Automatically choosing source color images for coloring grayscale images

Luiz Filipe M. Vieira Rafael D. Vilela Erickson R. do Nascimento
Fernando A. Fernandes Jr. Rodrigo L. Carceroni Arnaldo de A. Araújo

Departamento de Ciência da Computação – Universidade Federal de Minas Gerais
Av. Antônio Carlos, 6627, Pampulha – Belo Horizonte, MG, CEP 31270-010, Brazil
{*lfvieira, vilela, erickson, junior, carceron, arnaldo*}*@dcc.ufmg.br*

Abstract

This paper introduces a methodology for adding color to grayscale images in a way that is completely automatic. Towards this goal, we build on a technique that was recently developed to transfer colors from a user-selected source image to a target grayscale image. More specifically, we eliminate the need for human intervention in the selection of the source color images, which can be regarded as a first step towards real-time video colorization. To assess the merit of our methodology, we performed a survey where volunteers were asked to rate the plausibility of the colorings generated automatically for individual images. In most cases automatically-colored images were rated either as totally plausible or as mostly plausible.

1 Introduction

Adding color to grayscale images and movies in a way that seems realistic to most human observers is a problem that greatly challenged the motion picture industry in the 1980s and has recently attracted renewed interest within the Computer Graphics community [17]. While coloring of historic photos and classic movies has usually been done with the purported intent of increasing their visual appeal, there are certain coloring techniques – such as luminance-preserving pseudo-coloring – that have been specifically developed to facilitate the visualization of scientific, industrial and security images [13, 7].

In any of those cases, the coloring problem amounts to replacing a scalar value stored at each pixel of a grayscale image (e.g. luminance) by a vector in a multi-dimensional color space (e.g. a three-dimensional vector with luminance, saturation and hue). Thus, this is in general a severely under-constrained and ambiguous problem for which it makes no sense to try to find an *optimum* solution, and for which even the obtainment of *reasonable* solutions requires some combination of strong prior knowledge about the scene depicted and decisive human intervention. In this article, we present a methodology to color grayscale images in a fully automatic way that compensates the lack of human intervention by using a database of color images as a source of *implicit* prior knowledge about color statistics in natural images.

The invention of colorization – the process of "enhancing" monochromatic movies through the addition of color – is widely attributed to C. Wilson Markle, whose company Colorization, Inc. released the first full-length colorized movie in August 1985. Although colorization became a very profitable business by the late 1980s, the technology used at that time was painstakingly manual and, thus, expensive and time-consuming. For instance, the colorization of the classic movie Casablanca in 1988 took more than two months and nearly US$ 450,000 to complete, in a process that involved – among many other labor-intensive steps – researching historical wardrobe notes from the original movie's set to discover the specific colors that the actors and actresses wore in most of the scenes [3]. Eventually, as the novelty of colorization wore off and audiences dwindled, such high costs grew unsustainable and colorization of classic movies became an unfashionable and generally unprofitable activity. In reality, colorization of artworks such as Casablanca has always been a very controversial subject with delicate moral, ethical and legal implications [11].

However, there are nowadays more "mundane" applications that could benefit from fully automatic, inexpensive, efficient colorization techniques, even if these techniques turn out to be somewhat less reliable that those used in the 1980s. For instance, consider a scenario where two people that chat regularly through the Internet decide to enhance their virtual meetings with live video. If fully automatic, real-time colorization software was available to them, they might buy less expensive monochromatic webcams instead of color ones, use limited bandwidth by transmitting monochromatic video, but still be able to view fully colored video streams. Even techniques that work only with static pictures – such as those that we will present in this article – can be used to increase of visual appeal of scientific, educational and commercial presentations (and of some domestic pictures), by adding color to various types of single-band images, such as those obtained with X-rays, MRI, CT, ultrasound, electronic microscopy, and near-IR, thermographic and UV cameras.

In the real-time colorization scenario above, traditional techniques where segmented image regions are hand-colored one by one are grossly inadequate. An important step towards minimizing the amount of human intervention needed to color grayscale images was the recent work of Welsh et al. [17], who developed techniques to transfer the chromatic information from a source color image to a target grayscale image. Unfortunately, empirical evidence suggests that the degree of similarity between these source and target images has a strong influence on the quality of the results obtained. Thus, obtention of reasonable coloring with the techniques developed by Welsh et al. is, in principle, still strongly dependent on human selection of an appropriate source color image for each given grayscale image.

Here, we move another step towards automatic coloring of grayscale images, with a methodology where source color images are automatically selected from an image database. More specifically, we designed, implemented and experimentally assessed four techniques to choose images from a database, without any human intervention, to be used as a source images in color transferring. These techniques leverage on some of the most traditional ideas [16] on the very active area of content-based image retrieval.

In a broader context, our work is related to a recent surge of interest on the general problem of transferring properties from one image (or movie) to another. For instance, Hertzmann et al. [9] developed a technique that can learn various types of filtering transformations from pairs of images and then apply the learned filters to other images that, as a result of the process, inherit properties from the original images. In addition to this general technique, a variety of methods specifically tailored for transferring a single property such as radiometric texture [1], geometric texture [10], illumination [2] or motion [6] exist. In particular, Reinhard et al. [14] and Greenfield and House [8] have recently introduced methods for transferring color between images. However, they assume that the target images already contain chromatic information and one merely wants to change this information in some meaningful way, which is a problem that is in general easier than the one we address in this article.

2 Overview of image coloring system

Figures 1 to 3 illustrate the proposed methodology to color a grayscale image using information from a database of color images. Initially it is necessary to index the database by performing some type of ideally semantics-preserving compression [12] of each image in the database into a signature vector that must be as small as possible to maximize the efficiency of queries to the database later on. Since these signature vectors will eventually be compared against those of the grayscale images passed as input in the queries, they are extracted solely from the luminance component of each color image in the database (Figure 1).

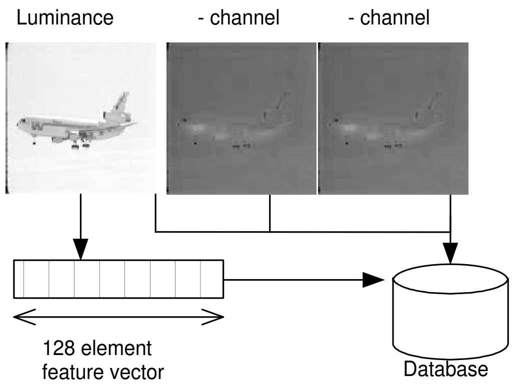

Figure 1. Image insertion in the database.

Once signature for all images in the database have been collected, the database is ready to start answering queries. Each grayscale image received in this phase is similarly reduced to a signature vector. Using a pre-defined similarity metric, this signature is then compared to those stored in the database index and the content of the database image with the most similar feature vector is returned (Figure 2).

Finally, the color transfer process takes two images, the grayscale one passed as input in the query and the color

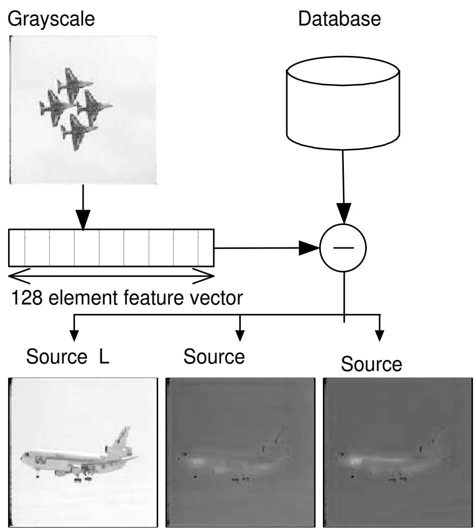

Figure 2. Query to find the a source color image for a given grayscale image.

one returned as the query answer, and adds to each scalar pixel of the former, the chromatic components of an automatically chosen pixel of the later. The actual results of this process, using one of the retrieval techniques described in Section 3, the color transfer process described in Section 4 and images from the database described in Section 5 is shown in Figure 3.

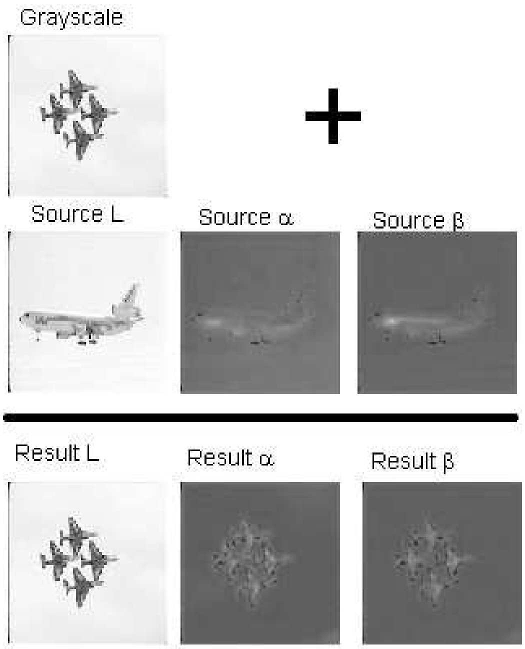

Figure 3. The color transfer process.

3 Image retrieval techniques

In this article we study four techniques – Luminance histogram, Luminance subsampling, Luminance + gradient histogram, and Luminance + gradient subsampling – to select the source image for color transfer automatically. As explained in the previous section all of them build database indices that depend exclusively on the luminance component of the database images. Thus, we start this section by describing a pre-processing step common to all these techniques: the decomposition of the source images in a component that has only luminance information and two mutually uncorrelated chromatic components. Then, on separate subsections, we focus on the particular aspects of each one of these four techniques: the way in which they process the luminance information to create the image signatures. In a final subsection, we describe another common point among all techniques in study: the way in which they compare the feature vectors, once these vectors have all been assembled.

3.1 Conversion to the $l\alpha\beta$ color space

The $l\alpha\beta$ color space was proposed by Ruderman et al. [15], as a result of an analysis of the statistics of human photosensor responses to natural images. More specifically, it was observed that while the responses on the bands of the three types of cones present in the human eye are strongly correlated, if these responses are converted to the $l\alpha\beta$ space, the existing correlation disappears. Reinhard et al. [14] realized the implication of this for color transfer: by converting images to the $l\alpha\beta$ space, each one of these channels can be altered independently, with low occurrence of undesirable cross-channel artifacts. In the case of transferring color for graylevel images we not necessarily need independence between the two chromatic channels, but conversion to the $l\alpha\beta$ space is still and effective way of decoupling luminance from all chromatic information.

In addition, the $l\alpha\beta$ space has a property that is very useful in the case of image databases: because it is logarithmic, it is not affected by gamma correction [14], a non-linear transformation that is often applied to images to improve exhibition in monitors. Because this transformation has a parameter that may vary from image to image, it is prone to cause trouble in linear color spaces.

Thus, we pre-process every database image by converting it to the $l\alpha\beta$ space, where the l component is a measure of luminance, α measures variations in the green-red chromatic axis (with positive values meaning red and negative ones meaning green) and β measures variations in the blue-yellow axis (with positive values meaning yellow and negative meaning blue). More specifically, each image, originally coded in the RGB space is initially converted to the LMS space, which corresponds to the bands of sensitivity of the human cones and was thus used by Ruderman et al. as the basis to define the $l\alpha\beta$ space. This initial transformation corresponds to a multiplication by a 3×3 matrix, as expressed in Equation (1):

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1967 & 0.7244 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

The conversion from LMS to $l\alpha\beta$ is the performed through the sequential application of the logarithmic transformation expressed in Equation 2 and the linear transformation expressed in Equation 3:

$$[\mathbf{L}, \mathbf{M}, \mathbf{S}] = [\log \mathrm{L}, \log \mathrm{M}, \log \mathrm{S}] \quad (2)$$

$$\begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{L} \\ \mathbf{M} \\ \mathbf{S} \end{bmatrix} \quad (3)$$

Finally, after the color-space conversion is finished, a 128-level quantization is performed on each channel.

3.2 Luminance histogram

The use of similarity between intensity histograms as a relevant criterion to perform content-based image retrieval is a well-established idea [16] that has been used in a myriad of works. Thus, its choice as one of the methods to be evaluated for suitability to the novel application that we address in this paper is a natural one. Formally, the histogram of an $m \times n$ monochromatic digital image $\mathcal{I}$ is a discrete function that maps each value k in the image's intensity range to the fraction of pixels in image $\mathcal{I}$ that have intensity k:

$$h_{\mathcal{I}}(k) = \frac{|\lambda_{\mathcal{I}}(k)|}{m \times n}, \qquad (4)$$

where $\lambda_{\mathcal{I}}(k) = \{(x, y) : \mathcal{I}(x, y) = k\}$, and $\mathcal{I}(x, y)$ is the intensity of pixel (x, y) in image $\mathcal{I}$.

Here, we use this function to generate image signatures by applying it to the l channel obtained after the color-space conversion described in the previous section. The only precaution that we have to take is to perform a linear normalization of the channel before computing its histogram, to factor out fluctuations in the illumination conditions under which the various images were acquired and the effects of gamma correction [14]. Since we defined that the l channel admits 128 possible values after the post-conversion quantization, its histogram will consist of a vector of 128 pixel counts, scaled to unit L_1 norm. Moreover, since the l is uncorrelated with the chromatic bands, comparable histograms can be computed for the graylevel images received as queries, as long as they are pre-processed with a logarithmic transformation followed by a linear normalization.

Thus, for each image on the database, a signature that consists of the 128 values of its luminance histogram is created. By comparing each of these signatures against the one obtained from a query image and selecting the best match, the color image that will be used as the source for coloring the query image is identified (Figure 4).

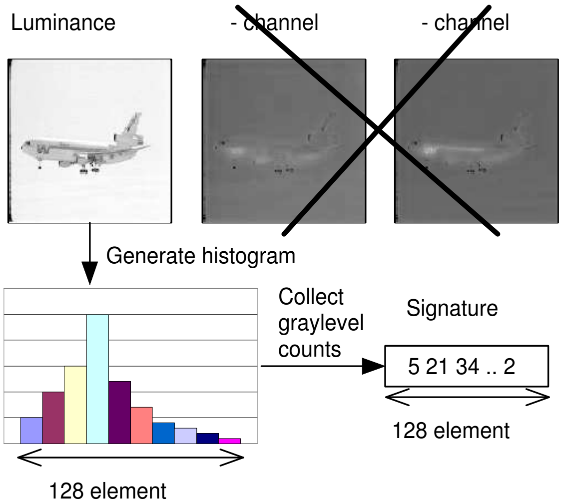

Figure 4. Luminance histogram.

3.3 Luminance subsampling

Another quite intuitive and broadly-used way of compressing an image into a signature is plain subsampling. In this technique the properly pre-processed (i.e., quantized, linearly normalized) l channel is convolved with a low-pass filter, the resulting image is divided in a regular grid of 128 regions of the same size, and the mean intensity within each such region is computed. Contrary to histograms that are invariant to the positions and orientations of the various objects within an image, subsampling preserves information related to the spatial distribution of intensities within the image – which may be important to disambiguate, for instance, sky from water – but on the other hand it obliterates all clues about the existence or not of fine texture in an image. The process of identifying the source color image for a given query through Luminance subsampling is illustrated in Figure 5.

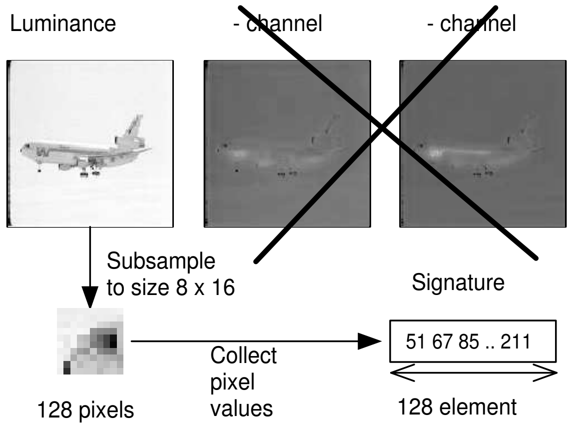

Figure 5. Luminance subsampling.

3.4 Luminance+gradient histogram

Texture is a property with high semantic content in many application domains. In order to ameliorate the loss of this information in the Luminance histogram and, especially, in the Luminance subsampling, we considered the use of signatures that are computed not only from the luminance channel, but also from its spatial gradient.

One technique to generate these signatures uses what we call a Luminance-and-gradient histogram, It starts by convolving the luminance image with a Sobel kernel, to estimate the image's spatial gradient. Then, it computes histograms both for the luminance image and for its gradient, but this time considering a more coarse quantization of only 64 values on each image, so that size of the signature does not need to be increased. The final signature for each image is then obtained simply by a concatenation of the normalized pixe counts from the two histograms, in a pre-defined order (Figure 6).

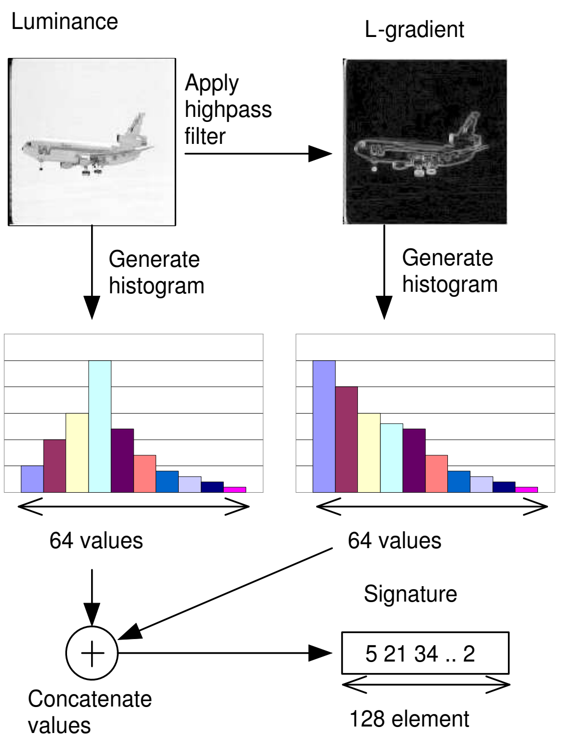

Figure 6. Luminance+gradient histogram.

3.5 Luminance+gradient subsampling

The last technique studied also aims at preserving information about fine textures in the image signatures, but it combines gradient extraction with subsampling rather than with histograms. Once more, the luminance image is initially convolved with a Sobel kernel, but now the pre- and post-filtering images are simply subsampled to a size of 8×8 pixels each. The two 64-element vectors formed by the pixels of these subsampled images are then simply concatenated to generate a signature that has the same number of bits than those generated by the other three techniques (Figure 7).

3.6 Similarity metric

Regardless of which of the techniques described in the previous four subsections is used, the criterion used to compare the image signatures and thus to determine the best match is the same: the correlation between the query and database signature vectors, calculated using their internal product:

$$\text{match}(\mathcal{Q}) = \text{argmax}_i \frac{\mathbf{q} \cdot \mathbf{d}_i}{||\mathbf{q}|| \cdot ||\mathbf{d}_i||}, \tag{5}$$

where $\mathcal{Q}$ and $\mathbf{q}$ are, respectively, the query image and its signature, and $\mathbf{d}_i$ is the i-th signature in the database.

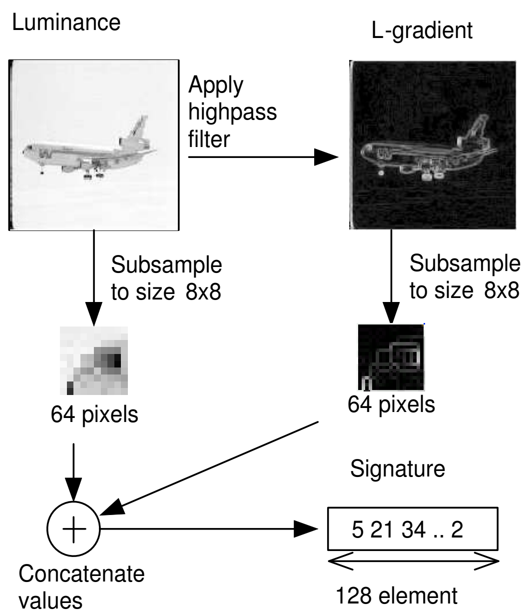

Figure 7. Luminance+gradient subsampling.

4 Transferring Color to Grayscale Image

Once an appropriate source color image has been selected for a given target grayscale image by one of the techniques described in the previous section, we use the method of Welsh et. al. [17] to transfer the chromatic information from the source to the target image, while keeping the luminance of the later unchanged.

Welsh's technique compares small pixel neighborhoods on the luminance channels of the source and target images, in order to try to identify, for each pixel in the target image, a matching pixel in the source image that has similar neighborhood statistics. Thus it also requires a conversion of all database images to the $l\alpha\beta$ color space. However, since this conversion is already done a priori, during the indexing of the database, this is actually a trivial requirement.

To speed up the pixel-to-pixel matching process, the method considers only a very sparse set of pixels in the source image – typically around 200 pixels selected by jittered sampling. The target image is scanned in scan-line order so that, for every one of its pixels, the source sample that is closest in terms of intensity value (50% weight) and intensity standard deviation within the surrounding 5×5 neighborhood (50% weight) is selected as a match. To create better correspondence between the two images, luminance remapping [9] is performed. This linearly shifts and scales the luminance histogram of the source image to fit the histogram of the target image. Once the best source pixel is identified for a given target pixel, the α and β values of the former are added to the later, which eventually leads to the complete colorization of the target image. This approach reportedly works well on image pairs where luminance correspondence between neighborhoods is correlated with color correspondence.

However, since there are quite a few image pairs that, in spite of being semantically close, do not satisfy this requirement, Welsh et. al. [17] also developed an interactive variant of their technique where pairs of matching source and target image swatches are hand-picked by a human operator. Once the target swatches are colored with the α and β channels of the source swatches, colors can be propagated to the rest of the target image with texture synthesis techniques [4, 5]. Obviously, this in general yields more convincing results than automatic coloring, but the need for human intervention eliminates the possibility of using this alternative process in real-time video applications, for instance. So, we did not perform any experiments involving manual selection of swatches. Nonetheless, there is nothing in this alternative scenario that prevents the use of any automatic source image selection technique that we have discussed so far. We simply regard this as a topic for future work.

5 Experimental evaluation

In order to assess the four techniques in study we used them with the same database of natural images, developed at the ENSEA University. This database has 14 semantically distinct classes, with a total of 1200 color images of natural scenes (Figure 8). The query images were also taken from the same database but, of course, each image submitted in a query was temporarily deleted from the database while the query was processed.

The tests were conducted in two phases. In the first phase, we observed the results produced by the four techniques on the coloring of a subset of the database. Based on these observations we made certain hypotheses about the behavior and appropriateness of the techniques for different types of images. In the second phase, these hypotheses were validated on queries with a distinct subset of the database, whose results were presented to over 40 volunteers, who were asked to grade them according to their plausibility.

Figure 8. Sample images from database used in all experiments. Here we show three of the 14 different classes of images (faces, sunsets and cars).

More specifically, in the first phase we chose one image of each class. Since classes have different numbers of images, a number i between one and six (the smallest number of images presented in a class) was randomly selected and the i-th image of each class was selected to be colored automatically. By observing the results obtained in this phase, we noticed that the methods that do *not* use information derived from the image gradient tended to work much better in uncluttered scenes, where there is a single object in the foreground and this object is clearly discernible from a mostly homogeneous background. We named the set of all database images that conform to this loose description the *homogeneous* group, and the set of images that do not conform to it (and thus have multiple objects on the foreground, or have a cluttered background, or are illuminated in an uneven way) the *heterogeneous* group. We also observed that the methods that *do* use image gradients tended to outperform their counterparts in these more complex, heterogeneous images.

Based on this preliminary assessment, we selected 15 new query images from the database, at random. Seven of these 15 images were classified as *homogeneous* and eight of them were classified as *heterogeneous*. For each technique, and for each one of the 15 images, we obtained the resulting color-enhanced image by applying the color transfer method described in Section 4. We developed a web site to assess the plausibility of the resulting colored images. Each volunteer that accessed this site was asked to choose among four possible classifications for each image, as shown in Figure 9. In addition each volunteer was also shown eight "placebo" images that had real colors and had never been colorized, as well as eight images colored using randomly-chosen color sources.

From the votes cast for each image, we computed net statistics on the plausibility of the output of each method. As one can observe in Figure 10, all four techniques produce images that are not as good as real color images but are still much better than those obtained by transferring color from a randomly selected source. Among the four methods, the ones that do *not* use gradients have a slight edge in plausibility.

If we look specifically at the images in the *homogeneous* group (Figure 11), then the results generated with the Luminance histogram are noticeably better than the ones generated with the other three methods This is reasonable: since

Figure 9. Partial screen capture from the coloring evaluation website.

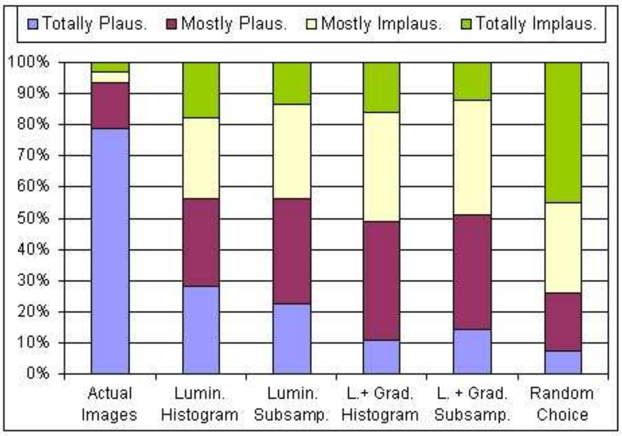

Figure 10. Breakdown of votes for each technique, on all 15 images.

the homogeneous images do not vary much, spatial location of the colors within them matters less that in the *heterogeneous* group and the gradient carries little information. An example of coloring of an image in this group is the picture with the set of fighter planes displayed in Figure 3. Luminance histogram matching identifies the picture of a commercial jet as a good source for color transfer in a large part due to the blue background that corresponds to the sky. This blue hue is then correctly transferred to the query image's background (notice the dark α and β channels in most of the resulting image). Of the 41 people that evaluated this image, 29 votes were for *totally plausible*, 10 votes for *mostly plausible* 2 votes for *mostly implausible* and no one thought it was *totally implausible*.

In the *heterogeneous* group (Figure 12), on the other hand, both spatial localization information and gradients matter and, consequently, the Luminance+gradient subsampling is the technique that performs better. Examples of how the Luminance histogram and the Luminance+gradient subsampling techniques perform on this type of image are given in Figures 13 and 14, respectively. Because of the strong illumination in the background, Luminance histogram matching returns the image of a polar bear, which generates a final coloring with very little saturation, evaluated as *totally* or *mostly plausible* by only 1 and 8 voters, respectively, out of a total of 42. Luminance+gradient subsampling, on the other hand, finds a picture that is not only of lions, but also has the same kind of non-uniform illumination, generating a more saturated coloring that is classified as *totally* or *mostly plausible* by most respondents.

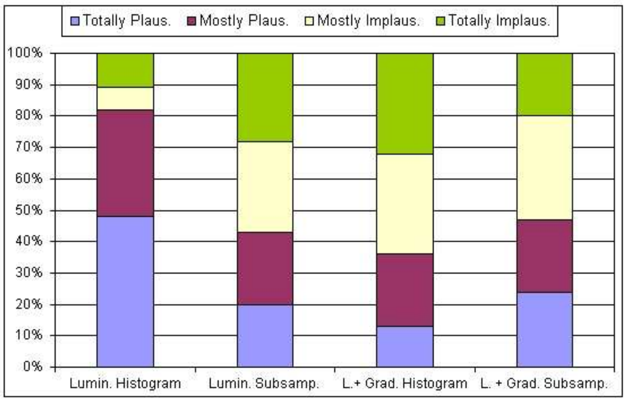

Figure 11. Breakdown of votes for each technique, on the homogeneous images only.

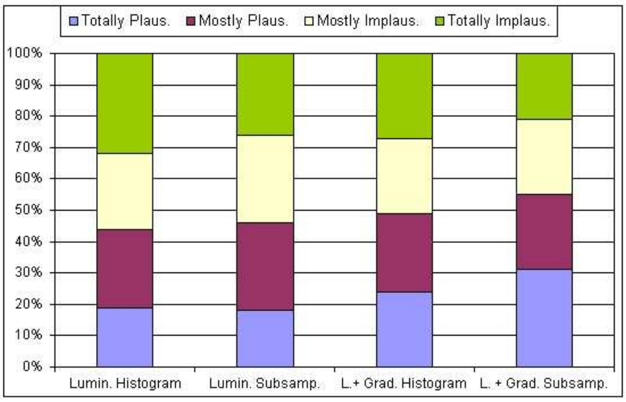

Figure 12. Breakdown of votes for each technique, on the heterogeneous images only.

6 Conclusion

In the current article we have demonstrated that it is possible to color many grayscale images in a way that is completely automatic and looks plausible to most people. The proposed coloring methodology blends existing works on color transferring with techniques from the very active area

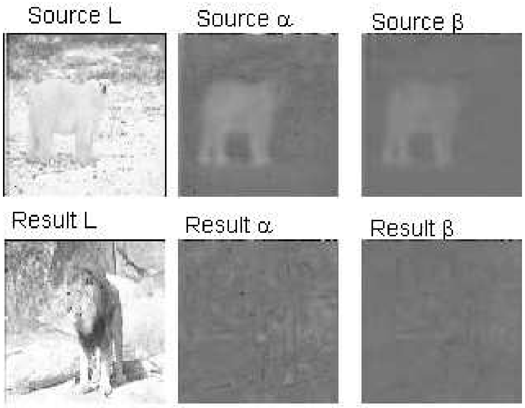

Figure 13. Coloring of an heterogeneous image with the Luminance histogram technique.

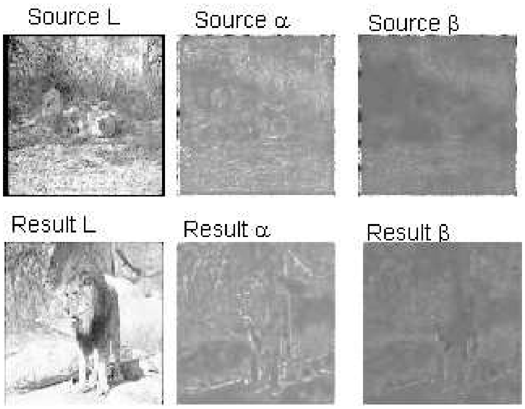

Figure 14. Coloring of an heterogeneous image with the Luminance+gradient subsampling technique.

of content-based image retrieval, in order to use an entire image database as the color source, rather than an individual user-selected image. While this methodology still produces some grossly miscolored images, we have to stress that in many applications automatic coloring is not meant to replace techniques that rely on human intervention completely, but only to reduce the number of images where these more labor-intensive techniques have to be applied. For instance, if an image generated automatically is found to be *mostly plausible*, perhaps all that the user needs to do is to go ahead and change the color in one or two segmented regions manually, which will take much less time than a fully manual coloring of the entire image.

Moreover, we have found that different techniques are appropriate for different types of images. Thus, there is a lot of room for improvement if we look deeper into the vast literature on content-based image retrieval, and incorporate the various state-of-the-art techniques in this topic within the framework that we proposed here.

This research has been supported by CNPq, by Fapemig and by PRPq-UFMG (Fundo Fundep RD).

References

[1] M. Ashikhmin. Fast texture transfer. *IEEE Computer Graphics and Applications*, 23(4):38–43, 2003.

[2] P. Debevec, T. Hawkins, C. Tchou, H.-P. Duiker, W. Sarokin, , and M. Sagar. Acquiring the reflectance field of a human face. In *Proc. ACM SIGGRAPH*, 2000.

[3] G. R. Edgerton. The germans wore gray, you wore blue. *Journal of Popular Film and Television*, Winter 2000.

[4] A. Efros and T. Leung. Texture synthesis by non-parametric sampling. In *Proc. 7th Int. Conf. Comp. Vis.*, pages 1033–1038, 1999.

[5] A. A. Efros and W. T. Freeman". Image quilting for texture synthesis and transfer. In *Proc. ACM Siggraph*, pages 341–346, 2001.

[6] M. Gleicher. Retargetting motion to new characters. In *Proc. ACM SIGGRAPH*, pages 33–42, 1998.

[7] R. C. Gonzales and R. E. Woods. *Digital Image Processing*. Addison-Wesley Publishing, 1987.

[8] G. R. Greenfield and D. H. House. Image recoloring induced by palette color associations. In *Proc. Int. Conf. in Central Europe on Computer Graphics, Visualization and Computer Vision*, 2003.

[9] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin. Image analogies. In *Proc. ACM SIGGRAPH*, pages 327–340, 2001.

[10] A. Hertzmann, N. Oliver, B. Curless, and S. M. Seitz. Curve analogies. In *Proc. Eurograpics Workshop on Rendering*, 2002.

[11] L. of Congress. U.S. Copyright Office. Technological alterations to motion pictures and other audiovisual works : implications for creators, copyright owners, and consumers, 1989.

[12] A. Pentland, R. Picard, and S. Sclaroff. Photobook: Content-based manipulation of image databases. In *Proc. SPIE Storage and Retrieval for Image and Video Databases*, 1994.

[13] W. K. Pratt. *Digital Image Processing*. John Wiley & Sons, 1991.

[14] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley. Color transfer between images. *IEEE Computer Graphics and Applications, special issue on Applied Perception*, pages 34–41, 2001.

[15] D. L. Ruderman, T. W. Cronin, and C.-C. Chiao. Statistics of cone responses to natural images: implications for visual coding. *J. Opt. Soc. Am. A*, 15(8):2036–2045, 1998.

[16] M. Swain and D. Ballard. Color indexing. *Int. J. Comp. Vis.*, 7(1):11–32, 1991.

[17] T. Welsh, M. Ashikhmin, and K. Mueller. Transferring color to greyscale images. In *Proc. ACM SIGGRAPH*, pages 277–280, 2002.