Journal of Imaging Science and Technology® 54(6): 060405–060405-6, 2010.

Error-Correcting Code (ECC) and Module Size Considerations in 2D Aztec Barcode Readability

Steven J. Simske▲ and Marie Vans
Hewlett-Packard Laboratories, Fort Collins, Colorado 80528, USA
E-mail: E-mail: steven.simske@hp.com

Guy B. Adams
Hewlett-Packard Laboratories, Long Down Avenue, Stoke Gifford, Bristol BS34 8QZ, United Kingdom

Abstract. *Error-Correcting Code (ECC) provides robust readability to both linear and two-dimensional (2D) barcodes, particularly for localized damage. Many ECC approaches, however, are based on assumptions about the types of damage or the communication channel used. As the applications for barcodes rapidly evolve with the increasing ubiquity of mobile cameras, an evaluation of the trade off between ECC and simply increasing the size of the bar code modules is required. This article compares the impact of simply changing the module size versus using error correction—which exhausts a percentage of the symbol area without adding payload data. For three typical nondestructive types of damage—the print-scan cycle, low quality printing, and blurring—investigated here, there is compelling support for increasing the size of the barcode modules and foregoing ECC. Freeing barcodes from the need for ECC provides an additional advantage: namely, the barcodes can be scrambled to be unreadable under the defined barcode specification without a change in size or appearance. This is in contrast to the use of ECC, for which only a small minority of rearrangements of the data in the barcode would be "decodable." Thus, without the use of ECC, it is much harder for a would-be counterfeiter or other dishonest party to determine the scrambling approach. Additionally, we can create 2D barcodes that are not "readable" using commercially available reading software, except where so desired. These results are discussed in light of destructive damage and for different applications of 2D bar codes. © 2010 Society for Imaging Science and Technology.*
[DOI: 10.2352/J.ImagingSci.Technol.2010.54.6.060405]

INTRODUCTION

Mobile image capture devices are ubiquitous, providing an ever-increasing ease and saliency for consumers, retailers, supply chain managers and manufacturers[1–4] to interrogate products,[1,2] labels,[1–3] and even signage.[4] A typical data vehicle is the barcode, which consists of black and white modules arranged in a one or two-dimensional array pattern. There are several well known symbologies, such as Aztec and DataMatrix two-dimensional (2D) codes or the UPC one-dimensional code used for automated grocery check-out systems. Many barcode implementations, however, rely on error-correcting code (ECC), to add robustness to the barcode reading process. The robustness model, however, historically derives from the printing of one-dimensional (1D) barcodes; mailing applications; environmental damage associated with smeared ink on low-quality paper; and abrasion or puncture damage.

For the reading of barcodes with cameras on mobile devices, localized damage is arguably a less important consideration than overall low image quality during capture. Poor or non-uniform illumination, blur due to poor focus and/or motion, and poor quality printing can cause low quality image capture. However, ECC is not necessarily designed to overcome these distortions. It has been argued that the selection of ECC, based on communication theory, is a largely misplaced focus, particularly for defects introduced in the process of printing or of scanning.[5] The same reference argues for "finessing the size of the spots and cells [to] minimize the effects of printing defects," and continues "by increasing the size of the data features…virtually any anticipated problem in these domains can be compensated for, so that information can be perfectly communicated."[5] In this article, we introduce the term Error Correction by Percentage of Symbol Area, or ECPSA. We compare the readability of Aztec barcodes with variable ECPSA to uncorrected barcodes with the same density of data bits (payload) per unit area. Separately, print-scan (PS), low quality printing, and blurring conditions are used to provide test cases for ECC/non-ECC comparisons. The next section describes the experiments performed. We then provide the results, and conclude with a discussion of the results together with recommendations for deployment of 2D barcodes.

METHODS AND MATERIALS

A series of barcode readability tests were performed using Aztec symbology[12] high-capacity 2D matrix barcodes. Aztec is able to encode both ASCII and Extended ASCII characters, and when using its full range mode of 151 modules combined with 25% ECPSA, the Aztec symbology is able to encode up to 3000 characters or 3750 numeric digits (that is, its size ranges between 15×15 modules and 177×177 modules). Aztec symbology uses Reed-Solomon error correcting code, which is a cyclic, multilevel, variable-length digital ECC used to correct multiple random error patterns (i.e., it is based on BCH code). All experiments were performed

▲IS&T member

Received Jan. 28, 2010; accepted for publication Jul. 23, 2010; published online Nov. 4, 2010.

1062-3701/2010/54(6)/060405/6/$20.00.

Table I. Specifications of the barcodes printed and tested in this article. All barcodes were Aztec code, and so 81 of the $27\times27=729$ possible bits were used for the invariant center target, leaving 648 bits for payload and ECC. Module sizes tested ranged from 8 to 30 mils (0.2–0.75 mm) in integral sequence (8, 9, 10,…, 30), where $1\ \text{mil}=10^{-3}$ in.

Target ECC (as ECPSA)	Size (modules)	Payload bits	ECC bits	Actual ECPSA
0% ECPSA	27×27	648	0	0.0%
10% ECPSA	27×27	584	64	9.9%
20% ECPSA	27×27	520	128	19.8%
30% ECPSA	27×27	456	192	29.6%
40% ECPSA	27×27	392	256	39.5%
50% ECPSA	27×27	328	320	49.4%

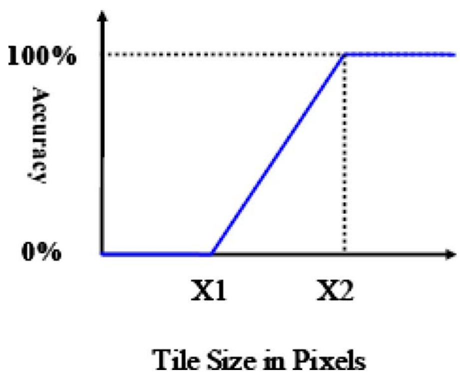

Figure 1. Piecewise-linear model for authentication used to fit barcode reading success data. The best fit minimizes the least-squared error between the predicted (solid blue line segments) and measured BRS data at tile sizes $8, 9, 10, \ldots, 30^6$.

using a 27×27 module configuration, so that comparisons between images with the same module size were not affected by overall barcode size. For every test, module size was varied from 8 to 30 mils (0.2–0.75 mm) in 1 mil increments ($1\ \text{mil}=10^{-3}$ in). Each module is either black (100% opacity) or white (100% transparency). B-Coder© Professional software (TAL Technologies, Inc., Version 4.0) allowed us to vary the ECPSA and payload—this is, in fact, why Aztec symbology rather than DataMatrix was chosen for the experiments. Through iterative adjustment of these settings, we were able to obtain 27×27 module Aztec barcodes with 0%, 10%, 20%, 30%, 40%, and 50% ECPSA settings (Table I). The number of payload modules (or "bits"), M_P, was equal to 648, 584, 520, 456, 392, and 328, respectively, for these ECPSA settings, so that the number of ECPSA, or nonpayload modules, M_{NP}, was equal to 0, 64, 128, 192, 256, and 320 bits, respectively.

The relative effectiveness of ECPSA depends on the 100% intercept, $X2$, of the piecewise-linear approximation[6] to the barcode reading success (BRS) curve shown in Fig. 1. This best fit minimizes the least-squared error between the predicted (solid blue line segments) and measured BRS data at tile sizes $8, 9, 10, \ldots, 30^6$.

For all of the following tests, the barcodes were read using an InData Systems™ 9500LDS portable terminal with add-on optic "shroud" for 405 nm LED (light-emitting diode) Light Delivery System (LDS-V2), hereafter "IDS-LDS." This system provides uniform lighting conditions (405 nm illumination) for all barcode reading performed (so that we are sure consistent illumination was used throughout the experiments). The IDS-LDS also tracks the time it takes for a successful read to occur, which was recorded throughout. Multiple pages (20 or more barcodes at each of the 23 module sizes) were printed under the following experimental conditions:

Figure 2. Example of original and damaged Aztec Code 2D barcodes (ch=648 indicates maximum bits to use for encoding characters). Original size was 20×20 mils per module. Payload is 456 out of 648 payload bits (30% ECPSA). Images show no damage (left), 12.5% damage (center) and 25% damage (right).

(1) Print using an HP 3600 Color LaserJet (hereafter "CLJ") with grayscale-only settings. This is the baseline experiment.
(2) Print using the CLJ, scan and print twice using the HP 6280 ink jet all-in-one (hereafter "IJ-AIO"). This degrades the barcodes printed in (1) by two print-scan (hereafter "PS") cycles, and constituted the PS-channel distortion experiment.
(3) Print directly using the IJ-AIO with grayscale-only settings. Since the IJ-AIO print quality is lower for barcodes than the CLJ, this constituted the "reduced print quality" (hereafter "RPQ") distortion experiment.
(4) Place a large external lens on the optic shroud of the IDS-LDS to directly blur the images from experiment (2). This was used in place of, for example, simulated blur,[7] and the presence of image blur was verified in software using the image capturing capability of the IDS-LDS and measurement of PSNR,[7] and using the value of $X2$ (Fig. 1), which substantially increased when the lens was in place (evidence of lower quality, blurred image). This constituted the blur distortion experiment. This resulted in a high degree of blur suitable for testing image deterioration, but is likely not representative of all blurs (e.g., motion blur) encountered in real-world barcode reading.
(5) Add damage through filling in of all white modules in increments of 1/8 of the overall payload area of the barcode, as shown in Fig. 2. This constituted the destructive damage (hereafter designated "DD") distortion experiment.

Since the value of $X1$ in Fig. 1 is in mils, or 10^{-3} in, the units of $(X2)^2$ are 10^{-6} in^2, and the Payload Density (PD) in bytes/in^2 for each test scenario was computed from:

Table II. Piecewise linear 100% accuracy value $X2$ value in mils and resulting Payload Density (PD, in parentheses) in bytes/in^2 for the original CLJ printing, PS-distortion and RPQ-distortion experiments.

Target ECC (as ECPSA)	CLJ original	CLJ+PS distortion	IJ-AIO RPQ distortion
0% ECPSA	8.5 (1730)	8.8 (1610)	10.2 (1200)
10% ECPSA	8.0 (1760)	8.8 (1450)	10.1 (1100)
20% ECPSA	8.0 (1570)	9.1 (1210)	9.3 (1160)
30% ECPSA	8.0 (1370)	9.1 (1060)	8.8 (1140)
40% ECPSA	8.0 (1180)	9.2 (890)	9.3 (870)
50% ECPSA	8.0 (990)	8.7 (840)	8.8 (820)

Table III. Piecewise linear 100% accuracy value $X2$ value in mils and resulting Payload Density (PD, in parentheses) in bytes/in^2 for the blurred CLJ printing, PS-distortion and RPQ-distortion experiments.

Target ECC (as ECPSA)	CLJ+Blurring	CLJ+PS+Blurring	IJ-AIO-RPQ+Blurring
0% ECPSA	13.7 (670)	14.4 (600)	14.8 (570)
10% ECPSA	13.3 (640)	14.4 (540)	14.2 (560)
20% ECPSA	13.1 (580)	14.4 (480)	14.2 (500)
30% ECPSA	13.6 (480)	14.2 (440)	13.7 (470)
40% ECPSA	13.2 (430)	13.8 (400)	13.9 (390)
50% ECPSA	13.0 (370)	13.4 (350)	13.8 (330)

$$PD = 10^6 \times \frac{M_P}{(M_P + M_{NP})} \times \frac{bytes}{bit} \times \frac{1}{X2^2} \qquad (1)$$

where M_P is number of payload modules, M_{NP} is the number of non-payload modules (used for ECC), bytes/bit is by definition 1/8 (the inverse of bits/byte, which is eight), and $10^6/X2^2$ converts mil^2 to in^2.

RESULTS

The results for the CLJ, PS-channel distortion, and IJ-AIO RPQ distortion are shown in Table II. Printing with the CLJ resulted in low values (below, in fact, the minimum specification of 10 mils for the Aztec symbology) for $X2$ and thus relatively high values for PD. For the CLJ, PD was above 1700 bytes/in^2 for 0% and 10% ECPSA. Increasing ECPSA dropped PD by a mean of nearly 20 bytes/in^2 for every 1% increase in ECPSA.

After two print-scan cycles, the $X2$ values increased by a mean (for the six ECPSA values tested) of nearly 0.9 mils, with the PD dropping by a mean of nearly 260 bytes/in^2. The PS distortion resulted in a more uniform drop in PD of approximately 15 bytes/in^2 for every 1% increase in ECPSA.

The results for the reduced print quality (RPQ) distortion differed from those for the CLJ and PS distortions in that a similar PD was observed for 0%, 10%, 20%, and 30% ECPSA. Above 30% ECPSA, however, a drop in PD of approximately 15 bytes/in^2 for every 1% increase in ECPSA was also observed. At 0% and 10% ECPSA, RPQ distortion resulted in a lower PD than PS distortion. Thus, for all three experiments reported in Table II, the highest PD was obtained for 0% alone or in combination with 10% ECPSA.

The effects of blurring on these three experiments are given in Table III. For the CLJ printing, blurring resulted in a 5.2±0.2 (mean±std.dev.) increase in $X2$ for the six different settings of ECPSA. PD was reduced by more than 60% at each setting of ECPSA after blurring was added. The trend for CLJ, however, was the same as without blurring—above 10% ECPSA, there was a drop in PD of approximately 7 bytes/in^2 for every 1% increase in ECPSA. Relative to the PD at 10% ECPSA, this was a quite similar percentage drop as observed for the (non-blurred) CLJ data in Table II.

Table IV. Time (mean of 10 or more successful barcode reads using the IDS-LDS, in msec) for 12 and 15 mil module Aztec 2D barcodes with ECPSA=30%, 40%, 50% and 75%. Damage is induced as shown in Fig. 2 to CLJ+PS distortion samples.

Module size (% damage)	30%	40%	50%	75%
12 mils (0%)	176	215	216	218
12 mils (12.5%)	605	862	676	230
12 mils (25%)		1356	680	489
12 mils (37.5%)				870
15 mils (0%)	126	145	114	165
15 mils (12.5%)	433	123	255	270
15 mils (25%)		167	297	281
15 mils (37.5%)				1358

The trends for CLJ+PS+Blurring in Table III are similar to the results for CLJ+PS in Table II. There is a 5.3±0.4 (mean±std.dev.) increase in $X2$, and PD is reduced by roughly 60% at each ECPSA setting, and monotonically decreases as ECPSA increases.

Blurring of the IJ-AIO RPQ Distortion results in a 4.7±0.4 (mean±std.dev.) increase in $X2$ (Table III) over the values in Table II. Above 10% ECPSA, there is a monotonic increase in $X2$ and decrease in PD.

The Destructive Damage (DD) distortion test (Table IV) shows that the barcode reading time increases significantly with the amount of DD added. When 50% ECPSA is used, the 37.5% DD barcodes are unreadable. This implies that on the CLJ printer, the PS distortion effectively removes at least one-fourth of the ECPSA added; that is, at least 12.5% of the ECPSA is required simply to overcome this PS distortion. When 40% ECPSA is used, however, barcodes with 25% DD can still be read, which delimits the effect of PS distortion on the CLJ+PS to less than or equal to 15% ECPSA.

DISCUSSION

Summary

The primary findings for payload density (PD) for the five experiments performed are as follows:

(1) For the CLJ experiment, peak PD was obtained for the 0% and 10% ECPSA barcodes. Increasing ECPSA above 10% resulted in significantly decreasing payload density. At 50% ECPSA, PD was reduced 42.8%, implying that 85.6% of the ECC added was "wasted," or at least inappropriate to the CLJ printing distortion.
(2) When PS distortion is added to the CLJ experiment, the best results are obtained when ECC is not employed. PS distortion results in a significant decrease in PD at any setting for ECPSA; however, the 0% ECPSA barcodes suffered the least deleterious effects due to PS distortion. At 50% ECPSA, PD was reduced 47.8%, implying that 95.6% of the ECC added was inappropriate to the CLJ+PS distortion.
(3) Ink jet printing resulted in lower barcode quality when compared to CLJ or even CLJ+PS distortion. However, for this test, a relatively similar PD was observed for 0%–30% ECPSA. Still, at 50% ECPSA, PD was reduced by 31.7%, implying 63.3% of the ECC added was inappropriate to recovering from the RPQ distortion.
(4) Physical blurring through the addition of a second lens to the optical path of the barcode reader resulted, predictably, in significantly reduced PD for all experiments performed. 50% ECPSA resulted in 44.8%, 41.7%, and 42.1%, reduction in PD for the CLJ, CLJ+PS and IJ-AIO-RPQ experiments, respectively. This implies that between 80–90 % of the added ECPSA is "wasted," inasmuch as it is not useful in overcoming blur combined with the three experiments. Note that the need for 42.9% ECPSA (the mean required to overcome blur on these three tests) translates into an increase of 19.5% in require module dimension. That is, otherwise 10×10 mil modules must instead be printed at 12×12 mils to accommodate this ECPSA.
(5) The addition of destructive damage (DD) distortion implies that CLJ printing plus PS distortion "equals" in some sense the equivalent of somewhere in the range of 12.5–15 % ECPSA. This implies that an effective ECPSA must be at least 12.5%. The Aztec default ECC is approximately 23%, likely providing an "additional" 10% safety factor. However, at every ECPSA above 12.5% tested herein, excepting the IJ-AIO RPQ 20% ECPSA, the significantly reduced PD contraindicated the deployment of ECC. ECC also results in increased decoding time for damaged barcodes (Table IV), which may be significant for human-to-device interaction.

The maximum payload density (PD) values observed were roughly 1750 bytes/in^2 for the CLJ, and 1200 bytes/in^2 for the IJ-AIO. These are consistent with reported values for the same or similar barcode symbologies. Table IX of Ref. 8, for example, provides bar code PD in bytes/in^2 as: Data Matrix, 1555; Aztec Code, 1888; QR Code, 1941; Multilevel 2D Bar Code (p_1-s_1), 2211; and Multilevel 2D Bar Code (p_5-s_1), 2397. A similar high density of 2200 bytes/in^2 was reported for six-color barcodes.[6]

Alternative bar code approaches include the color triangle approach.[9] For this color triangle, up to 2000 bytes/in^2 has been reported,[3] though the mobile barcode instantiation[3] carries only 30–40 bytes/in^2, implying a 98% allowance for mobile camera distortion. Further approaches to increasing PD include using halftone dot modulation/orientation,[10] spectral precompensation[6] and color calibrating indicia.[6]

Regardless, the Aztec barcodes used here provide high density even without structural precompensation, which compensates well for blur.[11] Since blurring had the strongest negative impact on PD, it appears that a barcode compensation geared to reduce the effect of blurring is an appropriate deployment strategy.

The results of these experiments imply that under print-scan (PS) channel distortion and blurring conditions, ECC is contraindicated for Aztec (and related 2D) barcodes, especially in the case of blur that typically occurs uniformly across the target. Once the threshold of readability is reached the rate of degradation is steep, thus requiring a large amount of ECPSA just to extend readability fractionally. The results in Table IV indicate that approximately 62.5% ECPSA is required to provide robustness to Destructive Damage (DD) of half the barcode. Also, further experiments are needed to determine the exact settings of ECPSA needed for different levels of DD. Future experiments are needed to quantify the effect of PS distortion and blurring on ECPSA; that is, how much ECPSA is "eaten up" by these other effects.

The deployment recommendations, in short, are based on the use of the barcodes. If DD is not allowed (e.g., if it is taken to imply tampering or damage of the barcode), these data imply that for 2D barcodes, increasing the size of the barcode modules is a better strategy than at least some forms of ECC (e.g., the Reed-Solomon ECC used in Aztec barcodes). If some DD is to be tolerated, then a minimum ECPSA (a recommended 15%) is needed simply to overcome the PS, RPQ, and/or blurring distortions. However, this requirement can be addressed by using larger module sizes, which is a more effective way of compensating for global, nondestructive errors. This conclusion implies that, for the types of barcodes we have tested, any ECPSA to be applied should be targeted at recovery from DD and can occupy less of the payload density, and module size increase emphasized as much as possible. Moreover, the types of error correction algorithms used should be reconsidered to target the types of errors most likely to occur during mobile capture.

Additional Advantages of Eschewing ECC

Freeing barcodes from the need for ECC provides a system security deployment advantage; namely, the barcodes can be scrambled. With ECC, only certain rearrangements of the

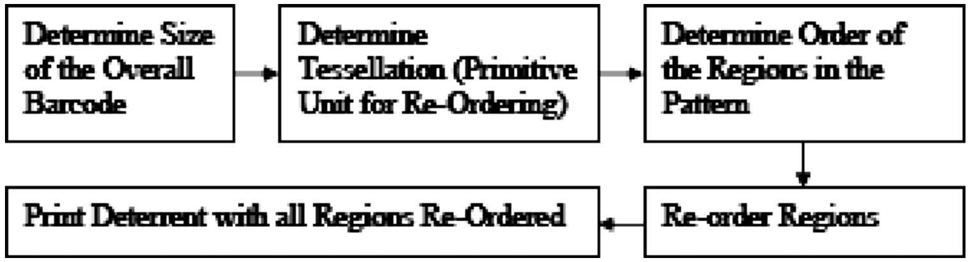

Figure 3. Outline of a barcode re-ordering (scrambling) approach that is possible when ECC is not used.

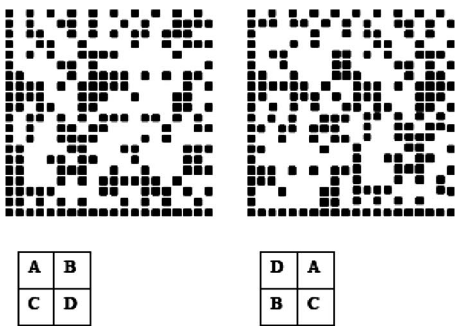

Figure 4. Example 2D (in this case, DataMatrix) barcode (left) and the 4 9×9 tile tessellation rectangles scrambled (right), rotated by reading order (top). Pattern of tessellation showing before and after arrangement of four large 9×9 tile sections A–D (bottom). The pattern on the left is readable using existing barcode reading hardware (e.g., the IDS-LDS barcode reader), while the pattern on the right is not readable.

A	E	A	E
B	F	B	F
C	G	C	G
D	H	D	H

G	B	G	B
C	D	C	D
F	H	F	H
E	A	E	A

Figure 5. Sample of 16-square tessellation of a square overall pattern, with the 8 equally sized fields rearranged and replicated.

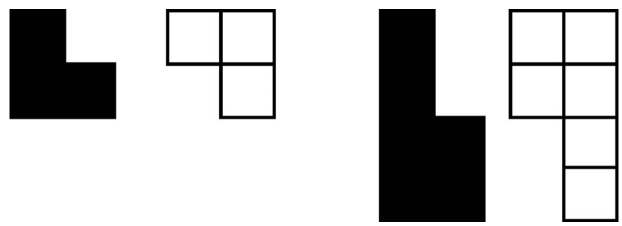

Figure 6. Two module high and four module high “L” tessellation pairs.

data in the barcode would be “decodable” due to the nature of the ECC algorithms—e.g., Reed-Solomon and other BCH codes, Gallagher codes, Hamming codes, etc., which depend on the structural arrangement of the black and white tiles in the barcode. Thus, when ECC use is eschewed, it is more difficult for a dishonest party to determine the scrambling approach.

The general outline of such an ECC-free scrambling approach is given in Fig. 3. First, the overall size of the barcode is determined. Then, the tessellation (manner in which the overall data area is subdivided into smaller, data-containing elements) is decided. The ordering of the data bits in each of these elements from the tessellation is then determined, and the deterrent generated and printed.

A straightforward tessellation pattern that can be chosen to map the 2D barcode is a square tessellation as shown in Fig. 4; however, any mapping that provides equal sized region tesselation will suffice. Examples include a right-side up “L” and upside down “L,” other rectangular tessellations, or combinations of different shapes of the same size—e.g., the Tetris set of seven different shapes comprised of four square tiles—so long as the shapes all fit together to create the overall pattern.

One alternative to ECC for robustness to damage is simple data replication. This can be combined with tessellation and scrambling as shown in Fig. 5. In this case, the replication effectively uses an ECPSA of 50%.

Rather than using the rectilinear (square) tessellations shown above, the actual tessellations can be indicated by smaller marks within the main marks of the barcode. For example, interwoven “L” patterns, one right side up and the next upside down, can be used to tessellate a plane. Similarly, alternating “I” and “−” patterns can be used to “map” the overall barcode. Many other tessellation patterns (including brickwalls, basketweave tiling, pixellated hexagons, and digital versions of Escher and Escher-like patterns) can be used—and any “spare” pixels on the sides of the tessellations can be used as check bits.

For example, with the tessellation of right side up and upside down “L” sections, additional bits can be used to define the size, starting condition (upside down/right side up) for each row, and “secondary payload portion” of each “L” (e.g., the long arm, short arm, elbow or combinations). As an enabling example, if the barcode is 24×24 modules, and “L”s are allowed to be 2 or 4 modules high (3 modules wide, as in Fig. 6), then a minimum number of bits to encode this deterrent are:

(1) one bit to distinguish between two and four module high implementation.
(2) six bits, for four module high (12 bits for two module high) “L”’s, to tell whether each pairing starts with an upside down or right side up “L,” multi-

plied by the eight pairs across each row. An additional bit per pairing is gained if we allow the "L"'s to rotate by 90°.

(3) Three bits for how extra data is represented in the "L" shapes (e.g., "1" or "0" for each of the three sections of the "L."

Using the above outlined approach, an 18×18 payload module barcode (e.g., as shown in Fig. 4) using two module high "L"'s contains $6 \times 9 = 54$ pairs of "L"'s, with each pair using up to 6 bits to define the secondary payload, and two bits to define the "L"'s. The overall number of bits for encoding can thus be as high as $54 \times (2+6) = 432$ bits, which is one-third greater than the 324 (that is, 18×18) bits in the barcode itself. This example shows that there are readily more means of representing the secondary payload than the number of bits in the primary payload when traditional ECC is eschewed.

It should be noted that we only investigated the Reed-Solomon ECC in this article. It is possible that the results would not be reproduced when other ECC codes, such as low-density parity check (LDPC), are employed. However, we did not have the option to use LDPC when creating Aztec barcodes, as they are not part of the standard.

Combined, the experimental results and scrambling approaches described in this article show that foregoing Reed-Solomon ECC appears to be advantageous for 2D Aztec barcodes. Additional payload modules are obtained for Aztec-compliant arrangements of the modules, with no observed diminution of robustness to localized damage, print-scan (PS) deterioration or blurring. In the case of scrambling, additional bits suitable for increased security protection are obtained.

REFERENCES

[1] http://www.gs1.org/productssolutions/mobile/

[2] http://www.epcglobalinc.org/home

[3] High Capacity Color Barcodes – Microsoft Research, http://research.microsoft.com/en-us/projects/hccb/default.aspx (last visited November 2009).

[4] http://www.openmobilealliance.org/

[5] Paperdisk, http://www.paperdisk.com/ibippap2.htm

[6] S. J. Simske, M. Sturgill, and J. S. Aronoff, "Effect of Copying and Restoration on Color Barcode Payload Density", *Proc. ACM Symposium on Document Engineering 2009* (DocEng 2009) (ACM, New York, 2009) pp. 127–130.

[7] D. Li, R. M. Mersereau, and S. J. Simske, "Blur Identification Based on Kurtosis Minimization", *Proc. IEEE ICIP* (IEEE, Piscataway, NJ, 2005) pp. 905–908.

[8] R. Villán, S. Voloshynovskiy, O. Koval, and T. Pun, "Multilevel 2D Bar Codes: Towards High Capacity Storage Modules for Multimedia Security and Management", IEEE Trans Information Forensics Security **1**(4), 405–420 (2006).

[9] D. Parikh and G. Jancke, "Localization and Segmentation of a 2D High Capacity Color Barcode", *WACV 2008* (IEEE, Copper Mountain, CO, 2008), pp. 1–6; (http://research.microsoft.com/en-us/projects/hccb/about.aspx).

[10] O. Bulan, V. Monga, and G. Sharma, "High Capacity Color Barcodes Using Dot Orientation and Color Separability", Proc. SPIE **7254**, 7254171.1–725417.7 (2009).

[11] S. J. Simske, J. S. Aronoff, M. M. Sturgill, and G. Golodetz, "Security Printing Deterrents: a Comparison of Thermal Ink Jet, Dry Electrophotographic, and Liquid Electrophotographic Printing", J. Imaging Sci. Technol. **52**(5), 050201.1–050201.7 (2008).

[12] ISO/IEC 24778:2008, Information Technology–Automatic identification and data capture techniques–Aztec Code bar code symbology specification, International Organization for Standardization, Geneva, Switzerland, http://www.iso.org/iso