

Fast and Efficient Method for Fire Detection Using Image Processing

Bianca Bramati y Joaquín Laks

1. Introduction

falta hacer

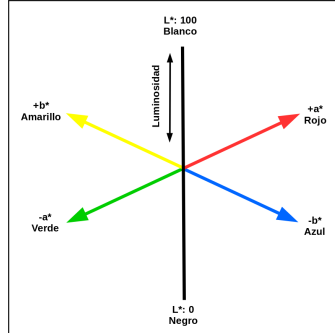
2. Metodología

2.1. Análisis de color

El primer paso del método trabaja sobre cada frame del video independientemente. Consiste en realizar un análisis de la escena teniendo en cuenta solamente sus colores. Utilizamos el modelo de color **CIE L*a*b***.

Este modelo consiste en representar cada píxel con tres componentes:

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



En base a este modelo de color, se definen los valores L_m^* , a_m^* y b_m^* , siendo cada uno de ellos el valor promedio de dicha componente en la imagen. A partir de estos promedios, calculamos las máscaras R_1, R_2, R_3, R_4 , con la intención de definir un espacio de color posible para el fuego:

$$R_1(x, y) = \begin{cases} 1, & \text{si } L^*(x, y) \geq L_m^*, \\ 0, & \text{caso contrario,} \end{cases}$$

$$R_2(x, y) = \begin{cases} 1, & \text{si } a^*(x, y) \geq a_m^*, \\ 0, & \text{caso contrario,} \end{cases}$$

$$R_3(x, y) = \begin{cases} 1, & \text{si } b^*(x, y) \geq b_m^*, \\ 0, & \text{caso contrario,} \end{cases}$$

$$R_4(x, y) = \begin{cases} 1, & \text{si } b^*(x, y) \geq a^*(x, y), \\ 0, & \text{caso contrario,} \end{cases}$$

Definimos, además, $P(L^*(x, y), a^*(x, y), b^*(x, y))$, como la probabilidad que L^* , a^* , b^* pertenezcan a una región de fuego. Se calcula para cualquier valor de L^* , a^* y b^* en base a un análisis con imágenes previamente segmentadas y etiquetadas. Usando este concepto, definimos una última máscara R_5 , que mantiene píxeles con mayor probabilidad de pertenecer a una región de fuego:

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

siendo α es un threshold.

Luego, la máscara final, F , mantiene aquellos píxeles que han sido mantenidos por R_1, R_2, R_3, R_4 y R_5 :

$$F(x, y) = \begin{cases} 1 & \text{if } \sum_{i=1}^5 R_i(x, y) = 5, \\ 0 & \text{caso contrario.} \end{cases}$$

Solo este procedimiento no alcanzaría para detectar fuego con certeza, habiendo una gran variedad de objetos rojos que no son fuego e igual se detectarían de esta forma.

2.2. Análisis de movimiento de píxeles

Como segundo paso, se sigue la idea que los píxeles pertenecientes a una región de fuego presentarán cambios (movimiento) entre frames consecutivos.

Para determinar si un píxel (x, y) está en movimiento en un instante t , generamos dos máscaras binarias para cada frame: **Foreground Difference** (FD), y **Background Difference** (BD).

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

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

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. Los thresholds T_{FD} y T_{BD} se calculan de forma dinámica para cada frame:

- T_{FD} es la suma de la media μ y la desviación estándar σ de $|L^*(x, y, t) - L^*(x, y, t-1)|$ para el frame del tiempo t .
- T_{BD} es la suma de la media μ y la desviación estándar σ de $|L^*(x, y, t) - BG(x, y, t-1)|$ para el frame del tiempo t .

Con estas dos máscaras, definimos la matriz de movimiento:

$$M(x, y) = FD(x, y, t) \oplus BD(x, y, t)$$

Es decir, consideramos que un píxel está en movimiento si $FD(x, y, t) = 1$ ó $BD(x, y, t) = 1$.

Juntando el análisis de color y de movimiento, un píxel "candidato" a ser parte de una región de fuego cumple:

$$CF(x, y, t) = F(x, y) \otimes M(x, y)$$

2.3. Análisis de componentes conexas

Como último paso, se analizan las **componentes conexas** de los píxeles (x, y) que cumplen $CF(x, y, t)$.

La motivación es que, en sus etapas tempranas, el fuego suele crecer espacialmente, por ende el número de píxeles detectados en una región de fuego debería incrementar. Entonces, se tienen en cuenta componentes conexas que crecieron en área en los últimos frames.

Llamamos $O(t)$ a la componente conexas 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:

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

3. Evaluación

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.

Realizamos la evaluación del método de dos formas:

Primero, comparamos solamente el resultado del análisis de color, con una segmentación realizada por el algoritmo de Otsu sobre la componente a^* .

Este es un 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.

Luego, evaluamos el método completo con videos, comparando para cada frame estas tres opciones:

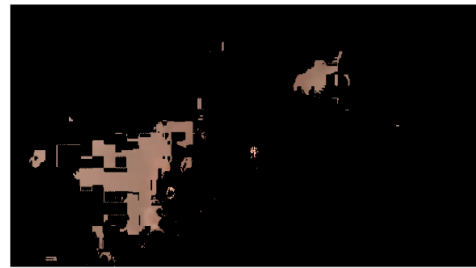
- El método original, es decir, R_1, \dots, R_5 .
- Otsu sobre la componente a^* .
- Otsu sobre la componente a^* combinado con R_5 .

4. Resultados

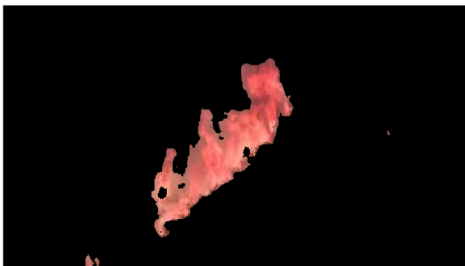
falta escribir un poco. Las tablitas tendría que rehacerlas pero no me dio el tiempo



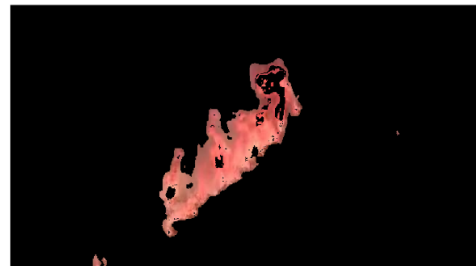
(a) Imagen original



(b) Imagen segmentada con R_1, R_2, R_3, R_4, R_5

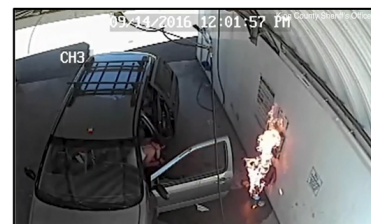


(c) Imagen segmentada con Otsu



(d) Imagen segmentada con Otsu y R_5

	Aciertos (%)	Falsos positivos	Falsos negativos
Método del paper	20%	10	46
Otsu sobre a^*	84%	12	0
Otsu sobre a^* y R_5	83%	12	1



	Aciertos (%)	Falsos positivos	Falsos negativos
Método del paper	100%	0	N/A
Otsu sobre a*	19%	21	N/A
Otsu sobre a* y R5	34%	17	N/A



	Aciertos (%)	Falsos positivos	Falsos negativos
Método del paper	100%	N/A	0
Otsu sobre a*	100%	N/A	0
Otsu sobre a* y R5	100%	N/A	0



	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



5. Conclusión

falta hacer

Referencias

- [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*.