++

Politecnico di Torino

Energy Management for IoT

ENERGY EFFICIENT DISPLAYS LABORATORY

Alessandro Landra (s284939)
Dalmasso Luca (s281316)

Course: Embedded Systems (Computer Engineering branch)
Date: 20/12/2021

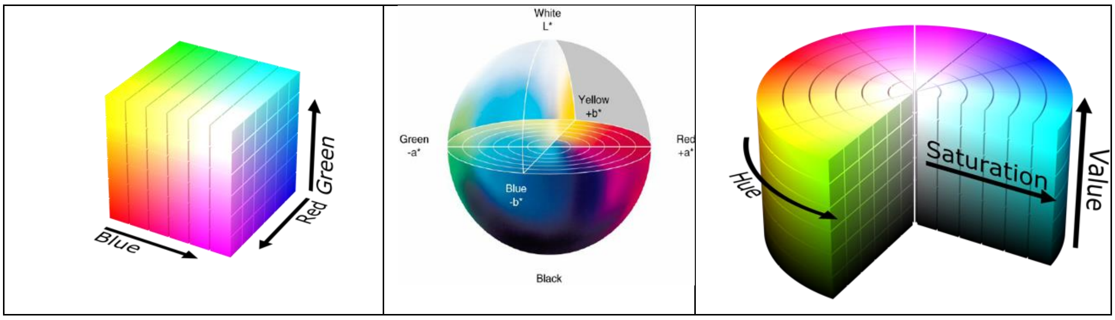
Goal of the lab

1. Demonstrate how different manipulations of image can be used to trade off image quality and power saving in emissive displays.
2. Learn how to manipulate images in the LAB, RGB and HSV color models.

Color Models

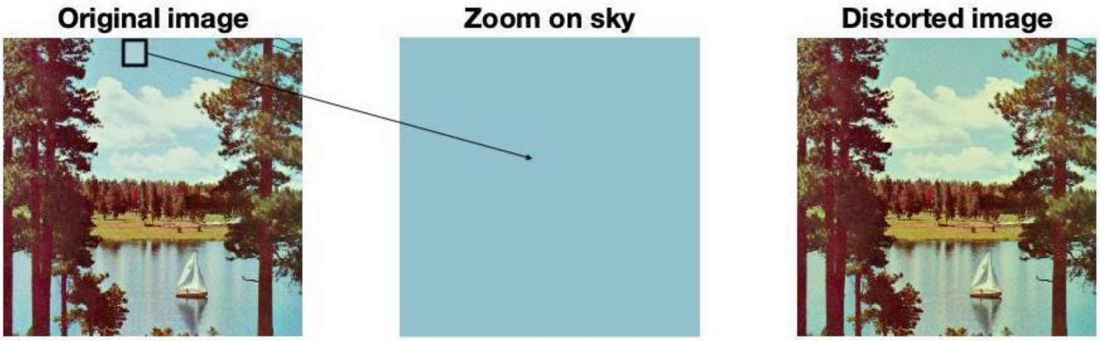
A color model is a visualization that depicts the color spectrum as a multidimensional model, in figure1 there are the 3 color models used for this experiment.

Figure1:
From left to right: RGB space, LAB space, HSV space.



A widely used color model is the RGB, where a pixel is a result of the overlap of 3 colours channels: Red, Green and Blue (Figure1).
It is possible to produce a distortion to an image by just changing the intensity of a color channel, but in general this approach is not very good because even a little change in the intensity of a color can produce a very different shade.
The following example in Figure2 shows how a shade of blue is completely different, basically has become a green shade, by just changing of 10% the blue channel intensity.

Figure2: The blue shade in that portion of the sky is characterized by this values:
R=147, G=194, B=204.
The new image has just a distortion of the blue channel of 10% (B=B*0.9).



Due to this problem, the approach used in the experiment is to reason in term of brightness and saturation.

In general, is possible to save power by acting directly on the brightness of the image, without acting directly on the colours.
In order to be able to modify the brightness and the saturation it is necessary to change the color model from RGB to HSV.

The HSV color model (Figure1) is still a 3D model that is based on the same RGB primary colours, the difference is that every color, or hue (H), is characterized with two parameters that are: saturation (S) and Brightness (V).

The distortion produced by a transformation can be measured in the LAB color space in terms of Eucledian distance between two images, that is computed in percentage with the following formula:

$$Distortion(\%) = \frac{\sum_{k=1}^{N}\left((L_i,k-L_j,k\right)+\left((A_i,k-A_j,k\right)+\left((B_i,k-B_j,k\right))}{N*\sqrt{(100^2+255^2+255^2)}} *100$$

$$where: \begin{cases} N = size\ of\ image \\ Denominator = maxium\ distance\ between\ 2\ colours \\ Numerator = Eucledian\ distance\ in\ LAB\ coordinates \end{cases}$$

OLED Display Power model

The power mode used for this experiment is based on a OLED emissive display, that as been characterized using RGB model.
OLED are a particular type of emissive displays that have the following characteristics:

1. Do not require external lights. (No Backlight like in LED displays).
2. Pixels are composed of 3 organic emitting diodes corresponding to red, green and blue colours.
3. The light that a pixel emits is in response to an electric current.

As a consequence of the points 2 and 3, the colour of a pixel in an OLED display is dependent on the currents absorbed by the 3 organic diodes and so the power consumption is dependent on the colours of an image.

From a quantitative point of view the following formula express the power consumption of a generic OLED display:

$$P_{image} = C + \sum_{i=0}^{n}\{f(R_i) + h(G_i) + k(B_i)\}$$

Where:

- C is a static value which is independent of the pixels values.
  → C is a characteristic of the display, must be measured experimentally by turning off all the pixels (Black image).
- f, h, and k are the functions that express the power consumption of a single pixel, their values are determined experimentally as well.
  →they are a characteristic of the display, for example f can be determined experimentally by turning off B and G values and measuring the power consumption of the pixel and by varying R.
- n is the number of pixels of the screen.

For our experiment, the OLED displays has the following power model:

$$P_{image} = w_0 + \sum_{i=1}^{n}(w_R * R_i^{\gamma} + w_G * G_i^{\gamma} + w_B * B_i^{\gamma})$$

$$where: \begin{cases} \gamma = 0.7755 \\ w_0 = 1.4817 \exp(-6) \\ w_R = 2.1364 \exp(-7) \\ w_G = 1.7775 \exp(-7) \\ w_B = 2.1435 \exp(-7) \end{cases}$$

From the following pictures is possible to better understand how different colours intensities can impact the power.

Figure1: primary colours in the RGB space are Red, Green and Blue.
Is possible to obtain all the colours by varying the intensities of the 3 channels (Red, Green, Blue).
Ex: Pure White is the combination of the 3 channels (R, G, B) at the maximum intensity.

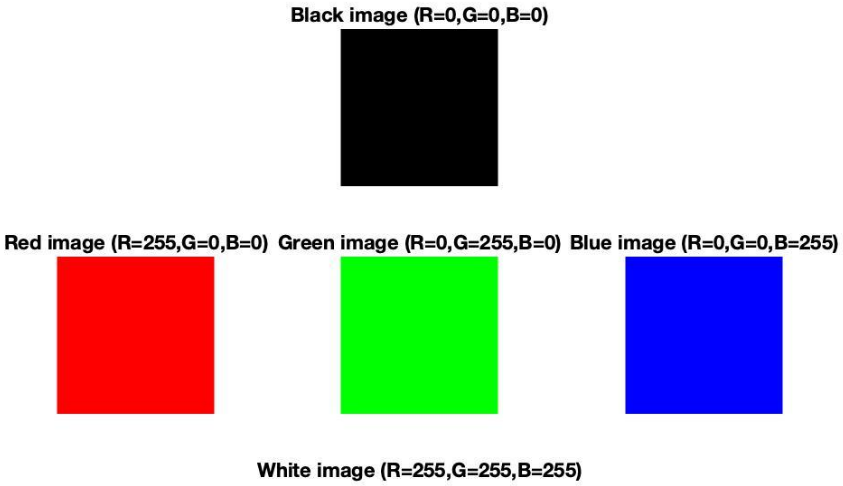
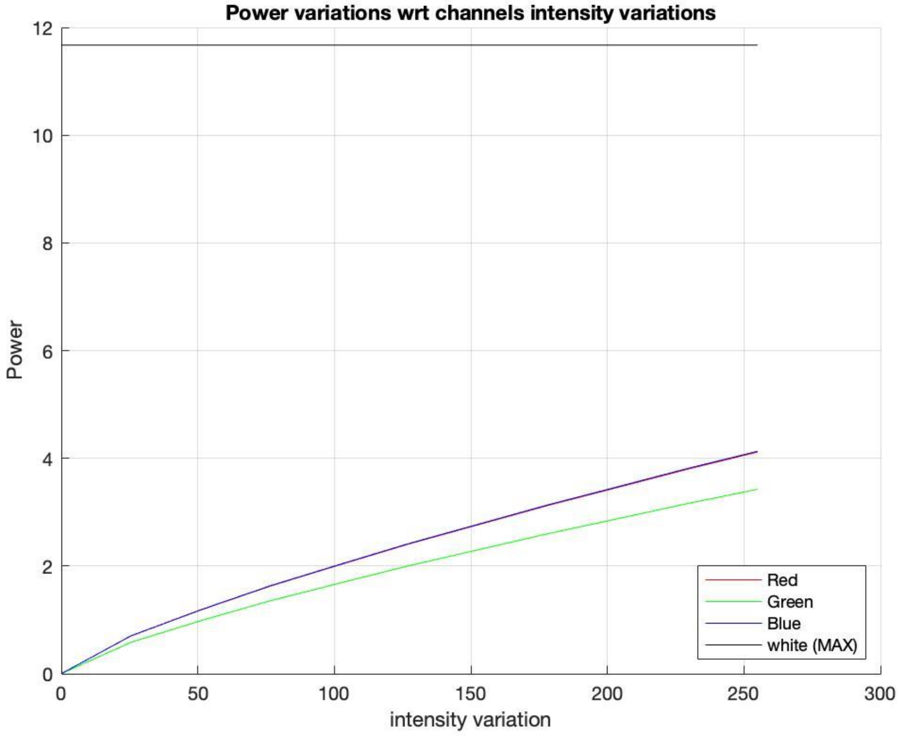

Black image (R=0,G=0,B=0)
Red image (R=255,G=0,B=0)
Green image (R=0,G=255,B=0)
Blue image (R=0,G=0,B=255)
White image (R=255,G=255,B=255)

Figure2: OLED power model characterization for a 512x512 RGB image.
Pure white is the most consuming colour, Green channel has less impact on power with respect to Blue and Red channel.
Hungry Blue is a characteristic of all OLED display.



DVS dependent OLED Display Power model

The power model used for the second part of this experiment introduces a dependency from the voltage used to power on a cell and takes into account the sum of all the currents flowing in each cell. Below both the current and power formulae can be analysed:

$$Icell = \frac{p1 * Vdd * Drgb}{255} + \frac{p2 * Drgb}{255} + p3$$

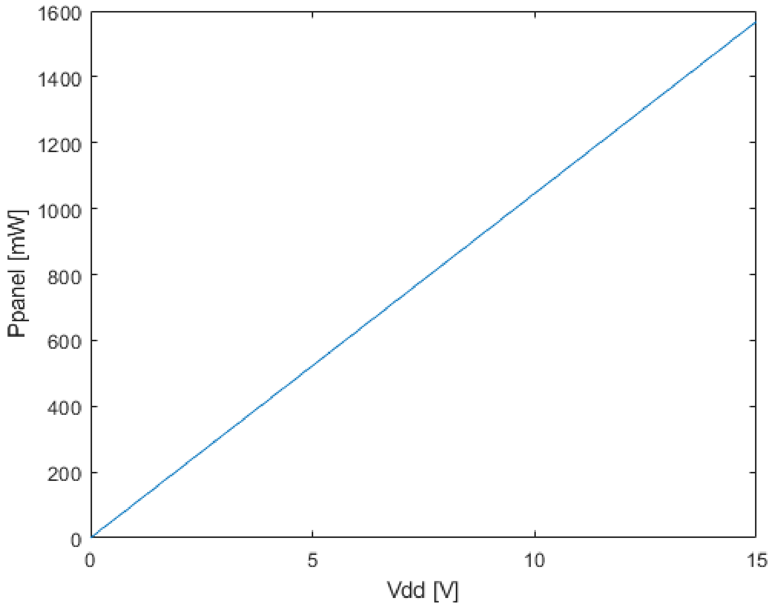
Where:

- Vdd is the voltage given to the whole display, so the potential difference applied to each cell.
- Drgb is the RGB color value of the current pixel.

$$and: \begin{cases} p1 = 4.251\exp(-5) \\ p2 = -3.029\exp(-4) \\ p3 = 3.024\exp(-5) \\ \end{cases}$$

$$P_{panel} = Vdd \sum_{i=1}^{W} \sum_{j=1}^{H} Icell(i,j)$$

Figure1: effect of voltage scaling of a randomly selected image on the power consumption of the whole panel.



Part I: Image transformations and distortion

In this experiment we used two techniques for power saving: Histogram Equalization and Thresholding.
Those techniques have been tested on 2 sets of images, the first set is very small (contains 14 images) and it has been used for preliminary tests, just to see how the different algorithms behave.
The second test set is very large, contains 200 images, it's a more large and heterogeneous dataset where to test the real effectiveness of the algorithms.

1. Histogram Equalization:
   The Histogram equalization algorithm is a method that is used to increase the global contrast of many images by flattening the histogram.
   This method is very effective especially when the image is represented by a narrow range of intensities, in this case the histogram flattening produce a better distribution of the colours in the full color scale range which will result in a better global contrast, especially when both background and foreground are dark or bright.
   Histogram equalization can also be used for power saving purposes but the results are in general not very effective, especially when used for images that are dark or in general when the subject of the image is a dark scene.

Figure3: (tested on small data set) highlighted are the best and the worst case. Is possible to see that in some images the power consumption drastically get worst as well as the distortion.

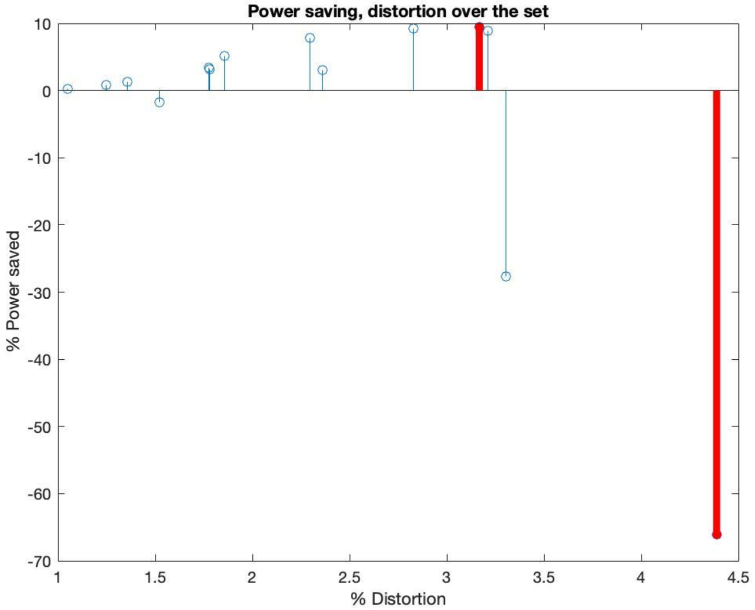


Figure4: (tested on small data set) best case.
This case produce a 10% power saving with a distortion around 3%.
The flattening process basically moved the histogram more to the left part (Darker hues), thanks to that is possible to save some power (Image is less bright).

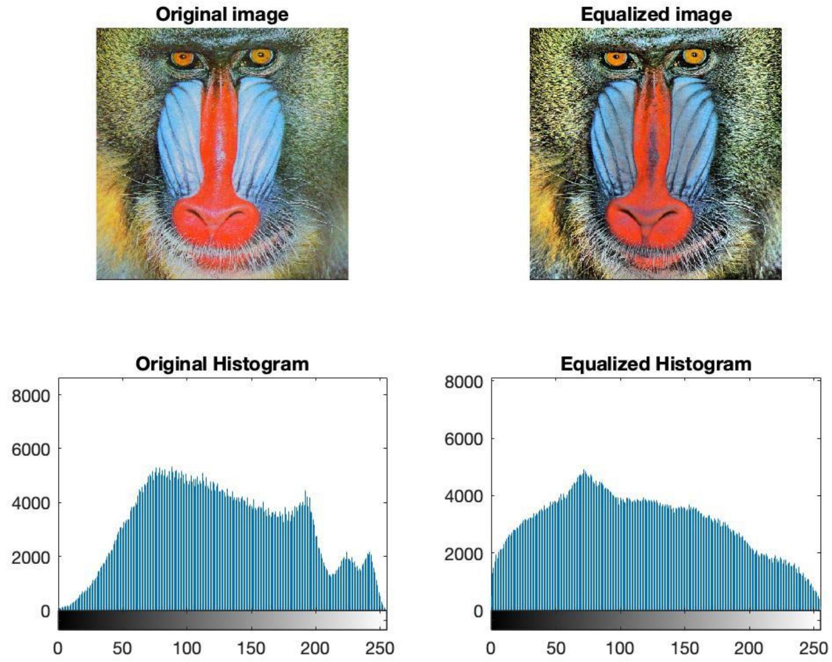
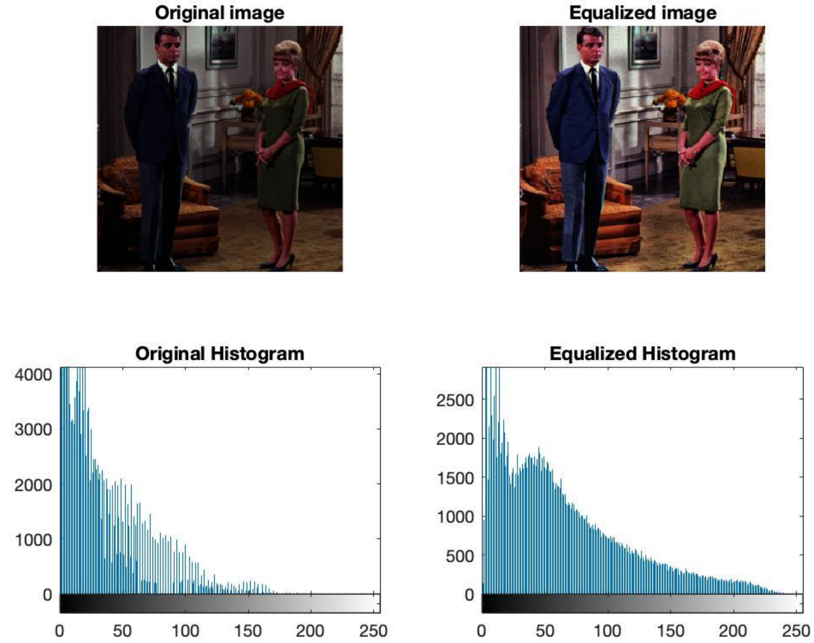


Figure5: (tested on small data set) worst case
This case is very bad both for power and for distortion, this is due to the fact that the original image has a very narrow distribution around dark hues.
The equalization moved a little the distribution to the right, in this way there is more contrast but now the image is brighter and more power hungry.



The following figures are the results coming from the large data set, and the conclusions are the same as the one before, infact is possible to see that in general the equalization does not produce any power saving (Figure6)

Figure6: (tested on big data set), it is not showed but the average power saving is below zero, so equalization on 200+ images increased the average power consumption.

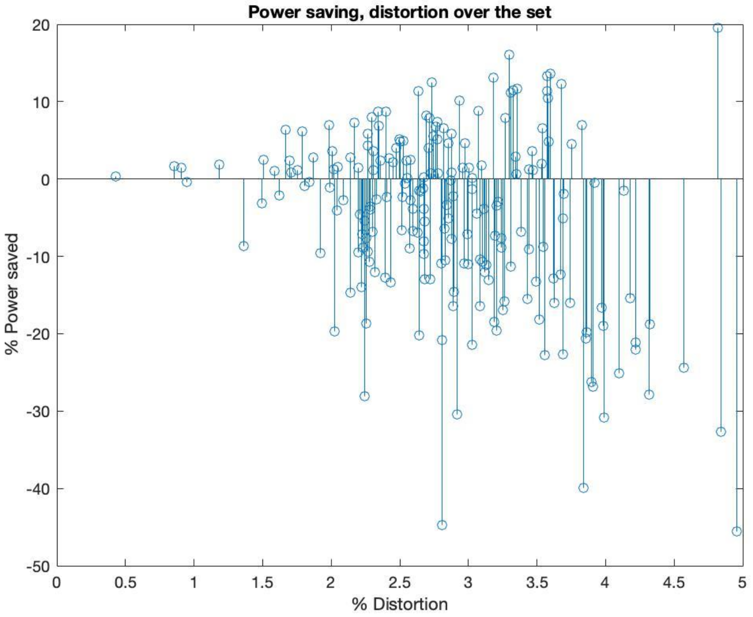


Figure7: (tested on big data set) best case.
Better contrast, best power saving because original image trend are the brighter hues that are flattened after equalization.

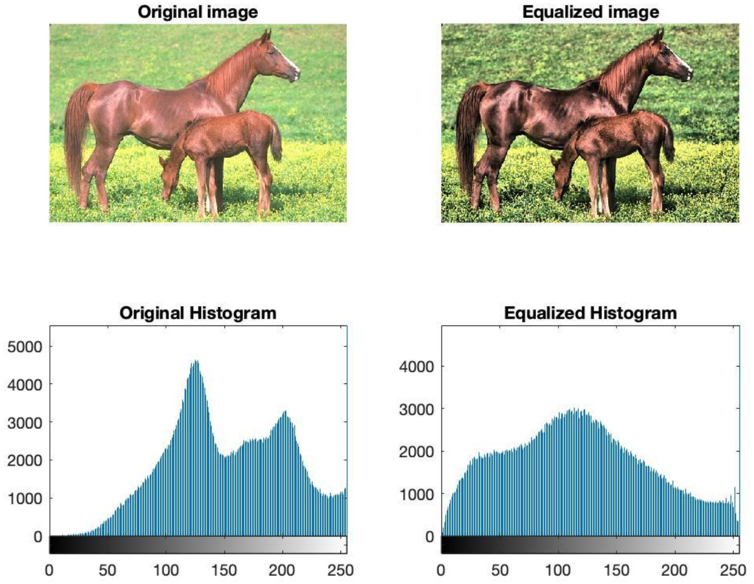
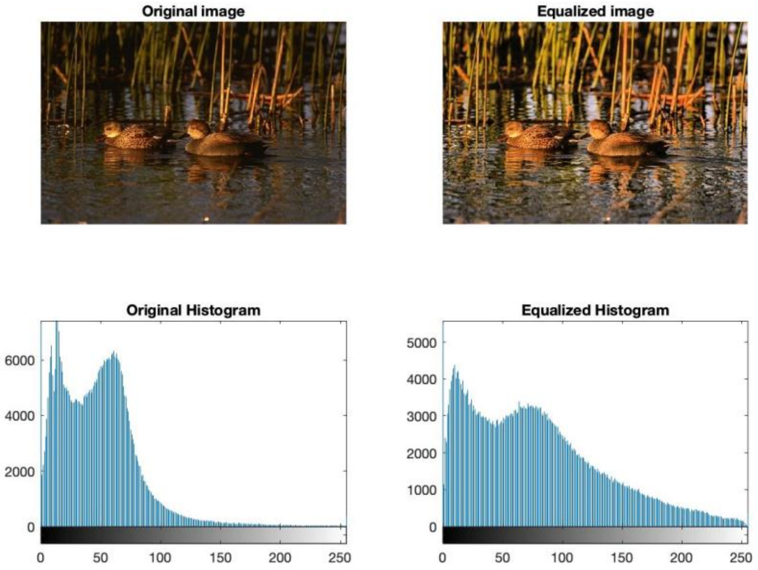


Figure8: (tested on big data set) worst case.



2. Thresholding: in this case the distortions are produced by modifying the brightness intensity (V) in the HSV space.
   In particular our algorithm divides the image into a number of sub-regions and compute the average brightness into them, whenever the average brightness (Vavg) is above a certain threshold Vth_{high} than it is saturated that value.
   Example: $Vth_{high} = 70\%$, Vavg=87.9% → Vavg = $Vth_{high} = 70\%$
   At the same time if the average brightness is less than a lower threshold the pixels in that regions are turned off, this does not produce a big visible distortion for dark hues and the advantage can become very effective because in OLED the black pixels are the lower power consuming.
   Example: $Vth_{low} = 20\%$, Vavg=15.45% → Vavg =$Voff = 0\%$

As said before is possible to act on the two thresholds Vth_{high} and Vth_{low}, but also in the number of sub-regions by changing the size of the sub-matrix (Ms x Mn) that divides the image.

Of course by using Ms=Mn=1 the algorithm will basically scan pixel per pixel ad a better result is expectable, but the computational effort is also greater.

In order to choose a proper threshold is possible to do some practical tests, as we did, using for example this tool: https://programmingdesignsystems.com/color/color-models-and-color-spaces/index.html , where you can tests how much will a hue change with respect to changes in brightness and saturation in HSV.

The parameters that we choose are: $Vth_{high} = 70\%, Vth_{low} = 20\%, SubRegion_{size} = 2x2\ matrix.$

Figure9: (tested on small data set), the distortion is contained into the 3% range and the power saving is always positive, tends to be around 0% in the worst cases.

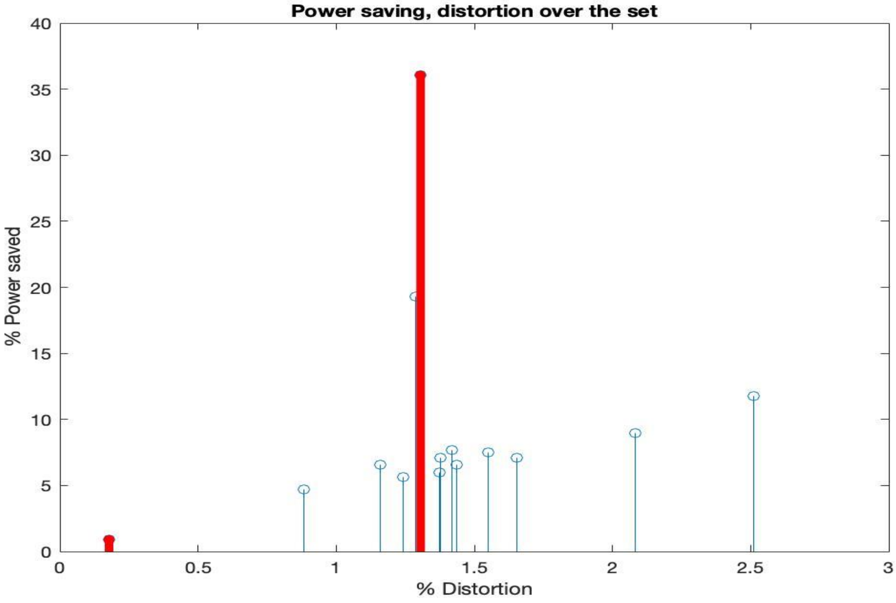


Figure10: (tested on small data set) best case.
Image is very dark and so there are a lot of regions under the Vth_{low} threshold, so their pixels are turned off.
The image is for sure distorted but still the original quality is preserved.

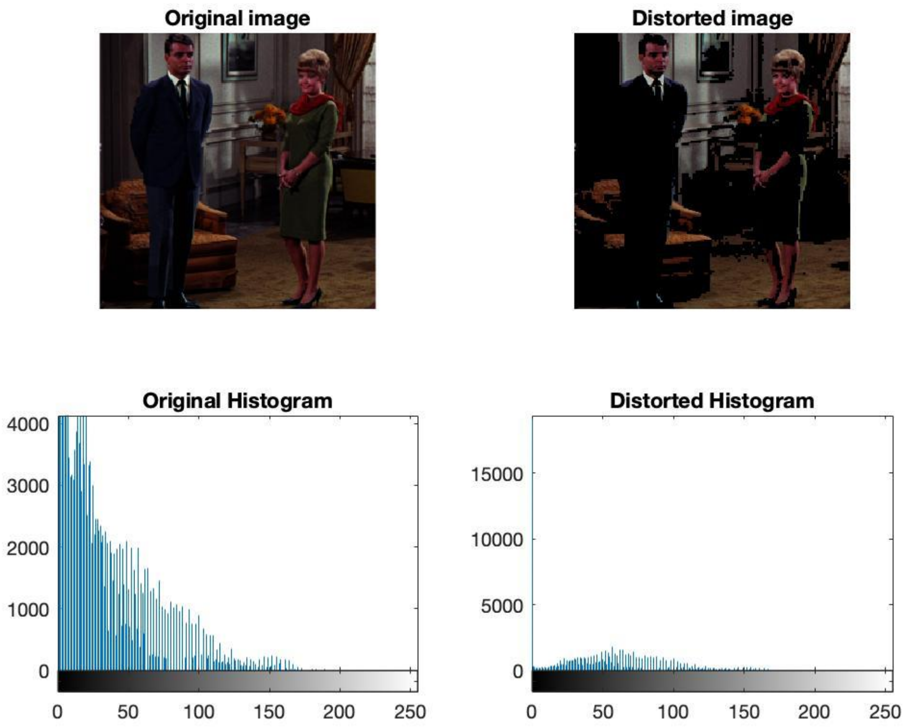
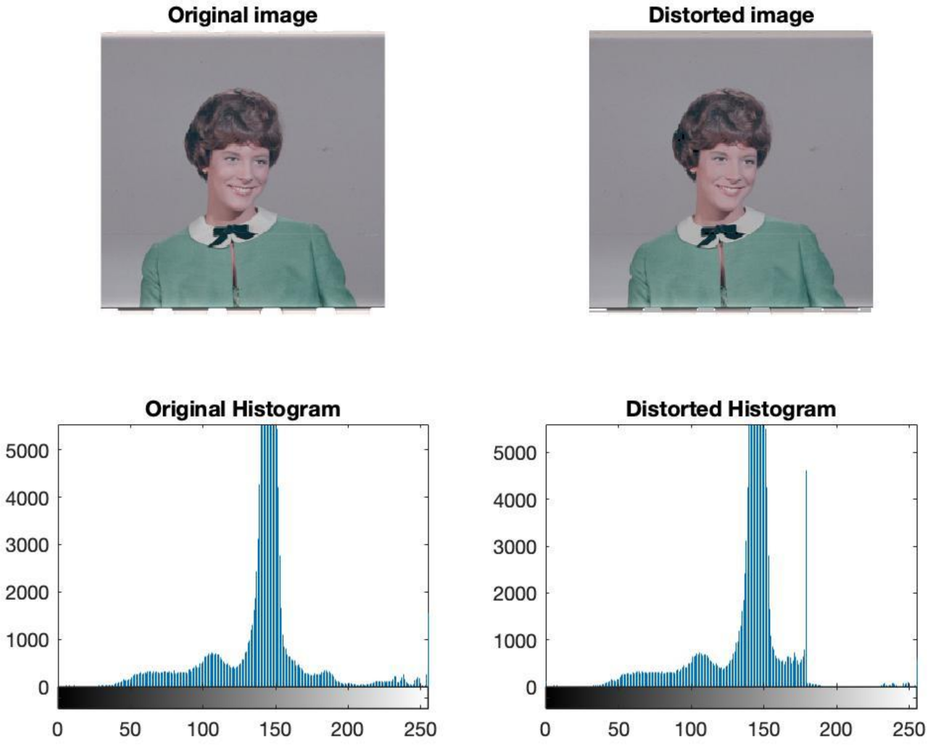


Figure11: (tested on small data set) worst case.



The following are the results for the large data set.

Figure12: the distortions are in the majority of the cases contained into the 3%.

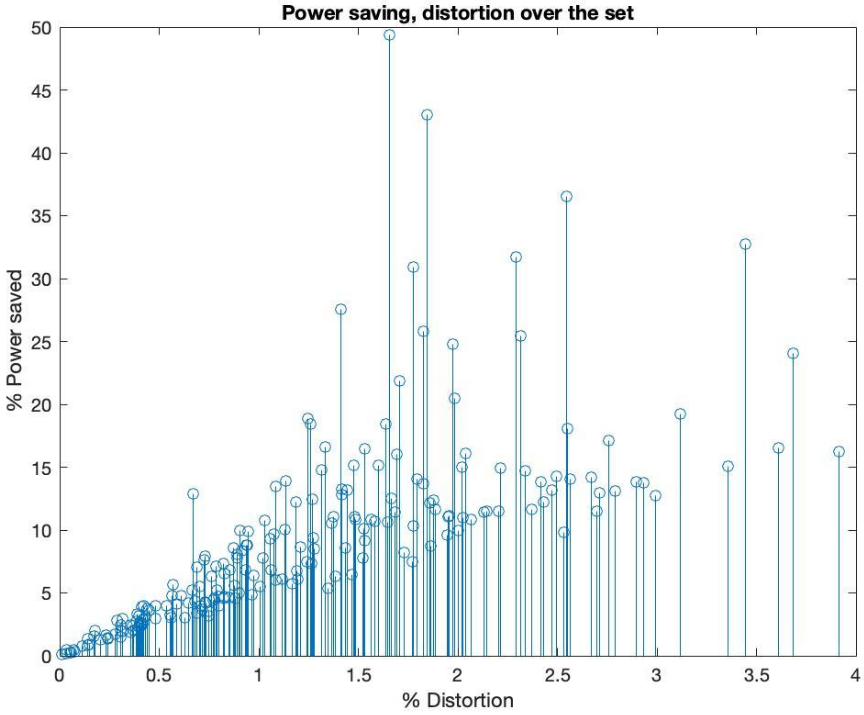


Figure13: best case

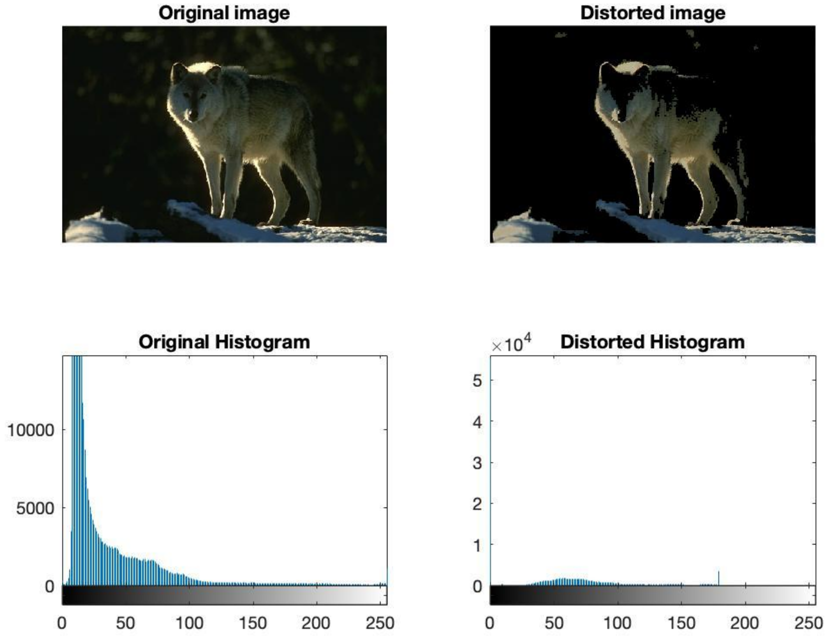
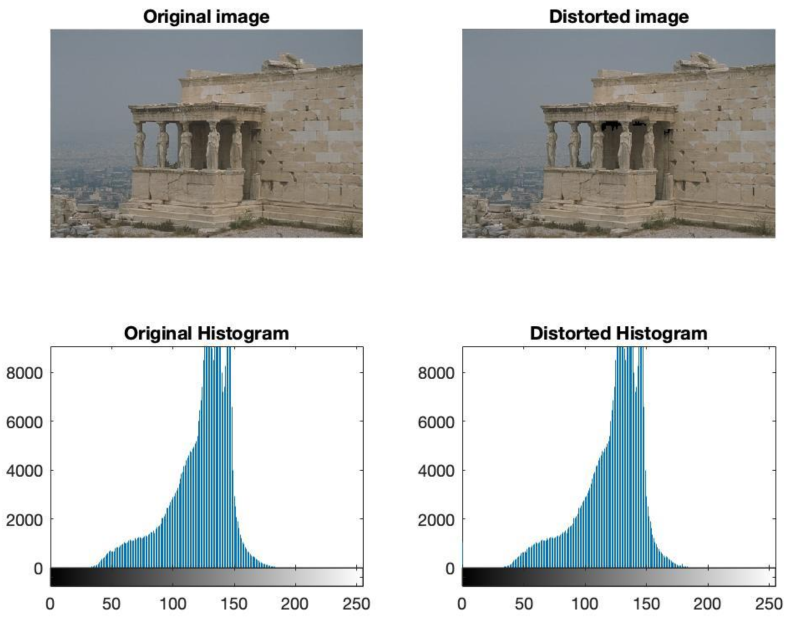


Figure14: worst case



Part II: Dynamic voltage scaling

In this experiment we applied voltage scaling to the images contained in the 2 sets used for part 1, taking advantage of the already implemented function ***displayed_image()*** to determine the maximum current and RGB value of the voltage-scaled image.

Our aim was to find a way to compensate the quality loss of the final picture and to do so, we applied the brightness scaling technique, both with fixed and variable increasing parameter.

$$V' = V + \boldsymbol{b}$$

Where:

- V is the brightness value in HSV colour model.
- ***b*** is the increasing parameter to compensate the voltage scaling effect.

We started determining a fixed value for ***b***, trying to maintain the structural similarity index (SSIM) over 0.9. For this purpose, we developed a script that increases the b parameter while maintaining the previous constraint with different Vdd, considering the average SSIM, analysing all the images contained in Set 1.

Below is possible to see the resulting parameters:

Vdd	14	13	12	11	10	9	8
b	0.175	0.175	0.2	0.2	0.2	0.225	0.25

Where:

- Vdd is the voltage applied to the panel after voltage scaling

Then, thresholding has been applied to ***b***, based on the brightness of each pixel. This, since Vdd reduction impacts more on brighter pixels than on darker ones (as described in ***Dynamic voltage scaling of OLED Displays*** paper), so a high parameter has been chosen over a certain luminance (0.7), a lower one in a range below that (0.3-0.7) and an even lower under that limit.

As a starting point, the previously determined b value, ***startingB*** has been used, to keep the dependency on the target Vdd:

$$\boldsymbol{b} = \begin{cases} \text{bHIGH: startingB} + 0.1, \text{with V} > 0.7 \\ \text{bLOW: startingB} - 0.1, \text{with } 0.3 < \text{V} \leq 0.7 \\ 0, \text{with V} \leq 0.3 \end{cases}$$

Concerning the first and second set together, of random images analysed, the results of the application of the previously mentioned techniques can be summarized as follow:

- Fixed brightness compensation parameter

  Average distortion: **2.67%**

  Average power saving: **-3.06%**

- Thresholded brightness compensation parameter

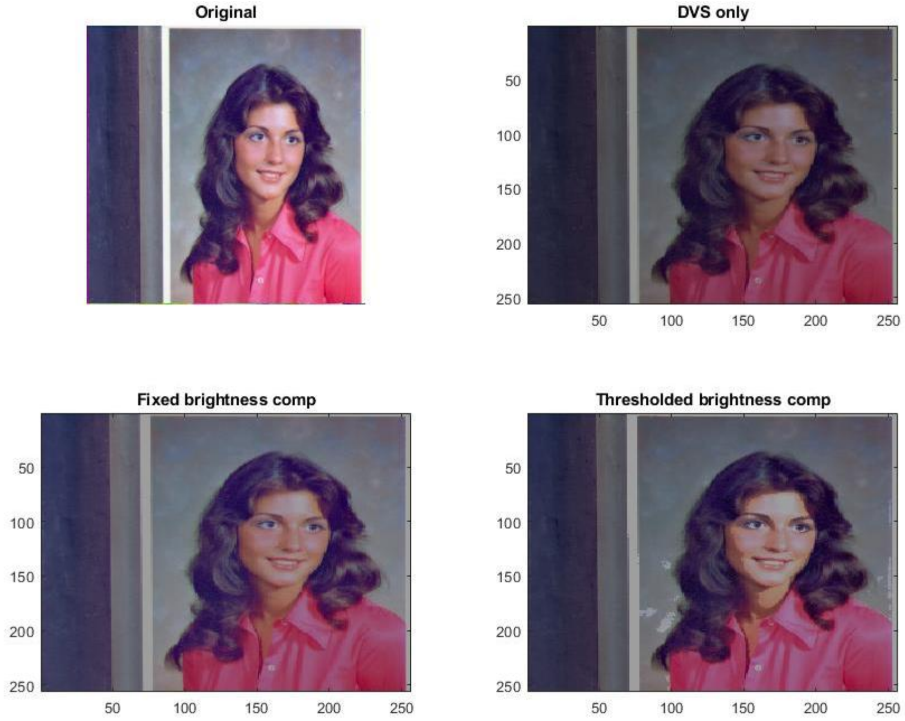
  Average distortion: **3.64%**

  Average power saving: **8.34%**

N.B. Vdd has been set to **12** (the original was **15**), to maintain both the distortions in LAB domain below **5%** (with Vdd=**11** the distortions were respectively **4.48%** and **5.66%**)

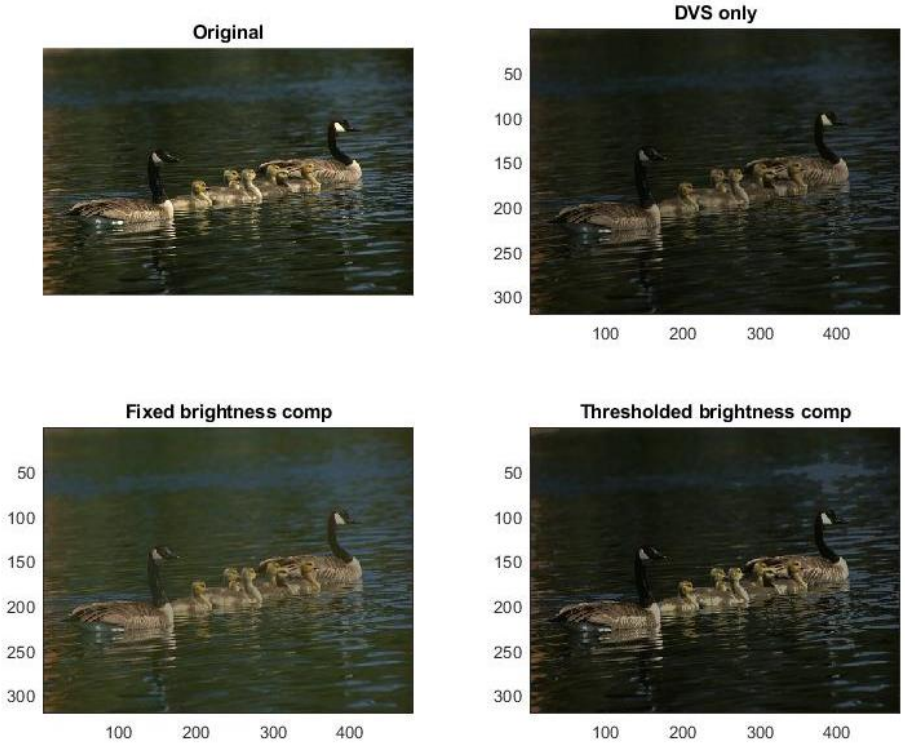
Therefore, the second approach leads to images that are a little bit less close to the original ones, in terms of differences in LAB domain, but which still take advantage of the voltage scaling as a power saving technique (with a fixed parameter, instead, we face an even higher power consumption than the one without voltage scaling, due to the brightness increase).

Is possible to notice, in the following pictures, the impact of DVS on brighter pixels, that is the problem that led us to thresholding. In this case the second approach produces an image of little bit lower quality with respect to the fixed parameter one, but consuming a considerably reduced amount of power.



Fixed parameter
Image distortion: *2.52%*
Power saving: *-0.03%*

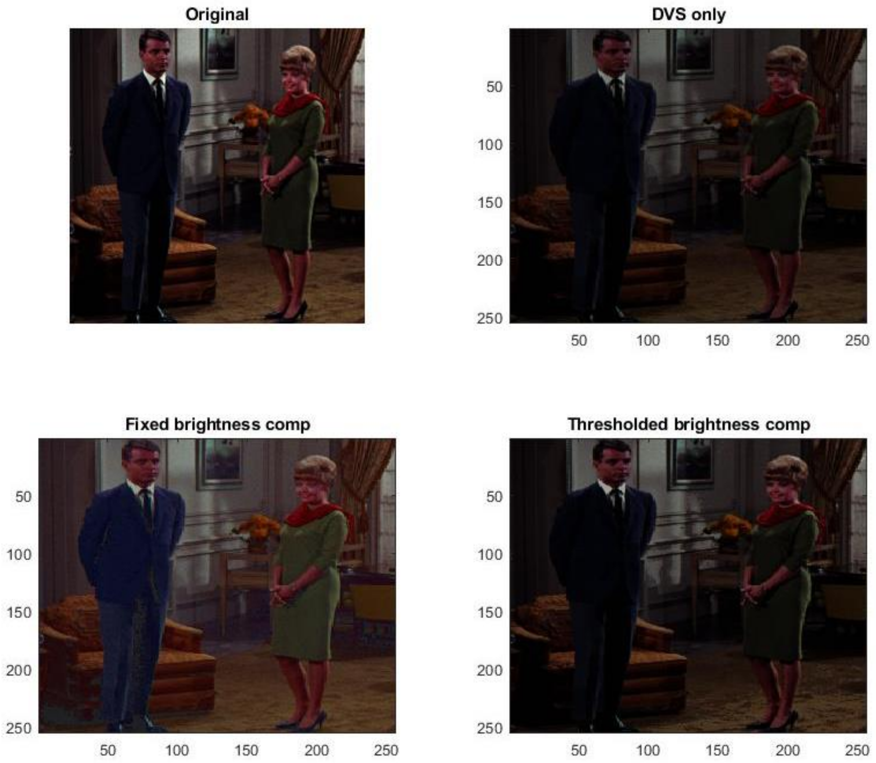
Thresholded parameter
Image distortion: *3.78%*
Power saving: *7.29%*



Fixed parameter
Image distortion: *1.71%*
Power saving: *-19.10%*

Thresholded parameter
Image distortion: *2.56%*
Power saving: *16.19%*

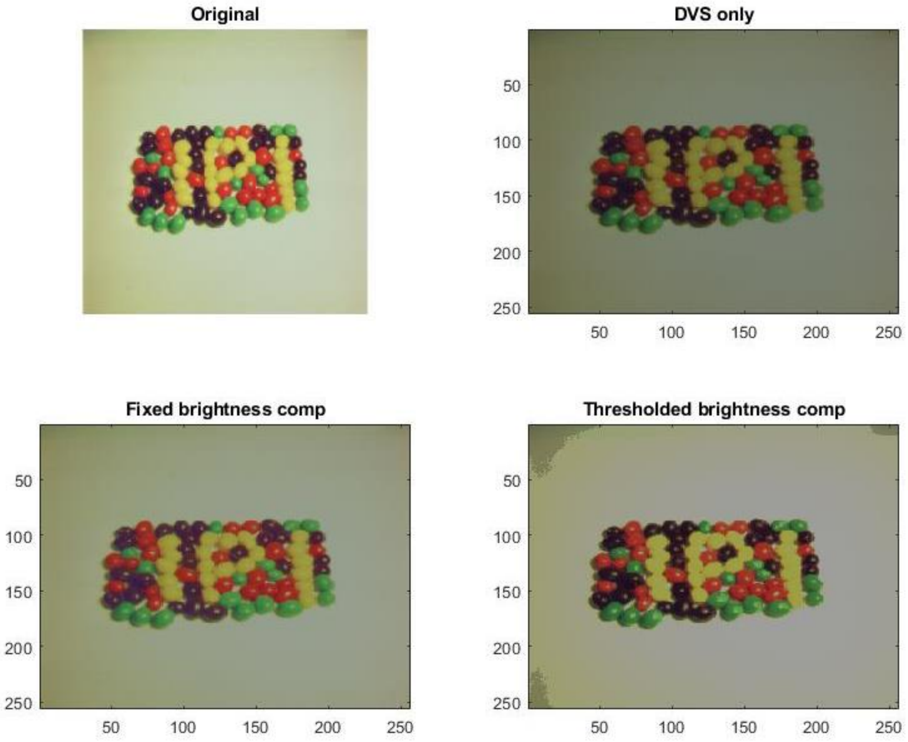
Below is possible to see another aspect: the Fixed compensation parameter produces a more human visible image, but it totally vanishes all the advantages of applying DVS, while the second approach is a good compromise.



Fixed parameter
Image distortion: *2.19%*
Power saving: *-22.04%*

Thresholded parameter
Image distortion: *1.60%*
Power saving: *15.93%*

By the way, is possible to see cases in which the thresholded parameter is not the best choice:



Fixed parameter	Thresholded parameter
Image distortion: *4.14%*	Image distortion: *4.46%*
Power saving: *2.48%*	Power saving: -*0.71%*

This unwanted behaviour comes from the high density of bright pixels in the original image, which cause ***bHIGH*** to be applied in most of the cases.