

Fast and Efficient Method for Fire Detection Using Image Processing

Joaquín Laks, Bianca Bramati

Procesamiento de Imágenes
2° Cuatrimestre 2024

El paper *Fast and Efficient Method for Fire Detection Using Image Processing*¹ presenta un método para **detectar fuego en videos**, en dos pasos:

- 1 Segmentar el fuego en cada frame de forma estática, en base a un modelo de color.
- 2 Detectar movimiento de píxeles entre frames contiguos.

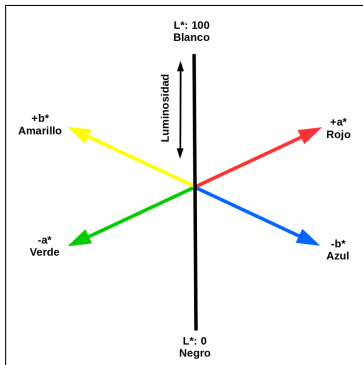
- 1 Replicar los resultados obtenidos en el trabajo.
- 2 Modificar la metodología de segmentación de la imagen, luego comparar los resultados.

El primer paso consiste en segmentar posibles regiones de fuego en la imagen, utilizando el modelo de color **CIE L*a*b***.

Modelo CIE $L^*a^*b^*$

Cada píxel cuenta con tres ejes:

- 1 L^* representa la **luminosidad** del píxel.
- 2 a^* representa la presencia de **verde-rojo** (valores negativos hacia verde, valores positivos hacia rojo).
- 3 b^* , la presencia de **azul-amarillo** (valores negativos hacia azul, valores positivos hacia amarillo).



Análisis de color

Se definen las máscaras R_1, R_2, R_3, R_4 , para definir un espacio de color posible para el fuego, basado en los promedios de cada canal:

$$R1(x, y) = \begin{cases} 1, & \text{si } L^*(x, y) \geq L_m^*, \\ 0, & \text{caso contrario,} \end{cases}$$

$$R2(x, y) = \begin{cases} 1, & \text{si } a^*(x, y) \geq a_m^*, \\ 0, & \text{caso contrario,} \end{cases}$$

$$R3(x, y) = \begin{cases} 1, & \text{si } b^*(x, y) \geq b_m^*, \\ 0, & \text{caso contrario,} \end{cases}$$

$$R4(x, y) = \begin{cases} 1, & \text{si } b^*(x, y) \geq a^*(x, y), \\ 0, & \text{caso contrario,} \end{cases}$$

donde L_m^*, a_m^*, b_m^* son los valores promedio de dicha componente en la imagen.

Se define una última máscara R_5 , de píxeles con mayor probabilidad de pertenecer a una región de fuego:

$$R_5(x, y) = \begin{cases} 1, & \text{si } P(L^*(x, y), a^*(x, y), b^*(x, y)) \geq \alpha, \\ 0, & \text{caso contrario,} \end{cases}$$

donde

- α es un threshold
- $P(L^*(x, y), a^*(x, y), b^*(x, y))$ es la probabilidad que L^* , a^* , b^* pertenezcan a una región de fuego, calculada en base a un análisis con imágenes previamente segmentadas y etiquetadas.

- Para determinar si un píxel (x, y) está en movimiento en un tiempo t , generamos dos máscaras binarias para cada frame: **Foreground Difference** (FD), y **Background Difference** (BD).
- Un píxel está **en movimiento** si $FD(x, y, t) = 1$ o $BD(x, y, t) = 1$

$$FD(x, y, t) = \begin{cases} 1, & \text{si } |L^*(x, y, t) - L^*(x, y, t - 1)| \geq T_{FD}, \\ 0, & \text{caso contrario,} \end{cases}$$

$$BD(x, y, t) = \begin{cases} 1, & \text{si } |L^*(x, y, t) - BG(x, y, t - 1)| \geq T_{BD}, \\ 0, & \text{caso contrario,} \end{cases}$$

donde

- $BG(x, y, t - 1)$ es la luminosidad del fondo de la imagen en el instante $t - 1$, obtenida analizando valores estáticos del frame previo.
- T_{FD} es la suma de la media μ y la desviación estándar σ de $|L^*(x, y, t) - L^*(x, y, t - 1)|$
- T_{BD} es la suma de la media μ y la desviación estándar σ de $|L^*(x, y, t) - BG(x, y, t - 1)|$

- Se analizan las **componentes conexas** de los píxeles (x, y) candidatos: aquellos segmentados tanto en el análisis de color como el de movimiento.
- Se tienen en cuenta componentes conexas que **crecieron en área** en los últimos frames.
- **Idea:** en sus etapas tempranas, el fuego debe crecer espacialmente, por ende el número de píxeles detectados debería incrementar.

- Llamamos $O(t)$ la componente conexa a analizar en el instante de tiempo t , y $NO(t)$ la cantidad de píxeles en $O(t)$.
- Definimos contador $CGO(t)$, que aumentará respecto al instante $t - 1$ si en t aumenta la cantidad de píxeles. Se actualiza como:

$$CGO(t) = \begin{cases} CGO(t-1) + 1, & \text{si } NO(t) \geq NO(t-1) \\ CGO(t-1), & \text{caso contrario.} \end{cases}$$

Dados estos tres pasos, consideramos que el **algoritmo detectó fuego** si luego de aplicarlos a un frame, aún tenemos **píxeles de valor mayor a cero**.

Evaluamos la metodología presentada en dos instancias:

- Comparamos el resultado del análisis de color, con una segmentación realizada por el **algoritmo de Otsu sobre la componente a^*** .
- Para cada frame, comparamos estas tres opciones:
 - El método original, es decir, R_1, \dots, R_5 .
 - Otsu sobre la componente a^* .
 - Otsu sobre la componente a^* combinado con R_5 .

- Método para segmentar imágenes, propuesto por Nobuyuki Otsu²
- En su versión más simple, separa la imagen en dos clases, y calcula un threshold que clasifica los píxeles en alguna de ellas.
- El threshold se calcula minimizando la varianza entre los píxeles de cada clase.

Utilizamos 4 videos (en total 763 frames)³, buscando evaluar el método con videos de distintas características:

- 1 Incendio en estación de servicio
- 2 Autopista sin fuego
- 3 Incendio en bosque
- 4 Incendio en patio, donde la iluminación de la escena es naranja

Experimento sobre método de segmentación

Como primer experimento, realizamos solamente el paso de análisis de color sobre frames estáticos.

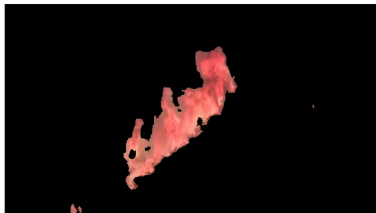
Resultados



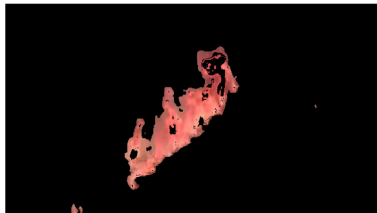
(a) Imagen original



(b) Imagen segmentada con R1, R2, R3, R4, R5



(c) Imagen segmentada con Otsu



(d) Imagen segmentada con Otsu y R5

Experimento sobre videos

A continuación, realizamos el método completo sobre los 4 videos mencionados.

El video [1], de 69 frames, tuvo como resultados:

	Aciertos (%)	Falsos positivos	Falsos negativos
Método del paper	20%	10	46
Otsu sobre a^*	84%	12	0
Otsu sobre a^* y $R5$	83%	12	1



Experimento sobre videos

El video [2], de 26 frames, tuvo como resultados:

	Aciertos (%)	Falsos positivos	Falsos negativos
Método del paper	100%	0	N/A
Otsu sobre a^*	19%	21	N/A
Otsu sobre a^* y R_5	34%	17	N/A



Experimento sobre videos

El video [3], de 68 frames, tuvo como resultados:

	Aciertos (%)	Falsos positivos	Falsos negativos
Método del paper	100%	N/A	0
Otsu sobre a^*	100%	N/A	0
Otsu sobre a^* y R_5	100%	N/A	0



Experimento sobre videos

El video [4], de 600 frames, tuvo como resultados:

	Aciertos (%)	Falsos positivos	Falsos negativos
Método del paper	18%	N/A	495
Otsu sobre a*	99%	N/A	7
Otsu sobre a* y R5	98%	N/A	7



- Para la tarea de **segmentación**, encontramos mejoras usando **Otsu** por sobre el método del paper.
- Algunas suposiciones del método original no son siempre ciertas, en particular las relacionadas a su color que se usan para calcular R_1, \dots, R_4 .
- El método original sigue teniendo mayor éxito
 - Sin embargo, notamos incidencia de **falsos negativos**, lo cual es poco deseable si el método se usa como refuerzo a una alarma de incendios.

- [1] Celik, T. (2010). *Fast and efficient method for fire detection using image processing*. ETRI journal, 32(6), 881-890.
- [2] Otsu, N. (1975) *A threshold selection method from gray-level histograms*. Automatica, 11(285-296), 23-27.
- [3] A. E. Çetin (2014) *Computer Vision Based Fire Detection Dataset*.