**Báo cáo giữa kỳ ĐATN**

**Đề xuất phương pháp cải tiến thuật toán K-Means cho bài toán phân vùng ảnh.**

**Xây dựng mô hình để phát hiện đám cháy qua ảnh và video.**

**K-Means:**

* Là một phương pháp phân cụm không mờ
* Theo đó mỗi điểm dữ liệu chỉ có thể thuộc về một trung tâm tại một thời điểm
* Thuật toán K-means là một phép lặp được sử dụng để phân vùng một hình ảnh thành các cụm K.

Cho X = {x1, x2 ... xn} biểu diễn một tập hợp các điểm ảnh của ảnh đã cho, trong đó n là số pixel. V = {v1,v2 ... vk} là tập hợp các trung tâm cụm tương ứng, trong đó k là số cụm. Mục đích của K-means là để giảm thiểu hàm mục tiêu J (V):



Trong đó,  là khoảng cách Euclide giữa xij và vj.ki là số điểm ảnh trong cụm i.

Sự khác biệt thường dựa trên màu pixel, cường độ, kết cấu và vị trí hoặc kết hợp trọng số của những yếu tố này.

Vi trung tâm cụm thứ i có thể được tính như sau:



For i=1,…,k.

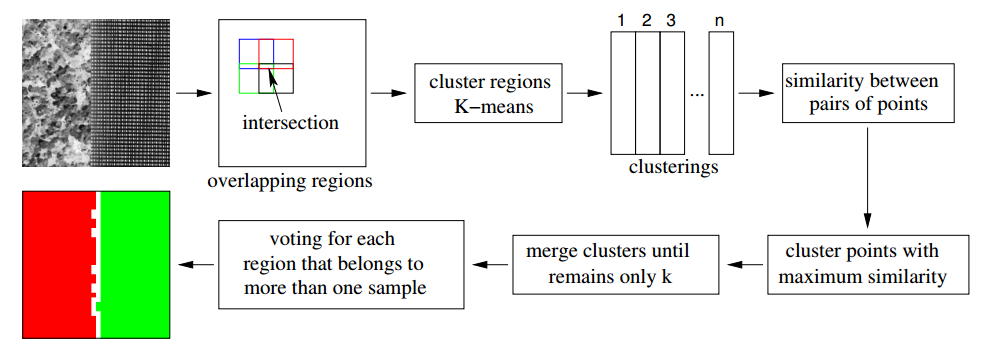
Nhược điểm của K-Means trong bài toán phân vùng ảnh:

* Không tạo ra kết quả tương tự cho mọi lần chạy
* Không khắc phục được hình ảnh nhiễu, ảnh chất lượng thấp
* Thời gian chạy chậm đối với ảnh chất lượng cao
* Kết quả phụ thuộc nhiều vào bước khởi tạo cụm ban đầu

**K-Means++:**

Segmentation ảnh đầu vào bằng việc segmentation từng pixel feature của ảnh.

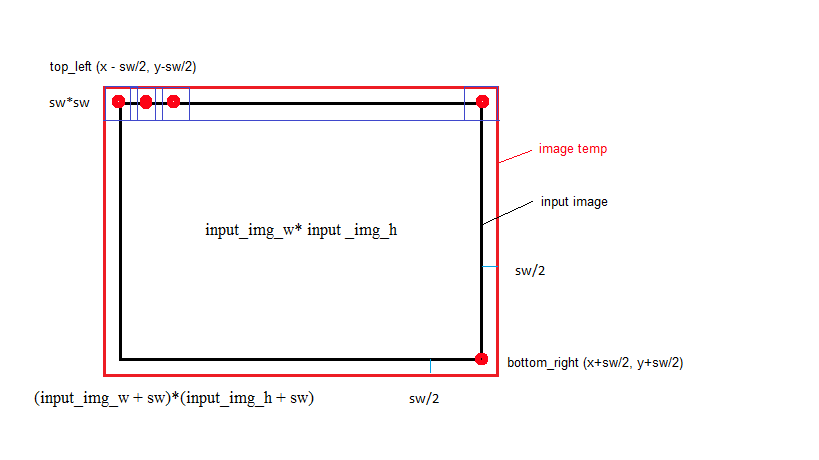
Ảnh đầu vào được chia thành một số cửa sổ vuông (sliding windows) tâm của mỗi winSize là 1 pixel, sao cho có sự chồng chéo giữa các vùng riêng biệt. Trích xuất từ ​​mỗi cửa sổ để tạo một vector đặc trưng. Sau đó, các vectơ đặc trưng kết quả này được sử dụng để quyết định vùng nào của hình ảnh được nhóm lại với nhau:



Các bước được thực hiện trong thuật toán phân cụm được đề xuất K-Means++. Sau nhiều lần thực hiện K-means++, các điểm được phân cụm theo sự giống nhau của chúng. Số lượng điểm bị phân loại sai có thể giảm, dựa vào nhãn feature của các cửa sổ Sliding Windows chồng chéo nhau.

Mỗi ảnh đầu vào có kích thước là ***input\_img\_w\* input \_img\_h*** sẽ được feature bằng các ô *Sliding Windows* vuông có kích thước là ***sw\*sw***.

Một ảnh tạm img\_temp có kích thước là ***(input\_img\_w + sw)\*(input\_img\_h + sw),*** khi đó ảnh đầu vào nằm trọn chính giữa bên trong ảnh tạm này, và cách viền của ảnh tạm này 1 đoạn padding có độ dày sw/2.

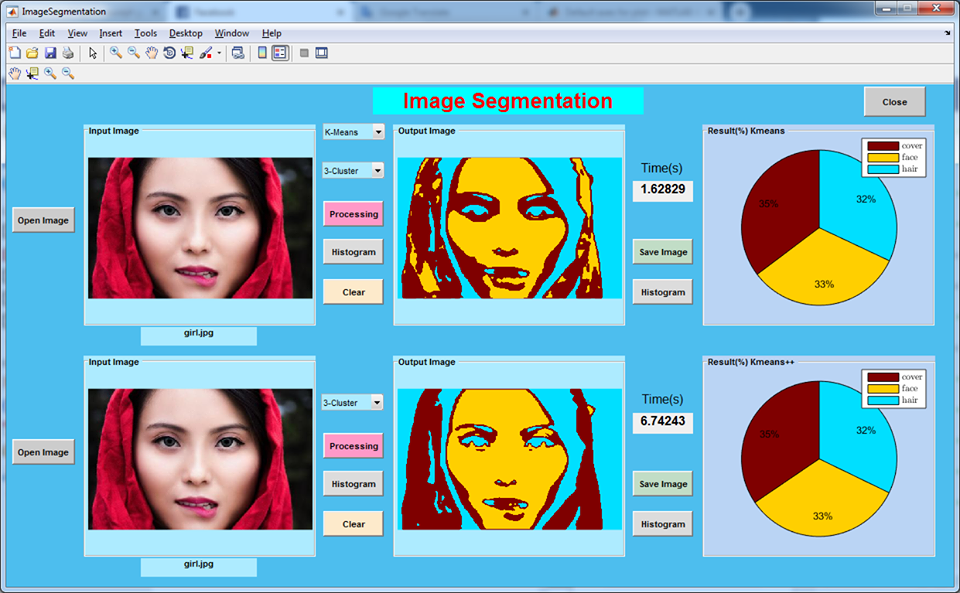


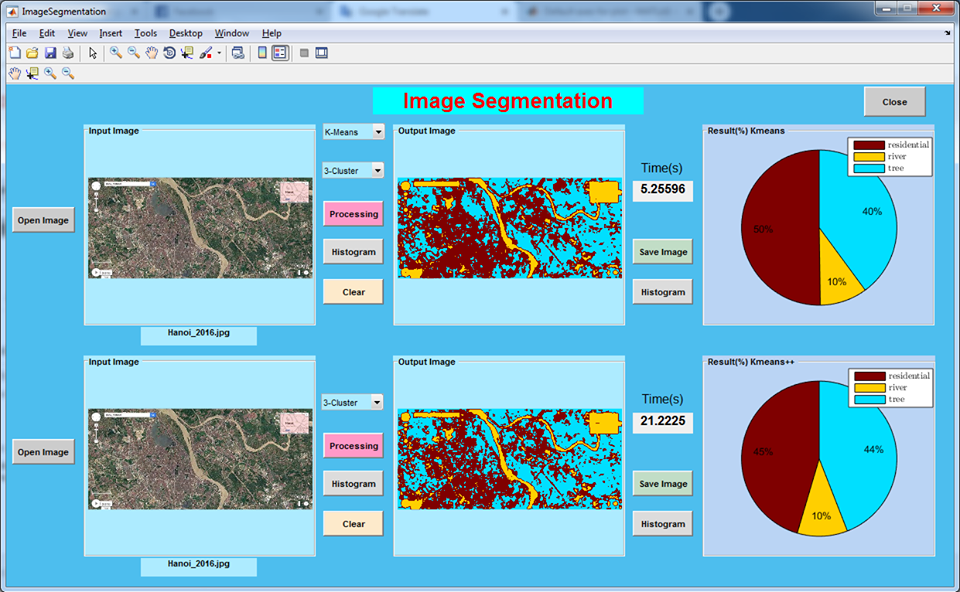
Với x, y là tọa độ của mỗi pixel ảnh đầu vào, với mỗi 1 điểm duyệt qua này, nó sẽ khởi tạo 1 ma trận 3 chiều có kích thước tương ứng là cửa sổ và giá trị cũng là các pixel nằm bên trong cửa sổ đó (ma trận nên đảm bảo luôn nhận các giá trị từ 0 đến 1). Sử dụng tham số tăng cường feature của vùng màu là nBins bằng cách mỗi pixel tương ứng với màu Red được tăng cường nBins^3, Green là nBins^2, Blue là nBins (tăng cường bằng việc nhân với ma trận 3 chiều ở trên) sao cho tổng của chúng luôn nằm trong khoảng nBins^3 phần tử, có nghĩa là mỗi 1 pixel sẽ được ép về 1 đặc trưng trong mảng nBins^3 phần tử. Tại mỗi 1 điểm có thể sinh ra các đặc trưng khác hoặc giống nhau, nhưng tựu chung lại sẽ bị ép về nBins^3 đặc trưng, và như thế nó được coi là segmentation trên từng pixel.

Mỗi 1 pixel tương ứng với 1 window feature, mỗi windows feature sẽ lấy tổng đặc trưng của các pixel trong window đấy, sau đó ta chia cho sw để lấy trung bình và đảm bảo đặc trưng của từng pixel trong input image nằm trong nBins^3 đặc trưng.

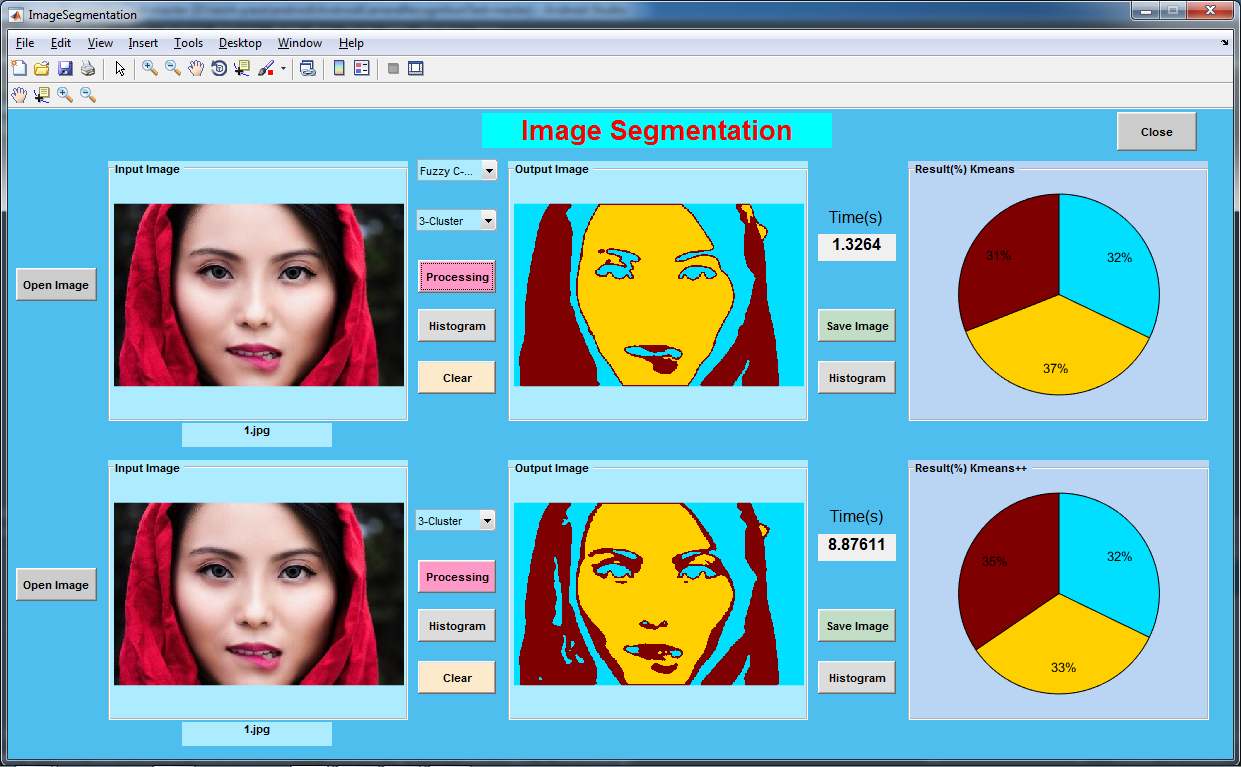
**Kết quả khi thực hiện**

So sánh K-Means và K-Means++





So sánh K-Means++ và Fuzzy C-Means

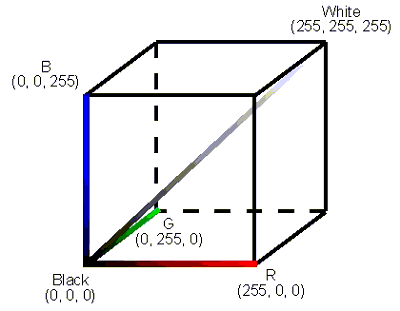


**Bài toán phát hiện đám cháy trong ảnh (video)**

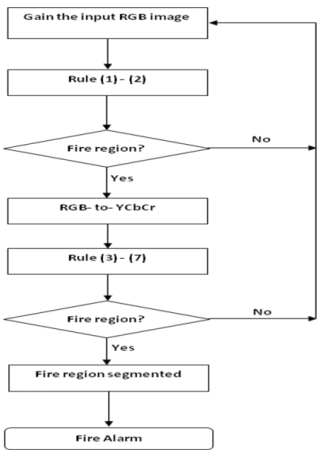
**Phát hiện vùng lửa**

Sử dụng không gian màu RGB và YcbCr, ưu điểm khi sử dụng không gian màu YcbCr là nó có thể tách độ sáng hiệu quả hơn không gian màu RGB.

Một ảnh số trong không gian màu RGB có 3 mặt phẳng R, G, B. Các sự kết hợp giữa các mặt phẳng màu RGB cho phép các thiết bị hiển thị biểu diễn màu trong môi trường kỹ thuật số. Mỗi mặt phẳng màu được lượng hóa thành các mức rời rạc. Nói chung có 256 (8 bit cho mỗi mặt phẳng màu) lượng tử hóa được sử dụng cho mỗi mặt phẳng, ví dụ trắng được biểu thị bằng (R, G, B) = (255,255,255) và màu đen được biểu thị bằng (R, G, B) = (0, 0, 0). Một ảnh màu bao gồm nhiều pixel, trong đó mỗi pixel được biểu thị bằng vị trí không gian trong lưới hình chữ nhật (x, y) và vector màu (R (x, y), G (x, y), B (x, y)) tương ứng với vị trí không gian (x, y) là vị trí tọa độ của pixel.



Sơ đồ thuật toán:

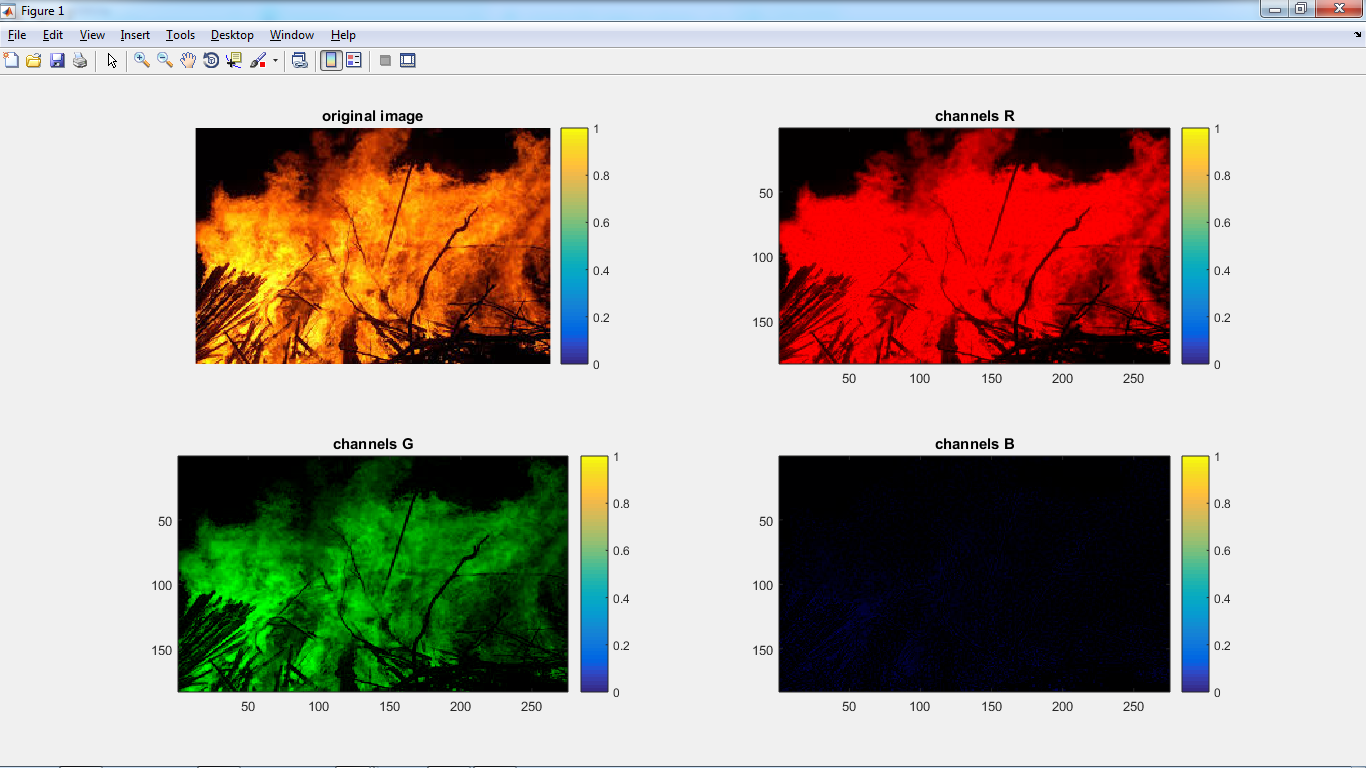


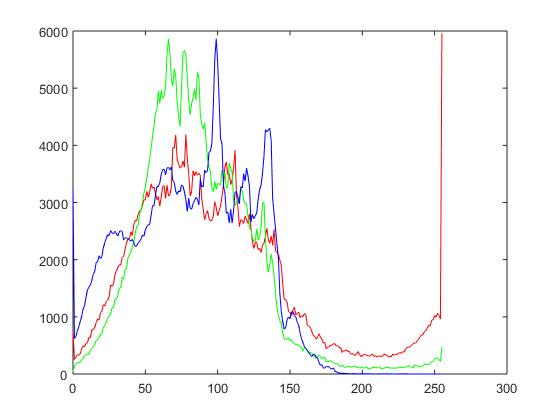
Nếu 1 điểm là điểm ảnh cháy khi nó thỏa mãn đồng thời 7 quy tắc sau đây:

**Quy tắc 1**

Có thể nhận thấy từ hình ảnh cho các vùng cháy, kênh R có giá trị cường độ cao hơn kênh G và kênh G có giá trị cường độ cao hơn giá trị cường độ kênh B.

Hình ảnh được phân vùng các điểm chớp cháy của nó như hình 1 với kênh màu green. Sau đó chúng ta tính giá trị trung bình của các mặt phẳng R, G và B trong các vùng lửa được phân vùng của ảnh gốc. Vì vậy, đối với pixel tại vị trí không gian (x, y) sẽ là pixel lửa nếu quy tắc bên dưới phải được thỏa mãn với pixel này.





**Quy tắc 2**

Xác định một số giá trị ngưỡng cho điểm ảnh lửa:

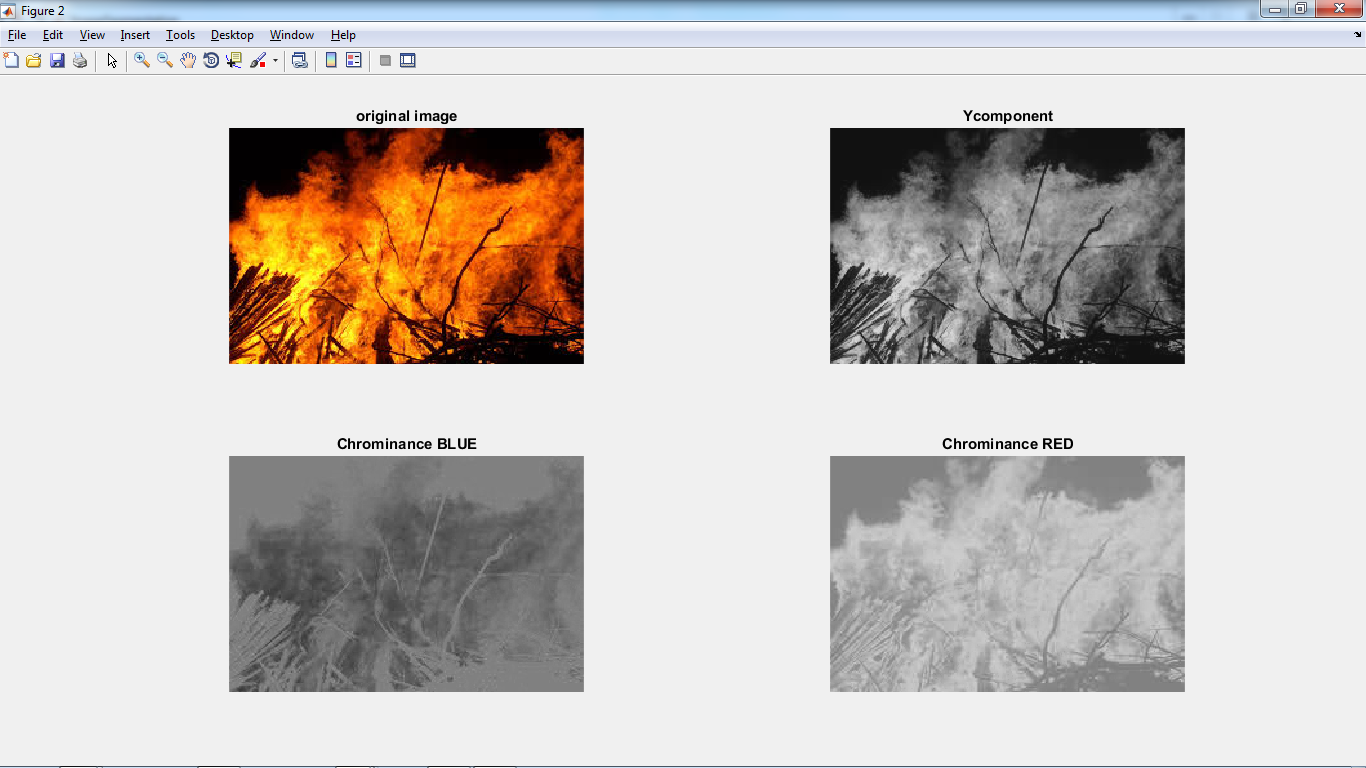
Khi hình ảnh được chuyển đổi từ không gian màu RGB sang YCbCr, cường độ và không gian màu sẽ dễ ràng phân biệt. Điều này giúp việc nhận dạng vùng lửa dễ dàng trong không gian màu YCbCr.

Trong đó Y là độ sáng, Cb và Cr là các thành phần độ màu Blue và độ màu Red tương ứng. Với ảnh màu RGB, nó được chuyển thành hình ảnh màu YCbCr bằng cách sử dụng RGB chuyển YCbCr.

Giá trị trung bình của ba thành phần Y, Cb và Cr, được biểu thị bằng Ymean, Cbmean và Crmean tương ứng được tính như sau:

Trong đó, (x, y) là vị trí tọa độ của pixel trong ảnh, M × N

là tổng số pixel trong hình ảnh đã cho



**Quy tắc 3 và 4**

**Quy tắc 5**

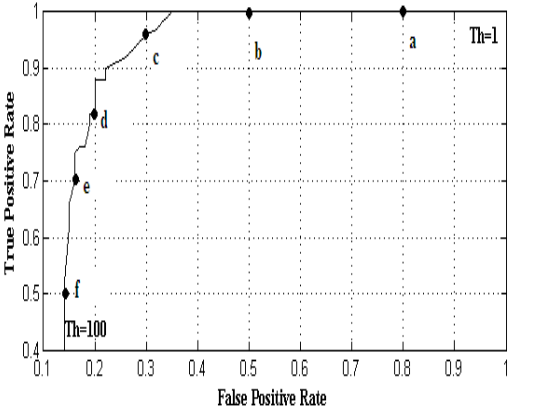
Vùng ngọn lửa nói chung là vùng sáng nhất trong bối cảnh nhìn thấy của bức ảnh hay camera. Giá trị trung bình trên toàn bộ tấm ảnh của ba kênh Ymean, Cbmean và Crmean chứa thông tin có giá trị. Đối với vùng ngọn lửa, giá trị của thành phần Y là lớn hơn thành phần Y trung bình trên toàn bộ bức ảnh. Trong khi giá trị của thành phần Cb tại vùng lửa nhỏ hơn giá trị Cb trung bình của trên toàn bộ bức ảnh. Hơn nữa, thành phần Cr của vùng lửa lớn hơn Cr trung bình trên toàn bộ bức ảnh:

Trong đó, R5 (x, y) chỉ ra rằng bất kỳ điểm ảnh nào thỏa mãn điều kiện được đưa ra trong phương trình (5) được coi là điểm ảnh cháy.

**Quy tắc 6**

Có sự khác biệt đáng kể giữa Cb và các thành phần Cr của các điểm ảnh cháy. Đối với pixel lửa, thành phần Cb chủ yếu là "đen" (cường độ thấp hơn) trong khi thành phần Cr chủ yếu là "trắng" (cường độ cao hơn). Thực tế này có thể được dịch sang một quy tắc khác như sau:

Giá trị của Th được chọn sao cho tỷ lệ phát hiện vượt quá 95% và tỷ lệ báo động giả nhỏ hơn 30% (điểm c) tương ứng với Th = 70.



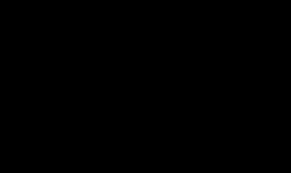
**Quy tắc 7**

Chúng ta có đã xác định một số giá trị ngưỡng cho điểm ảnh lửa. Ta chỉ xét giá trị ngưỡng cho 2 mặt phẳng Cb và Cr, không xét mặt phẳng Y vì nó là thành phần độ sáng và nó phụ thuộc vào điều kiện chiếu sáng.

Một điểm ảnh là pixel vùng lửa nếu thỏa mãn đồng thời 7 quy tắc trên.

|  |  |
| --- | --- |
| Color space | Rules |
| RGB | 1. RGB |
| 1. R Rmean G Gmean B Bmean |
| YCbCr | 1. Y(x,y) Cb(x,y) |
| 1. Cr(x,y) Cb(x,y) |
| 1. Y(x,y) Ymean Cb(x,y) Cbmean Cr(x,y) Crmean |
|  |
| 1. (Cb(x,y) 120) (Cr(x,y) 150) |

Kết quả khi thử nghiệm 1 số ảnh:

**Phát hiện vùng khói**

Tương tự như phát hiện vùng lửa, ta có thể mô hình hóa các điểm ảnh khói. Nhưng điểm ảnh khói không hiển thị các đặc điểm màu sắc như pixel lửa. Lúc đầu, khi nhiệt độ khói thấp, dự kiến khói sẽ hiển thị từ màu trắng-xanh đến trắng. Về sau nhiệt độ của khói tăng lên và màu chuyển từ màu xám đen đến màu đen. Vì vậy, chúng tôi có thể xây dựng các điểm ảnh khói như sau:

*Th*

*Th*

*Th*

Trong đó Th là ngưỡng toàn cầu dao động từ 15 đến 25, các điểm ảnh khói nên có tương tự cường độ trong các kênh màu RGB của chúng. Vì thông tin khói sẽ được sử dụng cho hệ thống phát hiện cháy sớm nhất, nên phát hiện các mẫu khói khi khói có nhiệt độ thấp. Đây là trường hợp, mà các mẫu khói có màu từ trắng-xanh đến trắng, có nghĩa là độ bão hòa của màu sắc nên càng thấp càng tốt. Sử dụng ý tưởng này, quy tắc sau được sử dụng khi không gian màu HSV được sử dụng:

S(x, y) 0.1

Để phân biệt được khói và các hình ảnh tương tự như sương mù hay mây trời, ta sẽ phát hiện sự chuyển động của khói và lửa trong video đám cháy.