International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol. 10, No. 12 (2017), pp.35-62
http//dx.doi.org/10.14257/ijsip.2017.10.12.04

Contrast Enhancement Techniques: A Survey

Karishma Rao[1], Archana Agarwal[2] and Sangeeta Dhall[3]

Student[1], Assistant Professor[2, 3], Electronics Engineering Department
YMCAUST, Faridabad, India
Karishmar.93@gmail.com[1]
archana_aggrawal@yahoo.co.in[2] and sangeeta_dhall@yahoo.co.in[3]

Abstract

Image Contrast enhancement is one of the important requirement in the field of image processing. It is a process of redistribution the image pixels count over the full intensity range so that image objects stand out more clearly and as a result perception quality improves. Many techniques, for the same are proposed by different researchers in literature and evaluated on the basis of diverse performance metrics such as Absolute Mean Brightness Error (AMBE), Average Information Content (AIC), Mean Square Error (MSE), Peak Signal to Noise Ratio (PSNR) and many more. This paper is an effort to perform exhaustive comparison of various Global and Local Contrast enhancement techniques existing in the literature, describing strong and weak areas of various techniques with respect to given performance metrics. These all techniques are implemented in MATLAB 7.0.2. Non-Parametric Modified Histogram Equalization (NPBHE) has shown excellent results in term of various performance metrics. These results can be quite beneficial for the researchers working in this direction.

***Keywords**: AIC, AMBE, Contrast Enhancement, MSE, NPBHE, PSNR*

1. Introduction

Image quality improvement is the foremost important step of digital image processing, in order to have best quality pictures. Perceptual quality can be improved by number of ways such as: removal of noise, edge enhancement, contrast enhancement, defogging *etc.* Contrast enhancement is one of the vital needs in this area. It is a process in which the image pixels in an image are adjusted in such a way, that each and every object is distinguishable from each other. It has wide application areas such as remote sensing, radar, consumer electronics, digital photography, medical image processing, texture synthesis, and many more image/video processing applications. Figure 1 shows the effect of contrast enhancement on a dull image (original image) after applying contrast improvement process. It is clearly visible from the figure that the output image is much clear, informative and has better picture quality.

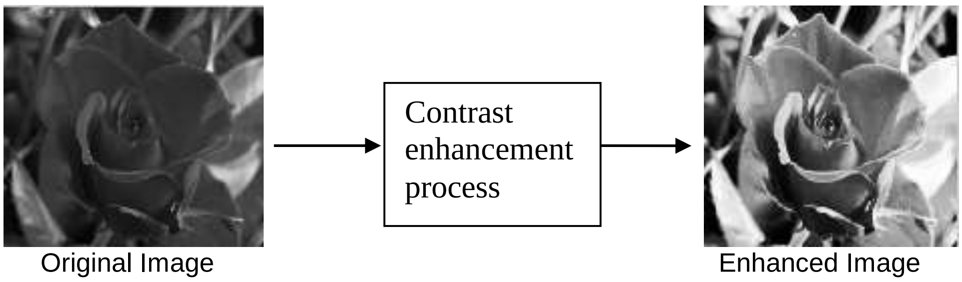

Figure 1. Contrast Enhancement Process

Received (May 19, 2017), Review Result (November 24, 2017), Accepted (December 1, 2017)

ISSN: 2005-4254 IJSIP

Contrast enhancement techniques can be divided into two sub-categories Global contrast enhancement and Local contrast enhancement as shown in Figure 2. The former uses the single transformation function to transform all the pixels of an input image. It is fast, simple and suitable for overall enhancement of an image but it is not able to enhance the local details of an image whereas, latter technique uses a small window that drifts through each and every pixel of an input image consecutively. Its computational time is high and sometimes it enhances noise. Both of these techniques have certain pros and cons. This paper is an effort to compare available contrast enhancement techniques in literature based on given performance metrics and identify the best technique as per results.

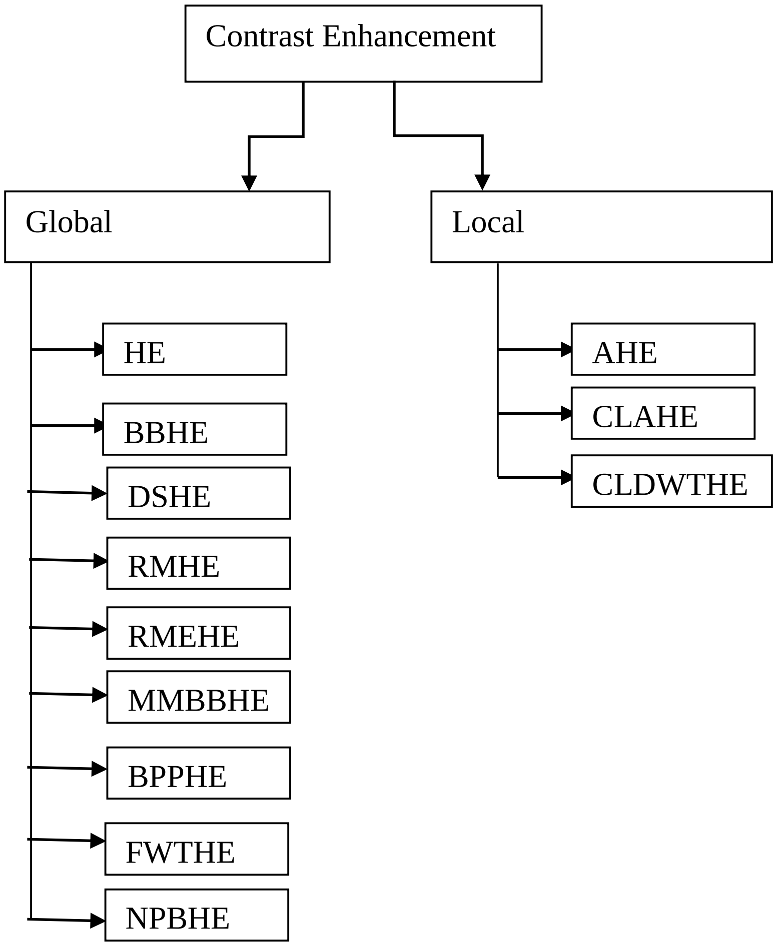

Figure 2. Classification of Contrast Enhancement Schemes

The rest of the paper is organized as follow: Section 2 gives the description of different image contrast enhancement techniques. Section 3 contains the simulation setup parameters. Section 4 shows experimental results based on given performance metrics and Section 5 gives overall conclusion which is followed by references.

2. Literature Survey

2.1. Histogram Equalization (HE)

This is a primal technique for image contrast enhancement. The first step that it requires is the formation of histogram. In the histogram of a low contrast image, pixels are either concentrated at low grey level or at high grey level [1]. Application of HE will flatten or stretch the histogram of low contrast image. But this flattening property leads to some other issues like formation of check board like image and more brighten image.

Algorithm for Histogram equalization

```
a=image read('grey.jpg')
[row col]=size(a)             %calculate rows and columns of image
n=row*col;                                  %no of pixels values in image
for i=1:row
for j=1:col
   value=a(i,j);
   f(value+1)=f(value+1)+1; %calculate count for each pixel value in image
   pdf(value+1)=f(value+1)/n; %calculate pdf for each pixel value in image
end
end
sum=0,l=255
for i=1:row
   sum=sum+pdf(i);
cdf(i)=sum;                                  %calculate cdf for each pixel value in
image
   out(i)=round((cdf(i)*l));   %calculate modified histogram
end
for i=1:row
for j=1:col
    ah(i,j)=out(a(i,j)+1);                   %formation of final enhanced image
 end
end
```

Where,
f stands for count of each pixel intensity
pdf stands for probability density function
cdf stands for cumulative distribution function
out represents variable that holds value of modified histogram at each grey level
n stands for total no of pixels in an image
sum represents variable that holds pdf value

In the histogram equalization, first step is to calculate count of each grey level values (0…256) which is designated as f. In the second step probability distribution function (pdf) is calculated by dividing count of each grey level value (f) by the total count (n). In next step cumulative distribution function is determined at each grey level by accumulation of pdf and is designated as cdf. Finally, after applying pre-defined transformation function final enhanced image is produced as equalized image.

2.2. Brightness Preserving Bi-Histogram Equalization (BBHE)

Histogram equalization improves the contrast of an image but it deteriorates image in other parameters like brightness and entropy. To overcome this, BBHE technique is used. It divides the histogram of input image into two sub-histograms based on mean value and then applies HE individually. As a result, enhanced image will be produced with Natural outlook and it also preserves brightness. [1]

Algorithm for Brightness Preserving Bi-Histogram Equalization

```
x=imread('airplane_gray2.jpg');
[row col]=size(x)                    %calculates rows and columns of image
image_mean = mean(x);   %calculate mean of input image
hist_lower=histogram(1:mean);                  %calculate lower histogram
hist_upper=histogram(mean+1, 255);                       %calculate upper histogram

pdf_lower = hist_lower/sum(hist_lower);   %calculate pdf for lower histogram
pdf_upper = hist_upper/sum(hist_upper);    %calculate pdf for upper histogram
hist_lower_cdf(1) = pdf_lower(1);  %assigning first pdf to first cdf of lower histogram
hist_upper_cdf(1) = pdf_upper(1);       %assigning first pdf to first cdf of upper histogram
 for k = 2:256
hist_lower_cdf(k)=hist_lower_cdf(k-1)+pdf_lower(k);%calculate cdf for lower histogram
hist_upper_cdf(k)=hist_upper_cdf(k-1)+pdf_upper(k);%calculate cdf for upper histogram
end
for i =1:row
  for j =1:col
g_val = x(i,j);
      if(g_val<=image_mean)
equalized_img(i,j)=0+round(((image_mean-0)
*hist_lower_cdf(g_val+1)));
      else
equalized_img(i,j)=(image_mean+1)+round(((255-
(image_mean+1))*hist_upper_cdf(g_val+1)));     %formation of final enhanced image
      end
  end
end
```

Where,
hist_lower stands for lower histogram before mean value
hist_upper stands for upper histogram before mean value
Pdf_lower stands for probability density function of lower histogram
Pdf_upper stands for probability density function of upper histogram
hist_lower_cdf stands for cumulative distribution function of lower histogram
hist_upper_cdf stands for cumulative distribution function of upper histogram
n stands for total no of pixels in an input image

In BBHE technique, firstly, mean of the input image is calculated as image_mean. Based on this mean value histogram of an input image is divided into two sub histograms which are hist_lower and hist_upper. Now, HE technique will be applied on each histogram independently by calculating pdf_lower, hist_lower_cdf for hist_lower and pdf_upper, hist_upper_cdf for hist_upper and after applying predefined transformation function formula final enhanced image has been obtained and it is represented as equalized image.

2.3. Dualistic Sub-Image Histogram Equalization (DSHE)

In BBHE technique Absolute Mean Brightness Error (AMBE) is improved in comparison to HE but it is not able to retain the Average Information Content (AIC). Higher values of Entropy or AIC indicates richness of the details in the output image [2]. Thus, for this improvement DSHE technique is used. This technique divides histogram of the image into two parts such as both parts have equal area under them. Median of an image is taken as basis of this division.

Algorithm for Dualistic Sub-Image Histogram Equalization

```
x=imread('airplane_gray2.jpg');
[row col]=size(x)                    %calculates rows and columns of image
image_median = median(x)
hist_lower=histogram(1:mean);                          %calculate lower histogram
hist_upper=histogram(mean+1, 255);                     %calculate upper histogram
pdf_lower = hist_lower/sum(hist_lower);   %calculate pdf for lower histogram
pdf_upper = hist_upper/sum(hist_upper);   %calculate pdf for upper histogram
hist_lower_cdf(1) = pdf_lower(1);%assigning first pdf to first cdf of lower histogram
hist_upper_cdf(1)=pdf_upper(1);%assigning first pdf to first cdf of upper histogram
 for k = 2:256
hist_lower_cdf(k) =  hist_lower_cdf(k-1) + pdf_lower(k);  %calculate cdf for lower
histogram
hist_upper_cdf(k) =  hist_upper_cdf(k-1) + pdf_upper(k);  %calculate cdf for upper
histogram
end
for i =1:row
   for j =1:col
g_val = x(i,j);
        if(g_val<=image_median)
equalized_img(i,j)=0+ round(((image_median -
                    0)*hist_lower_cdf(g_val+1)));
        else
equalized_img(i,j)= (image_median+1) +round(((255-
                    (image_median+1))*hist_upper_cdf(g_val+1))); %formation of final
enhanced image
        end
     end
  end
```

Where,
hist_lower stands for lower histogram before median value
hist_upper stands for upper histogram after median value
pdf_lower stands for probability density function of hist_lower
pdf_upper stands for probability density function of hist_upper
hist_lower_cdf stands for cumulative distribution function of hist_lower
hist_upper_cdf stands for cumulative distribution function of hist_upper
n stands for total no of pixels in an image

In DSHE technique, firstly median of the input image is calculated as image_median. Based on this median value histogram of an input image is divided into two sub histograms which are designated as hist_lower and hist_upper. Now, HE technique will be applied on each histogram independently by calculating pdf_lower, hist_lower_cdf for hist_lower and pdf_upper,hist_upper_cdf for hist_upperand after applying predefined transformation function formula final enhanced image has been obtained and it is represented as equalized image.

2.4. Recursive Mean Separate Histogram Equalization (RMSHE)

BBHE is not able to preserve mean brightness to that level, so that it can be used in applications like consumer electronics. RMSHE technique is the extension of the BBHE. As in BBHE input histogram is divided only once, in contrast RMSHE technique recursively divides the histogram into many sub-histograms [4]. This results in preservation of mean brightness. This recursion process can be repeated unless brightness is preserved up to the desired level.

Algorithm for Recursive Mean Separate Histogram Equalization

```
x=imread('airplane_gray2.jpg');
[row col] =size(x)
image_mean=mean(x);                          %calculate mean of image
hist_lower=histogram (1: mean);                      %calculate lower histogram
hist_upper=histogram(mean+1, 255);                  %calculate upper histogram
hist_lower_mean1 =mean(hist_lower);    %calculate mean of lower histogram
hist_lower1=histogram(1: hist_lower_mean1);%calculate lower histogram before
mean value of lower histogram
hist_lower2=histogram(hist_lower_mean1+1:image_mean);   %calculate upper
histogram after mean value of upper histogram
hist_upper_mean2 =mean(hist_upper);   %calculate mean of upper histogram
hist_upper1=histogram(image_mean+1:hist_lower_mean)       %calculate        lower
histogram
before mean value of upper histogram
hist_upper2= histogram(hist_lower_mean2+2:256); %calculate upper histogram after
mean value of upper histogram
pdf_lower1=hist_lower1/sum(hist_lower1);   %calculate pdf of hist_lower1
pdf_lower2=hist_lower2/sum(hist_lower2);   %calculate pdf of hist_lower2
pdf_upper1 = hist_upper1/sum(hist_upper1);   %calculate pdf of hist_upper1
pdf_upper2 = hist_upper2/sum(hist_upper2);    %calculate pdf of hist_upper2
hist_lower1_cdf(1) = pdf_lower1(1);      %assigning first pdf to first cdf of hist_lower1
hist_lower2_cdf(1) = pdf_lower2(1);      %assigning first pdf to first cdf of hist_lower2
hist_upper1_cdf(1) = pdf_upper1(1);     %assigning first pdf to first cdf  of hist_upper1
hist_upper2_cdf(1) = pdf_upper2(1);     %assigning first pdf to first cdf of hist_upper2

for k = 2:256
hist_lower1_cdf(k)=hist_lower1_cdf(k-1)+pdf_lower1(k);%calculate cdf of
hist_lower1
        hist_lower2_cdf(k=hist_lower2_cdf(k-1)+pdf_lower2(k);%calculate cdf of
hist_lower2
       hist_upper1_cdf(k)=hist_upper1_cdf(k-1)+pdf_upper1(k);%calculate cdf of
hist_upper1
       hist_upper2_cdf(k)= hist_upper2_cdf(k-1) + pdf_upper2(k);%calculate cdf of
hist_upper2
 end
for i =1:sz(1)
```



```
  for j =1:sz(2)
      g_val4 = x(i,j);
        if(g_val4<=hist_lower_mean1)
equalized_img(i,j)=0+round(((hist_lower_mean1-
                  0)*hist_lower1_cdf(g_val4+1)));
elseif(g_val4<=image_mean)
equalized_img(i,j)= [(hist_lower_mean1+1)  +round(((image_mean-
                  (hist_lower_mean1+1))*hist_lower2_cdf(g_val4+1)));
        end
  end
end
for i =1:row
  for j =1:col
      g_val5 = x(i,j);
        if(g_val5<=hist_upper_mean2&&g_val5>image_mean)
equalized_img(i,j)= image_mean+1+round(((hist_upper_mean2-
                  image_mean+1)*hist_upper1_cdf(g_val5+1)));
elseif(g_val5>=hist_upper_mean2)
equalized_img(i,j)= (hist_upper_mean2+1)+ round(((255-
                  hist_upper_mean2+1)*hist_upper2_cdf(g_val5+1))); %formation of
enhanced image
        end
    end
 end
```

Where,
hist_lower stands for sub histogram before mean
hist_upper stands for sub histogram after mean
hist_lower1 stands for sub histogram before mean of hist_lower
hist_lower2 stands for sub histogram after mean of hist_lower
hist_upper1 represents sub histogram before mean of hist_upper
hist_upper2 represents sub histogram before mean of hist_upper
pdf_lower1 represents probability density function of histogram hist_lower1
pdf_lower2 represents probability density function of histogram hist_lower2
pdf_upper1 represents probability density function of histogram hist_upper1
pdf_upper2 represents probability density function of histogram hist_upper2
hist_lower1_cdf represents cumulative density function of histogram hist_lower1
hist_lower2_cdf represents cumulative density function of histogram hist_lower2
hist_upper1_cdf represents cumulative density function of histogram hist_upper1
hist_upper2_cdf represents cumulative density function of histogram hist_upper2

In RMSHE technique firstly, the mean of a histogram of an input image is calculated which is represented as image_mean and divide histogram of an input image into two sub histograms hist_lower and hist_upper. Then these two sub histograms are again divided into two based on their mean hist_lower_mean1 and hist_lower_mean2. First two sub histograms are represented as hist_lower1, hist_lower2 and next two sub histogram are represented as hist_upper1, hist_upper2.[4] The second step is to perform histogram equalization of these four sub histograms independently by calculating pdf_lower1, hist_lower1_cdf, hist_lower2_cdf pdf_lower2 for sub histograms hist_lower1 and hist_lower2 and calculating pdf_upper1, hist_upper1_cdf, pdf_upper2, hist_upper2_cdf for sub histograms hist_upper1 and hist_upper2. Finally, these four sub histograms are composed into final enhanced image.

2.5. Recursive Dualistic Sub-Image Histogram Equalization (RDSHE)

DSHE is not able to preserve mean brightness up to that level so that it can use in applications where least brightness preservation is necessary. RDSHE technique is the extension of the DSHE technique. RMSHE technique, not only divide the histogram into two sub histograms as in the case of BBHE but it recursively separates the histogram into many sub-histograms. As a result, mean brightness reduces as long as recursion is performed. [5]

Algorithm for Recursive Dualistic Sub-image Histogram Equalization

---

```
x=imread('airplane_gray2.jpg');
[row col]=size(x)
image_median = median(x);   %calculate median of image
hist_lower=histogram(1:median);        %calculate lower histogram
hist_upper=histogram(median+1, 255);                    %calculate upper histogram
hist_lower_median1 =median (hist_lower);   %calculate mean of lower histogram
hist_lower1=histogram(1: hist_lower_median1 ); %calculate lower histogram before
                                                 medianvalue of lower histogram
 hist_lower2=histogram(hist_lower_median1+1:image_median );  %calculate upper
histogram after median value of upper histogram
hist_upper_median2 =median(hist_upper);   %calculate median of upper histogram
hist_upper1=histogram(image_mean:hist_   upper_mean2);   %calculate   lower
histogram
before median value of lower histogram
 hist_upper2=histogram(hist_  upper_mean2+1:255);  %calculate  upper  histogram
after
median value of upper histogram
pdf_lower1=hist_lower1/sum(hist_lower1);                %calculate pdf of hist_lower1
 pdf_lower2=hist_lower2/sum(hist_lower2);               %calculate pdf of hist_lower2
pdf_upper1 = hist_upper1/sum(hist_upper1);              %calculate pdf of hist_upper1
pdf_upper2 = hist_upper2/sum(hist_upper2);              %calculate pdf of hist_upper2
hist_lower1_cdf(1) = pdf_lower1(1)       %assigning first pdf to first cdf of hist_lower1
hist_lower2_cdf(1) = pdf_lower2(1);      %assigning first pdf to first cdf of hist_lower2
hist_upper1_cdf(1) = pdf_upper1(1);%assigning first pdf to first cdf of hist_upper1
hist_upper2_cdf(1)  =  pdf_upper2(1);            %assigning  first  pdf  to  first  cdf  of
hist_upper2

for k = 2:256
    hist_lower1_cdf(k)=hist_lower1_cdf(k-1)+pdf_lower1(k);%calculate cdf of
hist_lower1
    hist_lower2_cdf(k)=hist_lower2_cdf(k-1)+pdf_lower2(k);%calculate cdf of
hist_lower2
    hist_upper1_cdf(k)=hist_upper1_cdf(k-1)+pdf_upper1(k);%calculate cdf of
hist_upper1
    hist_upper2_cdf(k)=hist_upper2_cdf(k-1)+pdf_upper2(k);%calculate cdf of
hist_upper2
end
for i =1:sz(1)
   for j =1:sz(2)
       g_val4 = x(i,j);
           if(g_val4<=hist_lower_median1)
```



```
equalized_img(i,j)=0+round(((hist_lower_median1-
                    0)*hist_lower1_cdf(g_val4+1)));
elseif(g_val4<=image_median)
equalized_img(i,j)= (hist_lower_median1+1) +round(((image_median -
                    (hist_lower_median1+1)*hist_lower2_cdf(g_val4+1)));
        end
    end
end
 for i =1:row
   for j =1:col
       g_val5 = x(i,j);
         if(g_val5<=hist_upper_median2&&g_val5>image_median)
equalized_img(i,j) = image_median+1 + round(((hist_upper_median2-
image_median+1)*hist_upper1_cdf(g_val5+1)));
elseif(g_val5>=hist_upper_median2)
equalized_img(i,j) = (hist_upper_median2+1)+ round(((255-
(hist_upper_median2+1))*hist_upper2_cdf(g_val5+1)));
%formation of enhanced
image
            end
        end
     end
```

Where,
hist_lower stands for sub histogram before median
hist_upper stands for sub histogram after median
hist_lower1 stands for sub histogram before median of hist_lower
hist_lower2 stands for sub histogram after median of hist_lower
hist_upper1 represents sub histogram before median of hist_upper
hist_upper2 represents sub histogram after median of hist_upper
pdf_lower1 represents probability density function of histogram hist_lower1
pdf_lower2 represents probability density function of histogram hist_lower2
pdf_upper1 represents probability density function of histogram hist_upper1
pdf_upper2 represents probability density function of histogram hist_upper2
hist_lower1_cdf represents cumulative density function of histogram hist_lower1
hist_lower2_cdf represents cumulative density function of histogram hist_lower2
hist_upper1_cdf represents cumulative density function of histogram hist_upper1
hist_u2_cdf represents cumulative density function of histogram hist_upper2

In RDSHE firstly, the median of a histogram of an input image is calculated as image_median, which is used to divide histogram of an input image into two sub histograms named as hist_lower, hist_upper [5]. The next step is to calculate median of hist_lower and hist_upper and these are designated as hist_lower_mean1, hist_lower_mean2. These two histograms are again subdivided into four sub histograms named as hist_lower1, hist_lower2, hist_upper1, hist_upper2. Next step is to calculate pdf_lower1, pdf_lower2, hist_lower1_cdf, hist_lower2_cdf for hist_lower1 and hist_lower2 and calculate pdf_upper1, pdf_upper2, hist_upper1_cdf, hist_upper2_cdf for hist_upper1 and hist_upper2 in order to perform HE on these four sub histograms. Finally, enhanced image is obtained.

2.6. Minimum Mean Brightness Error Bi-Histogram Equalization (MMBEBHE)

MMBEBHE technique is broad version of the BBHE. MMBEBHE has been introduced to have better results in term of Absolute Mean Brightness Error as compared to BBHE, DSHE, RMHE and RMEHE [3]. Like BBHE, this method also separate input histogram into two sub-histograms, but the separating point is set by finding the minimum mean brightness error between the input and the enhanced image.

Algorithm for Minimum Mean Brightness Error Bi-Histogram Equalization

---

```
a=imread('grey.jpg')
 [row col]=size(a)                    %calculate rows and columns of image
n=row*col;   %no of pixels values in image
for(i=1:row)
   for(j=1:col)
        value=x(i,j);
        f(value+1)=f(value+1)+1;                    %calculate count for each pixel value in
image
        pdf(value+1)=f(value+1)/n;       %calculate upper histogram after mean value
  end
end
for(i=1:sz(1))
 s=s+(i*pdf(i));   %calculate mean of image
end
EX=round(s);
for(i=1:1:256)
   if(i==1)
      EY(i)=(l*(1-pdf(i)))/2;   %calculating mean of output image
      MBE(i)=(EY(i)-EX);   %calculate AMBE for each grey level
   else
       EY(i)=EY(i-1)+(1-(l*pdf(i)))/2;  %calculating mean of output image
      MBE(i)=(EY(i)-EX);    %calculate AMBE for each grey level
   end
 end
XT=round(abs(min(MBE)))    %calculating threshold XT that yield minimum MBE
hist_lower=histogram(1:XT);        %calculate lower histogram before threshold
hist_upper=histogram(XT+1, 255);          %calculate upper histogram after threshold

pdf_lower = hist_lower/sum(hist_lower);   %calculate pdf for lower histogram
pdf_upper = hist_upper/sum(hist_upper);    %calculate pdf for upper histogram
hist_lower_cdf(1) = pdf_lower(1);  %assigning first pdf to first cdf of lower histogram
hist_upper_cdf(1) = pdf_upper(1); %assigning first pdf to first cdf  of lower histogram
for k = 2:row
hist_lower_cdf(k)=hist_lower_cdf(k-1)+pdf_lower(k);%calculate    cdf    for    lower
histogram
hist_upper_cdf(k) =  hist_upper_cdf(k-1) + pdf_upper(k); %calculate cdf for upper
histogram
 end
for i =1:row
     for j =1:col
g_val = x(i,j);
             if(g_val<=XT)
```



```
equalized_img(i,j)=0 +round(((XT-
                  0)*hist_lower_cdf(g_val+1))); %formation of final enhanced image
         else
equalized_img(i,j)=XT+1+round(((255-
                  XT+1)*hist_upper_cdf(g_val+1))); %formation of final enhanced image
        end
     end
 end
```

Where,
hist_lower stands for lower histogram before threshold value
hist_upper stands for upper histogram after threshold value
f stands for count of each pixel intensity
pdf stands for probability density function
Pdf_lower stands for probability density function of histogram1
Pdf_upper stands for probability density function of histogram2
hist_lower_cdf stands for cumulative distribution function of histogram1
hist_upper_cdf stands for cumulative distribution function of histogram2
n stands for total no of pixels in an image

In MMBEBHE technique firstly, the mean of input histogram is calculated which is designated as image_mean, then for each gray level (0…256) Absolute Mean Brightness Error (MBE) is calculated [3]. The second step is to find out threshold value XT that yields minimum Absolute Mean Brightness Error (MBE). The third step is to divide input histogram into two sub histogram hist_lower, hist_upper based on threshold value XT. And next step is to perform histogram equalization on two sub histograms independently. Finally, two sub histograms are composed into final enhanced image.

2.7. Bi-Histogram Equalization with a Plateau Limit (BBPHE)

HE, BBHE, RMHE, RDSHE, MMBBHE all these contrast enhancement techniques are complex to implement and require a large computation time and in addition to this, most of the enhancement techniques shows large amount of enhancement in enhanced image. BBPHE is a broad version of the BBHE. Similar to BBHE, BBPHE first divide input histogram into two sub-histograms. Then these two sub histograms are clipped based on plateau limit. This results in less computation time and fewer enhancements in enhanced image [6].

Algorithm for Bi-Histogram Equalization with a Plateau Limit

```
x=image read('airplane_gray2.jpg');
[row col]=size(x)
image_mean = mean(x);   %calculate mean of input image
hist_lower=histogram(1:mean);        %calculate lower histogram
hist_upper=histogram(mean+1, 255);                %calculate upper histogram
TL =mean(hist_lower);   %calculate lower plateau limit
TU = mean(hist_upper)   %calculate upper plateau limit
 for i = 1:row
    if(hist_lower(i)<=TL)
histclip_lower(i)=hist_lower(i);
   else
```

```
histclip_lower(i)=TL;                              %clipping the lower histogram
  end
end
for i = 1:row
   if(hist_upper(i)<=TU)
histclip_u(i)=h_u(i);
   else
histclip_upper(i)=TU;                              %clipping the upper histogram
   end
end
pdf_lower = hc_l/sum(hc_l);   %calculate pdf for lower histogram
pdf_upper = hc_u/sum(hc_u);   %calculate pdf for upper histogram
hist_lower_cdf(1) = pdf_lower(1); %assigning first pdf to first cdf of lower histogram
hist_upper_cdf(1) = pdf_upper(1);   %assigning first pdf to first cdf of upper histogram
for k = 2:256
hist_lower_cdf(k)=hist_lower_cdf(k-1)+pdf_lower(k); %calculate cdf for lower
histogram
hist_upper_cdf(k)=hist_upper_cdf(k-1)+pdf_upper(k);      %calculate cdf for upper
histogram
end
for i =1:row
   for j =1:col
g_val = x(i,j);
         if(g_val<=image_mean)
equalized_img(i,j)=0+round(((o_mean-
                    0)*hist_lower_cdf(g_val+1))); %formation of final enhanced image
         else
equalized_img(i,j) = (o_mean+1)+ round(((255-
                    (o_mean+1))*hist_upper_cdf(g_val+1))); %formation of final
enhanced image
         end
     end
end
```

Where,
hist_lower stands for sub histogram before mean
hist_upper stands for sub histogram after mean
pdf_lower stands for probability density function of hist_lower
pdf_upper stands for probability density function of hist_upper
hist_lower_cdf stands for cumulative distribution function of hist_lower
hist_lower_cdf stands for cumulative distribution function of hist_upper
n stands for total no of pixels in an image
histclip_lower clipped histogram before mean
histclip_upper clipped histogram after mean

In BBPHE, firstly mean of the input image is calculated as image_mean. Based on this mean value histogram of an input image is divided into two sub histograms which are denoted as hist_lower and hist_upper. Next step is to clip these two sub histograms so that histclip_lower is obtained by clipping lower histogram hist_lower based on plateau limit TL and histclip_upper is obtained by clipping upper histogram hist_upper based on plateau limit TU. Now, HE technique will be applied on each histogram independently by calculating pdf_lower, hist_lower_cdf for histclip_lower and pdf_upper, hist_upper_cdffor histclip_upperand after applying predefined transformation function final enhanced image has been obtained and it is represented as equalized image.

2.8. Fast image Contrast Enhancement based on Weighted Threshold Histogram Equalization (FWTHE)

FWTHE is improved version of standard histogram equalization in term of level of enhancement and unnatural look [10]. It basically adjusts the probability distribution function between upper and lower threshold values. After obtaining modified probability distribution function standard HE is applied.

Algorithm for Fast image contrast enhancement based on Weighted Threshold Histogram Equalization

```
a=image read('airplane_gray2.jpg');
v=0.5,r1=0.5;                   %assigning value to power law function
pl=0.0001;                                        %assigning value to lower threshold
[row col]=size(a)               %calculate rows and columns of image
n=row*col;                                        %no of pixels values in image
for i=1:row
   for j=1:col
      value=a(i,j);
      f(value+1)=f(value+1)+1;                    %calculate count for each pixel value in
image
      pdf(value+1)=f(value+1)/n;         %calculate pdf for each pixel value in image
   end
end
pmax=max(pdf); %calculating peak value of pdf
pu=v*pmax; %calculating upper threshold value
for i=1:row
   if(pdf(i)>pu)
pwt(i)=pu;
elseif(pdf(i)>=pl&&pdf(i)<=pu)
pwt(i)=(((pdf(i)-pl)/(pu-pl))^r1)*pu;
   else
pwt(i)=0;   %calculate modified pdf between upper and lower threshold
   end
end
sum=0,l=255
for i=1:row
   sum=sum+pdf(i);
cdf(i)=sum;                                       %calculate cdf for each pixel value in
image
   out(i)=round((cdf(i)*l));                      %calculate modified histogram
end
for i=1:row
   for j=1:col
      ah(i,j)=out(a(i,j)+1);                      %formation of final enhanced image
   end
end
```

Where,

pu stands for upper threshold level
pl stands for lower threshold level
f stands for count of each pixel intensity
pdf stands for probability density function
cdf stands for cumulative distribution function
out represents modified histogram of an input image
pwt stands for modified probability density function
n stands for total no of pixels in an image

In the FWTHE, first step is to calculate count ofeach grey level which is denoted as f. The second step is to find out probability distribution function (pdf), which is calculated by dividing count of each grey level by the total count. Next step is to calculate modified probability distribution function and which is calculated as

1) Calculate upper threshold Pu and lower threshold Pl.
2) The second step is checking:

If pdf value is above upper threshold Pu then it is equal to Pu, if pdf value is below lower threshold Pl then it is equal to zero. The values between upper threshold Pu and lower threshold Pl is modified using particular formula as indicated in algorithm [10].

In next step cumulative distribution function is determined at each grey level by accumulation of pdf and is designated as cdf. Finally, after applying pre-defined transformation function modified histogram is obtained which is designated as out and in last step final enhanced image is produced as equalized image

2.9. Non-Parametric Modified Histogram Equalization (NPMHE)

Histogram equalization is the most widely used technique for contrast enhancement but it's not able to sustain in many parameters such as a Degree of Entropy Un-preservation, Average Information Content, Absolute Mean Brightness Error, Edge Based Contrast Measure. NPMHE technique basically modifies histogram of an input image and then applies HE [8]. NPMHE shows better results as compared to HE, BBHE, DSHE, RMHE, RDSHE, MMBBHE in most of the parameters.

Algorithm for Non-Parametric Modified Histogram Equalization

```
A=read image ('cameraman.jpg').
[row col] = size(A)    %calculating rows and columns of an input image
hist = imhist(A)                              %calculate histogram of an image
sume =0;L=255;
F = abs(int16(a(:,3:n))-int16(a(:,1:n-2)));
[r,c,v] = find(F>6)
d = a(sub2ind(size(a),r,c));
hist_modifiy = hist(double(d),256) /length(r);%calculate modified histogram
uniform_pdf = ones(L,1)/L                         %calculating uniform probability density
function
for(i=1:1:row)
   if(hist(i)>(1/L))
hist_clip(i) = (1/L);   %calculate clipped histogram
  else
hist_clip(i) = hist(i);                    %calculate clipped histogram
end
end
```



```
Mu = sum(uniform_pdf - hist_clip)        %calculate measure of un equalization
mod_pdf=(Mu*hist_modify)+((1-Mu)*uniform_pdf) %calculate modified pdf
for(i=1:1:row)
sume = sume+mod_pdf (i)
cdf(i) = sume;                        %calculate cdf for each pixel intensity
   out(i) = round((cdf(i)*l)+0.5)        %calculate final transformation function
end
for(i=1:row)
    for(j=1:col)
     ah(i,j)=  out(a(i,j)+1);            %formation of final enhanced image
end
end
```

Where,
f stands for count of each pixel intensity
hist_modify is modified histogram
uniform_pdf is uniform probability density function
hist_clip is clipped histogram
Mu is represents measure of unequalization
mod_pdf is modified probability density function
pdf stands for probability density function
cdf stands for cumulative distribution function
out represents final transformation function that holds values of modified histogram
n stands for total no of pixels in an image

In the NPMHE, firstly modified histogram of an input image is calculated which is represented as hist_modify. The second step is to calculate Measure of un-equalization Mu, which is defined as the difference between uniform probability distribution function uniform_pdf and Clipped histogram hist_clip. Clipped histogram (hist_clip) can be obtained by applying the condition: If values of original histogram (hist) are greater than 1/L then the values of hist_clip become equal to 1/L otherwise values of original histogram hist are assigned. Modified probability distribution function (mod_pdf) is calculated by including all these values. In next step cumulative distribution function is determined at each grey level by accumulation of pdf and is designated as cdf. Finally, after applying pre-defined transformation function modified histogram is obtained which is denoted as out and at last step final enhanced image is produced as equalized image [8].

2.10. Adaptive Histogram Equalization (AHE)

In the low contrast images, the important part may occupy only at relatively narrow range of gray level, majority of gray levels are occupied by “uninteresting areas” such as background and noise. These “uninteresting areas” may also generate large counts of pixels over the gray level and hence, large peaks in the histogram. In the case of global histogram equalization, it amplifies the image noise and does not adapt to local contrast requirements. AHE is a modified histogram equalization procedure that optimizes contrast enhancement based on local image data.

Algorithm for Adaptive Histogram Equalization

```
r=image read('airplane_gray2.jpg');
j = adapthisteq(r);  %applying function of AHE over the image
```

Where,
hist_lower stands for lower histogram before mean value
hist_upper stands for upper histogram before mean value
Pdf_lower stands for probability density function of lower histogram
Pdf_upper stands for probability density function of upper histogram
hist_lower_cdf stands for cumulative distribution function of lower histogram
hist_upper_cdf stands for cumulative distribution function of upper histogram
n stands for total no of pixels in an input image

AHE scheme first divide the image into the lattice of rectangular contextual regions, and a transformation function is obtained that performs gray scale mapping or it applies standard histogram equalization in each rectangular region. Desired number of contextual regions and the size of the regions depend on the type of an input image, and default size of the region in AHE function is 8x8 (pixels). In addition, a bi-linear interpolation is also used in the AHE function in order to assemble final AHE image.

2.11. Contrast Limited Adaptive Histogram Equalization (CLAHE)

CLAHE is an improved version of adaptive histogram equalization method. The AHE technique causes over-amplification of noise in the homogenous regions of image. It was developed to prevent this over-amplification of noise in homogenous regions. CLAHE limits the amplification of noise in the image with the use of Clip limit so that image appears more natural. [14]

Algorithm for Contrast Limited Adaptive Histogram Equalization

```
r=image read('airplane_gray2.jpg');
A=adapthisteq(r,'clipLimit',0.02,'Distribution','rayleigh');   %applying function of
CLAHE over the image
```

Where,
ClipLimit used for contrast limiting and its value is between(0 to 1).
Distribution indicates the desired histogram shape.

CLAHE technique first divide the image into the lattice of rectangular contextual regions, and it applies standard histogram equalization in each rectangular region. The obtained histograms for each region are clipped by a clip limit in order to get rid of over amplification. Desired number of contextual regions and the size of the regions depend on the type of an input image, and default size of the region in this function is 8x8 (pixels). A transformation function is obtained that performs gray scale mapping. In addition, a bi-linear interpolation is also used in the CLAHE function in order to assemble final image

2.12. Combination of Contrast Limited adaptive histogram equalization and Discrete Wavelet Transform (CLDWT)

CLAHE technique is able to enhance the local details of an image but it tends to over enhance the image and to solve this problem a novel technique has been introduced which is named as CLDWT technique. CLDWT is combination of CLAHE and DWT.

Algorithm for Combination of Contrast Limited Adaptive Histogram Equalization and Discrete Wavelet Transform

```
X=image read('cameraman.bmp');
A2=ones (256,256)
[row col] = size(X)  %calculating rows and columns of an input image
[cA1,cH1,cV1,cD1]=dwt2(X,'haar');  %compute DWT of input image
h1=adapthisteq(cH1,'clipLimit',0.03,'Distribution','rayleigh'); %performing CLAHE of
low frequency component of input image
A0 = idwt2(cA1,ch1,cV1,cD1,'haar',Sx) %reconstruct the image using inverse DWT
z=1.5;
b=1.5;k=0;
for i=1:r
   for j=1:c
      value=x(i,j);
      H(i,j)=(value-min(min(x)))/(max(max(x))-min(min(x)))^z;          %weighted
average of
reconstructed    and original image
   end
end
AH=X.*H+b*A0.*(A2-H);                    %formation of final enhanced image
```

Where,

Clip Limit used for contrast limiting and its value is between (0 to 1).

Distribution indicates the desired histogram shape.

dwt2 stands for discrete Fourier transform

idwt2 stands for inverse discrete Fourier transform

In the CLDWT firstly, Discrete Fourier transform (dwt) is taken of an input image then input image is decomposed into high and low frequency component. Then the low frequency component (cH1) is used for the CLAHE enhancement process. High frequency component is unchanged because it contains most of the noise of an input image. Finally, image is reconstructed using inverse DWT (idwt) and it is donated by AO. The reconstructed (AO) and original images(X) are used in weighting average formula to get final enhanced image which is donated by AH.

3. Simulation Setup Parameter and Performance Metrics

3.1. Experimental Setup

The simulation set up parameter are given in Table 1

Table 1. Set up Parameters

SPECIFICATIONS	
Processor	Intel(R)core(TM)I3
Memory	3 GB
Operating System	Windows 10 Pro(32 bit)
Tool used	MATLAB
Version	10
Images Type	.jpg, png
Resolutions of Images	256*256
Images Color	Grey scale image

3.2. Performance Parameters

The comparison between the results obtained from various contrast enhancement techniques is carried out for grayscale images. Besides the visual results, comparison is made based upon the various performance parameters such as Absolute Mean Brightness Error (AMBE), Entropy or Average Information Contents (AIC), Contrast Improvement Index (CII), Mean Square Error (MSE), Peak Signal to Noise ratio (PSNR). Measures of Enhancement (EME), Entropy Preservation (DEI) are the two important quantitative measures used here for the performance analysis of various enhancement techniques.

3.2.1. Absolute Mean Brightness Error (AMBE)

It is the difference between mean
Of original and mean of enhanced image and is given as [1].

$$AMBE = |M(x) - M(y)|$$

Where,
M(x) is average intensity of input image
M(y) is average intensity of enhanced image

3.2.2. Average Information Contents (AIC)

The AIC is used to measure the content of an image. Higher value of Entropy indicates richness of the details in the enhanced image. Higher value of the AIC indicates that more information is brought out from the images. The average information contents or entropy is defined as [2]:

$$AIC = -\sum_{k=0}^{L-1} P(k) \log P(k)$$

Where P(k) is the probability density function of the kth gray Level.

3.2.3. Contrast Improvement Index (CII)

In order to evaluate the competitiveness of the different contrast enhancement techniques, the most well-known benchmark is image enhancement measure. It is used to compare the results of contrast enhancement methods. Contrast improvement can be measured using CII as a ratio. CII is defined as:

$$CII = \frac{C_{Proposed}}{C_{Original}}$$

Where C is the average value of the local contrast measured with3×3 window as:

$$\frac{\max - min}{\max + min}$$

Cproposed and Coriginal are the average values of the local contrast in the output and original images, respectively.

3.2.4. Mean Square Error (MSE)

The MSE is obtained as a cumulative of the square of the errors between the image obtained after equalization and the original image. Lower the value of MSE means lower is the error.

$$\mathrm{MSE} = \frac{1}{\mathrm{size}} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [f(i,j) - K(i,j)]^2$$

Where,
f(i,j) is the pixel value of original image,
K(i,j) is the pixel value of equalized image.

3.2.5. Peak Signal to Noise Ratio (PSNR)

It is defined as the ratio between the maximum possible power of a signal and the power of corrupting noise. It is the ratio of peak square value of pixels by mean square error (MSE). It is expressed in decibel (db). The PSNR is defined as:

$$\mathrm{PSNR} = 10.\log_{10}(\frac{\mathrm{MAX_I}^2}{\mathrm{MSE}})$$

Where,
MAXI represents maximum value of pixel of the image, MSE is the mean square error.

3.2.6. Measure of Enhancement (EME)

Measure of Enhancement is used to measure the level of enhancement obtained using a given enhancement algorithm and EME should be low as possible.

3.2.7. Degree of Entropy Un-preservation (DEI)

In order to measure the entropy preservation in an output transformed image, a parameter called ‘degree of entropy un-preservation’ (DEU) is defined and It is defined as difference in entropy between the processed and the input image [10].

DEI= |E(x) -E(y)|

Where,
E(x) is entropy of input image
E(y) is entropy of enhanced image

4. Results

In this section, results are displayed after comparison between various contrast enhancement techniques based on various performance parameter such as AMBE, Entropy or AIC, CII, MSE, PSNR, EME and DEI. Results are taken from the two

Test images 'airplane' and 'cameraman'. Average of these results is used to display in the chart for each performance metric.

4.1. Snapshots

This section includes original 'airplane' and 'cameraman' images and enhanced images which are obtained after applying different algorithms, as shown in Figure 3 and Figure 4

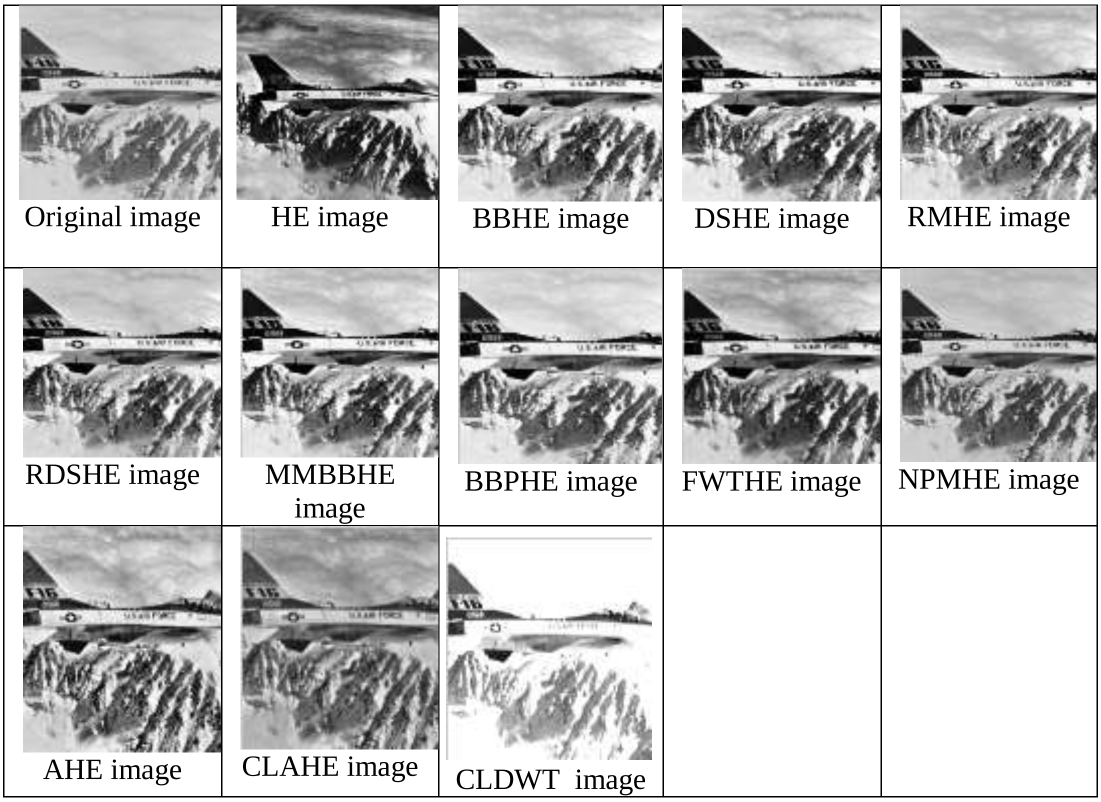

Figure 3. Shows the Snapshots of 'Airplane' Image

Figure 4. Shows the Snapshots of 'Cameraman' Image

4.2. Impact on AMBE

Impact of various Global and Local enhancement techniques on AMBE is shown in Figure 5. In this average result from two test images is taken. The followings point is noted

- It can be observed that in case of Global enhancement techniques both RMHE and RDSHE techniques have least AMBE value, in other words, it preserves mean brightness.
- MMBBHE and BBPHE also show superior results in term of brightness preservation.
- AHE has least AMBE value in case of Local enhancement schemes.

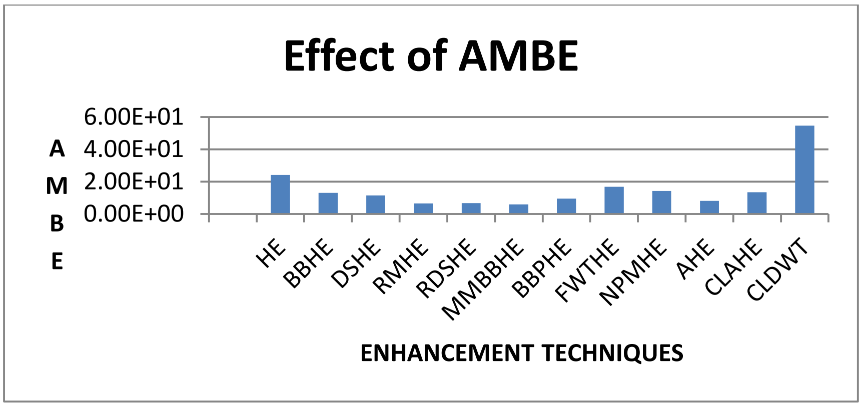

Figure 5. Absolute Mean Brightness Error chart

4.3. Impact on AIC

Impact of various Global and Local enhancement techniques on AIC is shown in Figure 6. It presents the average entropy values for various enhancement techniques applied to two standard images. AIC or Entropy is used to measure the richness of details in an image. Followings points can be noted

- BBPHE produces the highest entropy value, which indicates that the BBPHE is suitable to bring out average information content of the image.
- FWTHE, DSHE and NPMHE are second best techniques in term of entropy.
- AHE is best Local enhancement technique in term of AIC.

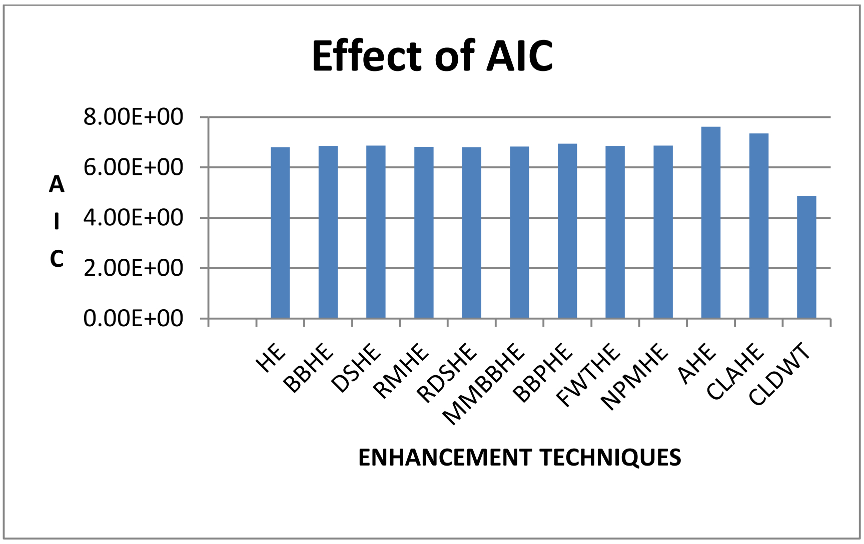

Figure 6. Average Information Content

4.4. Impact on Contrast Improvement Index

Impact of various in addition to brightness preservation and entropy of an image, there is also a parameter to measure the visual quality of an image and it is known as Contrast improvement index (CII). The graph of CII is indicated in Figure 7. Following points are observed:

- Contrast is more improved in the case of BBHE and BBPHE
- DSHE, RMHE performs better in term of Contrast improvement as compared to other enhancement techniques.
- Contrast is more improved in case of CLDWT Local technique.

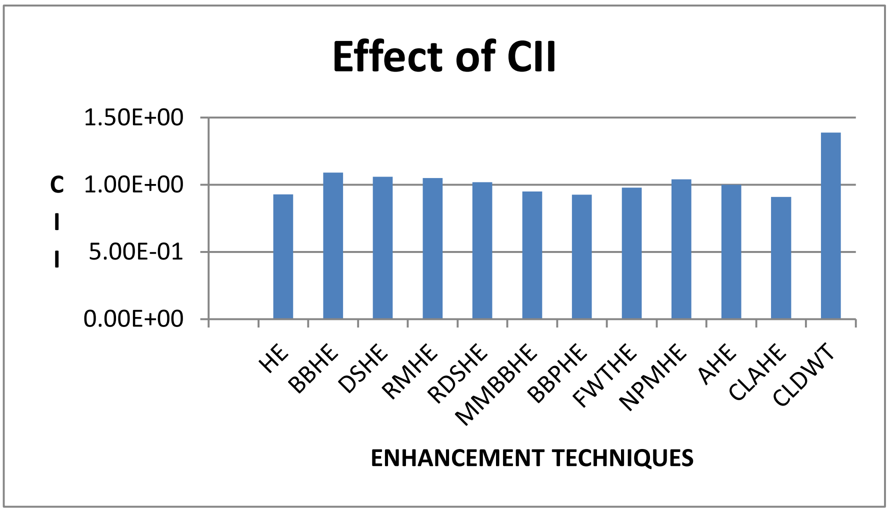

Figure 7. Contrast Improvement Index Chart

4.5. Impact on MSE

Impact of various enhancement techniques on MSE is shown in Figure 8. The MSE is the cumulative squared error between the enhanced and the original image. Following points can be noted

- MSE is least in the case of NPBHE and RMEHE techniques.
- RMHE and BBPHE also show better results as compared to other enhancement techniques exciting in the literature.
- AHE has least mean error.

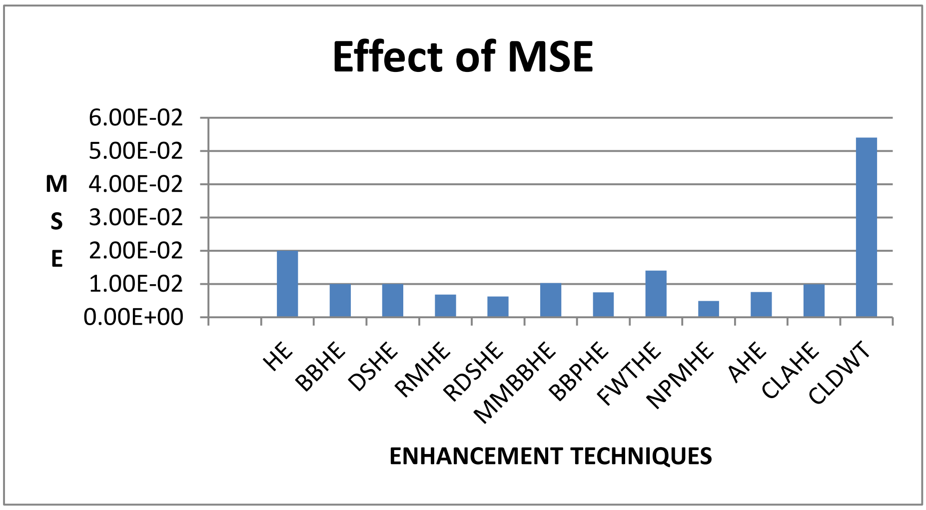

Figure 8. Mean Square Error Chart

4.6. Impact on PSNR

Impact of various enhancement techniques on PSNR is shown in Figure 9. It presents the average PSNR values for different enhancement techniques applied to two standard images. It represents the ratio between the maximum possible power of a signal and the power of corrupting noise. Following points can be noted

- NPMHE and RMEHE produce the highest value of a signal to noise ratio as compared to other techniques.
- PSNR is also maximum in the case of BBPHE and RMHE enhancement techniques.
- AHE is able to preserve signal to noise ratio.

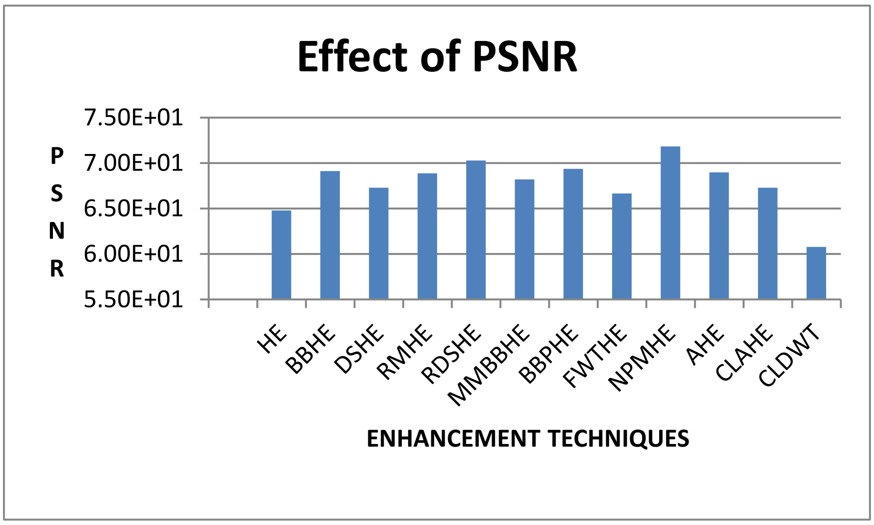

Figure 9. Signal to Noise Ratio Chart

4.7. Impact on DEI

Impact of various enhancement techniques on DEI is shown in Figure 10. It measures the entropy preservation in an output transformed image obtained from various enhancement techniques and it is the difference in entropy between the processed and the input image. Following points can be noted

- NPMHE shows outstanding results and it takes least DEI value as compared to all techniques exciting in literature.
- BBPHE is the second best technique that shows second least value of DEI.
- CLAHE has least DEI value as compared to all Local enhancement techniques.

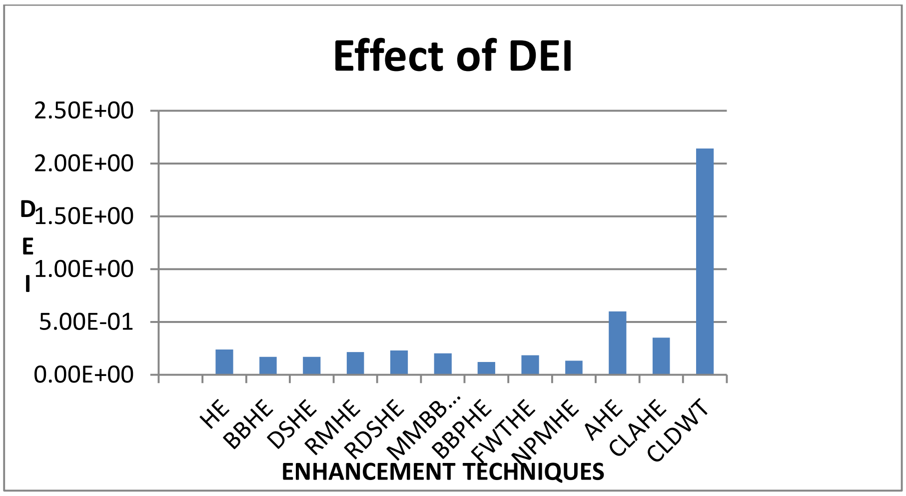

Figure 10. Degree of Preservation Chart

4.8. Impact on EME

Impact of various enhancement techniques on EME is shown in Figure 11. It presents the average EME values for various enhancement techniques applied on two standard images.

- HE has a large value of EME which shows that HE resultant image show a large amount of enhancement and which is undesirable.
- NPMHE and RMHE has the least value of EME
- RMEHE and BBPHE also show better results in term of EME.
- CLDWT shows least enhancement as compared to other Local enhancement techniques.

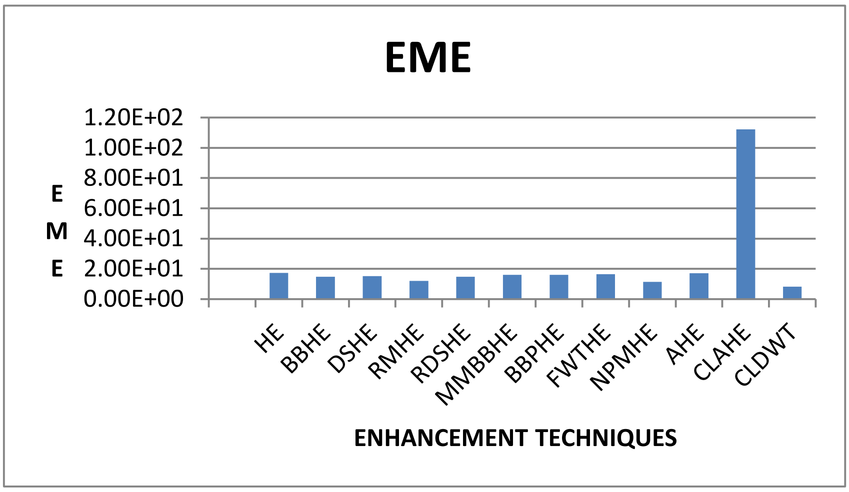

Figure 11. Measure of Enhancement Chart

5. Conclusion

This paper presents a comparative study of diverse Global and Local histogram equalization techniques in terms of AMBE, AIC, CII, MSE, PSNR, EME and DEI. Table-2 shows the overall comparison of various Global and Local contrast enhancement techniques present in literature. After the observation of experimental results of Global enhancement techniques, it is concluded that the brightness preservation, entropy is best handled by BBPHE at the same time BBPHE has the least error and highest PSNR. NPBHE has high entropy value, very low MSE and highest PSNR. Contrast is more improved in the case of BBPHE. Enhancement is moderate in case of BBPHE and very low in NPBHE. The degree of entropy preservation is low in case of BBPHE and very low in NPBHE. BBPHE takes less time to execute NPBHE produces the highest entropy value. Contrast is more improved by NPBHE. PSNR is maximum in NPBHE. In case of Local enhancement techniques AHE scheme is best in term of AMBE, MSE, PSNR and AIC. CLAHE has least DEI. CLDWT shows best results in term of enhancement. It is concluded that BBPHE and NPBHE show attractive results in most of the performance metrics as compared to other Global enhancement techniques existing in the literature. AHE is best in term of brightness and entropy whereas CLDWT is good in term of EME.

Table 2. Overall Comparison

GLOBAL							
	AMBE	**AIC**	**CII**	**MSE**	**PSNR**	**EME**	**DEI**
HE	VERY HIGH	VERY LOW	VERY LOW	VERY HIGH	VERY LOW	VERY HIGH	VERY HIGH
BBHE	MODERATE	MODERATE	VERY HIGH	MODERATE	HIGH	MODERATE	MODERATE
DSHE	MODERATE	HIGH	HIGH	MODERARE	LOW	LOW	MODERATE
RMHE	LOW	LOW	HIGH	LOW	MODERATE	LOW	HIGH
RMDHE	LOW	LOW	MODERATE	LOW	HIGH	MODERATE	VERY HIGH
MMBBHE	VERY LOW	MODERATE	LOW	MODERATE	MODERATE	HIGH	HIGH
BBPHE	LOW	VERY HIGH	HIGH	LOW	HIGH	MODERATE	LOW
FWTHE	VERY HIGH	MODERATE	MODERATE	HIGH	LOW	HIGH	MODERATE
NPBHE	HIGH	HIGH	MODERATE	VERY LOW	VERY HIGH	VERY LOW	VERY LOW
LOCAL							
AHE	LOW	VERYHIGH	HIGH	LOW	HIGH	LOW	LOW
CLAHE	MODERATE	HIGH	LOW	HIGH	LOW	VERY HIGH	VERY LOW
CLDWT	HIGH	LOW	VERY HIGH	HIGH	LOW	VERY LOW	VERYHIGH

References

[1] Y. T. Kim, “Contrast enhancement using Brightness Preserving Bi-histogram Equalization”, IEEE Transactions on Consumer Electronics, vol. 43, no.1, **(1997)**, pp.1-8.

[2] Y. Wang, Q. Chen and B. Zhang, “Image Enhancement Based On Equal Area Dualistic Sub- Image Histogram Equalization Method”, IEEE Transactions on Consumer Electronics, vol. 45, no. 1, pp. 68-75.

[3] S. D. Chen and A. R. Ramli, “Minimum mean brightness error bi- histogram equalization in contrast enhancement”, IEEE Transactions on Consumer Electronics, vol. 49, no. 4, **(2003)**, pp.1310–1319.

[4] K.S. Sim, C. P Tso and Y. Tan, “Recursive sub image histogram equalization applied to gray-scale images”, Pattern Recognit.Lett., vol. 28, no. 10, **(2003)**, pp. 1209–1221.

[5] C. H. Ooi, N. S. P. Kong and H. Ibrahim, “Bi-histogram with a plateau limit for digital image enhancement”, IEEE Trans. Consumer. Electronics, vol. 55, no. , **(2009)**, pp. 2072-2080.

[6] K. Singh and R. Kapoor, “Image enhancement using exposure based sub image histogram equalization”, Pattern Recognition Letters, vol.36, **(2014)**, pp.10-14.

[7] S. Poddar, S. Tewary, D. Sharma, “Non-parametric modified histogram equalization for contrast enhancement”, IET Image Processing, vol.7, no.7, **(2013)**, pp. 641-652.

[8] T. Arici, S. Dikbas and Y. Altunbasak, “A histogram modification framework and its application for image contrast enhancement”, IEEE Trans. Image Processing, vol.18, no.9, **(2009)**, pp. 1921-1935.

[9] Q. Wang and R. K. Ward, “Fast image/video contrast enhancement based on weighted threshold histogram equalization”, IEEE Trans. Consum. Electron, vol. 53, no.2, **(2007)**, pp. 757-764.

[10] A. Jabeen, M. MohsinRiaz, N. Iltaf and A. Ghafoor, “image transformation function using weighted transformation”, IEEE Sensors Journal, vol. 16, **(2016)**, no. 20.

[11] S. Wang, K. Gu, S. Ma, W. Lin and X. Liu, “Guided Image Contrast Enhancement Based on Retrieved Images in Cloud”, IEEE Trans on Multimedia, vol. 18, no. 2, **(2016)**.

[12] Z. Ling, Y. Liang, Y. Wang, H. Shen and X. Lu, “Adaptive extended piecewise histogram equalisation for dark image enhancement”, IET Image Process, vol. 9, iss. 11, **(2015)**, pp. 1012–1019.

[13] H. Lidong, Z. We, W. Jun and S. Zebin, “Combination of contrast limited adaptive histogram equalization and discrete wavelet transform for image enhancement”, IET Image Process, vol. 9, iss. 10, **(2015)**, pp. 908–915.

[14] L.-M. Jan, F.-C. Cheng, C.-H. Chang, S.-J. Ruan and C.-A. Shen, “A Power-Saving Histogram Adjustment Algorithm for OLED-Oriented Contrast Enhancement”, Journal of Display Technology, vol. 12, no. 4, **(2016)**.

[15] H. Liu and I. Heynderickx, “Visual attention in objective image quality assessment: Based on eye-tracking data”, IEEE Trans. Circuits Syst. Video Technol., vol. 21, no. 7, **(2011)**, pp. 971–982.

[16] X. Min, G. Zhai, Z. Gao and K. Gu, “Visual attention data for image quality assessment databases”, in Proc. IEEE Int. Symp. Circuits Syst. , **(2014)**, pp. 894–897.

[17] H. Hadizadeh and I. V. Bajic, “Saliency-aware video compression,” IEEE Trans. Image Process., vol. 23, no. 1, **(2014)**, pp. 19–33.

[18] X. Hou, J. Harel and C. Koch, “Image signature: Highlighting sparse salient regions”, IEEE Trans. Pattern Anal. Mach. Intell., vol. 34, no. 1, pp. 194–201, **(2012)**.

[19] X. Zhang and S. Lyu, “Blind estimation of pixel brightness transform”, in Proc. IEEE Int. Conf. Image Process., **(2014)**, pp. 4472–4476.
[20] K. Gu, G. Zhai, W. Lin and M. Liu, “The analysis of image contrast: From quality assessment to automatic enhancement”, IEEE Trans. Cybern, vol. 46, no. 1, **(2016)**, pp. 284–297.
[21] S. Winkler, “Analysis of public image and video databases for quality assessment”, IEEE J. Sel. Topics Signal Process., vol. 6, no. 6, **(2012)**, pp. 616–625.
[22] “Methodology for the Subjective Assessment of the Quality of Television Pictures”, ITU BT. 500-11, ITU, Geneva, Switzerland, **(2002)**.
[23] S. Wang, A. Rehman, Z. Wang, S. Ma and W. Gao, “Perceptual video coding based on SSIM-inspired divisive normalization”, IEEE Trans. Image Process., vol. 22, no. 4, **(2013)**, pp. 1418–1429.