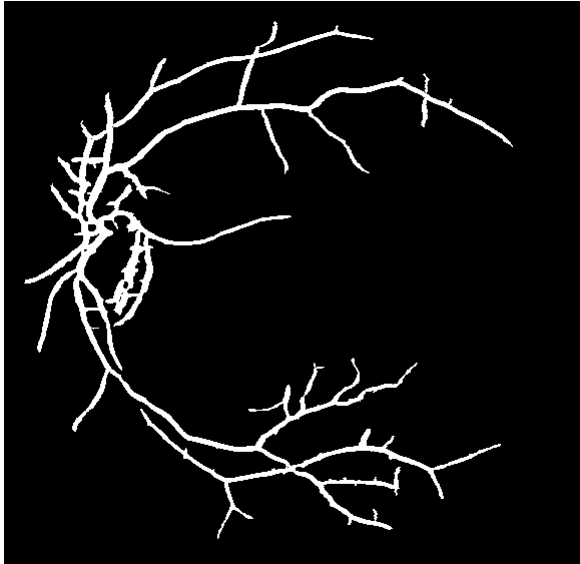


Figure 4: Componente conexa más grande

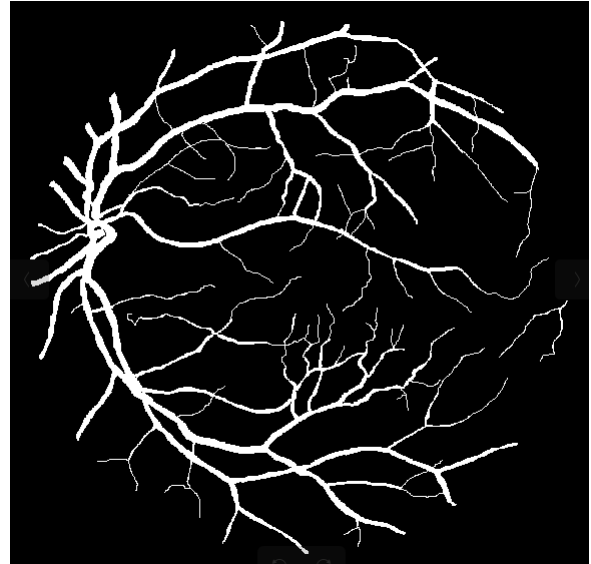


Figure 5: Ground truth

También tomamos la imagen de la vena temporal superior trazada para poder determinar que tan bueno es nuestro ajuste.

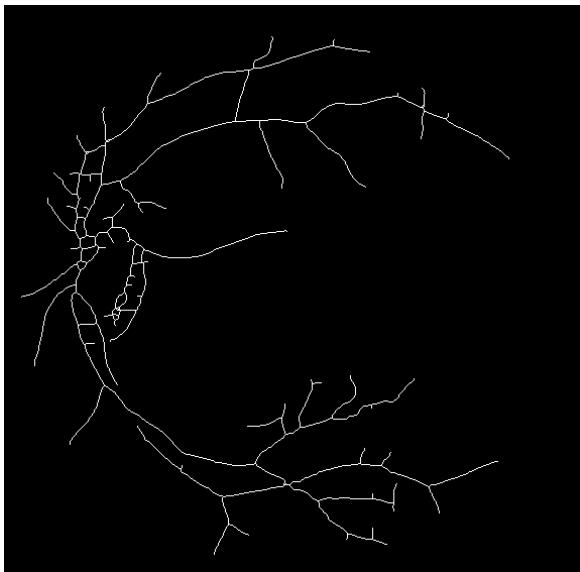


Figure 6: Esqueleto de nuestra componente conexa

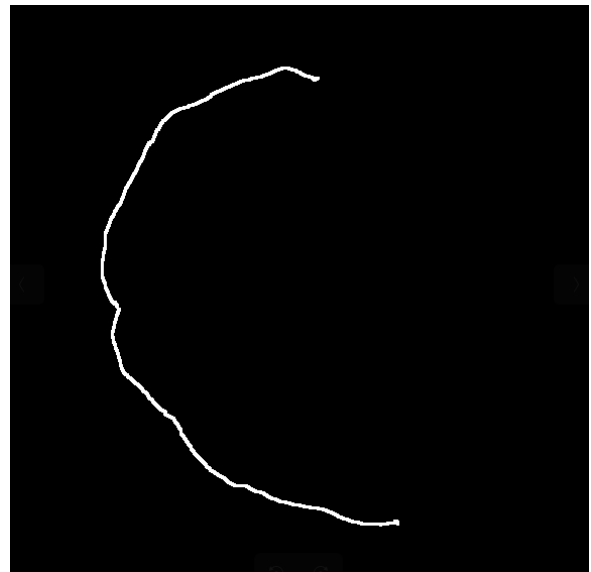


Figure 7: Vena temporal superior

3 Selección de los puntos

Para ajustar la parábola es necesario seleccionar 3 puntos de prueba, los cuáles se toman de manera aleatoria del esqueleto que generamos de nuestra imagen, con estos puntos se determinan los coeficientes A, B y C necesarios para calcular la ecuación cuadrática que pasa por estos puntos, y después, calculamos cuantos puntos de nuestro esqueleto pasan por la curva calculada con estos coeficientes, se repite para al menos 1000 iteraciones y se conservan los puntos y coeficientes que nos den la mayor cantidad de puntos coincidentes con el esqueleto.

Algorithm 1 Iteración para Ajuste de Curvas

```
1: iterations = 0
2: k = 0
3: while iterations < 1000 do
4:   contador = 0
5:   puntos = randomPoints(ske)
6:   A = coeficienteA(puntos)
7:   B = coeficienteB(puntos)
8:   C = coeficienteC(puntos)
9:   for i = 0 to n - 1 do
10:    if porRevisar[i].first == int(A × pow(porRevisar[i].second, 2) + B × porRevisar[i].second + C) then
11:      contador = contador + 1
12:    end if
13:  end for
14:  cantidad[k] = contador
15:  revisados[k] = puntos
16:  k = k + 1
17:  iterations = iterations + 1
18: end while
```

Para calcular los coeficientes se utilizaron las siguientes fórmulas [1]:

$$A = \frac{y_3(x_2 - x_1) + y_2(x_1 - x_3) + y_1(x_3 - x_2)}{(y_1 - y_2)(y_1 - y_3)(y_2 - y_3)}$$
$$B = \frac{y_3^2(x_1 - x_2) + y_2^2(x_3 - x_1) + y_1^2(x_2 - x_3)}{(y_1 - y_2)(y_1 - y_3)(y_2 - y_3)}$$
$$C = \frac{y_2 y_3(y_2 - y_3)x_1 + y_3 y_1(y_3 - y_1)x_2 + y_1 y_2(y_1 - y_2)x_3}{(y_1 - y_2)(y_1 - y_3)(y_2 - y_3)}$$

4 Resultados

Generando los puntos aleatorios y calculando los coeficientes iterativamente se consigue dar con una ecuación de parábola que tenga una gran cantidad de puntos coincidentes con el esqueleto generado de la imagen de entrada. Los coeficientes que generan la curva de mejor coincidencia y sus respectivos puntos se guardan en un archivo de texto, el cuál se carga en un código python y con la librería OpenCV se carga la imagen original de fondo de retina, sobre la cuál se dibuja la parábola calculada, finalmente se creó un código en bash para ejecutar en un solo comando ambos códigos de modo que los cálculos se realizan en C++ y posteriormente se grafica en python:

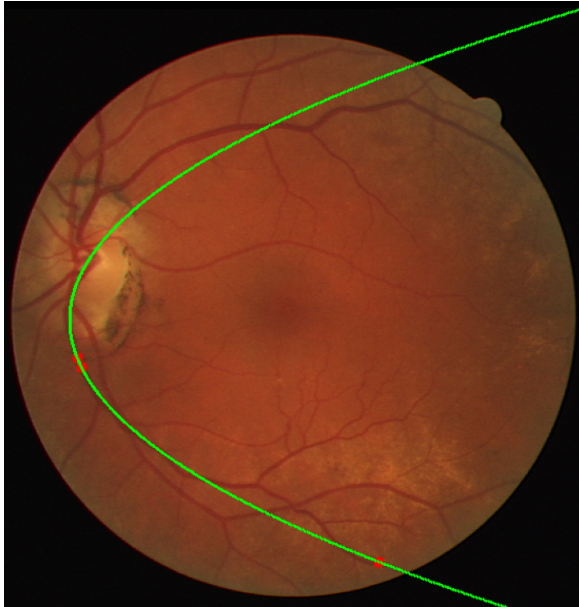


Figure 8: Ajuste 1 de la parábola (verde) y sus puntos (rojo).

Calculamos el ratio de coincidencias con el número total de píxeles dividiendo los píxeles que intersectan de la parábola y la imagen trazada de la vena temporal superior con el número total de píxeles de la vena, vemos que dada la naturaleza estocástica del método, para cada corrida se obtiene una parábola distinta, por lo que se puede concluir que el método utilizado no es muy robusto.



Figure 10: Ajuste 2 de la parábola (verde) y sus puntos (rojo).

```
La posición de mayor coincidencia es: 465
La cantidad de puntos coincidentes es 66
Puntos que generan la parábola con mayor coincidencias:
(74, 340)
(360, 534)
(78, 349)

A = 0.00556639
B = -3.3908
C = 583.397
Coincidencias con la vena temporal superior: 5011
Ratio de coincidencias con número total de píxeles: 1.67368
```

Figure 9: Datos calculados para el primer ajuste.

```
La posición de mayor coincidencia es: 169
La cantidad de puntos coincidentes es 48
Puntos que generan la parábola con mayor coincidencias:
(66, 314)
(83, 172)
(67, 209)

A = 0.00297823
B = -1.56714
C = 264.44
Coincidencias con la vena temporal superior: 4409
Ratio de coincidencias con número total de píxeles: 1.47261
```

Figure 11: Datos calculados para el segundo ajuste.



Figure 12: Ajuste 3 de la parábola (verde) y sus puntos (rojo).

```

La posición de mayor coincidencia es: 47
La cantidad de puntos coincidentes es 51
Puntos que generan la parábola con mayor coincidencias:
(69, 259)
(115, 402)
(171, 464)

A = 0.00283682
B = -1.55346
C = 281.049
Coincidencias con la vena temporal superior: 5137
Ratio de coincidencias con número total de pixeles: 1.71576

```

Figure 13: Datos calculados para el tercer ajuste.

5 Conclusión

Se logró establecer un método para dar una revisión cuantitativa al estado de la vena temporal superior de las imágenes de fondo de retina, sin embargo el método ha demostrado ser no muy robusto pues está basado en una generación de números aleatorios para construir un ajuste de una parábola, esto provoca que no se puedan reproducir los resultados obtenidos en una corrida para una misma imagen. Se pueden implementar métodos en el futuro para mejorar este aspecto del algoritmo y obtener resultados más confiables.

6 Referencias

[1] Parabolic Modeling of the Major Temporal Arcade in Retinal Fundus Images. Faraz Oloumi, Rangaraj M. Rangayyan and Anna L. Ells.