A Theory of Color Barcodes

Homayoun Bagherinia
Department of Electrical Engineering
University of California, Santa Cruz
hbagheri@soe.ucsc.edu

Roberto Manduchi
Department of Computer Engineering
University of California, Santa Cruz
manduchi@soe.ucsc.edu

Abstract

There is increased interest in the use of color barcodes to encode more information per area unit than regular, black-and-white barcodes. For example, Microsoft's HCCB technology uses 4 or 8 colors per patch. Unfortunately, the observed color of a surface depends as much on the illuminant spectrum (and other viewing parameters) as on the surface reflectivity, which complicates the task of decoding the content of the barcode. A popular solution is to append to the barcode a "palette" with the reference colors. In this paper, we propose a new approach to color barcode decoding, one that does not require a reference color palette. Our algorithm decodes groups of color bars at once, exploiting the fact that joint color changes can be represented by a low-dimensional space. Decoding a group of bars (a "barcode element") is thus equivalent to searching for the nearest subspace in a dataset. We also propose algorithms to select subsets of barcode elements that can be decoded with low error probability. Our experimental results show that our barcode decoding algorithm enables substantial information rate increase with respect to system that display a color palette, at a very low decoding error rate.

1. Introduction

1-D and 2-D barcodes have become tremendously popular as a means for information embedding. The goal of a barcode is to encode information with high spatial density while ensuring robust reading by an optical system. Typical barcode technology use dark ink on a light-colored surface (or vice-versa); the resulting spatial pattern encodes the information.

In order to increase the density of information, different ink colors could be used. For example, Microsoft's High Capacity Color Barcode (HCCB) technology [12] uses 2-D barcodes enhanced with four or eight different colors. This technology, marketed as "Microsoft Tag" (tag.microsoft.com), is becoming increasingly widespread in applications such as retail, publishing, and transit. The density of information (in bits per area unit) is proportional to $\log_2 N$, where N is the number of colors used. Thus, it would be desirable to use a large number of colors to embed more information in the same barcode. For example, using 16 colors instead of 4 would allow one to encode twice as many bits in the same area.

The main problem with using many colors in a barcode is that the observed surface color depends not only on the surface reflectance spectrum, but also on the (unknown) illuminant spectrum, which thus represents a nuisance parameter. Consequently, determining the color index of each surface patch in the barcode is a challenging operation, especially if multiple light conditions (indoor/outdoor) are expected. Other nuisance parameters include: specularities; color drift during printing; color fading; unknown or poorly calibrated camera color response; camera non-linearity; color mixing from two nearby patches due to blur; quantization and noise.

The strategy adopted by the HCCB detection algorithm [12] is to identify four clusters in color space using mean shift, and to assign each cluster center to one of the colors in a palette, contained in the barcode itself. Although this strategy has proven effective for the limited number of colors in an HCCB barcode, color clustering is not guaranteed to work well when many more colors are present. In addition, this strategy only works for dense barcodes, with the number of patches largely exceeding the number N of colors available, so that the addition of the color palette to the barcode has minimum impact on the spatial density of information.

This contribution takes a different approach, and considers color barcodes that can be decoded under multiple illuminants without the need to display a color palette or a reference color. Our strategy is to consider groups of k color patches (with $k \geq 2$), and model their evolution due to changing illuminant using a low-dimensional linear subspace. Thus, each group of k colors (a *barcode element*) is represented by one such subspace. When a group of k color patches is observed, our algorithm does not attempt to decode each color independently. Instead, the whole group is

2011 IEEE International Conference on Computer Vision Workshops
978-1-4673-0063-6/11/$26.00 ©2011 IEEE

Authorized licensed use limited to: Sterne Kessler Goldstein Fox. Downloaded on April 27,2026 at 14:53:59 UTC from IEEE Xplore. Restrictions apply.

Apple EX1018
IPR2026-00317
U.S. Patent No. 12,230,394

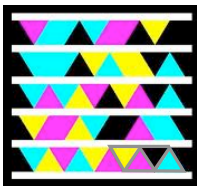

Figure 1. An example of HCCB code. The rightmost four patches in the last row (shown here with a gray border for display purpose) represent the palette of reference colors.

decoded by finding the nearest subspace in a dataset. We show experimentally that this algorithm enables relatively information rate (average number of bits per bar) with very low probability of incorrect detection. In particular, our results show that this algorithm has the potential to convey more information for the same area than technology such as HCCB that requires display of the reference colors for correct decoding.

Note that this article focuses only on algorithms for decoding the color information in a barcode. Other important issues such as detecting the presence of a barcode in the image, or localizing each patch individually, are outside the scope of this work.

2. Related Work

There is surprisingly little research work in the literature on the topic of color barcodes besides Microsoft's HCCB technology cited earlier. Perhaps the first reported attempt to use color in a 2-D barcode can be found in a patent by Han *et al.* [8], who used reference cells to provide standard colors for correct indexing. This technology, named ColorCode™, is marketed by Colorzip Media (colorzip.com). A different type of color barcodes [14] is marketed by ImageId (imageid.com). Bulan *et al.* [3] proposed to embed data in two different printer colorant channels via halftone-dot orientation modulation. Grillo *et al.* [7] used 4 or 16 colors in a regular QR code. PM codes [16] use color to define layers, each of which makes up a 2-D barcode. Kato *et al.* [9] select colors that are maximally separated in a plane of the RGB color cube. The same type of color barcode (named MMCC) was used in a study the effect of JPEG compression on decoding [15]. Pei *et al.* used four colors in a color barcode technology named "Continuous Color Barcode Symbols" [13].

None of these previous works tried to model the changes of the observed color due to changing illuminant, except for the patent of Sali and Lax [14], which uses a k-means classifier to assign the (R,G,B) value of a color patch to one reference color. Note that existing color constancy algorithms (e.g.[10, 6, 5, 4, 2]) are not of much use here. Classic color constancy assumes that neither the surface reflectance nor the illumination are known *a priori*, and aims to infer the surface colors under some specific scene hypotheses (e.g., gray world model, low-dimension illuminant/reflectance spectra). In our case, we have full control over the selection of the surface colors that can be part of a barcode. This facilitates detection, but also poses the problem of which surface colors and color combinations are best suited to the task. Closest to our work is a recent paper by Wang and Manduchi [17], who studied the problem of information embedding via printed color. Their algorithm used one or more reference patches of known color, seen under the same illuminant as the color to be decoded. Observation of the reference patch(es) produces an estimate of a parametric color transformation between a canonical illuminant and the current illuminant, which is then used to decode the information-carrying color patches. The reference patches thus play a similar role to the color palette attached to the HCCB barcode, without the need to display all colors in the palette (usually one or two reference color patches suffice).

Our study borrows the idea of statistical modeling of joint color changes from the work on color eigenflows by Miller and Tieu [11]. However, rather than trying to represent the variation of *all* printable colors as a function of illuminant, we concentrate on the variation of small groups of printed colors.

3. Information Rate

We assume that the patches in a color barcode are built from a set $\mathcal{C}_N$ of N *reference patches*[1] A length k *barcode element* is an ordered set of k reference patches extracted from $\mathcal{C}_N$. For reasons discussed in Sec. 5, we assume that the patches in a barcode element are selected without replacement (i.e., all color patches in a barcode element are different from each other). We also assume that only a subset $\mathcal{B}_{k,x}$ of the set of all possible length k barcode elements $\mathcal{B}_k$ can be used to build a barcode, where x denotes the proportion of elements of $\mathcal{B}_k$ in $\mathcal{B}_{k,x}$ (with $0 < x \leq 1$).

A *barcode* is the juxtaposition (in any spatial pattern) of n barcode elements, resulting in $K = nk$ bars. The *information rate* R of a barcode[2] (measured in bits per bar) is defined by the logarithm base 2 of the number of different symbol that can be represented by the barcode, divided by the number of bars:

$$R = \frac{1}{k} \log_2 \frac{xN!}{(N-k)!} \tag{1}$$

Note that if $k \ll N$, the following approximation holds:

$$R \approx \log_2 N + \frac{\log_2 x}{k} \tag{2}$$

[1] Note that the words "patch" and "bar" are used interchangeably in this paper to mean a region with uniform color.

[2] Note that in communication theory, "information rate" usually represents the average entropy per symbol. Our definition assumes that all symbols are equally likely.

Authorized licensed use limited to: Sterne Kessler Goldstein Fox. Downloaded on April 27,2026 at 14:53:59 UTC from IEEE Xplore. Restrictions apply.

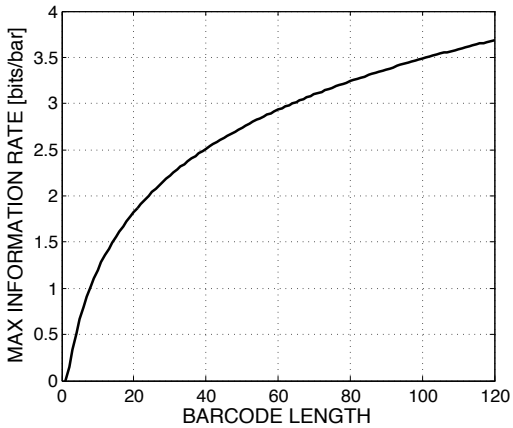

Figure 2. The maximum information rate $\bar{R}_{\max}(K)$ for a barcode that displays its reference colors as a function of the barcode length K.

The goal of a barcode decoder is to infer the index of each observed barcode element in $\mathcal{B}_{k,x}$. The mainstream approach to color barcode decoding (e.g. [8, 12]) assumes that N patches in the barcode are reserved to display the reference colors. This allows for faithful decoding by comparing the color of the patches in the barcode against the displayed reference colors. However, this solution comes at the cost of reduced information rate, since the N reference patches cannot be used to encode information. The information rate in this case is (assuming that each patch is decoded independently):

$$\bar{R}(K,N) = \left(1 - \frac{N}{K}\right) \log_2 N \tag{3}$$

It is instructive to compute the maximum information rate achievable with this system as a function of the barcode length K (shown in Fig. 2 for $K \leq 120$):

$$\bar{R}_{\max}(K) = \max_N \bar{R}(K,N) \tag{4}$$

For example, the maximum information rate of a length 60 barcode that displays its reference colors is of about 3 bits/bar, meaning that this barcode cannot carry more than 180 bits. This value of information rate is obtained for N=16 reference colors. Increasing the number of reference colors decreases the information rate for this barcode length, as the gain due to the higher number of symbols that can be represented by each bar is undermined by the fact that fewer patches are available for carrying information.

Along with the information rate, it is important to consider the error rate. Let $P_E(k,x)$ be the probability of decoding error (incorrect identification) of a generic barcode element in $\mathcal{B}_{k,x}$. The probability of decoding error for the whole barcode, assuming that decoding errors for the individual barcode elements in the barcode are statistically independent events, is:

$$P_E(K,k,x) = 1 - (1 - P_E(k,x))^{K/k} \tag{5}$$

Note that, though a convenient working hypothesis, the assumption of independent decoding error may not hold true in all situations, and should be tested experimentally.

Some qualitative considerations can be drawn from (1) and (5). The information rate R grows linearly with $\log_2 N$ (as long as $k \ll N$) and with $\log_2 x$. Increasing k increases the information rate (note that the second term in the r.h.s. of (1) is negative), and this increase is all the more noticeable for small values of x. For example, if $x = 0.02$, then increasing k from 3 to 4 adds almost 0.5 bits per bar. For what concerns the probability of incorrect decoding, it grows with the number of bars K in the barcode. The dependence of $P_E(k,x)$ on k and x is more complex, and is the object of the work described in the next sections.

In conclusion of this section, we point out that we are not considering any form of channel coding to reduce the decoding error rate at the cost of increased redundancy. Channel coding, which is used commonly for barcodes, could certainly be implemented in the barcodes considered here as well.

4. Barcode Element Decoding

4.1. The Dimensionality of Joint Color Spaces

Decoding a barcode element means finding its index in $\mathcal{B}_{k,x}$ based on the observed colors $\mathbf{c} = [c_1, \ldots, c_k]^T$ of its patches, where $c_i = [c_i^R, c_i^G, c_i^B]^T$ is an (R,G,B) color vector. As well known, a variation of the illuminant spectrum determines a variation of the perceived colors. A popular model to describe the observed color of a Lambertian surface [10] 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. Thus, the observed color of a surface s under a given illuminant is equal to

$$c(v) = \Phi_s v \tag{6}$$

where v is a vector of length N_{ill} containing the coefficients of the illuminant with respect to the chosen basis, and Φ_s is a full-rank $3 \times N_{\text{ill}}$ matrix whose entries are a function of the illumination and reflectance basis vectors as well as of the spectral sensitivities of the camera. Note that, since $N_{\text{ill}} \geq 3$ in general, the rank of Φ_s is 3, making the decoding of an individual color c hopeless without some prior knowledge of the scene or of the illuminant. If, however, *multiple* colors seen under the same illuminant are decoded at once, the task is less daunting. For example, consider the vector $\mathbf{c}(v)$ formed by the colors in the barcode elements as defined above. Then $\mathbf{c}(v) = \Phi v$ with

$$\Phi^T = [\Phi_1^T | \ldots | \Phi_k^T] \tag{7}$$

Authorized licensed use limited to: Sterne Kessler Goldstein Fox. Downloaded on April 27,2026 at 14:53:59 UTC from IEEE Xplore. Restrictions apply.

where

$$\text{rank}(\Phi) = \min(3k, N_{\text{ill}}) \tag{8}$$

Hence, the vector $\mathbf{c} \in \mathbb{R}^{3k}$ is constrained to live in a subspace $\mathcal{S}$ of dimension of at most N_{ill}. This observation is critical for our decoding algorithm, as discussed next.

4.2. Joint Subspace Generation

Our decoding algorithm begins by modeling the subspaces $\mathcal{S}(i)$ of all length k barcode elements. We have considered two approaches to build these subspace. In the first case, each barcode element is seen under a wide variety of illuminants, and a subspace of suitable dimension M is built from these observations via Principal Component Analysis. In practice, one only needs to take images under multiple illuminants of the N reference patches, build vectors $\mathbf{c}$ from these images for each barcode element, and compute the SVD of the resulting matrix. This procedure produces accurate subspace modeling; in practice, however, it may be unwieldy, because each subspace depends on the color patches as well as on the characteristics of the camera used to observe them. This means that the whole training procedure (including taking pictures of the patches under different illuminants) would have to be repeated each time a different camera is used or the colors are printed with a different printer. We thus consider a second approach to subspace modeling, which, albeit less accurate, enables a much simple training procedure. This approach relies on the diagonal (or Von Kries) model of color changes, which assumes that each color channel changes as a result of an illuminant change by a multiplicative factor that depends on the illuminant but not on the reflectant:

$$c_s(v_2) = D_{v_1 \to v_2} c_s(v_1) \tag{9}$$

where $D_{v_1 \to v_2}$ is a diagonal matrix. It is easy to see that, in this case, $c(v) = \Phi v$ with

$$\Phi^T = \begin{bmatrix} c_1^R & 0 & 0 & c_2^R & 0 & 0 & c_3^R & \dots \\ 0 & c_1^G & 0 & 0 & c_2^G & 0 & 0 & \dots \\ 0 & 0 & c_1^B & 0 & 0 & c_2^B & 0 & \dots \end{bmatrix} \tag{10}$$

The three columns of Φ form an orthogonal basis of the subspace. In practice, calibrating the algorithm for a new camera or a new set of reference colors can be performed simply by taking a single picture of all N reference patches *under any illuminant*. This allows the system to immediately generate a basis for each barcode element by building the corresponding matrix Φ.

4.3. Nearest Subspace Decoding

Given the observed color vector $\mathbf{c}$ of a barcode element, we decode it by assigning it to the subspace $\mathcal{S}(i)$ that has the minimum distance to $\mathbf{c}$ (where the distance of $\mathbf{c}$ to $\mathcal{S}(i)$ is defined in the usual way by the Euclidean distance between $\mathbf{c}$ and its projection onto $\mathcal{S}(i)$). Nearest subspace search is a common technique in Computer Vision. It was shown by Basri *et al.* [1] that it is possible to map subspace $\mathcal{S}$ and query item $\mathbf{c}$ to points in $\mathbb{R}^{d'}$ for some d', in such a way that the Euclidean distance between these two points increases monotonically with the distance of $\mathbf{c}$ to $\mathcal{S}$, thereby enabling the use standard nearest neighborhood techniques (e.g. k-d trees) for barcode element decoding.

4.4. Barcode Elements Subset Selection

Nearest subspace decoding produces a certain probability of error P_E, defined as the average probability of decoding error over all barcode elements. Note that P_E contributes to the probability of decoding error for a barcode formed by n barcode elements as by (5). If P_E is larger than desired, one may reduce the cardinality of the subspace elements by only selecting a subset $\mathcal{B}_{k,x}$ of $\mathcal{B}_k$. Intuitively, a smaller set provides fewer opportunities for misclassification, at the cost of reduced information rate.

Selection of a proportion x of barcode elements that minimizes the associated P_E is computationally very expensive. In particular, if one barcode element is removed from $\mathcal{B}_k$, the new empirical probability of error needs to be recomputed for all barcode elements. We have considered several techniques to reduce the complexity associated with subset selection. One possible approach is to use subspace distance as an indicator of the probability that two barcodes could be confused with each other. We adopt the following definition of distance between two subspaces $\mathcal{S}_1$ and $\mathcal{S}_2$ with dimension d_1 and d_2, respectively [18]:

$$\text{dist}^2(\mathcal{S}_1, \mathcal{S}_2) = \max(d_1, d_2) - \|\Phi_1^T \Phi_2\|_F \tag{11}$$

where Φ_1, Φ_2 are any orthonormal basis matrices of $\mathcal{S}_1$ and $\mathcal{S}_2$, respectively, and $\|\cdot\|_F$ represents the Frobenius norm. Fig. 3 shows the effect of pairwise subspace distance on the probability of error. More precisely, we considered all barcode elements that could be built with $N = 24$ patches and k bars (see Sec. 5 for details about our experimental dataset). We then estimated (via cross-validation over multiple illuminants) the probability that the barcode element i is incorrectly decoded as j with $i \neq j$. Obviously, the sum of all these probabilities, divided by the number of barcode elements, gives the probability of incorrect decoding P_E. To build the plot in Fig. 3, we ordered all length k barcode element pairs according to decreasing distance. Then we computed the cumulative sum of all probabilities of decoding i as j, divided by the number of barcodes. The plot clearly shows that the contribution to the overall probability of decoding error P_E is due for the most part to the barcode element pairs that are closest to each other. This suggests that barcode element subset selection could be accomplished based on pairwise subspace distance. For example, for small k, we adopt the following greedy strategy. Start

Authorized licensed use limited to: Sterne Kessler Goldstein Fox. Downloaded on April 27,2026 at 14:53:59 UTC from IEEE Xplore. Restrictions apply.

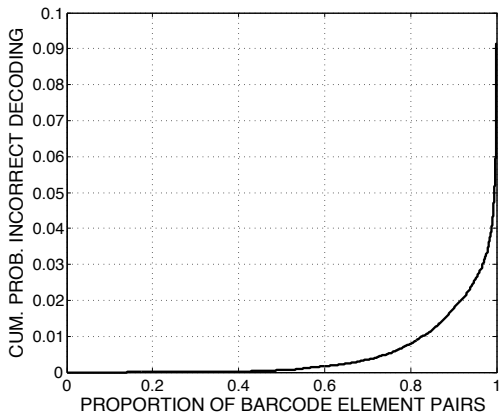

Figure 3. The cumulative probability of incorrect decoding as a function of the proportion of ordered barcode element pairs considered for $k = 3$ (see text). All subspaces have dimension of 2.

from a barcode element chosen at random. At each iteration, add to the subset the barcode element that maximizes the minimum subspace distance to all barcode elements already selected. For larger values of k, even this procedure may become too computationally intensive, and we resort to a simpler strategy. First, we compute the probability of incorrect decoding for each barcode element. Then, we build the subset from the barcode elements that have the smallest probability of incorrect decoding.

5. Experiments

The reference color patches for our experiments were selected from a checkerboard of 512 colors, uniformly sampled in (R,G,B) color space, printed on paper by a regular printer. Images were taken of the checkerboard with a Canon EOS 350D camera in raw (CR2) format under 69 different lighting conditions (including direct sunlight, diffuse skylight with overcast sky or under cast shadow, and various types of artificial light). We selected ten representative illuminants by k-means clustering of the observed color values of a white patch in the set.

We then selected two sets of reference color patches, $\mathcal{C}_{24}$ and $\mathcal{C}_{12} \subset \mathcal{C}_{24}$ for $N = 24$ and $N = 12$ respectively, using the greedy strategy introduced in Sec. 4.4. The selected colors are shown in Fig. 4. Synthetic images of all barcode elements for k ranging between two and five were built from the average color values of the images of the reference color patches seen under the ten representative illuminants (where all patches forming a barcode elements were seen under the same illuminant). For each set of length k barcode elements, we extracted subsets with various proportion x as discussed in Sec. 4.4. For each such subset, we computed the probability of incorrect decoding $P_E(k, x)$ as follows. We ran five rounds of cross-validation, each time

Figure 4. The 24 color patches selected for our tests. The first two columns contain the colors for the tests with N=12.

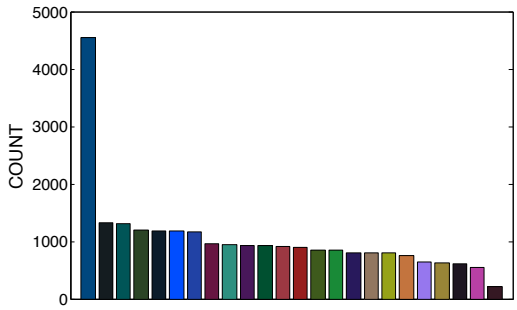

Figure 5. For each one of the 24 reference colors, the plot shows the number of barcode elements selected with $k = 5$ and $x = 0.003$ containing that color.

picking five illuminants at random, learning the subspaces for each barcode element considered based on its images under these illuminants, and testing each barcode element in turn on one of the remaining illuminants, randomly chosen. We counted the number of times any barcode element was incorrectly decoded, and divided the result by the number of cross-validation rounds (five) and by the number of barcode elements in the subset (equal to $x \cdot 24!/(24-k)!$). We also tested the decoding algorithm based on the diagonal model discussed in Sec. 4.2. In this case, the color subspaces were built from observation of the reference colors under just one illuminant. We ran five rounds of cross-validation, each time selecting one illuminant at random (without repetition), training our model on such illuminant and testing it with barcode elements seen under another randomly chosen illuminant.

Fig. 5 shows the number of barcode elements (selected for $k = 5$ and $x = 0.003$ using the selection algorithm discussed at the end of Sec. 4.2) containing each one of the 24 reference colors in $\mathcal{C}_{24}$. Note that 22 reference colors are chosen with comparable probability; one color had much higher probability of being selected, while another color was selected much less often than the others.

Fig. 6 shows the probability of incorrect decoding P_E for

Authorized licensed use limited to: Sterne Kessler Goldstein Fox. Downloaded on April 27,2026 at 14:53:59 UTC from IEEE Xplore. Restrictions apply.

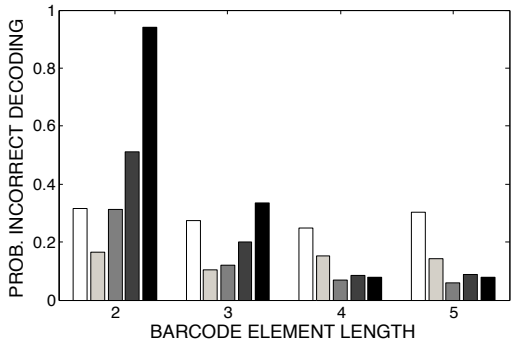

Figure 6. The probability P_E of incorrect decoding as a function of the barcode element length k and subspace dimension M from 1 (white bars) to 5 (black bars).

barcode length k between two and five and for subspace dimension M between one and four (see Sec. 4.2). Note that, for each k, there is an optimal value of the subspace dimension M, and that the error increases for larger values of M. This may be due to the fact that increasing the subspace dimension M may lead to overfitting and poor generalization. In the experiments presented below, we chose to use $M = 2$ for $k = 2, 3$ and $M = 3$ for $k = 4, 5$.

Fig. 7 shows the probability of decoding error $P_E(k, x)$ for a generic barcode element of length k between 2 and 5, using $N = 12$ and $N = 24$ reference colors, and for various values of the subset proportion x, as a function of the resulting information rate (as by (1)). Both types of subspace modeling (via PCA or via the diagonal model) are considered. Fig. 8 shows the probability of decoding error $P_E(K, k, x)$ for a K length barcode, computed using Eq. (5) for $K = 60$, 120, and 240.

The main observation that can be drawn from these results is that it is possible to reach relatively high information rate with very low error rate. For example, PCA-based subspace modeling for $k = 5$ and $N = 24$ results in a probability less than 0.001 of incorrect decoding of a length 60 barcode, while allowing one to encode information at a rate of about 3.8 bits per bar. To put this result in context, let us recall from Fig. 2 that the maximum information rate of a length 60 barcode that displays its reference colors is of less than 3 bits per bar. Thus, our system allows one to pack about 0.8 additional bits per bar (or 48 bits overall) in a length 60 barcode with very low decoding error probability. One can easily infer from Eq. (1) that an information rate of 3.8 bits/bar for a length 5 barcode element and $N = 24$ colors is achieved for $x = 0.103$, and thus decoding each barcode element requires finding the nearest subspace in a database of 524,288 elements.

The effect of the number of reference colors N, barcode element length k, and subspace modeling algorithm on the resulting information rate and decoding error probability are clear from Fig. 7 and 8. Increasing the number of reference colors N allows one to achieve higher values of information rate. Longer barcode elements result in lower decoding error probability for the same information rate. For this reason, we do not allow color repetition in a barcode element. Repeating a color in a length k barcode is equivalent (for what concerns decoding) to include $\mathcal{B}_{k-1}$ in $\mathcal{B}_k$, leading to a substantial increase of the decoding error for $\mathcal{B}_k$. Using the diagonal model to build joint color subspaces (which, as discussed in Sec. 4.2, allows for a fast calibration procedure) leads to an increase of the decoding error rate by a factor of 10. Even so, for moderately long barcodes (e.g. $K = 60$ bars), the decoding error probability remains low (the probability of decoding error for a barcode with 60 bars is equal to 0.01 at $R = 3.8$ bits per bar).

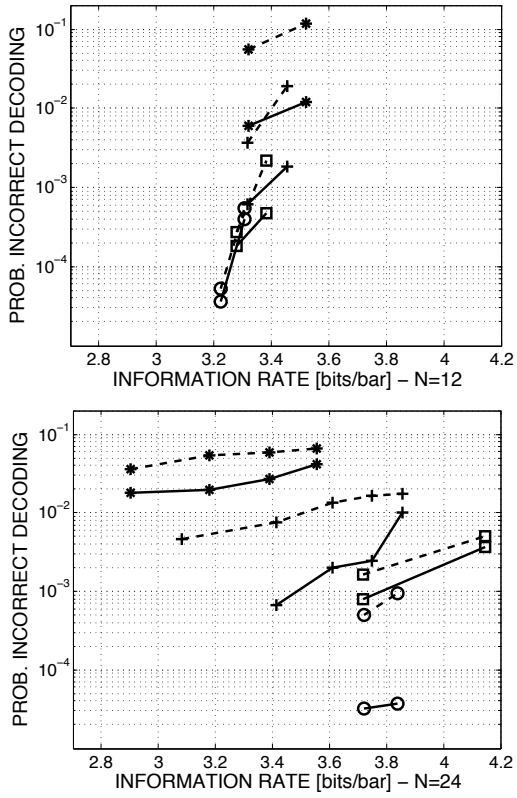

Figure 7. The probability incorrect detection $P_E(k, x)$ (on a logarithmic scale) for length k barcode elements versus the information rate R (1). For each value of the barcode element length k, a variable number of subset size proportions x were tested. '$*$': k=2; '+': k=3; '□': k=4; '○': k=5. Continuous line: subspaces learnt via PCA over five illuminants. Dashed line: diagonal model (10).

6. Conclusions

We have proposed a new algorithm for decoding barcode elements in a color barcode that does not display its reference colors. Our experiments have shown that, by care-

Authorized licensed use limited to: Sterne Kessler Goldstein Fox. Downloaded on April 27,2026 at 14:53:59 UTC from IEEE Xplore. Restrictions apply.

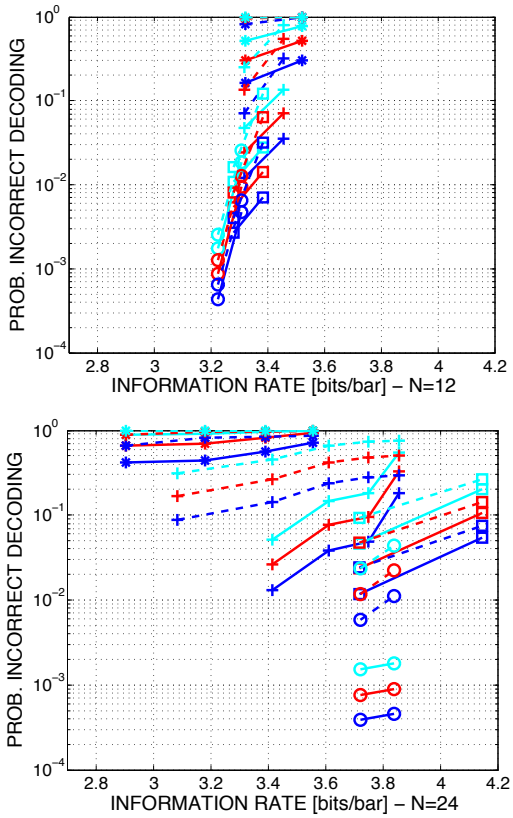

Figure 8. The probability incorrect detection $P_E(K, k, x)$(on a logarithmic scale) for a length K barcode element formed by multiple length k barcode elements versus the information rate (1). For each value of the barcode element length k, a variable number of subset size proportions x were tested. '$*$': k=2; '+': k=3; '□': k=4; '○': k=5. Blue line: K=60; red line: K=120; cyan line: K=240. Continuous line: subspaces learnt via PCA over five illuminants. Dashed line: diagonal model (10).

fully selecting a subset of barcode elements, it is possible to achieve good information rate at low decoding error probability. Thus, this approach represents a serious competitor to mainstream barcode technology that requires display of the reference colors, thereby limiting the effective information rate.

More research work is needed to compare the decoding error rate achieved by our system with other sources of error in practical situations (for example, errors due to blur-induced color mixing from two nearby patches or to printed color drift and fading). Our current research is also addressing the issue of fast decoding. As mentioned in Sec. 4.3, nearest neighbor search techniques can be used for finding the closest subspace to a color vector by means of a mapping into a higher dimensional space. Our goal is to implement a system that can reliably decode length 5 barcode elements on a smart phone at less than one tenth of a second, thus enabling reading of a complete barcode with 50 bars in less than one second.

Acknowledgements

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

References

[1] R. Basri, T. Hassner, and L. Zelnik-Manor. Approximate nearest subspace search with applications to pattern recognition. In *Proc. IEEE Computer Vision and Pattern Recognition*, 2007. 4

[2] D. H. Brainard and W. T. Freeman. Bayesian color constancy. *Journal of the Optical Society of America A*, 14:1393–1411, 1997. 2

[3] O. Bulan, V. Monga, and G. Sharma. High capacity color barcodes using dot orientation and color separability. In *Proc. SPIE-IS&T Electronic Imaging*, volume 7254, 2009. 2

[4] G. Finlayson, M. Drew, and B. Funt. Diagonal transforms suffice for color constancy. In *Proceedings of Fourth International Conference on Computer Vision*, pages 164–171, 1993. 2

[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.*, 23:1209–1221, November 2001. 2

[6] D. A. Forsyth. A novel algorithm for color constancy. *Int. J. Comput. Vision*, 5:5–36, September 1990. 2

[7] A. Grillo, A. Lentini, M. Querini, and G. Italiano. High capacity colored two dimensional codes. In *Proc. Int. Multiconf. on Comp. Science Inf. Tech.*, 2010. 2

[8] T. Han, C. Cheong, N. Lee, and E. Shin. Machine readable code image and method of encoding and decoding the same. U.S. Patent 7020327, 2000. 2, 3

[9] H. Kato, K. Tan, and D. Chai. Novel colour selection scheme for 2D barcode. In *Proc. International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS 2009)*, 2009. 2

[10] 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 293–297. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1987. 2, 3

[11] E. Miller and K. Tieu. Color eigenflows: Statistical modeling of joint color changes. In *Proc. of International Conference on Computer Vision(ICCV)*, volume 1, pages 607–614, 2001. 2

[12] 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 (WACV '08:)*, pages 1–6, Washington, DC, USA, 2008. IEEE Computer Society. 1, 3

[13] S. Pei, G. Li, and B. Wu. Codec system design for continuous color barcode symbols. In *Proceedings of the 2008*

Authorized licensed use limited to: Sterne Kessler Goldstein Fox. Downloaded on April 27,2026 at 14:53:59 UTC from IEEE Xplore. Restrictions apply.

IEEE 8th International Conference on Computer and Information Technology Workshops, pages 539–544, Washington, DC, USA, 2008. IEEE Computer Society. 2

[14] E. Sali and D. Lax. Color bar code system. U.S. Patent 7210631, 2006. 2

[15] K. Tan, S. Ong, and D. Chai. JPEG color barcode images analysis: A camera phone capture channel model with autofocus. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 2(4), 2009. 2

[16] O. Tatsuya and M. Kazuhiro. Layered two-dimensional code, creation method thereof, and read method. United States Patent Application 20090166418, 2006. 2

[17] F. Wang and R. Manduchi. Color-constant information embedding. In *Proc. IEEE Workshop on Color and Reflectance in Imaging and Computer Vision*, 2010. 2

[18] L. Wang, X. Wang, and J. Feng. Subspace distance analysis with application to adaptive bayesian algorithm for face recognition. *Pattern Recognition*, 39:456–464, March 2006. 4

Authorized licensed use limited to: Sterne Kessler Goldstein Fox. Downloaded on April 27,2026 at 14:53:59 UTC from IEEE Xplore. Restrictions apply.