High Information Rate and Efficient Color Barcode Decoding

Homayoun Bagherinia and Roberto Manduchi

University of California, Santa Cruz, Santa Cruz, CA 95064, USA
{hbagheri,manduchi}@soe.ucsc.edu
http://www.ucsc.edu

Abstract. The necessity of increasing information density in a given space motivates the use of more colors in color barcodes. A popular system, Microsofts HCCB technology, uses four or eight colors per patch. This system displays a color palette of four or eight colors in the color barcode to solve the problem with the dependency of the surface color on the illuminant spectrum, viewing parameters, and other sources. Since the displayed colors cannot be used to encode information, this solution comes at the cost of reduced information rate. In this contribution, we introduce a new approach to color barcode decoding that uses 24 colors per patch and requires a small number of reference colors to display in a barcode. Our algorithm builds groups of colors from each color patch and a small number of reference color patches, and models their evolution due to changing illuminant using a linear subspace. Therefore, each group of colors is represented by one such subspace. Our experimental results show that our barcode decoding algorithm achieves higher information rate with a very low probability of decoding error compared to systems that do display a color palette. The computational complexity of our algorithm is relatively low due to searching for the nearest subspace among 24 subspaces only.

1 Introduction

Color barcodes are becoming one of the technologies to embed information in recent years. The information density of conventional 1-D and 2-D barcodes is much lower than color barcodes in a limited area. The use of colors creates more symbols that lead to larger data capacity within the same symbol size.

One of the popular color barcodes in the market is HCCB (Microsoft's High Capacity Color Barcode) [8]. HCCB uses a grid of colored triangles with four or eight colors to encode data. The necessity of increasing information density in a given space encourages the idea of using more colors in color barcodes. For instance, using 24 colors instead of 8 would increase the information rate by 1.5 times in the same barcode area.

Decoding large number of colors is a challenging task. One of the main problems is that an observed color patch depends on the surface reflectance and unknown illuminant spectrums. One way to overcome this problem is to apply a pre-processing step such as color constancy operation to compensate with varying and unknown illuminants ([3], [5], [6], [7]). Another strategy is the clustering of a number of colors in color space [10]. Modeling the changes of the observed color due to changing illuminant could also solve this problem [1].

In this contribution, our focus is a solution for decoding the color information in a barcode. We consider color barcodes that can be decoded under multiple illuminants with small number of reference patches of known colors, seen under the same illuminant as the color to be decoded. The use of reference color patches enables robust decoding with relatively low computational complexity which is suitable for implementation on a low powered mobile device. Our algorithm considers each color patch and a set of reference color patches as a group of color patches, and models their evolution due to changing illuminant using a linear subspace. Therefore, each group of colors (one color and one or more reference patches) is represented by one such subspace. When a color barcode is observed, our algorithm decodes each color independently by building a group from the color and a set of reference color patches. Then, each group is decoded by finding the nearest subspace among all subspaces. Our experiments show that our approach produces higher information rate than existing technology such as HCCB and the works described in [1] and [8] with a very low probability of incorrect decoding using small number of reference color patches.

2 Related Work

There are limited numbers of research work in the literature on the topic of color barcode decoding. The common approach to color barcode decoding (e.g. [8], [10]) assumes that all N colors that are used to generate a color barcode are available in the barcode itself. HCCB technology also assumes that all colors in the palette are available. This makes the decoding task simpler by comparing the color of the patches in the barcode against the colors of N patches. However, since the N color patches are not information-carrying patches, the information rate will be reduced.

The method of Sali and Lax [10] uses a k-means classifier to assign the (R,G,B) value of a color patch to one reference color. Similarly, the HCCB detection algorithm [8] identifies a set of clusters in color space using mean shift clustering, and assigns each cluster center to one of the reference colors in a palette, contained in the barcode itself. These strategies usually work well for the limited number of colors in a barcode. Color clustering is not guaranteed to work well when many more colors are used. Moreover, these strategies only work for dense barcodes, with the number of patches considerably larger than the number of colors N, so that the attachment of the color palette to the barcode has insignificant impact on the spatial density of information.

Wang and Manduchi [12] studied the problem of information embedding via printed color. Their algorithms used one or more reference color patches of known colors, observed under the same unknown illuminant as the colors to be decoded. Known reference color patches enable an estimate of illuminant as a parametric color transformation between a canonical illuminant and the unknown illuminant. This transformation is used to render the unknown color patch under the canonical illuminant, which can be decoded using the training data.

Recent work by Bagherinia and Manduchi [1] showed a study of the variation of small groups of printed colors and modeled their evolution due to changing illuminant using a low-dimensional linear subspace. Their approach decodes length k barcode elements of a color barcode, and does not use reference colors. They showed that by carefully selecting a subset of barcode elements, it is possible to achieve good information rate at low decoding error probability. Since their system uses a subset of barcode elements and does not allow the repeat of colors within a barcode element, maximum information rate for N colors cannot be achieved. The number of subspaces also increases with the length k of barcode elements which leads to high computational complexity for large barcodes. Thus, this approach is not suitable for low powered mobile device implementation.

This contribution builds on the work by Bagherinia and Manduchi [1]. However, rather than considering groups of k color patches, we concentrate on the variation of one color patch and r reference patches as a group of color patches to model their evolution due to changing illuminant.

3 Information Rate and Probability of Decoding Error

The patches in a color barcode are created from a set $\mathcal{C}_N \subset \mathcal{C}$ of N color patches and $\mathcal{C}_M \subset \mathcal{C}$ of M reference color patches. A color barcode can be defined as the juxtaposition of n color bars for information encoding and r reference colors in any spatial pattern, resulting in a $K = n + r$ length barcode.

As defined in [1], the information rate R of a barcode (reference colors are not used) is defined by the logarithm base 2 of the number of different symbol that can be represented by the barcode and measured in bits per bar. The information rate for a barcode of K bars and r reference patches is defined as:

$$R(K, r) = \left(1 - \frac{r}{K}\right) \log_2 N \quad (1)$$

An approach to color barcode decoding assumes that r reference patches in the barcode are attached to the barcode. This may allow for more robust decoding. However, this solution comes at the cost of reduced information rate, since r reference patches cannot be used to encode information.

Fig. 1 illustrates the information rate for $N = 24$, $K = 120$ bars and varying r from 0 to 24. For example, the information rate of a length 120 barcode that displays all $r = 24$ reference colors is 3.67 bits/bar. The information rate using $r = 2$ reference colors is 4.51 bits/bar. A color barcode system that uses two reference colors can pack about 0.84 additional bits per bar (or 100 bits overall) than systems that use 24 reference colors ($r = N$). Thus, it is highly beneficial to use small number of reference colors.

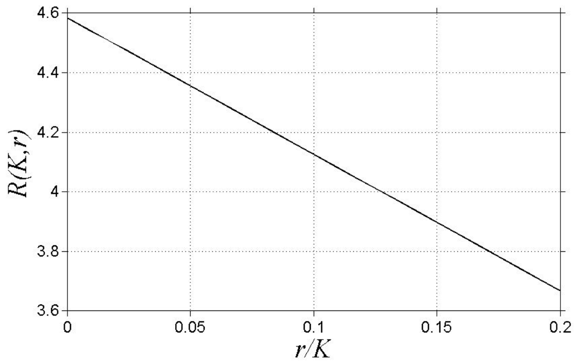

Fig. 1. The information rate for $N = 24$, a length $K = 120$ barcode, and varying r from 0 to 24 reference colors.

The probability of decoding error $P(r)$ is important to consider for a generic barcode with N colors and r reference colors. This is defined as the average probability of decoding error of a color patch over all colors. As described in [1], the probability of decoding error for a barcode of length K, assuming that decoding errors for the individual colors in the barcode are statistically independent events, is:

$$P_E(K, r) = 1 - (1 - P_E(r))^K \tag{2}$$

The probability of incorrect decoding grows with the number of bars K in the barcode. The dependence $P_E(K, r)$ on r can be determined experimentally as described in Sec. 5.

4 Color Barcode Decoding

4.1 Decoding Approach

Let the $3(1 + r)$-dimensional vector $\mathbf{e} = [c, d_1, \ldots, d_r]^T$ be the measured color of a surface for information decoding and r reference surfaces (a reference color group of r colors), where $c = [c^R, c^G, c^B]^T$ and $d_j = [d_j^R, d_j^G, d_j^B]^T$ $(j = 1 \ldots r)$ are (R, G, B) color vectors. A model to describe the observed color of Lambertian surfaces assumes that the spectra of the surface reflectances and of the illuminants live in finite-dimensional spaces of dimension N_{ref} and N_{ill} respectively [7]. Hence, the observed group of colors under a given illuminant is equal to

$$\mathbf{e} = \mathbf{\Phi} v \tag{3}$$

$\mathbf{\Phi}$ is a $3(1 + r) \times N_{\text{ill}}$ full-rank matrix whose entries are a function of the illumination and reflectance basis vectors as well as of the spectral sensitivities of the camera assuming Lambertian characteristics of surfaces. $v = [v_1, v_2, \ldots, v_{\text{N}_{\text{ill}}}]^T$

is a vector of length N_{ill} containing the coefficients of the illuminant with respect to the chosen basis.

The observed color c of a single surface under a given illuminant is equal to $c = \mathbf{\Phi_c} v$. The decoding of c would be very difficult if no reference color is used. This is due to the fact that $N_{\text{ill}} \geq 3$ in general ([4], [9]), and the rank of $\mathbf{\Phi_c}$ would be 3 which makes $rank(\mathbf{\Phi_c})$ smaller than N_{ill}. If, however, a color along with multiple reference colors seen under the same illuminant is decoded at once, the probability of correct decoding will be higher due to $\mathbf{e} \in \mathbb{R}^{3(1+r)}$ and $rank(\mathbf{\Phi}) = min(3(1+r), N_{\text{ill}})$. The vector $\mathbf{e}$ is constrained to live in a subspace S of dimension of at most N_{ill}.

Let S be the linear subspaces in $\mathbb{R}^D$ spanned by $\mathbf{e}^i$ over varying illuminant i. The decoding algorithm for an individual color represented by $\mathbf{e}$ begins by modeling the subspaces S_k of N colors ($k = 1 \ldots N$). Similar to the work described in [1], we have considered two approaches to build these subspaces. In the first case, the subspaces of suitable dimension D are built from observations under verity of illuminants via Principal Component Analysis (PCA). The second approach is based on the diagonal model of color changes, which assumes that each color channel changes as a result of an illuminant change by a multiplicative factor. The matrix $\mathbf{\Phi}$ is formed based on a single color and r reference color patches under a single illuminant as follows

$$\mathbf{\Phi}^T = \begin{bmatrix} c^R & 0 & 0 & d_1^R & 0 & 0 & d_2^R & \ldots \\ 0 & c^G & 0 & 0 & d_1^G & 0 & 0 & \ldots \\ 0 & 0 & c^B & 0 & 0 & d_1^B & 0 & \ldots \end{bmatrix} \tag{4}$$

The subspace modeling based on diagonal model is less accurate. However, it allows a fast calibration procedure (since it requires a single image of N colors and r reference colors under a single illuminant) and is suitable for the situation when a different camera is used or the colors are printed with a different printer.

Once N subspaces are built, we decode an observed color vector $\mathbf{e}$ by assigning it to the subspace that minimizes the distance to $\mathbf{e}$. The distance is defined as the distance between $\mathbf{e}$ and its projection onto S_k.

4.2 Reference Color and Subspace Selection

To produce a certain $P_E(r)$ smaller than desired, one needs to define reference colors within a set of colors. Selection of r number of reference colors that minimizes the associated $P_E(r)$ is computationally very expensive. In particular, the probability of incorrect decoding for all combinations of r reference colors from a large set of colors needs to be evaluated. We have considered a suboptimal greedy technique to reduce the complexity associated with r reference colors selection. Our approach is to insert one reference color at a time to compute the $P_E(r)$ as follows: we select a reference color from the set $\mathcal{R}_1 \subset \mathcal{C}_M$ that minimizes the $P_E(r = 1)$. Then we build a set $\mathcal{R}_2$ of groups of two reference colors of which contain the selected reference color in previous step. Then, we search for the group (of reference colors) that minimizes the $P_E(r = 2)$. We continue this procedure until r reference colors satisfy the desired $P_E(r)$. Basically, for $r > 1$ we build all possible combinations ($\mathcal{R}_r$) of r length groups of reference colors of which contain the reference colors selected in the previous steps.

We considered all possible reference colors that could build sets of reference color groups $\mathcal{R}_r$. We then estimated (via cross-validation over multiple illuminants) the probability that the color i (represented by the vector $\mathbf{e}$) is incorrectly decoded as j. The sum of all these probabilities, divided by the number of colors, gives the probability of incorrect decoding $P_E(r)$ for a given reference color group of length r. For PCA-based approach, we repeated this procedure for subspace dimensions from one to five. For each r length reference colors, we selected the subspace dimension D that minimizes $P_E(r)$ for chosen r reference colors.

Fig. 2 shows the probability of incorrect decoding $P_E(r)$ for the number of reference colors r between one and five and the subspace dimension between one and five. We select the subspace dimensions 3 and 4 for $r = 1, 2$ and $r = 3, 4, 5$ respectively.

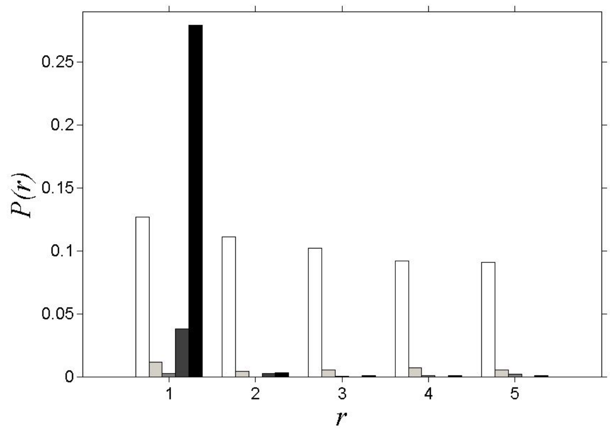

Fig. 2. The probability $P_E(r)$ of incorrect decoding as a function of the number of reference colors r and subspace dimension from 1 (white) to 5 (black).

5 Experiments

For our experiments, we printed a colorchecker with $5 \times 25 = 125$ colors on paper by a regular printer, uniformly sampled in (R, G, B) color space. Images were taken of the printed colors with a Canon EOS 350D camera in raw (CR2) format from zero degree (with respect to the normal to the colorcheckers surface) under 64 different lighting conditions including indoor and outdoor under direct sunlight, diffuse skylight with overcast sky, cloudy sky, or under cast shadow, and various types of artificial light. The color value of each patch was calculated by averaging over a few hundreds color values within the patch to reduce noise.

We then selected $N = 24$ color patches using images under all 64 illuminants. For each illuminant, we used k-means to select 24 colors. Commonly used initialization methods for k-means algorithm is Random Partition. To make sure that the k-means clustering selects 24 distinct colors, we run k-means 10 times with different starting point to select 24 colors with highest occurrences in all runs. Once we have 24 colors for each illuminant, we selected 24 colors with highest occurrences among all illuminants. The color of reference patches were selected from $M = 125 - 24$ colors as described in Sec. 4.2. The colors for reference colors are different than the $N = 24$ colors since it is desired to build each vector **e** with distinctive colors rather than allowing color repetition in the vector **e**. Synthetic images of all 24 colors and $r = 1 \ldots 5$ reference color patches

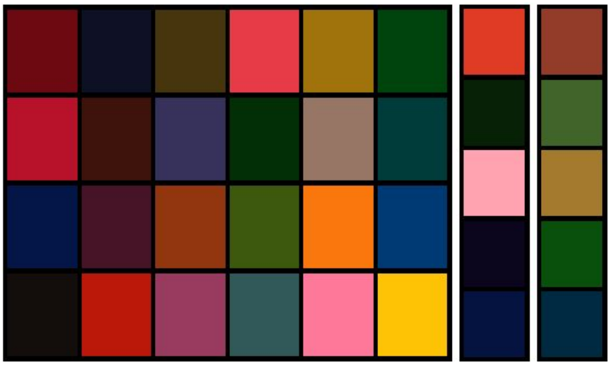

Fig. 3. The 24 color patches, 5 reference color patches selected for PCA-based and diagonal model.

were built from the average color values of the images of the color patches seen under the 64 representative illuminants. For PCA-based approach, we compute the probability of incorrect decoding $P_E(r)$ for number of reference colors r ranging between one and five as follows. We ran ten rounds of cross-validation, each time randomly selecting 32 illuminants, learning the subspaces for each vector **e** representing a color and r reference patches considered based on its images under these illuminants, and testing each vector **e** in turn on 32 of the remaining illuminants. We count the number of times any color was incorrectly decoded, and divided the result by the number of test illuminants (32), by the number of cross-validation rounds (10) and by the number of colors (24). We also tested the decoding algorithm based on the diagonal model discussed in Sec. 4.1 In this case, the color subspaces were built from observation of the colors under just one illuminant. We ran ten rounds of cross-validation, each time randomly selecting one illuminant (without repetition), training our model on such illuminant and testing it with vectors **e** seen under 32 illuminants randomly chosen.

We expanded our experiments to more realistic illumination situations. We showed the robustness of our algorithm when a color barcode is observed from different viewing angles under different illuminants. Typically, the colored surfaces produced by a color printer are far from being Lambertian. They may have a strong specular refection component. In the presence of specular refection, a viewpoint-dependent component with the same color as the illuminant adds to the perceived color of the surface from diffuse refection [11]. We took 32 test images from our printed colorchecker under four illuminants from multiple viewpoints ranging from -50 to 50 degrees with respect to the normal to the colorcheckers surface and from a constant distance about 1.5 to 2 meter. We extracted eight images taken under the same illuminant from these 32 test images. Then, similar to the scatter-plot in [2], we showed the scatter-plot of the normalized values $R/(R+G+B)$ vs. $G/(R+G+B)$ of the color of $N = 24$ patches. Under an ideal Lambertian model, the normalized color for a color patch should be constant regardless of the viewpoint under the same illuminant. The scatter-plot in Fig. 4 suggests that normalized colors from most of color patches are scattered. The scattering within each color cluster may be due to the specular component as a portion of illuminant color is captured along with the surface color. In particular, a subset of point clusters are oriented towards the white surface point (the white cross in the scatter-plot), whose color is similar to the color of the illuminant. We also compute the probability of incorrect decoding $P_E(r)$

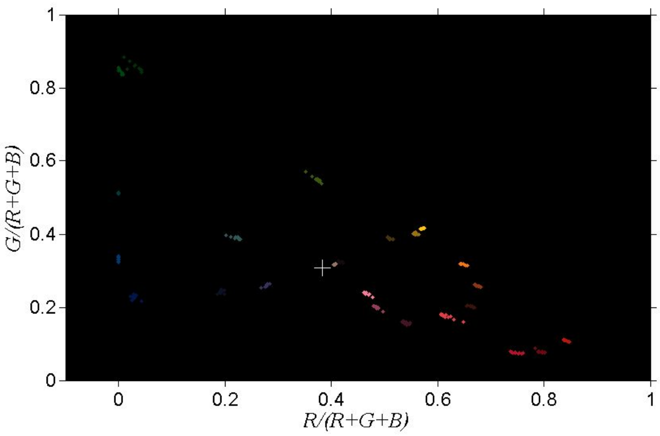

Fig. 4. The scatter-plot of the normalized color values of $N = 24$.

for number of reference colors r by running ten rounds of cross-validation, each time randomly selecting 32 illuminants (from 64 training illuminants), learning the subspaces, and testing on the data of these 32 images.

Fig. 5 shows the probability of decoding error $P_E(r)$ and $P_E(K, r)$ (for a $K = 120$ length barcode, computed using Eq. (2)) for number of reference colors

r using $N = 24$ colors. We tested the decoding method on the data with zero viewing angle and the data with different viewing points ranging from -50 to 50 degrees. Both types of subspace modeling (via PCA or via the diagonal model) are considered.

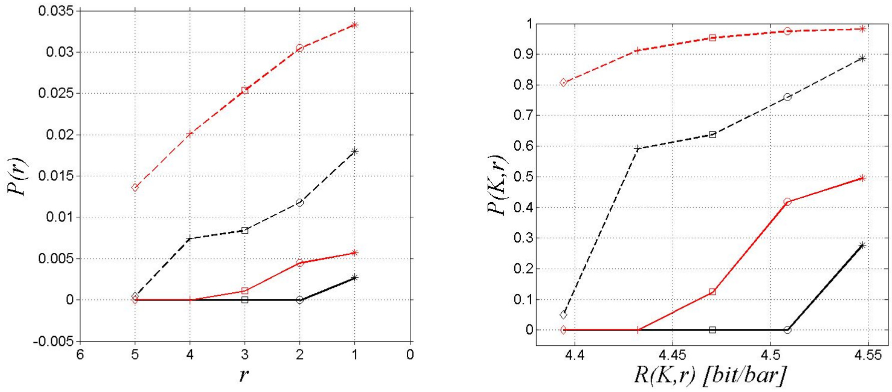

Fig. 5. Left: the probability of incorrect decoding $P_E(r)$ to decode a single color patch as a function of the number of reference colors r. Right: the probability incorrect detection $P_E(K, r)$ for a length $K = 120$ barcode versus the information rate compute using Eq. (1) and (2). '$*$': $r = 1$; '$\circ$': $r = 2$; '$\square$': $r = 3$; '$+$': $r = 4$; '$\diamond$': $r = 5$. Subspaces learnt via PCA (solid line) over 32 illuminants, and diagonal model (dashed line). Black line: the algorithm tested on data with zero viewing angle. Red line: the algorithm tested on data with different viewing points ranging from -50 to 50 degrees.

These results suggest that it is possible to reach high information rate with a very low probability of decoding error. For example, PCA-based subspace modeling for $r >= 2$, $N = 24$ colors, and a length $K = 120$ barcode results in a probability of zero incorrect decoding of any length of barcode, while allowing one to encode information at a rate of about 4.51 bits per bar if only two reference colors are used. In presence of specular reflection, the information rate is 4.43 bits per bar with a probability of zero incorrect decoding if four reference colors are used. Using the diagonal model to build the subspaces leads to less accurate decoding. The probability of decoding error using five reference colors is equal 0.048 at 4.39 bits per bar. The probability of decoding error for smaller barcode such as $K = 60$ bars ($N = 24$ and $r = 5$) reduces to 0.02 with an information rate of 4.2 bits per bar using Eq. (2). The subspace modeling based on diagonal model does not produce promising results in presence of specular reflection.

Decoding each color patch requires finding the nearest subspace in a database of only 24 elements regardless of the number of reference colors chosen. This makes this algorithm suitable for implementation on a low powered mobile device.

6 Conclusions

We introduced a new color barcode decoding approach that requires few reference colors attached to the color barcode. Our experiments have shown that, by carefully selecting a set of reference colors, it is possible to achieve a high information rate at the low probability of decoding error with relatively low computational complexity which is suitable for implementation on low powered mobile device. In the future, we hope to extend our decoding strategies in more realistic situations such as errors due to color mixing from two neighboring patches, to printed color drift and fading, or to color barcodes printed from different printers.

7 Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No. IIS - 0835645.

References

1. Bagherinia H, Manduchi R. 2011. A theory of color barcodes. IEEE Color and Photometry in Computer Vision Workshop (CPCV 2011)
2. Bagherinia H, Manduchi R. 2011. Robust real-time detection of multi-color markers on a cell phone. Journal of Real-Time Image Processing
3. D. H. Brainard and W. T. Freeman. Bayesian color constancy. Journal of the Optical Society of America A, 14:13931411, 1997
4. E.R. Dixon, Spectral distribution of Australian daylight, Journal of the Optical Society of America, 68, pp. 437-450, (1978)
5. G. D. Finlayson, S. D. Hordley, and P. M. Hubel. Color by correlation: A simple, unifying framework for color constancy. IEEE Trans. Pattern Anal. Mach. Intell., 23:12091221, November 2001
6. D. A. Forsyth. A novel algorithm for color constancy. Int. J. Comput. Vision, 5:536, September 1990
7. L. T. Maloney and B. A.Wandell. Color constancy: a method for recovering surface spectral reflectance. In M. A. Fischler and O. Firschein, editors, Readings in computer vision: issues, problems, principles, and paradigms, pages 293297. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1987
8. D. Parikh and G. Jancke. Localization and segmentation of a 2D high capacity color barcode. In Proceedings of the 2008, IEEE Workshop on Applications of Computer Vision (WACV08:), pages 16, Washington, DC, USA, 2008. IEEE Computer Society
9. J. P. S. Parkkinen and J. Hallikanen and T. Jaaskelainen, Characteristic spectra of Munsell Colors, Journal of the Optical Society of America A, 6, pp. 318-322, (1989)
10. E. Sali and D. Lax. Color bar code system. U.S. Patent 7210631, 2006
11. Shafer SA (1985) Using color to separate reflection components. Color Research and Application 10:210-218
12. F. Wang and R. Manduchi. Color-constant information embedding. In Proc. IEEE Workshop on Color and Reflectance in Imaging and Computer Vision, 2010