Predicting image quality using a modular image difference model

M. Orfanidoua, S.Triantaphillidou*a, E. Allena

*aUniversity of Westminster, Watford Road, HA13TP, Harrow

ABSTRACT

The paper is focused on the implementation of a modular color image difference model, as described in [1], with aim to predict visual magnitudes between pairs of uncompressed images and images compressed using lossy JPEG and JPEG 2000. The work involved programming each pre-processing step, processing each image file and deriving the error map, which was further reduced to a single metric. Three contrast sensitivity function implementations were tested; a Laplacian filter was implemented for spatial localization and the contrast masked-based local contrast enhancement method, suggested by Moroney, was used for local contrast detection. The error map was derived using the CIEDE2000 color difference formula on a pixel-by-pixel basis. A final single value was obtained by calculating the median value of the error map. This metric was finally tested against relative quality differences between original and compressed images, derived from psychophysical investigations on the same dataset. The outcomes revealed a grouping of images which was attributed to correlations between the busyness of the test scenes (defined as image property indicating the presence or absence of high frequencies) and different clustered results. In conclusion, a method for accounting for the amount of detail in test is required for a more accurate prediction of image quality.

Keywords: image difference model, iCAM, color difference, image quality, JPEG, JPEG2000

1. INTRODUCTION

Research on objective image quality measures and metrics is nowadays more active than ever, with input from many various fields relating to imaging. Traditionally, research on color and on spatial image evaluation were undertaken by different scientific groups, somehow relating to different expertise, despite the fact that image quality attributes are unlikely to be considered independently of each other. In the last decade there has been a great effort to unify color difference modeling and spatial aspects of the human vision into a single model that can, amongst other applications, be employed in image evaluation [2].

Zhang and Wandell [3] were amongst the first recognizing the limitations of simple CIELAB color difference metric for use with complex stimuli. They proposed a pre-processing step to CIELAB that accounts for certain spatial properties of vision. The resulting S-CIELAB spatial color difference metric employs a series of convolution kernels to approximate the contrast sensitivity function, which are used to blur spatial frequencies in images that cannot be perceived. The contrast sensitivity function was an obvious inclusion in earlier image quality metrics and image difference models, which were concerned mainly with spatial image attributes, such as Barten’s SQRI [4] metric and Daly’s VDP [5] model. The application of S-CIELAB was proven very useful for deriving an image difference map between image pairs, representing spatially weighted CIELAB pixel-by-pixel color differences.

Johnson and Fairchild presented an extension of S-CIELAB, a modular framework developed for calculating image differences and predicting image quality. The modular color image difference framework [1,6,7] was designed to be used with advanced color difference formulae, such as the CIEDE2000, and performs spatial filtering in the frequency, rather than the spatial domain. Further independent pre-processing modules are added for adaptation, spatial localization and local contrast detection, before performing pixel-by-pixel color difference calculations. The framework allows for great flexibility in the choice of color spaces used to evaluate the color differences. Examples are the CIELAB color space, the CIECAM02 color appearance model, or the IPT color space.

*trian@wmin.ac.uk; phone 44 (0)20 79115000; fax 44 (0)20 79115943; wmin.ac.uk
Using global image statistics, such as the mean, median, standard deviation or others, the color difference map can be further reduced to a single metric that can be used as a perceptual image difference measure between pairs of images. A single value is bound to hide useful information that can be obtained by examining error maps after the implementation of each individual module, or the final color difference map - such as the location in the image where the difference is more prominent, or the origin of the image differences (i.e. whether the error is due to lightness, chroma or hue) [2]. On the other hand, a metric is useful when threshold detection or the overall perceived image quality needs to be predicted. Work on the modular color image difference framework along with the parallel evolution in color difference formulae and color appearance modeling lead to an image appearance model, iCAM. Various components of iCAM can be used to predict image quality and image differences, complex image appearance and the rendering of high dynamic range images [2,8].

1.1 Image differences between original and compressed images

Several modules from the iCAM were implemented in this work, with the aim to predict visual magnitudes between pairs of uncompressed images and images compressed at four different compression rates using lossy JPEG and JPEG 2000 compression schemes. The work involved programming each step, processing each image file and deriving the error map, which was further reduced to a single number. The metric was tested against relative quality differences between original and compressed images, derived from psychophysical investigations on the same dataset [9]. The independent steps that were implemented in this work are illustrated in figure 1 - as described in reference 6.

![Fig. 1. Flowchart of the Modular Color Image Difference Metric as shown in [6]](image-url)
2. MODEL IMPLEMENTATION

2.1 Spatial Filtering

In this first step, the contrast sensitivity function (CSF) is used for simulating the decrease in human visual sensitivity that occurs with increased spatial frequency by blurring unperceived higher frequencies and enhancing lower frequencies that are visually more important. Different CS functions are applied on opponent color channels, achromatic, red – green, yellow – blue. Spatial filtering is carried out in the frequency domain, as it is shown to produce more accurate approximations than in spatial domain [10]. The CSF can be further modified to take into account the frequency information of the images, a step known as a spatial frequency adaptation (section 2.2).

The sRGB encoded test images (see section 3) were converted to the CIELAB color space before any manipulation. The CSFs were represented by a band-pass function for the luminance and a low-pass for the chrominance channels. The luminance CSF is a simple three parameter exponential function [10,11] described by equation 1:

\[ \text{csf}_{\text{lum}} (f) = a \cdot f^c \cdot e^{-bf} \]  

\[ \text{eqn 1} \]

\[ a, b \text{ and } c \text{ were set to the constants 75, 0.2 and 0.8 respectively, shown to fit reasonably well with published CSFs. The functions used for filtering the chrominance channels are sums of two Gaussian functions, as shown in equation 2. The values for the relevant parameters are given in [7]:} \]

\[ \text{csf}_{\text{chrom}} (f) = a_1 \cdot e^{-b_1 \cdot f^a} + a_2 \cdot e^{-b_2 \cdot f^2} \]

\[ \text{eqn 2} \]

The spatial frequency, \( f \), in equations 1 and 2 is represented in cycles-per-degree (cpd) of visual angle and thus takes into account both the addressability and viewing distance (see section 3 for display ppi and viewing distance). The CSFs were represented as two dimensional matrices of the size of the image. This was achieved by creating a mesh grid of the size of the padded image which was meant for Fourier transform and filtering [12].

In the implementation of the CSFs attention should be given to the DC component of the filter, which corresponds to the mean value of the image channel. The mean value needs to be kept constant after each manipulation, so that uniform patches are not affected by the filtering and their differences are predicted accurately. This can be accomplished by, either truncating the luminance CSF into a low-pass filter or normalizing the CSFs so that the DC component is equal to 1.0. By subtracting the mean luminance component and then adding it back after filtering we can avoid normalization in the luminance channel. This method was implemented here, which keeps the DC component to zero and creates a smoother luminance CSF which prevents artifacts commonly resulting from Fourier filtering. The chrominance CSF was normalized to unity for the DC component. The two dimensional filters were then multiplied with the discrete Fourier transform of the individual channels.

2.2 Spatial Frequency Adaptation

Spatial frequency adaptation is due to the selective response of the eye to various spatial frequencies [13]. In other words, when the eye adapts to a high frequency, other frequencies in the visual field appear to be of lower frequency, and vice versa. To compensate for spatial frequency adaptation, which is inevitable when looking at complex scenes, the CSFs have to be treated before implementation. There are several models that alter the nature of the CSF based upon either assumptions of the viewing conditions, or based upon the information contained in the images themselves [10].

Three implementations of the CSF were used in this work. In the first, the CSFs were not adapted with respect to spatial frequencies present in the visual field. The second involved altering the CSFs according to a non-linear \( 1/f \) approximation which is shown to suit well imaging applications [10]. It is represented by equation 3 below:

\[ \text{csf}_{\text{adap}} = f^{1/3} \cdot \text{csf} (f) \]

\[ \text{eqn 3} \]
where \( f \) represents the frequency map. Such a variation results in a shift of the peak of the contrast sensitivity and an increase in the gain of mid-to-high frequencies.

The third implementation was based on the spatial frequencies included in the image itself, in which the CSF was normalized by the luminance spectrum of the image. Its general description is shown in equation 4 [10]. Before normalization the luminance spectrum was blurred by filtering with a 7-band pass filter to approximate the behavior of the human visual system and avoid large error due to otherwise noisy spectrum. It was further treated in a non-linear fashion (raised to 1/3) to modulate lower frequencies and boost higher image frequencies [12].

\[
\text{csf}_{\text{adap}} = \frac{\text{csf}(f)}{\text{image}_{\text{hist}}(f)}
\]  

(4)

### 2.3 Spatial Localization

Filtering the image with the CSF results in the minimization of image frequencies which are not perceptible, but also in the removal of high frequencies from edges. This loss is undesirable, considering the sensitivity of the human visual system to edge information. To accommodate for this loss, edges have to be enhanced after spatial filtering, a task that can be achieved using simple spatial filters or, better, frequency filters that take into account the viewing distance and concentrate on frequencies in selected ranges. Alternatively, the CSF can be modified to boost certain high frequency information [1]. The aim is to sharpen approximately in the range of 20-30 cpd [12].

In this implementation a 3x3 Laplacian was applied on the RGB channels, with form \([1 \ 1 \ 1; 1 \ -8 \ 1; 1 \ 1 \ 1]\). It was selected visually after trial and error. The filtered image was subtracted from the original (as the central co-efficient was selected to be negative) to obtain the enhanced image.

### 2.4 Local Contrast Detection

Local contrast detection is used to adjust differences in contrast in small image areas judged in relation to the average value of a localized region. This module essentially creates a family of gamma curves used to map one input value to many different output values, depending on the values of the neighboring pixels. Its implementation is considerably easy and is based on the non-linear masking suggested by Moroney [14]. The gamma curves are based on the low frequency information in the image as well as its global contrast, as shown in equation 5 below:

\[
gamma = \max\left(\frac{\text{input}}{\text{max}}, \frac{\text{median}}{\text{median-max}}\right)
\]  

(5)

Equation 5 was applied separately in each of the opponent CIELAB channels, with the maximum and median values corresponding to those of each channel. The mask for each channel was computed by applying a low-pass Gaussian with cut-off 0.05 in the frequency domain and then by inverting back to the spatial domain. Calculations in the chrominance channels, where values are both positive and negative, were carried out by taking the absolute input values.

### 2.5 Obtaining color differences and a metric

Pixel by pixel color differences were calculated between original image and each compressed version, using the CIEDE2000 formula [15]. Each of the ten originals was tested against all eight compressed versions (see section 3 - test images), resulting to a total number of 80 image pairs. A difference map was produced for each image pair. To make predictions on the overall relative quality of each compressed image, the error map was reduced to a single number. Several statistical measures (mean, median, standard deviation, coefficient of variation) were tested to ensure that the
option selected was the most suitable measure for the purpose. From an early stage the median of the error map appeared to best describe the changes between the different implementations and was the one which was finally selected to test the model predictions against subjective results.

3. IMAGE DATASET AND PSYCHOPHYSICAL DISPLAY

3.1 Test images

Images selected for this study were part of a larger test, employed in psychophysical investigations dealing with perceived image quality of JPEG and JPEG2000 compression schemes [9]. Ten high quality originals were used, shown in Appendix A. Nine of them were color images, chosen from a Master PhotoCD, opened at a resolution of 512 by 768 pixels at 16-bit per channel in CIELAB, converted to sRGB and down-sampled to 8-bit per channel and 317 by 476 pixel resolution (approximately 445Kb TIFF files). The 10th original was the monochrome version of ‘Lena’, commonly used in evaluation of image compression. Original scenes were chosen purposely to contain different amounts of detail, low varying areas, various degrees of local intensity and chromaticity, and a variety of dominant colours, strong and weak edges.

Eight different manipulations were applied on each original. They involved compressing at four compression rates - 20:1, 40:1, 60: and 80:1, using both JPEG and JPEG 2000. Compression rates were based upon JPEG rather than JPEG 2000 because of the relatively limited compression capabilities of JPEG. This set of compression rates was selected to cover a range used in everyday imaging across a variety of applications, particularly consumer imaging applications and the internet. JPEG compression was carried out in Advanced JPEG Compressor v4.1 [16] using baseline JPEG standard compression. JPEG 2000 compression was carried out in Lurawave SmartCompress 3.0 [17]. Default settings were used in both methods of image compression.

3.2 Psychophysical display and viewing conditions

Subjective ratings were obtained from a paired comparison experiment described in [9], in which observers were asked to select the preferred image from a pair displayed on screen, based upon the overall image quality. Ten observers with normal color vision and normal or corrected-to-normal visual acuity carried out the tests. For each particular test scene, each compressed version was compared with all the others and with the uncompressed original, making the total number of unique pairs 36 per scene.

Image pairs were displayed on a 15 inch CRT NEC Multisync M500 monitor, with a Matrox Graphics MGA Millenium graphics-cart adapter set at addressable resolution of 1024 by 768 pixels. The effective pixel pitch was approximately 0.29mm (approx. 85 ppi). To ensure correct color rendition, the monitor was characterized using the GOG model [18] and then calibrated to approximate the sRGB reference display; the viewing conditions were set accordingly [19]. Observers sat at a viewing distance of 60 cm from the display faceplate. That gave a visual resolution of 35 pixels per degree.

The experiment produced interval quality scales. Scaled values for each image were employed to derive subjective quality differences between original and each compressed version. The values were signed as positive or negative for images observed as better or worst with respect to the original.

4. OBJECTIVE VS SUBJECTIVE QUALITY

Color difference predictions were tested against subjective quality differences. Plots of the median of each difference map versus subjective quality are shown in Figures 2, 3 and 4, which represent implementations of 1- the CFS without spatial frequency adaptation, 2- the adapted CSF using the model described in equation 3, and 3- the adapted CSF using the model of which a general representation is given by equation 4, respectively.
Fig. 2. Model prediction vs relative subjective quality for the 1st CSF implementation.

Fig. 3. Model prediction vs relative subjective quality for the 2nd CSF implementation.
There are several aspects that are represented in plots in figures 2, 3 and 4. Any point on the left of the origin represents an image with worse quality than the original. Although compressed images are generally expected to have worse image quality than the original, it is noticed in the plots that there are exceptions. More specifically, ‘Lena’ compressed at 20:1 with JPEG and ‘africantree’ compressed at 20:1 and 40:1 with JPEG were preferred from the respective originals. This can be due to the fact that slight sharpening is produced by DCT compression at low compression levels [20], which was probably viewed as an improvement in both scenes which have inherently soft edges. Another aspect is related to the spreads of the predictions which generally gave more coherent results for the first and the third implementations of the CSF.

4.1 CSF – 1st implementation

Figure 2 shows the plot obtained by the first implementation of the CSF, where there spatial frequency adaptation was not applied. Median color differences have a certain distribution throughout the plot area. Firstly, there is a distinctive separation between the points of certain images. Points are split into two different image groups, each following a different linear trend indicating reasonable model predictions for the particular group. Each group comprises of points only from five out of the ten images, as listed in table 1:

<table>
<thead>
<tr>
<th>Group A</th>
<th>Group B</th>
</tr>
</thead>
<tbody>
<tr>
<td>bike*</td>
<td>kids*</td>
</tr>
<tr>
<td>boats*</td>
<td>formula*</td>
</tr>
<tr>
<td>motorace</td>
<td>Lena</td>
</tr>
<tr>
<td>chinatown*</td>
<td>africantree</td>
</tr>
<tr>
<td>glasses</td>
<td>yellowflowers*</td>
</tr>
</tbody>
</table>

Table 1: Observed image groups from the first and third CSF implementations

This observation is supported by deriving the individual group’s correlation coefficients for linear regression, as well as the correlation coefficient obtained when all data are considered as one group for comparison. $r^2$ coefficients are listed in

![Graph showing model prediction vs relative subjective quality for the 3rd CSF implementation.](image-url)
table 2. They indicate a better correlation when dividing the data points into two image groups than when they are considered as one group.

<table>
<thead>
<tr>
<th>Grouping</th>
<th>Correlation coefficients</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>0.3241</td>
</tr>
<tr>
<td>Group A</td>
<td>0.8329</td>
</tr>
<tr>
<td>Group B</td>
<td>0.6731</td>
</tr>
</tbody>
</table>

Table 2: Correlation coefficients for linear fitting on data obtained from the first CSF implementation

Further, the clusters created represent, for most images, groups of different compression ratio. Clusters were organized for ratios 80:1, 60:1, 40:1 and 20:1 from left to right – including data for both JPEG and JPEG2000. This second classification however is approximate and is not representative for all images in each cluster, because subjective ratings for each compression ratio were greatly scene dependent i.e. each image’s scale value was more or less deviated from the average scale value for each compression ratio and this was dependent on original scene content [9,21]. Image names accompanied by * in Table 2 indicate images of which image ratings were closer to mean subjective ratings and as a result they fall in this clustering.

4.2 CSF – 2nd implementation

Figure 3 shows the plot obtained by using the second implementation of the CSF, in which spatial frequency adaptation is based on the probability of frequencies present in the visual field - the “natural world assumption” [10]. In this plot the distribution of points is less structured. The clustering occurs in three levels and three different regression lines of images best describe the distribution of points for different image sets. \( r^2 \) correlation coefficients for linear fit were derived for the different groups and can be seen in the table 3. Again, the groups seem to have a better correlation than when all images are considered together.

<table>
<thead>
<tr>
<th>Grouping</th>
<th>Correlation coefficients</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>0.2465</td>
</tr>
<tr>
<td>Motorace &amp; bike</td>
<td>0.844</td>
</tr>
<tr>
<td>Chinatown, glasses, boats &amp; kids</td>
<td>0.6742</td>
</tr>
<tr>
<td>Formula, tree, lean &amp; yellow flowers</td>
<td>0.4805</td>
</tr>
</tbody>
</table>

Table 3: Correlation coefficient for linear fitting on data obtained from the second CSF implementation

4.3 CSF – 3rd implementation

Figure 4 represents the plot obtained when using the image dependent spatial frequency adaptation, where the CSF is altered based on the frequency information included in each image. There is, again, a definite separation between image points which belong to two distinct image groups, the same groups as in fist CSF implementation (see table 1). Compression ratio clusters are again obvious - although they are slightly less tight than the ones in the first implementation. Table 4 shows \( r^2 \) coefficients obtained from the linear regressions.

<table>
<thead>
<tr>
<th>Grouping</th>
<th>Correlation coefficients</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>0.2579</td>
</tr>
<tr>
<td>Group A</td>
<td>0.7133</td>
</tr>
<tr>
<td>Group B</td>
<td>0.6412</td>
</tr>
</tbody>
</table>

Table 4: Correlation coefficient for linear fitting on data obtained from the third CSF implementation
5. COMPENSATING FOR SCENE DEPENDENCY

Results from predictions indicate that, although the model is designed and was implemented to utilize information contained in the images themselves, it cannot compensate entirely for scene dependency present in subjective image quality results. This is demonstrated by the specific distribution of the images throughout the plots, for all three different CSF implementations. The results led to more research to discover the reasons behind the grouping. A commonality in images of group A and those of group B - obtained by the first and third implementations of the CSFs, where the results appeared more coherent - was sought, using image measures from reference 21. In this, scene analysis tools were employed to quantify scene content and explain the dependency in the subjective ratings for image compression. The measure that appeared to explain best the present grouping was related to the ‘busyness’ of the scene, defined as the image property indicating the presence or absence of details in the scene.

5.1 Calculating original image busyness

A metric for busyness in the scenes is obtained with a simple image segmentation technique, employed to separate slow varying areas from busier areas within scene [22] – in this case the uncompressed original. An illustration of the steps involved is shown for one test-original in figure 5. The steps include:

- the calculation of the gradient image of the CIELAB L* channel, by applying the Sobel edge detector in both horizontal and vertical orientations, and using a very low threshold of 0.04,
- the dilation of the binary image to amplify the detail, using flat linear structuring elements,
- the use of a flood filling operation to fill the holes in the dilated image,
- the erosion of the binary image to get rid of spurious noise.

The Sobel edge detector is dependent on viewing distance. Thus, the common threshold for the calculation of the gradient image was derived empirically, after careful observations and is considered appropriate for the segmentation process of images, displayed at the given visual resolution. (This potentially can be related to specific visual spatial frequencies.) The value $b$, expressed as a percentage, is finally used as a metric of scene busyness. It is calculated from the ratio of the number of white pixels in the threshold image - that indicated the busy image areas - to the total number of pixels.

![Fig 5: Stages of the image segmentation process used to separate slow varying areas from busier areas of the test image ‘yellowflowers’ [21]](image-url)
Table 5 lists the $b$ values, along with their rank order with respect to all the test scenes in original study [9] and for the test scenes employed in the current work (a sub-set).

<table>
<thead>
<tr>
<th>Group</th>
<th>Busyness, $b$</th>
<th>$b$ rank</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Group A</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bike</td>
<td>83.65</td>
<td>8 (13)</td>
</tr>
<tr>
<td>boats</td>
<td>60.29</td>
<td>7 (10)</td>
</tr>
<tr>
<td>chinatown</td>
<td>84.60</td>
<td>9 (14)</td>
</tr>
<tr>
<td>glasses</td>
<td>44.20</td>
<td>5 (5)</td>
</tr>
<tr>
<td>motorace</td>
<td>89.25</td>
<td>10 (15)</td>
</tr>
<tr>
<td><strong>Group B</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>africantree</td>
<td>2.38</td>
<td>1 (1)</td>
</tr>
<tr>
<td>formula</td>
<td>32.73</td>
<td>2 (2)</td>
</tr>
<tr>
<td>kids</td>
<td>40.48</td>
<td>4 (4)</td>
</tr>
<tr>
<td>Lena</td>
<td>50.10</td>
<td>6 (8)</td>
</tr>
<tr>
<td>yellowflowers</td>
<td>35.41</td>
<td>3 (3)</td>
</tr>
</tbody>
</table>

Table 5: ‘Busyness’, $b$, and rank for the ten test scenes.

It can be seen from table 5, that Groups A and B formed in the results from the model predictions represent images with more or less detail, respectively - with a 50% threshold. The one exception in Group A is ‘glasses’ that has less than 50% busyness. The inverse is almost true for Group B, in which only ‘Lena’ has (just) more than 50% busyness. The ranks in parenthesis indicate image order with respect to all 15 scenes in the original study.

### 6. DISCUSSIONS & CONCLUSIONS

The work focused on the implementation of relevant modules included in the iCAM, to obtain predictions on the overall image quality of compressed images with JPEG and JPEG2000. Results from the model implementation were correlated with subjective ratings and overall trends were satisfactory. Observation of the relevant plots leads to the indication that there is a scene dependency factor affecting the distribution of predictions for the ten different test images. This, despite the fact that the applied model included steps that utilize information contained in the images themselves.

More specifically, a measure relating to scene busyness demonstrated good correlations with the two groups of image points shown in results from the first and third CSF implementations of the model. Busyness relates directly to the spatial frequencies in images themselves. So, one would expect that the 3rd implementation of the CSF, which uses a spatial frequency adaptation model that accounts for the image frequencies, could compensate for this scene dependency – especially when compared with the first implementation which did not model spatial frequency adaptation. Further research should pursued toward describing how various amounts of different spatial frequencies (or busyness) in a scene might affect predictions from the modular image color difference model on compressed image quality.

The authors would like to thank G.M. Johnson for useful communication and MATLAB code.

### REFERENCES


G. M. Johnson (private communication and relevant code provided).


Code provided by: triants@wmin.ac.uk