

Universidad Distrital Francisco José de Caldas

Curso: Cibernética 3

Nombres :

Daniel Alejandro Rodríguez Suarez - Código: 20172020009

Sebastian Salazar - Código: 201720202016

Grupo: 3



UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

Prueba de concepto

Planteamiento del problema

La propuesta es un sistema contador de rayos capturados en imágenes, en el que se emplea un sistema de lógica difusa que evalúa diferentes estadísticas obtenidas del procesamiento de imágenes realizado. El problema consiste en que los rayos tienen ramificaciones que pueden ser confundidas y ser contadas como rayos, adicionalmente en una tormenta pueden caer más de un rayo por lo que el sistema tiene que ser capaz de hacer un estimado de cuantos rayos hay presentes.

Para realizar el ejercicio se tomaron las siguientes cuatro imágenes de prueba:



Procesamiento de imágenes

El procesamiento fue adaptado del vídeo tutorial de procesamiento hecho por Andy Thé, representante de MATLAB (2014) tiene los siguientes pasos:

1. Carga de imagen

```
I = imread('img/rayos_4.jpg');  
imshow(I);
```

2. Cambio a escala de grises y reducción de elementos en la imagen con filtro de imagen binaria

```
g_img = rgb2gray(I);  
imshow(g_img);  
lv_gray=0.8;  
i1=im2bw(g_img,lv_gray);  
imshow(i1);
```

3. Rellenar agujeros: estos pueden presentarse en ramificaciones que no hayan quedado filtradas o luces de la ciudad en las imágenes

```
Ifilled = imfill(i1,'holes');  
figure, imshow(Ifilled);
```

4. Segunda rellena de agujeros: usando el comando strel se rellenan más agujeros de la imagen

```
se = strel('disk', 1);  
Iopenned = imopen(Ifilled,se);  
% figure,imshowpair(Iopenned, I);  
imshow(Iopenned);
```

5. Extraer datos estadísticos imagen: Se rodean los elementos de la imagen según su centroide y se obtiene información del área, excentricidad y el contorno de las cajas.

```
Iregion = regionprops(Iopenned, 'centroid');  
[labeled,numObjects] = bwlabel(Iopenned,4);  
stats = regionprops(labeled, 'Eccentricity', 'Area', 'BoundingBox');  
areas = [stats.Area];  
eccentricities = [stats.Eccentricity];
```

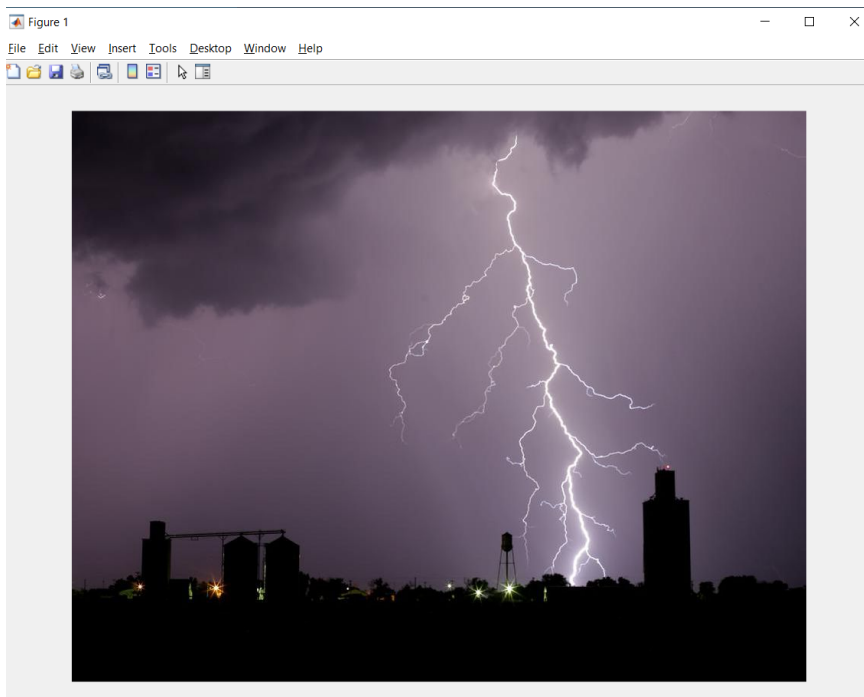
6. Usar estadísticas para determinar los bordes de los rayos: Se colocan cajas en la imagen y se cuenta el número de objetos presente

```
idxOfSkittles = find(eccentricities);  
statsDefects = stats(idxOfSkittles);  
  
figure, imshow(I);  
hold on;  
for idx = 1 : length(idxOfSkittles)  
    h = rectangle('Position',statsDefects(idx).BoundingBox,'LineWidth',2);  
    set(h,'EdgeColor',[.75 0 0]);  
    hold on;  
end  
if idx > 10  
    title(['There are ', num2str(numObjects), ' objects in the image!']);  
end  
hold off;
```

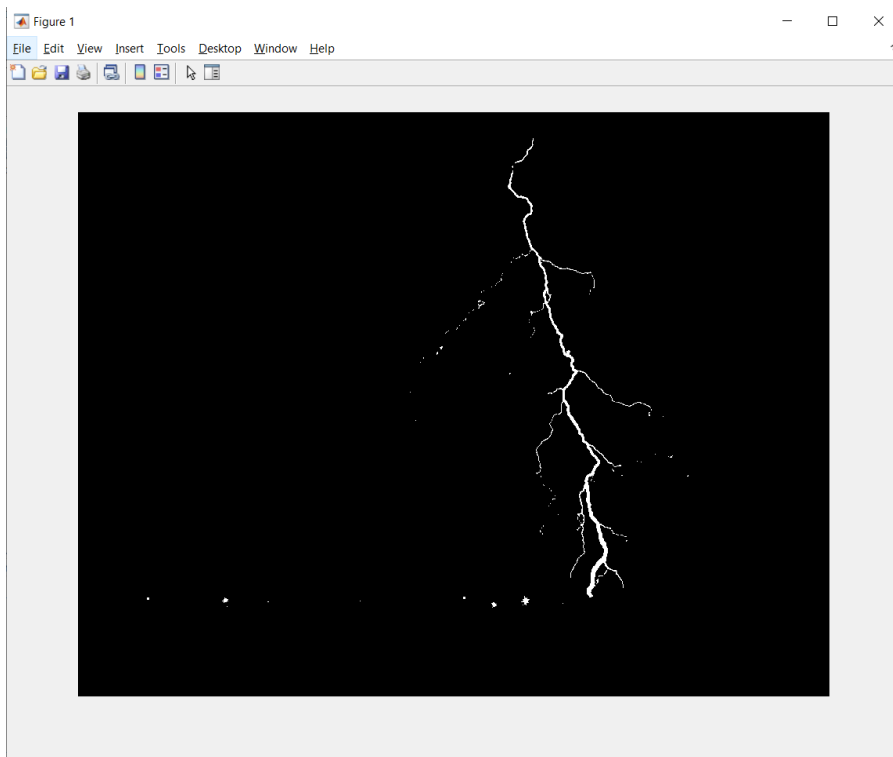
Ejemplo

A continuación se mostrarán los resultados obtenidos paso a paso del procesamiento de la imagen usando la imagen número dos.

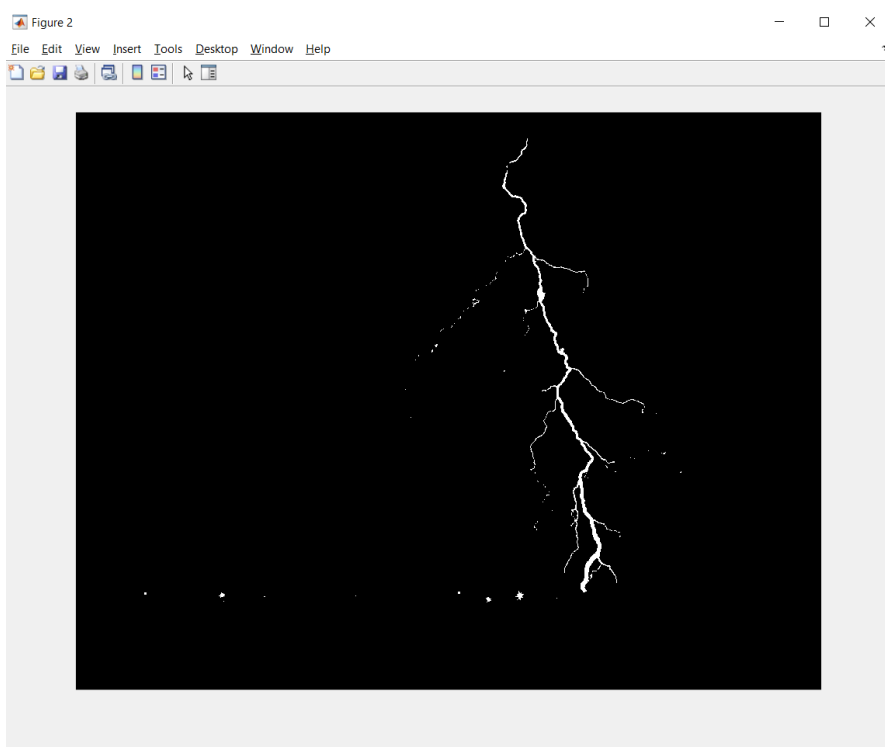
1. Carga de imagen



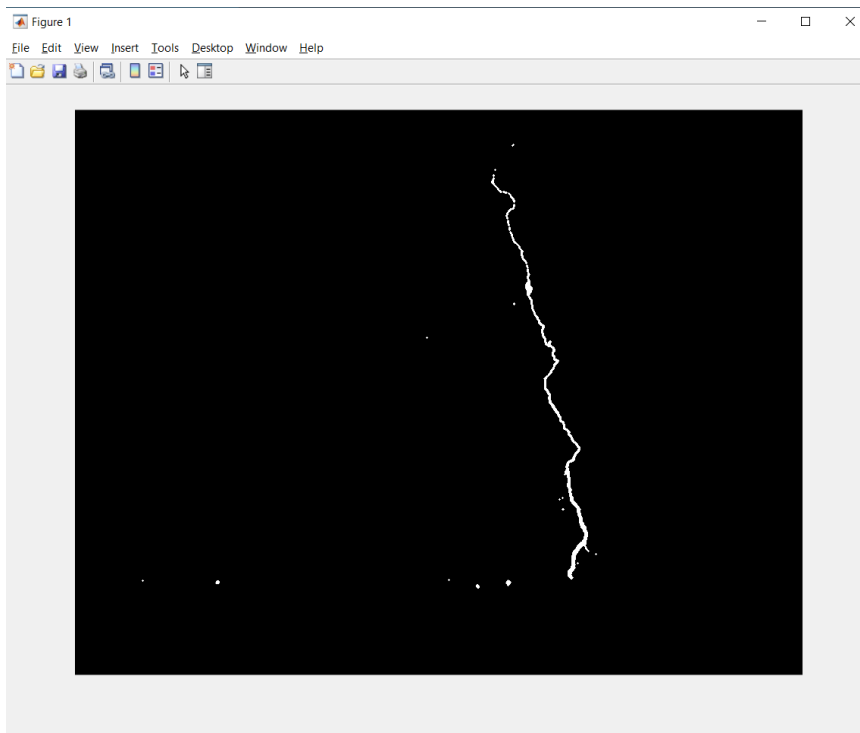
2. Cambio a escala de grises y reducción de elementos en la imagen con filtro de imagen binaria



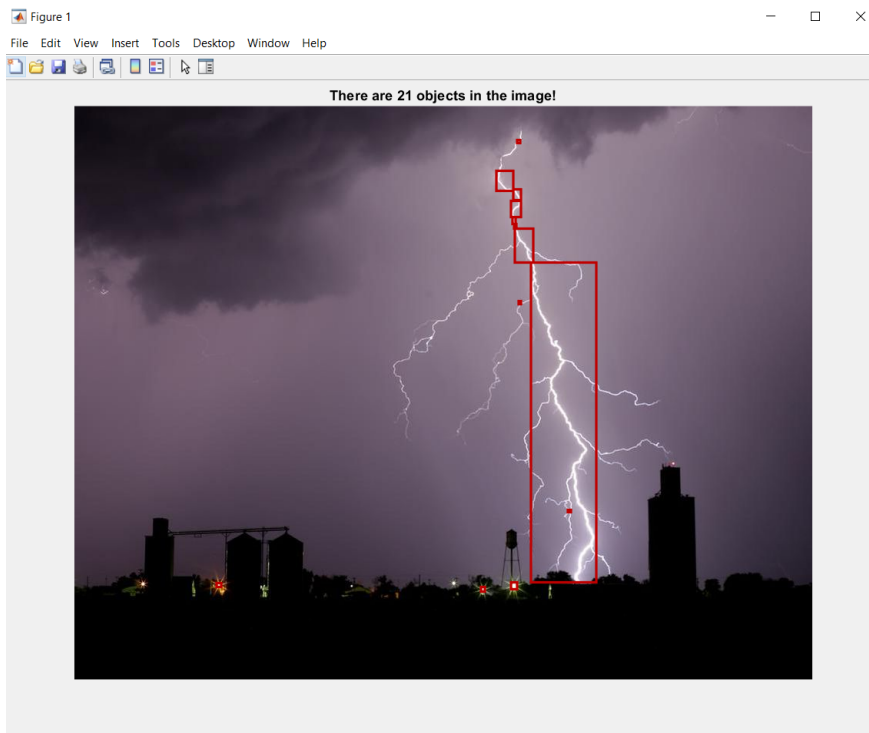
3. Rellenar agujeros: estos pueden presentarse en ramificaciones que no hayan quedado filtradas o luces de la ciudad en las imágenes



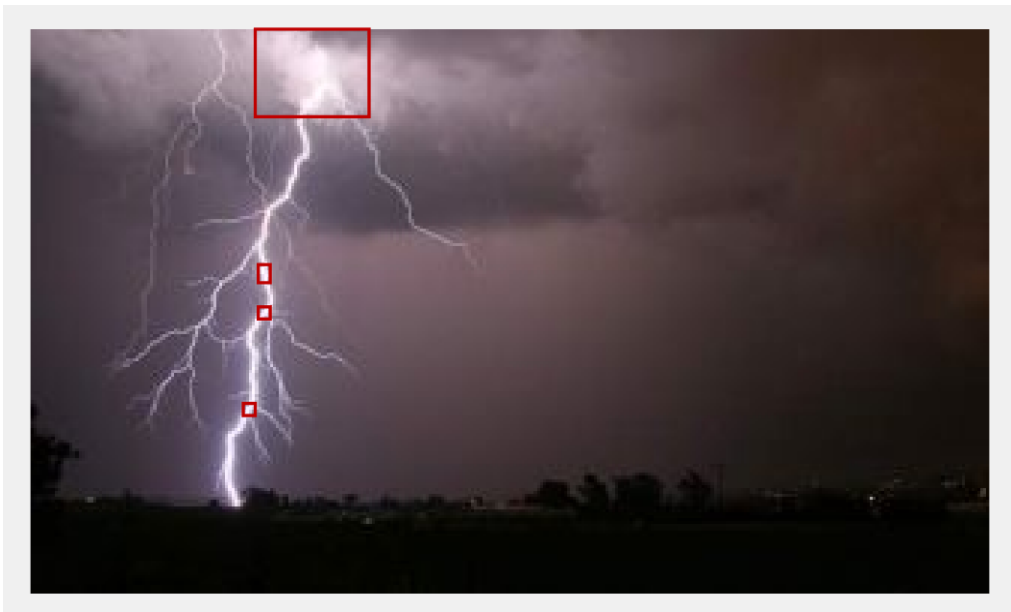
4. Segunda rellena de agujeros: usando el comando strel se rellenan más agujeros de la imagen



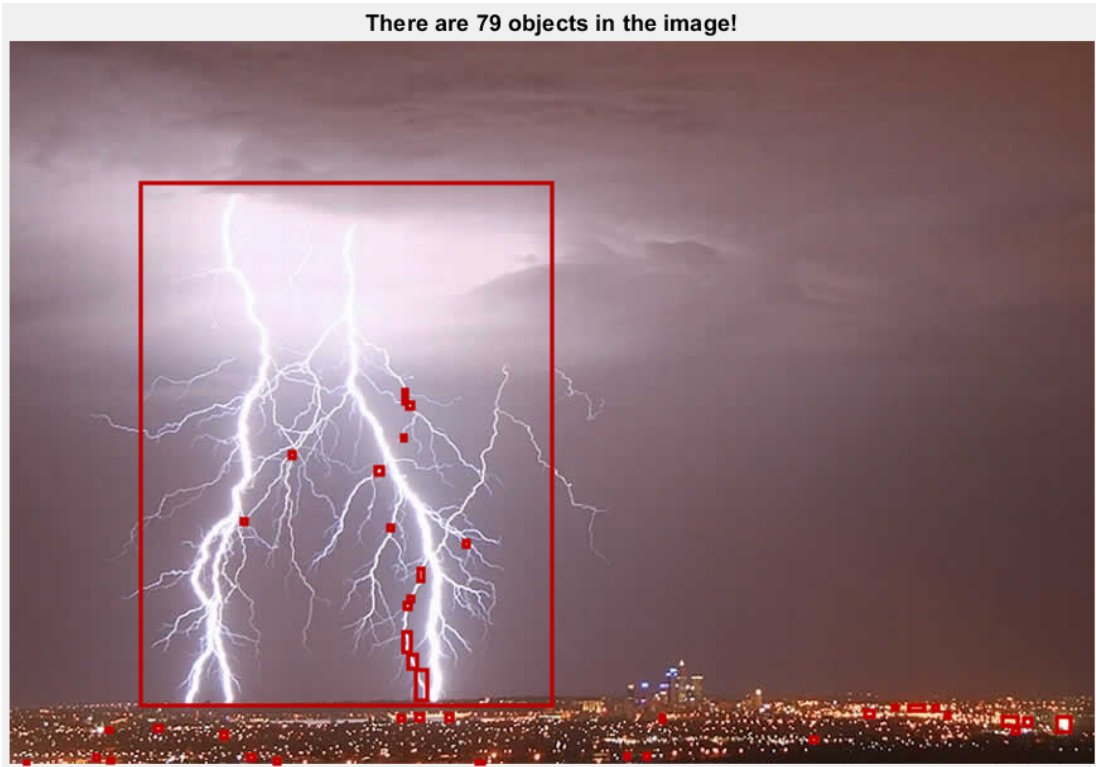
5. Extraer datos estadísticos imagen: Se rodean los elementos de la imagen según su centroide y se obtiene información del área, excentricidad y el contorno de las cajas.
6. Usar estadísticas para determinar los bordes de los rayos: Se colocan cajas en la imagen y se cuenta el número de objetos presente

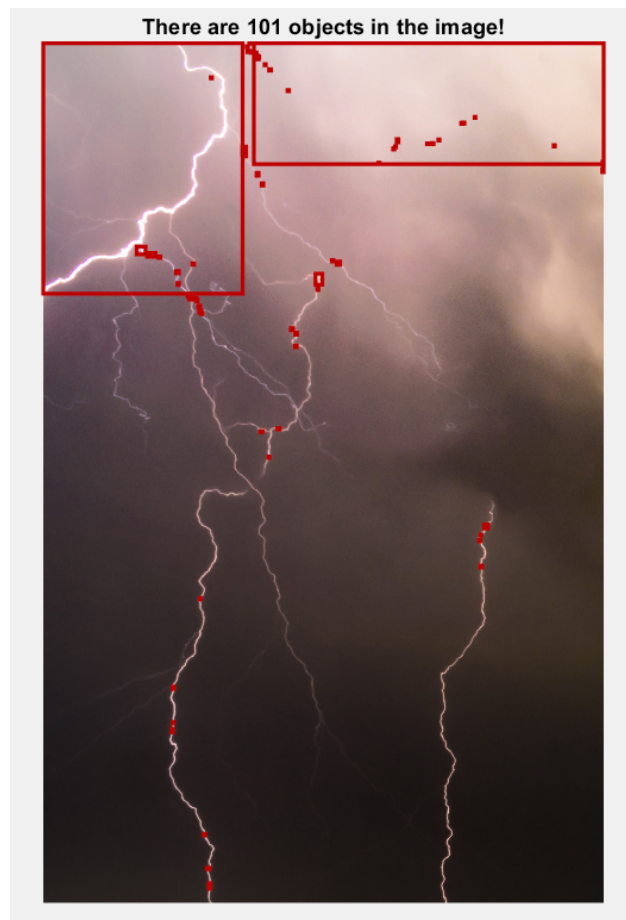


En la imagen anterior se obtuvo que habían 21 objetos en la imagen, cuando idealmente debería haber un gran rectángulo que cuente un rayo, también se puede ver que hay pequeños rectángulos que cuentan una pequeña región del rayo.



There are 79 objects in the image!

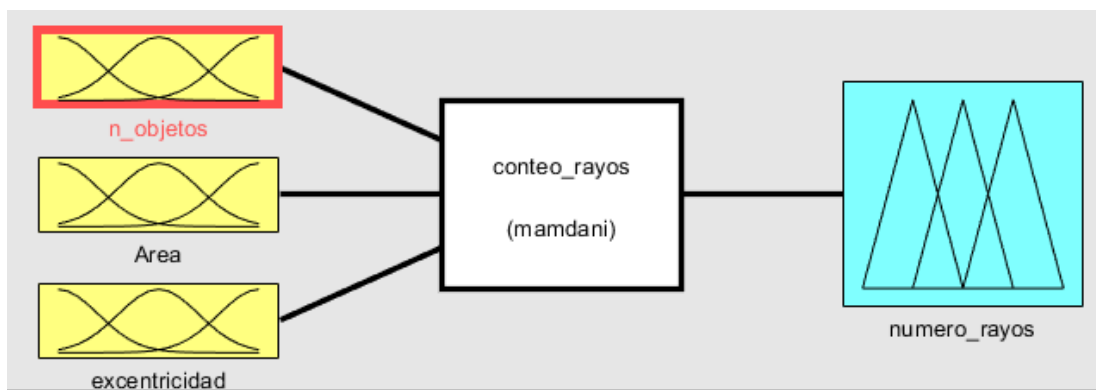




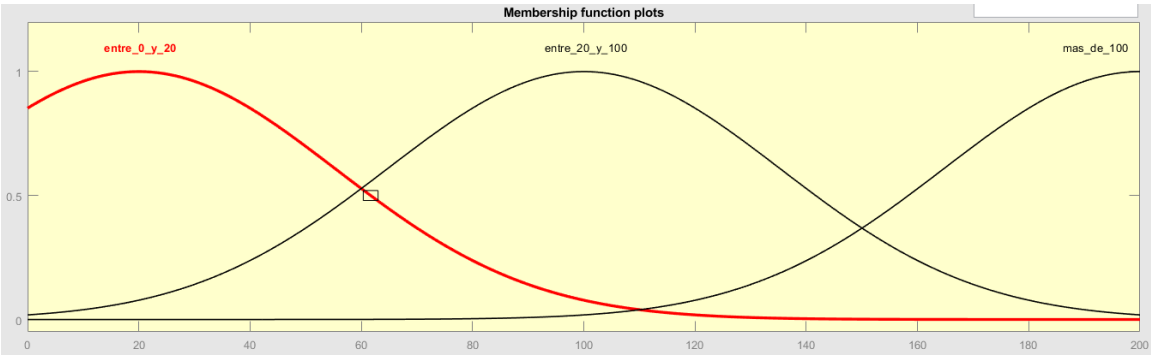
Sistema de lógica difusa

El sistema tiene que ser capaz de diferenciar los cuadros que contienen un rayo y los que no, por lo que se evalúan las siguientes reglas

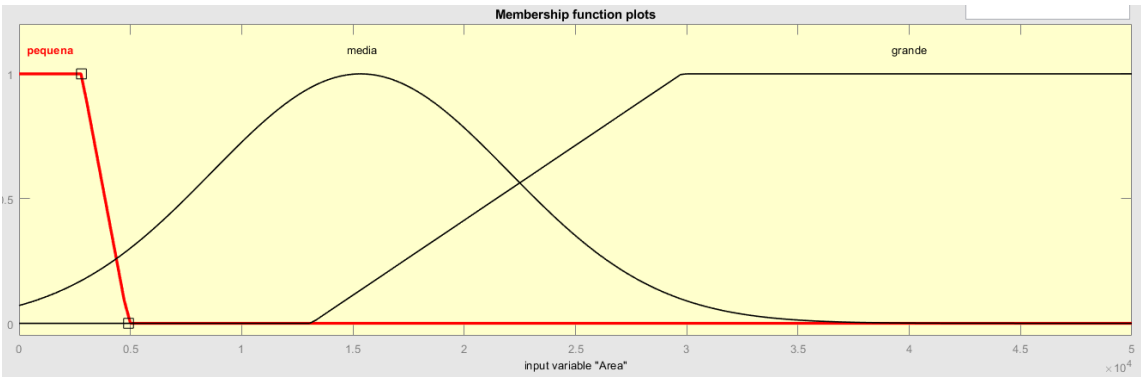
Reglas



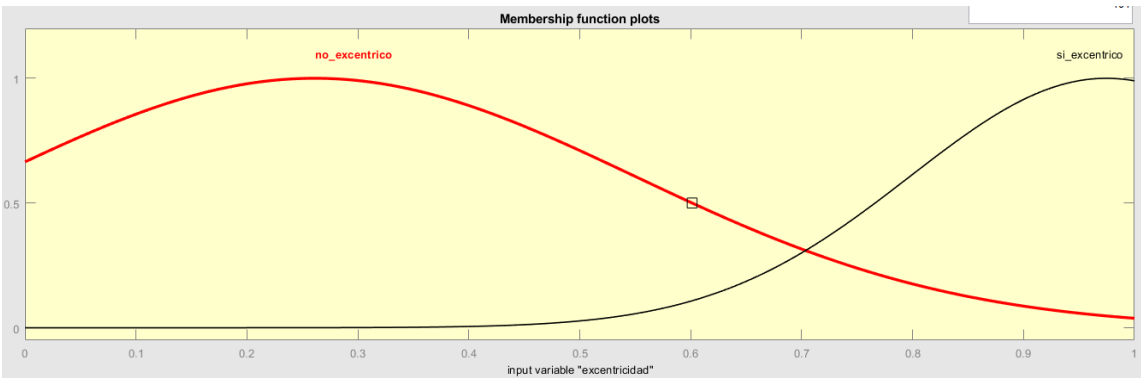
Número de objetos en la imagen



Área de recuadro evaluado



Excentricidad del recuadro de la imagen



Código

```
%% Sistema de lógica difusa
warning('off')
sistema = newfis('conteo_rayos');

%Variable de entrada: Número de objetos
sistema=addvar(sistema,'input','n_objetos',[0 200]);

%Funciones de pertenencia
sistema=addmf(sistema,'input',1,'entre_0_y_20','gaussmf',[35.4 20]);
sistema=addmf(sistema,'input',1,'entre_20_y_100','gaussmf',[35.4 100]);
sistema=addmf(sistema,'input',1,'mas_de_100','gaussmf',[35.37 200]);
plotmf(sistema,'input',1)

%Variable de entrada: Area
sistema=addvar(sistema,'input','Area',[0 5e+04]);

%Funciones de pertenencia
sistema=addmf(sistema,'input',2,'pequena','trapmf',[-1.64e+04 -6130 2800 4915]);
sistema=addmf(sistema,'input',2,'media','gaussmf',[6677 1.534e+04]);
sistema=addmf(sistema,'input',2,'grande','trapmf',[1.312e+04 2.977e+04 5.057e+04 5.067e+04]);
plotmf(sistema,'input',2)

%Variable de entrada: excentricidad
sistema=addvar(sistema,'input','excentricidad',[0 1]);

%Funciones de pertenencia
sistema=addmf(sistema,'input',3,'no_excentrico','gaussmf',[0.289 0.261]);
sistema=addmf(sistema,'input',3,'si_excentrico','gaussmf',[0.177 0.9746]);
plotmf(sistema,'input',3)

%Variable de salida: Numero rayos
sistema=addvar(sistema,'output','numero_rayos',[0 2.5]);

%Funciones de pertenencia
sistema=addmf(sistema,'output',1,'0_rayos','gbellmf',[0.6158 3.278 0.06342]);
sistema=addmf(sistema,'output',1,'1_rayo','trimf',[0.9327 1.014 1.095]);
sistema=addmf(sistema,'output',1,'mas_de_1_rayo','gbellmf',[0.7022 3.28 2.19]);
plotmf(sistema,'output',1)
```

```

%Reglas de inferencia
ruleList=[
    1 1 1 1 1 2 % 0 rayos
    1 1 2 1 1 2 % 0 rayos
    1 2 2 1 1 2 % 0 rayos
    1 3 2 2 1 2 % 1 rayo
    2 1 1 1 1 2 % 0 rayos
    2 2 1 1 1 2 % 0 rayos
    2 2 2 1 1 2 % 0 rayos
    2 3 2 2 1 2 % 1 rayo
    3 1 1 1 1 2 % 0 rayos
    3 2 1 1 1 2 % 0 rayos
    3 2 2 1 1 2 % 0 rayos
    3 3 2 3 1 2]; % + de 1 rayo

sistema = addrule(sistema,ruleList);

%Sistema difuso
fuzzy(sistema)

```

Evaluación de sistema

El sistema de lógica difusa tiene la intención de clasificar cada uno de los cuadros encontrados en la imagen, para ver si contienen un rayo o no.

```

%% Evaluar sistema difuso
%Para evaluar varias entradas
Datos = [8 510 0.766957779668603 % rayo
    8 5 0 % no rayo
    21 2002 0.997533893110550 % rayo
    79 43284 0.516318159560453 % rayo
    101 26621 0.965097480846280 % rayo
    101 10 0.592257740151389 % no rayo
    101 11 0.732333837236824 % no rayo
    ];

Y = evalfis(Datos, sistema)

```

Los resultados obtenidos fueron

¿Contiene un rayo?	Valor calculado	Función piso entero
Si	1.0665	1
No	0.4285	0
Si	1.3072	1
Si	1.3579	1
Si	1.3093	1
No	0.6268	0
No	0.9818	0

Análisis de resultados

El sistema de lógica de difusa es capaz de identificar si un recuadro identificado en el procesamiento de imágenes contiene un rayo si se le aplica la función piso entero, pero es incapaz de diferenciar entre dos rayos que estén superpuestos, por lo que es necesario mejorar el modelo para que si el número es mayor que uno, se tome el valor decimal dado por el sistema de lógica como la probabilidad de que el recuadro contenga más de un rayo.

Referencias

- Thé., A. (2014). "Image processing made easy - previous version". MatLab. Recuperado de: <https://youtu.be/1-jURfDzP1s>