Using perceptually uniform color spaces for image steganography

An enhancement of the Least Significant Bit method

Michael Specht
Department of Computer Science
Otto-von-Guericke-Universität Magdeburg, Germany

Abstract. *The steganographic capacity of a lossless compressed color image which can be gained by using the Least Significant Bit method can be further improved when performed in a perceptually uniform color space. Furthermore, providing a mechanism which allows steganographic capacity to be expressed by a real rather than an integral number of bits per pixel allows for potentially higher capacities because fine-tuning the parameters of the algorithm becomes possible. This paper shows how the transition of the LSB method from the RGB to the CIELab color space can be achieved and compares both methods.*

1 Introduction

If a secret, possibly encrypted message should be transmitted and the transmission itself should be kept secret, steganographic methods can be used to conceal the mere presence of the secret message. In order to hide a *payload*, a *carrier medium* is slightly altered in such a way that the difference is not perceivable but the payload is contained in the resulting *package* and can be completely restored. In this paper, the carrier and the resulting package are lossless compressed RGB color images and the payload is arbitrary digital data.

In image steganography, a common method to embed a payload into a carrier image is the Least Significant Bit method [3], in which a small number of least significant bits of each channel (typically red, green and blue with a size of 8 bits each) are replaced with the bits of the payload. The more bits are replaced, the higher the resulting steganographic capacity but at the same time, more noise is appearing in the resulting image, rendering it visually suspicious. Thus, the visible noise represents an upper limit to the steganographic capacity.

The non-linearity of the RGB color space raises the question if the steganographic capacity of a carrier image can be further increased by using a perceptually uniform color space. In this paper, a transition of the LSB method from the RGB to the CIELab color space is presented.

2 Method

During the encoding process of the LSB method, a number of least significant bits of each channel are replaced with the bits of the payload. The difference between the carrier and the package image cannot be recognized by the human visual system if the number of bits being replaced is very small – that is, if the noise introduced by the payload bits is not too strong.

In this paper, a color *c* is defined as a triplet of red, green and blue coefficients each ranging from 0 to 255, resulting in a total of $256^3 = 2^{24} = 16{,}777{,}216$ possible colors. The upper limit of the stegano—graphic capacity is 24 bits per pixel, which is reached when all bits of the pixel are replaced with payload bits, thereby completely destroying the package image.

The task at hand is to maximize the steganographic capacity while keeping the noise in the package image at an acceptable level which cannot be recognized by the human visual system.

2.1 Subdivision of the RGB color space

The method of replacing least significant bits corresponds to subdividing the RGB color space into a number of cuboid shaped color classes. A color class is a set of colors which can be exchanged freely because they cannot or can only hardly be distinguished from each other.

The parameters of this subdivision are the number of subdivisions applied to each color channel n_{red}, n_{green} and n_{blue}. The resulting number of color classes is $n_{classes} = n_{red} \cdot n_{green} \cdot n_{blue}$. If one bit of each channel is to be overwritten with payload bits, then seven original bits are retained in each channel, so $n_{red} = 128$, $n_{green} = 128$, $n_{blue} = 128$, and $n_{classes} = 128^3 = 2{,}097{,}152$. The number of possible colors in the carrier image is reduced from about 16.7 million to about 2 million and the resulting redundancy can be used for embedding the payload at three bits per pixel. Fig. 1 shows an example of a uniform RGB color space subdivision.

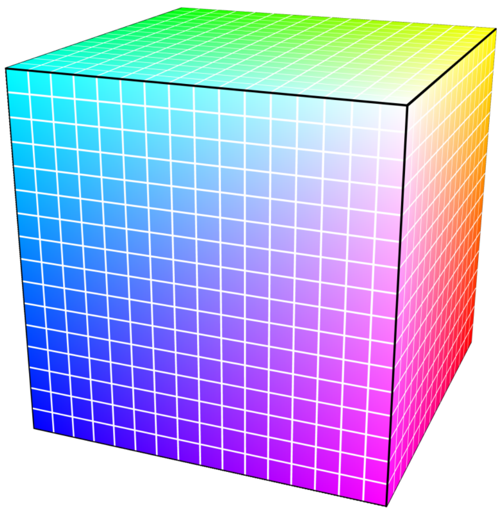

Fig. 1: A uniform 16x16x16 subdivision of the RGB color space, resulting in a total of $16^3 = 4{,}096$ color classes.

As a result of the classification process the following functions are defined:

- **class**(c), which returns a class number ranging from 0 to $n_{classes}$ - 1 for any given color c.
- **size**(**class**(c)), which returns the number of colors in a given class.
- **index**(c), which returns a number in the range from 0 to **size**(**class**(c)) - 1 and defines a unique index for each color c inside its color class.

The steganographic capacity of a pixel with the color c is equal to $\log_2$(**size**(**class**(c))), the steganographic capacity of a carrier image is the sum of the steganographic capacities of all its pixels. In the case of the LSB method, **size**(**class**(c)) is constant for every color c and equal to $2^{24} / n_{classes}$. Because $n_{classes}$ is a power of two, **size**(**class**(c)) is also a power of two for every color c, resulting in an integral steganographic capacity for each color.

2.2 Improving on the LSB method

Because the RGB color space is perceptually highly non-linear, steganographic capacity is wasted. The central idea presented in this paper is to perform the subdivision in a perceptually uniform color space to accommodate for the non-linearity inherent in the RGB color space and thus maximize the steganographic capacity of the carrier image.

Another factor which determines the resulting steganographic capacity is the choice of the parameters of the subdivision. In the method presented here, these parameters do not necessarily have to be powers of two but can be chosen freely, making the adjustment of the resulting steganographic capacity much more flexible.

The third issue contributing to the method presented in this paper is that the sensitivity of the human visual system is different for changes of luminance and for changes of chrominance. In the RGB color space, each channel represents a combination of luminance and chrominance, and the differences in sensitivity cannot be exploited for achieving higher steganographic capacities. In the CIELab color space, the L channel represents luminance, while the a and b channels both represent chrominance, making it possible to treat these two color properties differently and optimize the steganographic capacity of the carrier image.

2.3 Subdivision of the CIELab color space

From the total set of 2^{24} RGB colors, every possible color c is transformed into the CIELab color space[1] and then classified using a uniform subdivision. The parameters of this classification are the number of subdivisions of each channel n_L, n_a and n_b. The resulting number of classes is $n_{classes} = n_L \cdot n_a \cdot n_b$.

1 The color space transformation contains a gamma correction step for which a fixed gamma value has to be chosen because the conditions under which the package image will be displayed are unknown. In this paper, a gamma value of 2.2 is assumed which, according to Poynton, is suitable for displaying images on a monitor in a dimly lit environment [4]. Full details of the transformation are described in [5].

Fig. 2 shows a CIELab color space subdivision with $n_L = n_a = n_b = 16$. Some of the classes remain empty after the classification process due to the shape of the CIELab color gamut and the fact that unlike the RGB color gamut, it does not completely fill out the standard cube. Also, **size**(**class**(c)) is not constant and not equal to a power of two for every c, which in turn means that the capacity of a carrier image must be expressed by a real number of bits per pixel as opposed to an integral number in the case of the LSB method. This also means that the payload can not be embedded by simply replacing bits.

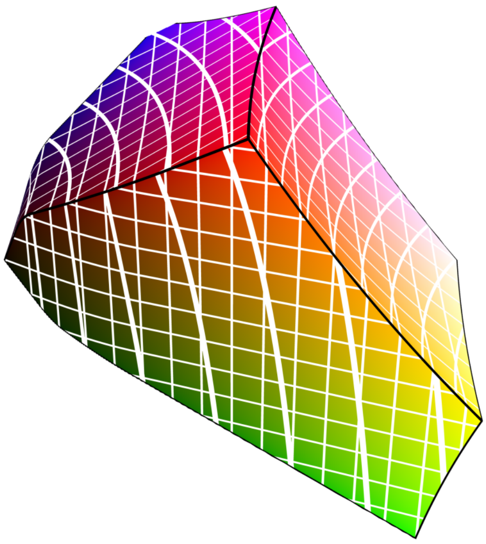

Fig. 2: A uniform subdivision of the CIELab color space. The color space has been transformed to fit tight into the standard cube for illustrative purposes.

The fact that **size**(**class**(c)) is not constant implies that the steganographic capacity of a pixel depends on the color it contains, which in turn means that the steganographic capacity of a carrier image depends on the colors used in the image. Some parts of the RGB color space such as light blue get heavily compressed during the color space transformation while the spatial expansion of other parts is retained. Colors which are located in those parts that are subject to high compression provide a higher capacity than other colors.

2.4 The transcoding process

Since the payload cannot be embedded into the carrier image by replacing certain bits, it has to be transcoded from one digital representation into another. The payload is regarded as an unsigned integer P which is transcoded into a numeral C with a variable radix during the encoding step. The target numeral C consists of n places (or pixels) p_0 to p_{n-1} that can be assigned any of **size**(**class**(c_i)) values each, where c_i is the color represented by the pixel p_i and n is equal to the number of pixels in the carrier image.

During the encoding process, P is repeatedly divided by **size**(**class**(c_i)) of the next carrier pixel's color c_i. The modulo of this division represents the next digit in C and is used to look up the resulting color in the package image with the help of the **index** and **class** functions. The process stops when P is zero.

```
i = 0;
while (P > 0) {
    s = size(class(c_i));
    p_i = c with index(c) = P mod s and
                 class(c) = class(c_i);
    P = P / s;
    i = i + 1;
}
```

In the decoding step, P is initialized with 0. The package pixels are traversed in reverse order and for every pixel p_i, P is multiplied with **size**(**class**(p_i)) and then **index**(p_i) is added.

```
P = 0;
for each p_i from last to first {
    P = P · size(class(p_i));
    P = P + index(p_i);
}
```

In practice, the calculations can be performed using the GNU Multiple Precision Library [1]. Because processing big numbers is slow, P can be divided into kilobyte-sized blocks before the encoding step. The resulting waste of capacity is minimal and overcompensated by the benefit of higher computational speed.

3 Results

Comparing the LSB method to the method presented in this paper is a difficult task, because in order to achieve equal steganographic capacities for both methods, the parameters of the CIELab subdivision must be carefully adjusted until the visual quality of the package image is sufficient and the resulting steganographic capacity is equal to the one achieved by the LSB method.

Fig. 4: A 32x32x32 subdivision of the RGB color space, resulting in a capacity of 9 bits per pixel. A banding effect is clearly visible in the background.

Fig. 5: A 74x28x26 subdivision of the CIELab color space, resulting in the same capacity of 9 bits per pixel. The banding effect is minimized due to the fact that more detail is preserved in the luminance channel than in the chrominance channels.

3.1 Minimum steganographic capacity

Assuming that the visual quality of a 24 bit-CIELab image with 8 bits per channel is equal to that of a 24 bit-RGB image, which means that no color is changed perceptually, the method presented in this paper outperforms the LSB method due to the shape of the CIELab color gamut. Because the gamut does not completely fill out the standard cube, some RGB colors are mapped to the same CIELab color which results in small steganographic capacities for most colors.

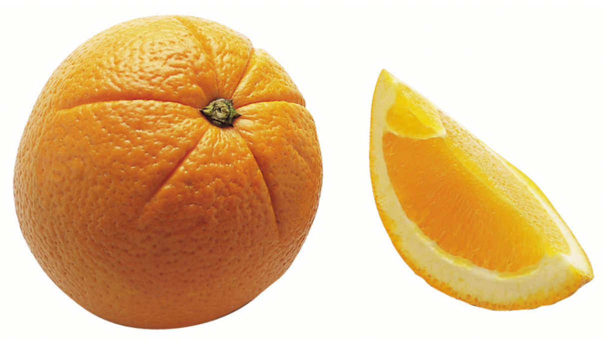

Fig. 3: A 256x256x256 subdivision of the CIELab color space, resulting in a steganographic capacity of 1.91 bits per pixel in the case of the orange image [6].

In Fig. 3, a 256x256x256 subdivision of the CIELab color space was used, resulting in a steganographic capacity of 1.91 bits per pixel. This capacity can be regarded as the *minimum steganographic capacity* which can be achieved without changing the visual appearance of the image. Note that this minimum capacity is not available in the RGB color space, where a subdivision of 256x256x256 results in a steganographic capacity of zero bits per pixel.

3.2 Capacity of 9 bits per pixel

The fact that the human visual system is more sensitive to changes in luminance than to changes in chrominance is exploited in Figs. 4 and 5 [2], which both have a steganographic capacity of 9 bits per pixel. The LSB method produces banding artifacts in the smooth background gradient, whereas the method presented in this paper performs better since more detail is preserved in the luminance channel and less detail in the chrominance channels.

4 Discussion and further work

The results presented in this paper show that applying a subdivision to a perceptually linear color space outperforms the LSB method in which the subdivision is applied to the RGB color space. The benefits are:

1. The steganographic capacity can be fine tuned because the parameters of the subdivision are not restricted to powers of two.
2. The steganographic capacity depends on the carrier image because colors are classified according to how they are perceived. The fact that some images are better suited for steganography than others is taken advantage of.
3. Differences in perception of luminance and chrominance can also be exploited.

However, the process of encoding and decoding a payload is much slower compared to the speed of the LSB method, because the color space conversion and the handling of big numbers during the transcoding process are very time-consuming steps.

The following issues should be further explored:

1. Higher steganographic capacities could be achieved by applying a hexagonal subdivision to the chrominance channels a and b, because the shape of the hexagon is the most circle-like of all gapless planar subdivisions, providing potentially higher steganographic capacities.
2. A mechanism which finds an optimal subdivision for any given carrier image would be helpful for creating package images with a capacity higher than the minimum steganographic capacity achieved by using a 256x256x256 subdivision.
3. The method presented in this paper should be examined for detectability by steganalysis methods.

5 Acknowledgments

The author would like to thank Andreas Engel and Henry Sonnet for their kind assistance in writing this paper.

---

6 References

[1] *Free Software Foundation*: **The GNU MP Bignum Library**, http://www.swox.com/gmp/.

[2] *Frey, Nik*: **Flaming cocktails**, http://commons.wikimedia.org/wiki/Image:Flaming_cocktails.jpg.

[3] *Katzenbeisser, Stefan and Petitcolas, Fabien*: **Information Hiding Techniques for Steganography and Digital Watermarking**. Artech House Publishers 2000.

[4] *Poynton, Charles*: **A Technical Introduction to Digital Video**. John Wiley & Sons 1996.

[5] *Poynton, Charles*: **Digital Video and HDTV Algorithms and Interfaces**. Morgan Kaufmann 2003.

[6] *United States Department of Health and Human Services*: **Citrus sinensis**, http://commons.wikimedia.org/wiki/Image:Citrus_sinensis.jpg.