

Adaptive Color Watermarking

Alastair Reed, Brett Hannigan
Digimarc Corp., Tualatin, Oregon

ABSTRACT

In digital watermarking, a major aim is to insert the maximum possible watermark signal while minimizing visibility. Many watermarking systems embed data in the luminance channel to ensure watermark survival through operations such as grayscale conversion. For these systems, one method of reducing visibility is for the luminance changes due to the watermark signal to be inserted into the colors least visible to the human visual system, while minimizing the changes in the image hue. In this paper, we develop a system that takes advantage of the low sensitivity of the human visual system to high frequency changes along the yellow-blue axis, to place most of the watermark in the yellow component of the image. We also describe how watermark detection can potentially be enhanced, by using a priori knowledge of this embedding system to intelligently examine possible watermarked images.

Keywords: Watermarking, color watermarking, human visual system, Digimarc

1. INTRODUCTION

At a recent conference, John Gage, chief researcher at Sun Microsystems, stated “every surface, floor, wall, curtain – even clothes – will carry information, and we can only guess how this capacity will be used.” [1] While this statement may be guilty of a bit of hyperbole, the field of digital watermarking takes full advantage of the fact that many every day articles can hold information in addition to serving their primary purpose. In particular, digital image watermarks function by imperceptibly tweaking pixel values within an image to encode some piece of digital information. This additional data can be used to carry content ownership information, uniquely identify the image for database storage, or help identify counterfeit documents.

In many instances the digital watermark embedder is faced with two conflicting goals: first, to make the watermark as robust as possible, to ensure that the watermark remains in the image through malicious attacks and non-malicious image manipulations, and secondly to make the watermark as imperceptible as possible, to preserve the original image fidelity. A number of researchers have suggested methods to exploit properties of the human visual system in order to increase the amount of watermark signal that can be embedded without noticeably altering the image content [2][3]. The human eye’s varying contrast sensitivity, color sensitivity, and edge perception, are just a few of the characteristics which have been used to help control how strong of a watermark signal image areas can imperceptibly hold.

In this paper, we develop a system that takes advantage of the low sensitivity of the human visual system to high frequency changes along the yellow-blue axis, to place most of the watermark in the yellow component of the image. The algorithms developed are careful to minimize perceived hue changes to which the human visual system is particularly sensitive [4]. We also describe methods of matching watermark detection with the image adaptive watermark embedding. We follow this discussion with results from embedding tests, and conclude with an analysis of the effectiveness of this system and possible extensions.

2. EMBEDDING METHOD

2.1 Color Spaces

Many watermark systems embed a signal by adjusting the digital RGB, CMYK, or other color space values of an image to achieve a given luminance change. The signal is often embedded as a luminance change so that the signal will survive color to grayscale conversions and also to allow detection when using grayscale scanners or cameras.

The signal visibility is reduced and robustness improved by using spread spectrum techniques to embed a given watermark message. Additionally, because luminance is just one-dimension of a multi-dimension color space, a given luminance change can be obtained by many different combinations of color changes. This effect is easier to visualize in two dimensions as shown in Figure 1 below:

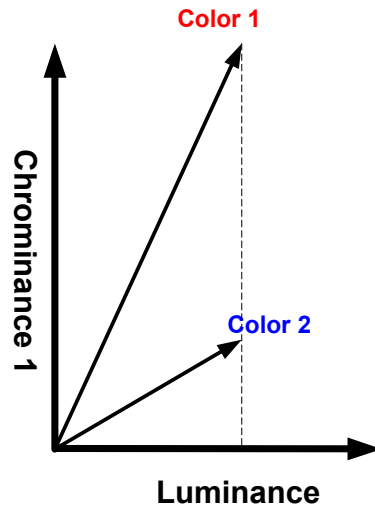


Figure 1: Two Dimensional “Luminance” Signal

In figure 1 we envision a two-dimension projection of a three dimensional color space consisting of a luminance component and two chrominance components. The two colors shifts represented, Color 1 and Color 2, may differ greatly in their appearance even though they represent the same luminance shift.

A similar concept is extended to 3 dimensions, in Figure 2, to envisage how color changes in different directions in color space will produce the same luminance change. Mathematically, the total luminance shift of a given color in color space can be determined by calculating the dot product of the color vector and the luminance unit vector.

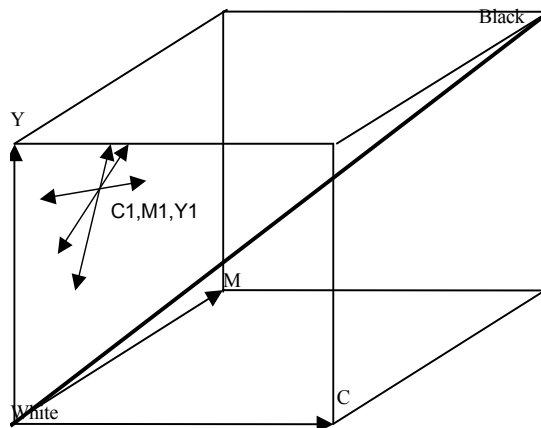


Figure 2: Directions in Color Space for Same Luminance

Colors have three main perceptual attributes, which are hue, saturation and brightness [4]. The most obvious is hue, denoting whether the color appears red, orange, yellow, green, blue or purple (or some mixture of neighboring pairs on this list). Saturation denotes the extent to which the hue is apparent; saturation is therefore zero for white, grays and black, is low for pastel colors, and is high for the colors of

the spectrum. Brightness denotes the lightness of a color; low for black, medium for grays and browns, and high for yellow and white. These three attributes are orthogonal to each other, thus saturation and brightness can be changed without affecting the hue of a color.

In this paper, saturation is approximated as a vector between white and the color as shown in Figure 3, and brightness is approximated as a vector between black and the color as shown in Figure 4.

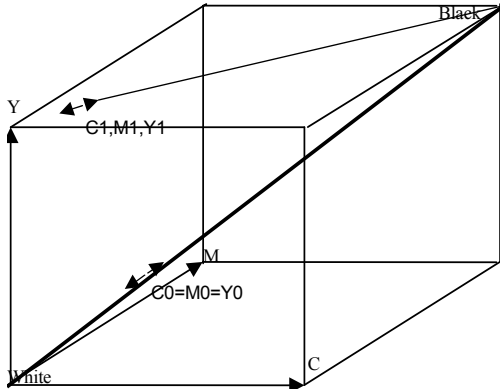


Figure 3: Vector Scaled Towards Black

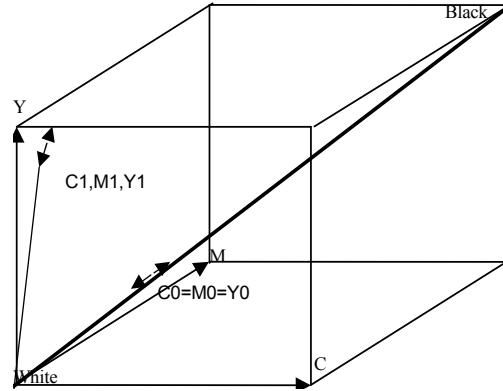


Figure 4: Vector Scaled Towards White

The watermark is applied by scaling either one, or a combination of these two vectors, in order to put the signal predominantly in the yellow channel while determining the other colors to modify to leave the hue approximately the same.

2.2 Human Visual System

When deciding the optimum direction to move a color in color space, our algorithm takes account of two main facts about the human visual system.

- 1) The eye is particularly sensitive to changes in image hue.
- 2) The eye is about a fifth less sensitive to high frequencies in blue-yellow than luminance [5]

The contrast sensitivity of the eye to changes along black to white, red to green and blue to yellow is shown in Figure 5.

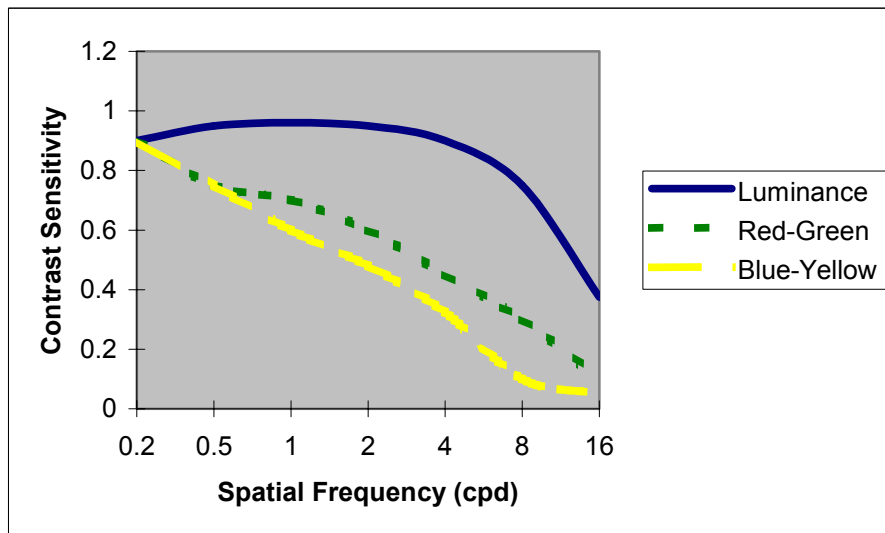


Figure 5: Eye Contrast Sensitivity Function

2.3 Adaptive Color Algorithm

Using these two facts about the Human Visual System, our algorithm is designed to preferentially place the watermark in the yellow channel, while minimizing any hue change. To accomplish this, the algorithm first calculates the amount of yellow contained in a given color. If yellow is a secondary color, the algorithm chooses to scale to black, which embeds the signal into the secondary colors. A secondary color component is the ink (or inks) with a lower percentage. Adjusting the secondary colors is similar to adjusting the brightness. If yellow is a dominant color, the algorithm chooses to scale to white, which embeds the signal into the dominant colors. A dominant color component is the ink (or inks) with a higher percentage. Adjusting the dominant colors is similar to adjusting the saturation. Since brightness and saturation are both orthogonal to hue, any linear combination of these two changes will leave the hue unchanged.

For example, consider the following CMY triplets:

- a) A yellowish color of 0,0,90 will have its signal applied solely in the yellow channel
- b) A reddish color of 0,90,90 will have the signal applied in the magenta and yellow channels
- c) A blue color of 90,90,0 will have the signal applied solely in the yellow channel
- d) A magenta color of 0, 90, 0 will have signal applied in both the cyan and yellow channels.
- e) A gray color of 50, 40, 40 will have the signal applied approximately equally in the cyan, magenta and yellow colors.

In each case the hue of the color, which is approximately the ratio of cyan to magenta to yellow, is preserved.

The first step to calculating the yellow content of a color is to move from a CMYK color system to simply a CMY color system by recombining black into the C, M, and Y components using the following equations:

$$C_1 = C_0 + K_0 - a * C_0 * K_0 \quad M_1 = M_0 + K_0 - a * M_0 * K_0 \quad Y_1 = Y_0 + K_0 - a * Y_0 * K_0$$

where

- C_1 , M_1 , and Y_1 are the equivalent cyan, magenta, and yellow ink percentages after black recombination
- C_0 , M_0 , and Y_0 are the original cyan, magenta and yellow values ink percentages
- K_0 is the original black ink percentage
- and a is a correction factor (.01) for ink additivity failure.

The yellow content can then be calculated using a color opponent calculation as in reference [6].

$$YBSat = (2 * Y_1 - 2 * C_1 - M_1) / 100.$$

YBSat is clipped to go between 0.0 and 1.0. When YBSat is 1.0 the pixel has high yellow content, and when it is 0.0 the pixel has low yellow content. Thus, when YBSat is high, we would like to scale to white, putting the watermark signal into the dominant colors. Likewise, when YBSat is low, we would like to scale to black, putting the watermark signal into the secondary colors. For values of YBSat between the 0 and 1 extremes we should place the watermark signal into some linear combination of scale to black and scale to white. This allows the algorithm to smoothly move between scale to black and scale to white depending on the pixel color.

$$Tweak = TweakToBlack * (1 - YBSat) + TweakToWhite * YBSat$$

2.4 Matched Detector

The watermark embedder algorithm developed in this paper modifies color pixel values to achieve a given luminance change. By choosing to modify luminance values, we better prepare the watermark to survive grayscale conversion. However, if the watermark detection algorithm has access to the color image data, it should be able to improve its reading ability by using knowledge of the embedding process.

Traditionally, when the watermark detector attempts to read a watermark, it estimates the luminance tweak

that was added to the regular image luminance. Essentially, this amounts to performing a dot product between the pixel's color value and the luminance unit vector. While this method will work for the system developed in this paper, we can increase the detected watermark strength by attempting to calculate the color tweak that was added to the image color. To accomplish this, the detector estimates each pixel's original color value. This estimation could be as simple as taking the pixel's original color value to be the same as the embedded color value. It then calculates the color tweak vector that the embedder would have used to tweak the luminance of this given color. By performing a dot product between the pixel's color value and the normalized color tweak vector, we should obtain a more accurate representation of the embedded tweak value. The following figure illustrates the potential gain in matching the detector to the embedder in two dimensions.

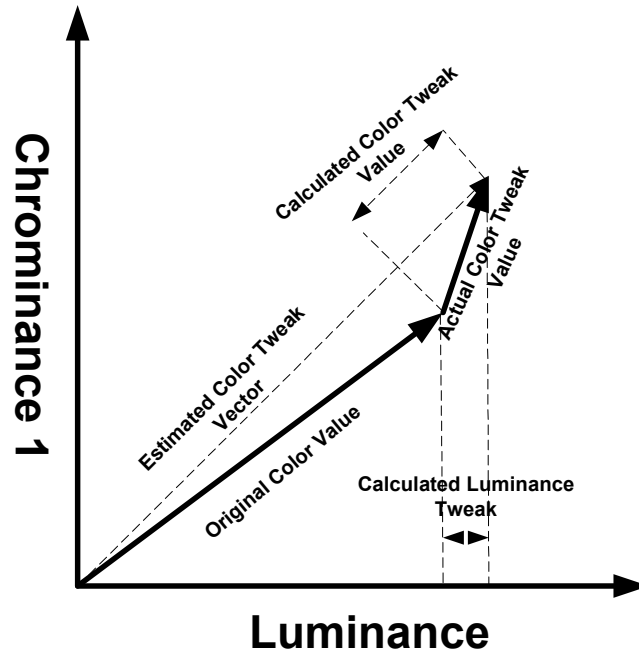


Figure 6. Matched Color Adaptive Detection

In practice, if the image supplied to the watermark detector has been through a print and image reacquisition process, the color measured by the detector is an approximation due to noise in the printer and image sensor. However, as long as the estimated color tweak vector is closer to being parallel with the actual color tweak vector than the luminance vector is, then the detector should realize some signal gain.

3. RESULTS

3.1 Embedder Results

For the same luminance change, the visibility changes in the two cases of treating the color as a vector between white and the color, or black and the color are shown in the Table below.

Color	Scale to Black Colors Modified/Visibility	Scale to White Colors Modified/Visibility
yellow	cyan/magenta - high	yellow - low
red	cyan - high	magenta/yellow - low
green	magenta - medium	cyan/yellow - medium
Blue	Yellow - low	Cyan/magenta - high
Cyan	Magenta/yellow - low	Cyan - high
Magenta	Cyan/yellow - medium	Magenta - medium

Table 1: Visibility of Same Luminance Changes

Please note that printed versions of the test images included in this paper show a more pronounced difference between the traditional scale to black embedding and the new adaptive color embedding. The six comparison images are on the following page.

Due to the lower sensitivity of the eye to high frequency changes in yellow, a lower visibility mark can be made with the same luminance change when the signal is inserted with the adaptive color algorithm, as shown in Image 1 and Image 2.

However, the luminance change seen by an ideal detector is approximately the same, as shown in Image 3 and Image 4 below. Because high spatial frequency yellow is harder to see than high spatial frequency black, the watermark has a lower visibility to the human visual system for the same luminance change.

Our standard watermarking algorithm, inserts the watermark signal by always scaling to black, as shown in Image 5. The adaptive color algorithm uses the input pixel color to determine the optimum direction in color space for inserting the watermark signal as shown in Image 6. Thus, while a similar magnitude watermark signal is inserted in Image 5 and Image 6, the watermark visibility is significantly reduced in Image 6.

In one usage scenario, watermarked images are printed on a press. These prints are then presented to a web camera, which attempts to detect the watermark. Detection rates decline for high ink percentages when this method is used. In practice this appears to be due to low blue sensitivity and auto-gain problems in web cameras (this problem is discussed in more detail in the following section). To compensate for these issues, we modify the algorithm at high ink percentages to include more scale to black tweak. This places more signal into the secondary colors, which are less affected by auto-gain saturation of dominant colors. While this makes the watermark a bit more visible, it is necessary to allow for acceptable detection rates. For other applications, such as scanners or purely digital images, auto-gain is not a problem, and our algorithm should work well unmodified.

3.2 Matched Detector Results

After implementing a version of our watermark detector that attempts to match the embedding process instead of simply looking for luminance tweaks, we found that the performance varied significantly depending upon the usage scenario. In the simplest case the watermarked image is kept in the digital domain, meaning the image is not printed. In this case we found that the matched detector performed more robustly than simply looking for luminance tweaks. Detection metrics indicating how strong of a watermark signal was detected were much higher, and in some cases images that would not have read normally were detected. However, this scenario represents the best possible conditions.

In more trying conditions, preliminary results were not as positive. In particular, we found that in a few tests where the watermarked image was printed on a press and then presented to a web camera for image reacquisition, the matched detector did not perform as well as the typical luminance tweak detector. This result was puzzling, as we expected at least equal performance. Upon closer examination of the camera system and the images it provides however, we soon saw a number of problems. First, a number of camera systems implement auto-white balance and auto-gain algorithms that directly conflict with our detection methodology. For image regions with a strong green component for instance, the camera delivers an image with green completely saturated. The detector sees that this is an area with a very strong green component and attempts to use that as the estimated color tweak vector. Since the color is saturated in this direction however, no watermark data is present along this vector. Essentially, the camera removed all watermark tweaks in the direction of this dominant color. The luminance tweak detector is still able to occasionally pull some of the watermark tweak out of the non-saturated color vectors, thus showing improved performance. Additionally, it appears that most camera manufacturers take advantage of the fact that the human visual system is less sensitive to contrast changes in the blue-yellow direction by compressing this channel more heavily when shipping data back to the computer. This compression further reduces the watermark tweaks that we place in the yellow channel, thus decreasing our ability to detect.



Image 1. Traditional Scale to Black Embedding
(Watermark accentuated for effect)

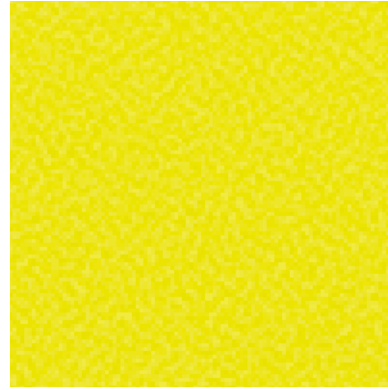


Image 2. Adaptive Color Embedding
(Watermark accentuated for effect)

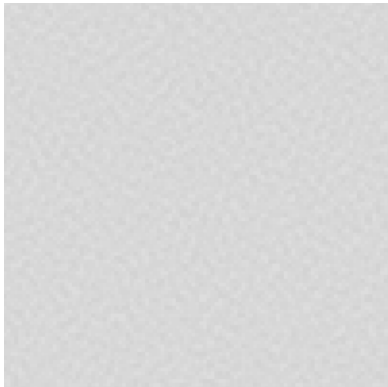


Image 3. Detector View of Scale to Black Watermark
(Watermark accentuated for effect)

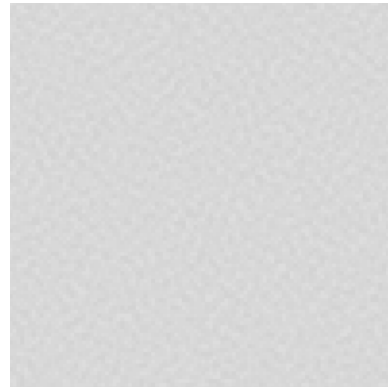


Image 4. Detector View of Adaptive Color Watermark
(Watermark accentuated for effect)



Image 5. Traditional Scale to Black Watermark
(Watermark accentuated for effect)



Image 6. Adaptive Color Watermark
(Watermark accentuated for effect)

It should be noted that we were only able to perform preliminary testing of the matched detector. Tests were performed on a limited test set and a limited number of usage scenarios. We hope to perform more in depth analysis in the near future.

4. CONCLUSION AND POSSIBLE EXTENSIONS

The algorithm developed in this paper successfully reduces watermark visibility in color images by adaptively choosing the optimum direction in color space to insert signal tweaks. For best results in practice, the transmission channel needs to be taken into account and the embedder algorithm may need modification for some usage scenarios. For example, the auto-gain problem in some web cameras requires the algorithm to be modified at high ink percentages.

Watermark detection can potentially be improved by matching the watermark reading process to the embedding process. Again, performance characteristics of the image acquisition tool, such as a web camera, need to be taken into account. The detector should be able to realize some improvement in watermark detection even with non-ideal color measurement.

In future work, we would like to further investigate matching the watermark detection algorithm to the adaptive color embedding algorithm. One method is to use standard tools such as ICC Color Profiles to allow the detector to more exactly match the embedder. The closer we are able to match the detector to the embedder, the better watermark reading performance we can expect.

ACKNOWLEDGMENTS

The authors would like to thank Tony Rodriguez, Steve Decker, Don Haaga, Lou Berkley, Eric Hudson, Matt Riley, Peter Harmon, Joel Meyer and Steve Stewart.

REFERENCES

1. Dean Delandreville, "In Your Face – Adobe vs. Macromedia," ZDNet News, <http://www.zdnet.com/devhead/stories/articles/0,4413,2624510,00.html>.
2. C.I.Podilchuk and W.Zeng, "Image-adaptive watermarking using visual models", *IEEE Journal of Selected Areas in Communications*, Vol 16, pp 525-539, May 1998
3. Brett T. Hannigan, Alastair Reed, Brett Bradley, "Digital watermarking using improved human visual system model," *SPIE Electronic Image 2001*, 2001.
4. R.W.G. Hunt, *The Reproduction of Colour*, pp. 69-70, Fountain Press, Tolworth, England, 1988.
5. Brian A. Wandell, *Foundations of Vision*, pp. 328, Sinauer Associates, Sunderland, Massachusetts, 1995.
6. Sebastiano Battiato, Dario Catalano, Giovanni Gallo, Rosario Gennaro, "Robust watermarking for images based on color manipulation," *Information Hiding Workshop 1999*, pp. 302-317, 1999.