

TRABAJO FINAL

Comparación entre HE y ACE

PROCESAMIENTO DE IMÁGENES (86.56)

Año 2019 - 2^{do} Cuatrimestre

INTEGRANTES:

Manso, Juan

- #96133

<juanmanso@gmail.com>

I. Introducción

En el presente trabajo se analizan dos métodos para la mejora de imágenes al variar el contraste de las mismas. En clase vimos los métodos de *stretching* y la *ecualización del histograma* o *HE*. El primer método busca aumentar el rango dinámico del histograma, mientras que el segundo busca distribuir los valores del histograma (estimación de la función densidad $f(x)$) de forma tal que la distribución resultante se asemeje a la de una uniforme.

Según el método propuesto por Srinivasan y Balram¹, la mejora de contraste puede realizarse con ecualizaciones locales a regiones particulares del histograma ponderadas por una función de pesos. Esta propuesta (*adaptive contrast enhancement* o *ACE*) muestra ser una mejora a la ecualización del histograma dado que se pueden obtener resultados que se adapten a la imagen de entrada, resaltando zonas oscuras o también mejorando el contraste entre objetos.

A continuación se realizará la comparación entre el método *ACE* y el método *HE*.

II. Desarrollo

II.1. Ecualización de histogramas

Una variable aleatoria cuya distribución es uniforme tiene la misma probabilidad de realizarse con un valor, que con cualquier otro. Sin embargo, si se uniformiza el histograma para que todos los valores de luminancia sean equiprobables, se perdería la característica de la imagen y

el resultado carecería de sentido en comparación a la original. Es por ello que se recurre a la linealización de la función de densidad acumulada $F_X(X)$.

De esta manera, al uniformizar la densidad, los picos de luminancia se esparcen para no quedar concentrados en un rango acotado, aumentando el contraste.

Dada una imagen digital de N píxeles con un rango de M valores de luminancia (típicamente $M = 256$) se puede estimar la función densidad de probabilidad de la luminancia x (*PDF*, *probability density function*) según:

$$f(x) = \sum_{i=0}^{M-1} \frac{n_i}{N} \mathbb{1}\{x = x_i\} \quad (1)$$

donde n_i corresponde a la cantidad de píxeles cuya luminancia es x_i (que se limita a M niveles). Éste es el fundamento detrás del uso de histogramas que representan la *PDF* en un gráfico de barras generado a partir de la ecuación (1).

Luego, se obtiene la función de densidad acumulada (*CDF*, *cumulative density function*):

$$F_X(X) = \sum_{i=0}^{M-1} f(x) \mathbb{1}\{x \leq x_i\} \quad (2)$$

Ahora, a partir de la *CDF* calculada en la ecuación (2), se realiza el ajuste de la imagen a través de *HE* operando según la siguiente ecuación para cada píxel pix_i :

¹Srinivasan, S & Balram, Nikhil. (2006). Adaptive contrast enhancement using local region stretching.

$$HE (pix_i) = \frac{R [f(pix_i) - \min\{f(x)\}]}{\max\{f(x)\} - \min\{f(x)\}} \quad (3)$$

En nuestro caso particular, donde se busca un rango máximo, $R = M - 1 = 255$, $\max\{f(x)\} = 1$ y $\min\{f(x)\} = 0$. Por lo tanto la ecuación (3) se puede simplificar:

$$HE (pix_i) = 255 \cdot f(pix_i) \quad (4)$$

Así, cada valor de luminancia se mapea según su *CDF* a un nuevo valor de luminancia, manteniendo la posición espacial original.

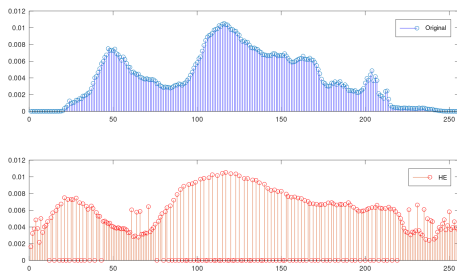


Figura 1: Ejemplo comparativo de histogramas de imágenes con y sin ecualización.

Como se puede ver en la siguiente comparación entre la imagen original y la ajustada, ésta última realza en exceso el contraste de la imagen generando efectos indeseados como por ejemplo nubes borrosas de píxeles sobre el personaje o en los arbustos.



(a) Original



(b) HE

Figura 2: Imágenes con y sin HE.

II.2. Adaptive Contrast Enhancement

El método de *Adaptive Contrast Enhancement* toma la técnica de *HE*, pero la aplica a diferentes regiones del histograma con el objetivo de disminuir la distorsión generada por el método original.

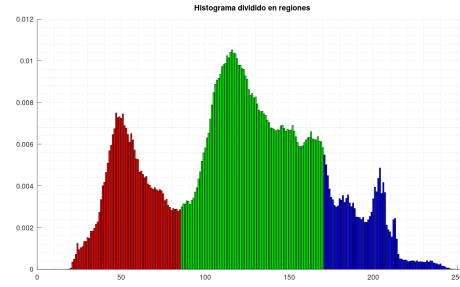


Figura 3: Diferenciación entre regiones de luminancia en el histograma de la imagen.

Se definen tres regiones de igual longitud en el histograma basadas en el nivel de luminancia: oscuro, medio y claro (Figura 3). El método aplica un *HE* independiente a cada región y luego las une, ponderándolas según algún criterio a definir.

La realización de dicho criterio se encuentra en el factor w_i definido para cada región i . El uso de un vector de pesos $W = [w_1 \ w_2 \ w_3]$ trae dos ventajas fundamentales: mejoras personalizadas para cada caso particular de imagen y la alternativa de modos de operación. Un caso de uso de los modos de operación son los filtros provistos por las aplicaciones móviles para retocar imágenes para subirlos a las redes sociales. Las mismas aplicaciones suelen tener además una opción para refinar la imagen de forma más detallada.

En el caso más genérico del método, los pesos son calculados a partir de la pseudo varianza de cada región dadas las medias de luminancia correspondientes. La varianza da una noción de la forma del histograma, en particular denota si los puntos se encuentran concentrados en un rango pequeño de valores o si están dispersos. Analíticamente obtenemos la pseudo varianza σ_i para cada región i según:

$$\sigma_i = \frac{1}{N_i} \sum_{j=0}^{m_i} h_j \cdot |y_j - \mu_i| \quad (5)$$

donde, N_i y m_i son la cantidad de píxeles y la cantidad de niveles de luminancia dentro del

rango i respectivamente. Los valores h_j corresponden a la cantidad de píxeles cuyo nivel de luminancia es y_j (equivalente al valor del histograma para j). Por último, μ_i es la media de valores de luminancia asociado al rango i .

Habiendo calculado la pseudo varianza para cada rango, se obtienen los pesos a partir de la siguiente ecuación:

$$w_i = \eta_i \left(1 - \left| \frac{2\sigma_i}{\sigma_{\max}} - 1 \right| \right) \quad \text{con } \sigma_{\max} = \frac{\max_{j \in I} y_j}{2} \quad (6)$$

con η_i siendo un factor de forma determinado por el usuario para refinamiento de la imagen. El valor de σ_{\max} se halla a partir del peor caso cuya distribución estaría concentrada sólo en los extremos de forma equiprobable.

La ecuación (6) nace del análisis de comportamiento esperado de los pesos en función de la pseudo varianza. Para nuestra aplicación, buscamos asignarle pesos mayores a aquellas regiones cuya pseudo varianza es moderada porque esto implica que la ecualización generaría poca distorsión. En los casos extremos (varianza máxima y mínima), la ecualización produciría grandes o nulos cambios. Gráficamente, la función para calcular los pesos se encuentra en la Figura 4.

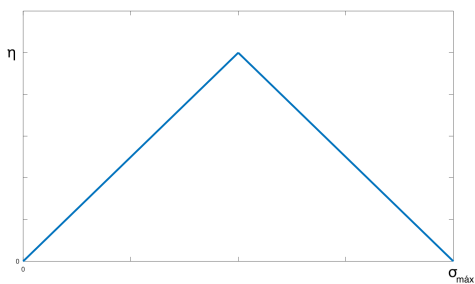


Figura 4: Función de pesos w_i para cada valor de varianza.

Una vez calculado el vector de pesos, se procede a realizar la ecualización para cada región. Para ello, se comienza tomando los valores de la función de densidad acumulada (CDF) correspondientes a la región asociada. Luego, basándose en la ecuación (3), se hace el promedio ponderado entre la ecualizada y la original para la región i -ésima según:

$$ACE_i = orig_i - w_i (orig_i - eq_i) \quad (7)$$

Tras la ecualización y la suma ponderada de las regiones, se unen las regiones obteniendo la imagen adaptada, como se muestra a continuación:



(a) Original



(b) ACE

Figura 5: Comparación entre imágenes con y sin ACE denotando una mejora de contraste entre los objetos.

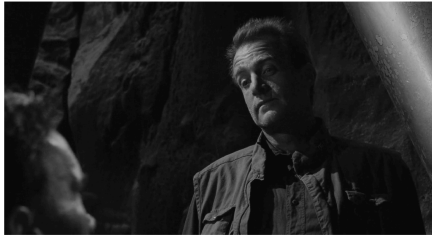
II.3. Resultados

Para el análisis de la efectividad del método, podemos hacer 3 tipos de comparaciones:

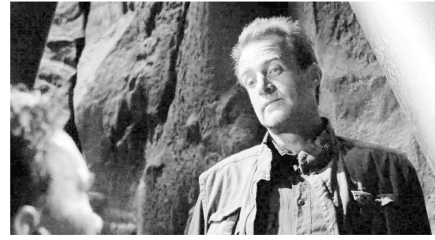
1. Imagen
2. Histograma
3. Relación señal-ruido pico ($PSNR$)

Como se puede ver en la Figura 6, la ecualización del histograma puede generar imágenes saturadas con respecto a las originales (con Figura 6b y Figura 6a respectivamente). En cambio, al hacer la comparación de la Figura 6a y la alterada por ACE (Figura 6c), esta última muestra un mejor contraste visual versus su contraparte de HE.

Para evaluar la variación del rango dinámico, se hallan los histogramas correspondientes para cada caso como se puede ver en la Figura 7.



(a) Original



(b) HE



(c) ACE

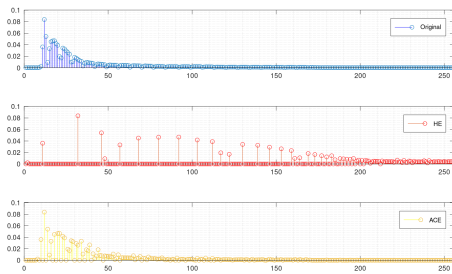
Figura 6: Comparación de resultados para la imagen de prueba *Total Recall*.

Figura 7: Histograma de las imágenes procesadas por HE y ACE, en comparación con la original.

Como se puede ver en la Figura 7, en ambos casos (*HE* y *ACE*) el rango dinámico aumenta. Sin embargo, se puede apreciar que el histograma del *HE*, aunque de mayor rango, contiene mayor cantidad de discontinuidades o picos, mientras que el del *ACE* tiene una envolvente más suave y concentrada en un rango dado. Esto implica que la imagen modificada por *HE* presenta mayores saturaciones, resultando en artefactos indeseados.

Finalmente, en la Tabla 1 se hallan los valores de *PSNR* de los dos métodos bajo análisis. Queda en evidencia de forma analítica, que la propuesta de *ACE* mejora la imagen en mayor

medida que el *HE*, al tener un *PSNR* mayor en todos los casos. Para la imagen *Shadowlands*, el *MSE* utilizado para el cálculo de *PSNR* es tan pequeño que genera un resultado infinito.

Imagen de prueba	$PSNR_{he}$	$PSNR_{ace}$
Total Recall	54,524	59,705
Shadowlands	∞	∞
Odín	28,232	32,429

Tabla 1: *Peak Signal-to-noise ratio* para las 3 imágenes de prueba

III. Conclusiones

El método de *Adaptive contrast enhancement* es una buena alternativa al tradicional método de ecualización del histograma (*HE*) para su aplicación en video dada su fácil configuración y versatilidad ante imágenes de variadas características.

A pesar de agregar mayor complejidad al método, el mismo provee mejoras y personalización que justifican su implementación tanto en sistemas de procesamiento de imágenes como de video.

A. Imágenes de prueba

Para generar las imágenes de prueba con el método ACE, se utilizó un modo de operación uniforme de $\eta^1 = [0,5 \ 0,5 \ 0,5]$.

A.1. Total Recall

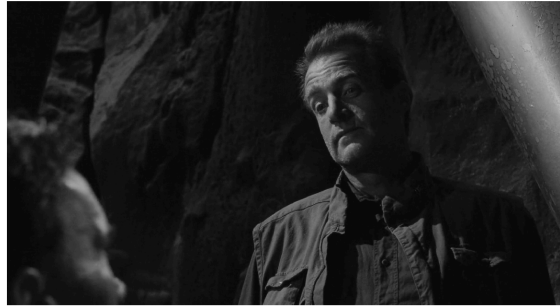


Figura 8: Imagen Original

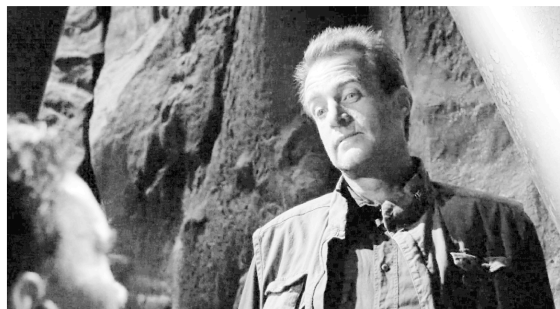


Figura 9: Imagen modificada por HE



Figura 10: Imagen modificada por ACE

A.2. Shadowlands



Figura 11: Imagen Original



Figura 12: Imagen modificada por HE



Figura 13: Imagen modificada por ACE

A.3. Odín



Figura 14: Imagen Original



Figura 15: Imagen modificada por HE



Figura 16: Imagen modificada por ACE

B. Imágenes oscuras con un modo de operación personalizado

Si se realizan modificaciones al modo de operación, las imágenes pueden mejorar su contraste considerablemente. En este ejemplo, se utiliza un $\eta^2 = [0,4 \ 0,5 \ 0,1]$ para concentrar la mejora en las zonas oscuras y medias. El contraste puede disminuir pero también lo hacen los artefactos en la imagen.

B.1. Total Recall



Figura 17: Imagen modificada por ACE



Figura 18: Imagen modificada utilizando η^2

B.2. Shadowlands



Figura 19: Imagen modificada por ACE



Figura 20: Imagen modificada utilizando η^2

C. Funciones principales

C.1. tpf.m

Listing 1: tpf.m

```

1  close all;
2  clear;
3
4
5  addpath(' ../img');
6  addpath(' ../utils');
7
8  max_range = 255;
9  myBlue = [0, 0.447, 0.741];
10 myRed = [0.85, 0.325, 0.098];
11 myYellow = [0.929, 0.694, 0.125];
12 myGreen = [0, 0.5, 0];
13
14 %% Cargo las imágenes
15 name1 = 'odin';
16 name2 = 'shadowlands-dark';
17 name3 = 'totalRecall';
18
19 img1 = imread([' ../img/' name1 '.jpeg']);
20 img2 = imread([' ../img/' name2 '.png']);
21 img3 = imread([' ../img/' name3 '.png']);
22 img1g = rgb2gray(img1);
23 img2g = rgb2gray(img2);
24 img3g = rgb2gray(img3);
25
26
27 %% Imagen de prueba
28 im = img3g;
29 eta = [0.5, 0.5, 0.5];
30 % eta2 = [0.4, 0.5, 0.1];
31 % eta = eta2;
32
33 figure
34 imshow(im)
35 title('Imagen Original')
36
37 figure
38 ho = histo(im);
39 stem(ho, 'markeredgecolor', 'b', 'color', myBlue);
40 grid minor
41 axis([0, 256]);
42 title('Histograma de la imagen original');
43
44 figure
45 cdfo = cdf(ho);
46 plot(cdfo, 'linewidth', 2);
47 grid on
48 axis([0, 256]);
49 title('Función de densidad acumulada para la imagen original');
50
51 close all;
52
53 %% Realizo ecualización del histograma
54 imeq = equalizer(im, max_range);
55
56 figure
57 imshow(imeq)
58 title('Imagen Ecualizada');
59
60 figure
61 he = histo(imeq);
62 stem(he, 'color', 'g', 'markeredgecolor', myGreen);
63 grid minor
64 axis([0, 256]);

```

```

65 title('Histograma de la imagen ecualizada');
66
67 figure
68 cdfe = cdf(he);
69 plot(cdfe, 'linewidth', 2, 'color', myGreen);
70 grid on
71 axis([0, 256]);
72 title('Función de densidad acumulada para la imagen ecualizada');
73
74 close all;
75
76 %% Aplico el metodo ACE
77 imace = ace(im, eta);
78
79 figure
80 imshow(imace)
81 title('Imagen mejorada con ACE');
82
83 figure
84 hace = histo(imace);
85 stem(hace, 'color', myRed, 'markeredgecolor', 'r');
86 grid minor
87 axis([0, 256]);
88 title('Histograma de la imagen mejorada con ACE');
89
90 figure
91 cdface = cdf(hace);
92 plot(cdface, 'color', myRed, 'linewidth', 2);
93 grid on
94 axis([0, 256]);
95 title('Función de densidad acumulada para la imagen ACE');
96
97 close all;
98
99 %% Resultados
100 sections = [85, 170];
101 dark = 1:sections(1); mid = (sections(1)+1):sections(2); bright = (sections(2)+1):
    length(ho);
102
103 figure
104 title('Histograma dividido en regiones');
105 hold on
106 bar(bright, ho(bright), 'b');
107 bar(mid, ho(mid), 'g');
108 bar(dark, ho(dark), 'r');
109 axis([0, 256]);
110 grid minor
111
112 figure
113 title('Comparación de histogramas');
114 subplot(313);
115 stem(hace, 'color', 'y', 'markeredgecolor', myYellow); legend('ACE'); grid minor;
116 axis([0, 256]);
117 subplot(312);
118 stem(he, 'color', myRed, 'markeredgecolor', 'r'); legend('HE'); grid minor;
119 axis([0, 256]);
120 subplot(311);
121 stem(ho, 'color', 'b', 'markeredgecolor', myBlue); legend('Original'); grid minor;
122 axis([0, 256]);
123
124 figure
125 title('Comparación de CDFs');
126 hold on
127 plot(cdfo, 'color', myGreen, 'linewidth', 2);
128 plot(cdfe, 'color', myRed, 'linewidth', 2);
129 plot(cdface, 'color', myYellow, 'linewidth', 2);
130 legend('Original', 'HE', 'ACE');
131 legend('location', 'southeast');
132 axis([0, 256]);
133 grid minor

```

```

134
135 figure
136 title('Comparación de CDFs con diferencial');
137 difovseq = cdfo-cdfe;
138 difovsace = cdfo-cdface;
139 hold on
140 stem(difovseq, 'color', myRed, 'markeredgecolor','r');
141 stem(difovsace, 'color', myYellow, 'markeredgecolor', 'y');
142 legend('HE', 'ACE');
143 % legend('location', 'southeast');
144 axis([0, 256]);
145 grid minor
146
147 figure
148 title('Comparación de imagenes');
149 subplot(313);
150 imshow(imace);
151 subplot(312);
152 imshow(imeq);
153 subplot(311);
154 imshow(im);
155
156
157 figure(10)
158 title('Comparación de histogramas (Original vs ACE)')
159 subplot(212)
160 stem(hace, 'color', myRed, 'markeredgecolor', 'r')
161 legend('ACE')
162 axis([0, 256])
163 subplot(211);
164 stem(ho, 'color', 'b', 'markeredgecolor', myBlue)
165 axis([0, 256])
166 legend('Original')
167
168 figure(11)
169 title('Comparación de CDFs');
170 hold on
171 plot(cdfo, 'color', myBlue, 'linewidth', 2);
172 plot(cdface, 'color', myRed, 'linewidth', 2);
173 legend('Original', 'ACE');
174 legend('location', 'southeast');
175 axis([0, 256]);
176 grid minor
177
178 figure
179 imshow(im)
180 figure
181 imshow(imace)

```

C.2. ace.m

Listing 2: ace.m

```

1 % El eta define el modo de operacion
2 % -> ace = img - eta (img - he)
3 % eta = 1 -> solo he
4 % eta = 0 -> solo img
5 function EQ = ace(img, eta)
6
7 % Defino las regiones
8 dark = 0:84; mid = 85:169; bright = 170:255;
9 index_dark = img>85; index_mid = img>84 & img<170; index_bright = img>169;
10 im_dark = img(index_dark); im_mid = img(index_mid); im_bright = img(index_bright)
11 ;
12 max_ranges = [dark(end)-dark(1) mid(end)-mid(1) bright(end)-bright(1)];
13 ymin = [dark(1) mid(1) bright(1)];
14
15 %%%%%%%%%%

```

```

16  %% HE Dark
17  im_region = im_dark;
18  region = 1;
19  wf = weighted_factor(histo(im_region, max_ranges(region)), dark, eta(region))
20  cdf_region = cdf(histo(im_region));
21
22  % Asigno a cada pixel su correspondiente punto en la CDF
23  ev_cdf = cdf_region(im_region+1); % +1 porque se indexa a partir de 1 y no 0
24
25  % Ecualizo la region
26  EQ_dark = uint8(ev_cdf * max_ranges(region) + ymin(region));
27
28  % Asigno los pixels ecualizados a su posicion correcta en la imagen
29  EQ(find(index_dark)) = EQ_dark;
30
31  % Idem pero utilizando el peso wf para la ecualizada y la original
32  aux = wf * EQ_dark;
33  auxIm = (1-wf) * im_region;
34  wEQ(find(index_dark)) = aux;
35  wIm(find(index_dark)) = auxIm;
36
37  %%%%%%%%%%
38  %% HE Mid
39  im_region = im_mid;
40  region = 2;
41  wf = weighted_factor(histo(im_region, [dark mid]), [dark mid], eta(region))
42  cdf_region = cdf(histo(im_region));
43
44  % Asigno a cada pixel su correspondiente punto en la CDF
45  ev_cdf = cdf_region(im_region+1); % +1 porque se indexa a partir de 1!
46
47  % Ecualizo la region
48  EQ_mid = uint8(ev_cdf * max_ranges(region) + ymin(region));
49
50  % Asigno los pixels ecualizados a su posicion correcta en la imagen
51  EQ(find(index_mid)) = EQ_mid;
52
53  % Idem pero utilizando el peso wf para la ecualizada y la original
54  aux = wf * EQ_mid;
55  auxIm = (1-wf) * im_region;
56  wEQ(find(index_mid)) = aux;
57  wIm(find(index_mid)) = auxIm;
58
59  %%%%%%%%%%
60  %% HE Bright
61  im_region = im_bright;
62  region = 3;
63  wf = weighted_factor(histo(im_region, [dark mid bright]), [dark mid bright], eta(
    region))
64  cdf_region = cdf(histo(im_region));
65
66  % Asigno a cada pixel su correspondiente punto en la CDF
67  ev_cdf = cdf_region(im_region+1); % +1 porque se indexa a partir de 1!
68
69  % Ecualizo la region
70  EQ_bright = uint8(ev_cdf * max_ranges(region) + ymin(region));
71
72  % Asigno los pixels ecualizados a su posicion correcta en la imagen
73  EQ(find(index_bright)) = EQ_bright;
74
75  % Idem pero utilizando el peso wf para la ecualizada y la original
76  aux = wf * EQ_bright;
77  auxIm = (1-wf) * im_region;
78  wEQ(find(index_bright)) = aux;
79  wIm(find(index_bright)) = auxIm;
80
81
82  %%%%%%%%%%
83  %% Convierto el arreglo de pixels en una imagen
84  EQ = reshape(wIm + wEQ, size(img));

```

```
85
86 end
```

D. Funciones auxiliares

D.1. cdf.m

Listing 3: cdf.m

```
1 function F = cdf(pdf)
2   F = cumsum(pdf);
3 end
```

D.2. histo.m

Listing 4: histo.m

```
1 function h = histo(img, hist_range)
2   if nargin < 2
3       hist_range=0:255;
4   elseif length(hist_range(:)) == 1
5       hist_range=0:hist_range;
6   else
7       hist_range=hist_range;
8   end
9
10  im = img(:);
11
12  h = sum(im==hist_range)/length(im);
13 end
```

D.3. equalizer.m

Listing 5: equalizer.m

```
1 function EQ = equalizer(img, max_range)
2   cdfo = cdf(histo(img, max_range));
3
4   cdfmax = max(cdfo);
5   cdfmin = min(cdfo);
6
7   ev_cdf = cdfo(img+1); % +1 porque se indexa a partir de 1!
8   EQ = uint8((ev_cdf - cdfmin) * max_range(end) / (cdfmax - cdfmin));
9   EQ = uint8(EQ * 255);
10 end
```

D.4. weighted_factor.m

Listing 6: weighted_factor.m

```
1 function wf = weighted_factor(h, region, eta)
2   ymax = max(region); % Máximo valor de luminancia de la región
3   N = sum(h); % Cantidad de puntos
4   m = mean(region.*h)*ymax/N; % Luminancia media de la región
5
6   % Calculo del sigma maximo, suponiendo que se concentra todo en los extremos
7   % -> max_sigma = (1/N) * N/2 * 2 * (ymax+m)*(ymax-m+1)/2;
8   max_sigma = ymax/2;
9
10  % Calculo la pseudo varianza
11  sigm = (1/N) * sum(h .* abs(region-m));
```

```
12  
13     wf = eta * (1 - abs(2*sigm/max_sigma - 1));  
14 end
```

D.5. psnr.m

Listing 7: psnr.m

```
1 function val = psnr(orig, new)  
2     mse = (orig-new).^2;  
3     mse = sum(sum(mse))/rows(orig)/columns(orig);  
4  
5     m = max(max(orig));  
6  
7     val = 20*log10(m) - 10*log10(mse);  
8 end
```