Journal of Imaging Science and Technology® 51(1): 86–95, 2007.

Qualification of a Layered Security Print Deterrent[1]

Steven J. Simske and Jason S. Aronoff
Hewlett-Packard Laboratories, 3404 E. Harmony Rd., Mailstop 85, Fort Collins, CO 80528
E-mail: Steven.Simske@hp.com

Abstract. *Variable data printing (VDP), combined with precision registration of multiple ink layers, empowers a layered deterrent using variable print strategies on each of the multiple layers. This shifts the need for specialized printing techniques to the need to accommodate variable ink approaches. Such layered deterrents can incorporate infrared/ultraviolet fluorescent inks, infrared opaque and transparent black inks, inks containing taggants, magnetic ink, and inks with differential adhesive properties to enable sandwich printing. Overt features printed as part of the same layered deterrent provide excellent payload density in a small printed area. In this paper, the statistical and hardware processes involved in qualifying two layers of such a deterrent for their deployment in product (e.g., document and package) security are presented. The first is a multicolored tiling feature that provides overt security protection. Its color payload is authenticated automatically with a variety of handheld, desktop, and production scanners. The second security feature is covert and involves the underprinting or overprinting of infrared information with the covert tiles. Additional layers using existing security deterrents are also described, affording the user information densities as high as 560 bits/cm² (70 bytes/cm²).*

[DOI: 10.2352/J.ImagingSci.Technol.(2007)51:1(86)]

INTRODUCTION

Counterfeiting, smuggling, warranty fraud, production overruns, product diversion, and related problems are a huge concern for brand owners. Conservative estimates place counterfeiting alone at 5–7% of world trade, or more than $300 billion/annum.[1] Because the harmful effects of counterfeiting extend to entire economies and societies,[2] fighting counterfeiting not only protects a brand name but also can add to brand value if the company is perceived as an agent in product security. Counterfeiting in the pharmaceutical industry is enabled by the practice of relabeling and repackaging,[3] increasing the need for item-level authentication. The US Food and Drug Administration (FDA) has created a Medwatch[4] program to provide up-to-the-minute reporting of adverse events in the pharmaceutical distribution chain, emphasizing the ubiquity and severity of the counterfeiting.

To deter counterfeiters, a layered deterrent is recommended. This is a printed deterrent that contains two or more layers of information in a single region. Higher density of layered deterrents is provided when multiple layers of ink are precisely registered, such as is possible with liquid electrophotographic (LEP) digital press technologies.

Product security begins with the package. If each package provides a unique identifier, which can be tracked and linked to a provenance record tracing its location throughout its distribution path, then even a modest level of customer/retailer authentication poses a significant exposure risk to a would-be counterfeiter.[5] The incentive for package reuse is also removed. Using this approach, the packages should provide overt security printing features that can be authenticated simply (e.g., with camera phones, digital cameras, scanners, and all-in-ones) and reliably. This approach will always be complemented by complex deterrents (color-shifting inks, layered deterrents,[6] etc.), electronic and active deterrents (RFID, etc.), tamper-evident deterrents, and other registry-based deterrents. Under some circumstances, a unique identifier can provide a level of security dictated by its density—the amount of information that can be reliably read using the deterrent. For this to happen, it must be reliably authenticated.

In this paper, two deterrents are considered (and salient portions of them qualified). The first is a 2D arrangement of color tiles,[7,8] which can provide branded colors, product-special colors, and/or be part of an overt deterrent. These color tiles can in turn be associated with overprinted microtext. Figure 1 demonstrates this feature in its two default deployments: without superimposed microtext (upper) and with superimposed microtext (lower). The upper color tile feature can also accommodate hidden ultraviolet/infrared (UV/IR) inks, as described below—or overprinted UV/IR inks—for additional, covert security. In Fig. 1, the default deployment of the upper feature is expanded to twice its size relative to the lower feature (the addition of microtext requires roughly a 2× increase in tile width and height to authenticate accurately).

Thirty-six characters (the 26 English letters A–Z and the 10 numerals 0–9) are associated with two consecutive color tiles (each taking on one of six possible colors—thus, the 36 characters are encoded exactly by 6×6 color combinations) in English reading order (left to right by row, top to bottom by consecutive rows). The color pairs mapped to these are $A=(R,R)$, $B=(R,G)$, $C=(R,B)$, $D=(R,C)$, $E=(R,M)$, $F=(R,Y)$, $G=(G,G)\ldots$, $9=(Y,Y)$, where *RGBCMY* are the colors red, green, blue, cyan, magenta, and yellow, respectively. Note that, for example, the letter "N" is always encoded as a blue followed by a green tile in the feature on the

[1]Presented in part at IS&T's Digital Fabrication Conference, Baltimore, MD, September, 2005.

Received Jan. 25, 2006; accepted for publication Aug. 15, 2006.
1062-3701/2007/51(1)/86/10/$20.00.

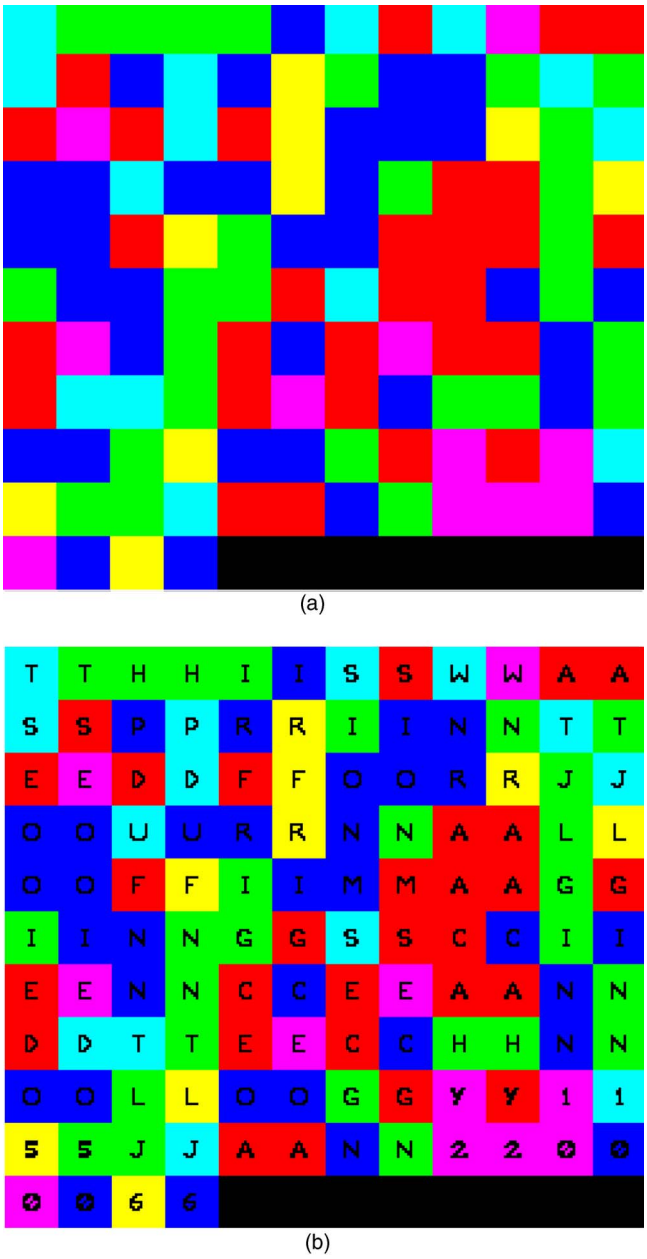

Figure 1. Color tile security printing feature in default deployment, without microtext (upper) and with microtext (lower). The upper feature is expanded to twice its size relative to the lower feature, as necessary for accurate authentication (the addition of microtext requires approximately a twofold increase in tile width and height to authenticate accurately). The letters A–Z and numerals 0–9 are associated with two consecutive color tiles in European reading order. The color pairs mapped to these are A $=(R,R)$, B$=(R,G)$, C$=(R,B)$, D$=(R,C)$, E$=(R,M)$, . . . , 9$=(Y,Y)$, where *RGBCMY* are the colors red, green, blue, cyan, magenta and yellow, respectively. Note that, for example, the letter "P" is always encoded as a blue followed by a cyan tile in the feature on the right above.

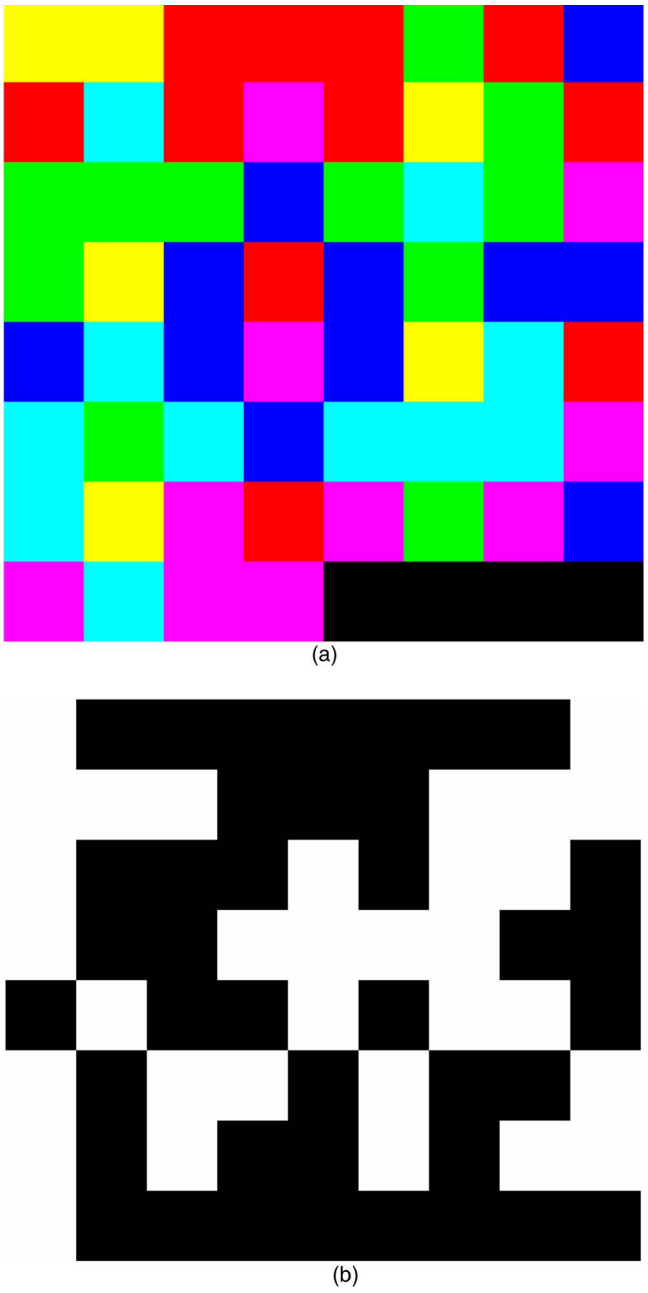

Figure 2. Security printing features: color tile (upper) and binary tile (lower) for testing differential IR-opaque inks. For the qualification described herein, the color tile feature was printed using CMY (cyan, magenta, yellow) inks, and the binary tile feature with spot color blue (C6170A) ink.

right in Fig. 1. Both features encode the string "THISWASPRINTEDFORJOURNALOFIMAGINGSCIENCEANDTECHNOLOGY15JAN2006."

The second layer (Fig. 2) is a binary covert tile produced by one of two approaches. The first approach is the combination of an infrared (IR) reflective ink layer overprinted by two types of black (or other spot color) ink,[9,10] making it appear to be a uniform (spot) colored area, but encoding a covert tile structure. This feature is produced using inks that have differential opacity to visible and infrared light excitation. In offset and other "static printing" technologies, process black ink can be used as the ink with opaque IR characteristics, and Anoto black ink[11] as the ink with transparent IR characteristics. Using a variable data printing front end, one can simply select between the two spot color inks and decide which sections of underprinted infrared ink to reveal. The second approach to providing a layered deterrent is to

simply overprint IR tile patterns on a color tile deterrent such as shown in Fig. 1. The second approach was simulated here with a blue ink tile.

SECURITY PRINTING FEATURE QUALIFICATION

To qualify a feature, the following steps are required

a. Design the feature. This includes specifying the variables in the feature and the ranges over which they should be varied. On the low end of the range, the feature should essentially never authenticate (or authenticate below any acceptable accuracy), whereas on the high end, the feature should authenticate at an acceptable level. For the color tile feature, the variables include (i) the set of colors printed, (ii) the width and height of, and thus number of bits in, the feature, (iii) the inclusion/exclusion of (visible) microtext, and (iv) the width and height of the tiles.

   For the binary tile, the variables include (i) the spectral characteristics of the inks used, (ii) the width and height of, and thus number of bits in, the feature, and (iii) the width and height of the tiles.

b. Determine the set of features to print. Based on the above set of variables, for the color tiles (i) the set of colors printed was {*RGBCMY*}, (ii) an 8×8 array of tiles was printed with at least six of each color, (iii) microtext was not printed visibly over the color tiles, and (iv) the width and height are equal and are varied from 0.125 to 1.25 mm (in 0.125 mm increments).

   For the binary tiles, (i) a single ink was selected to print, HP C6170A spot color blue ink, (ii) an 8×9 array of tiles was printed with 32 white spaces and 40 black spaces (including 8 black spaces on the lowest row, as in Fig. 2), and (iii) the width and height are equal and are varied from 0.125 to 1.25 mm in 0.042 mm increments.

c. Print the set of features. Thirty-six color tile features were printed at each of ten sizes, at 600 ppi. For purposes of testing, multiple security printing features are written to each letter-sized ($11'' \times 8.5''$) page, as shown in Fig. 3. A total of 360 (36 each at 0.125, 0.25, …, 1.25 mm in dimension) color tile features, each with 60 colored tiles (21 600 total tiles), were printed. The final four black tiles on the color tile features are ignored by the authentication algorithm. The color tile features were printed on a thermal inkjet printer at 600 dots per inch (dpi), or 240 dots/cm, resolution using default settings except for selecting "high quality."

   A total of 16 binary tile features were printed at resolutions of 0.125, 0.167, …, 1.25 mm (28 different sizes, 16 binary tiles each, 72 tiles each, for a total of 32 256 tiles). A sample page for these tile features is shown in Fig. 3. The binary tile features were printed on a thermal ink jet printer at 600 dpi (240 dots/cm) using default settings, except that the color cartridge was disabled (so only the blue ink printed) and "high quality" was selected.

   Each color tile sequence used 30 of the 36 characters in the set, and each character appeared in 30 of the 36 samples at each resolution (the same set of 36 features was printed at each resolution). Each binary tile included the 16 4-bit subsequences (0000, 0001, …, 1111), and once more the same set of 16 features was printed at each resolution.

d. Scan the pages of the features. The printed pages were all scanned using a commercial off-the-shelf desktop scanner (the pages were placed manually on the scanner, so that the automatic document feeder was not used) at 600 pixels per inch (ppi), or 240 dots/cm, using default settings, and stored with lossless compression. To accommodate all the features, 29 pages of color tiles and 37 pages of binary tiles were scanned.

e. Extract the features from the printed pages. A segmentation algorithm[12,13] was used to extract each feature automatically from the scanned page of multiple features. Where possible, whitespace was included around the feature. After this step, the 360 color tile features and 448 binary tile features are saved as individual image files.

f. Authenticate the features. The set of extracted features is then evaluated using the authentication algorithms described below. The output of the authentication algorithm is a sequence that can be directly compared to the intended sequence. The number of loci (single tile reading) errors is calculated for each feature.

g. Determine critical point in the authentication curves. Curves are then obtained showing the number and percentage of tiles read successfully along with the absolute number of tiles correctly read. From these data, one can recommend the security feature deployment parameters (size, in the case of the features tested herein). The error rate is used to define how many check bits, redundant bits, etc., must be added to prevent read errors.

AUTHENTICATION

Color tile authentication consists of the following steps, all of which are embedded in a single executable that performs near-real time analysis of an image:

a. Thresholding. Thresholding is performed on the saturation values of the scanned pixels, since the *Y* tiles have similar intensity values to white, and the six colors cover much of the hue gamut. Saturation is defined as

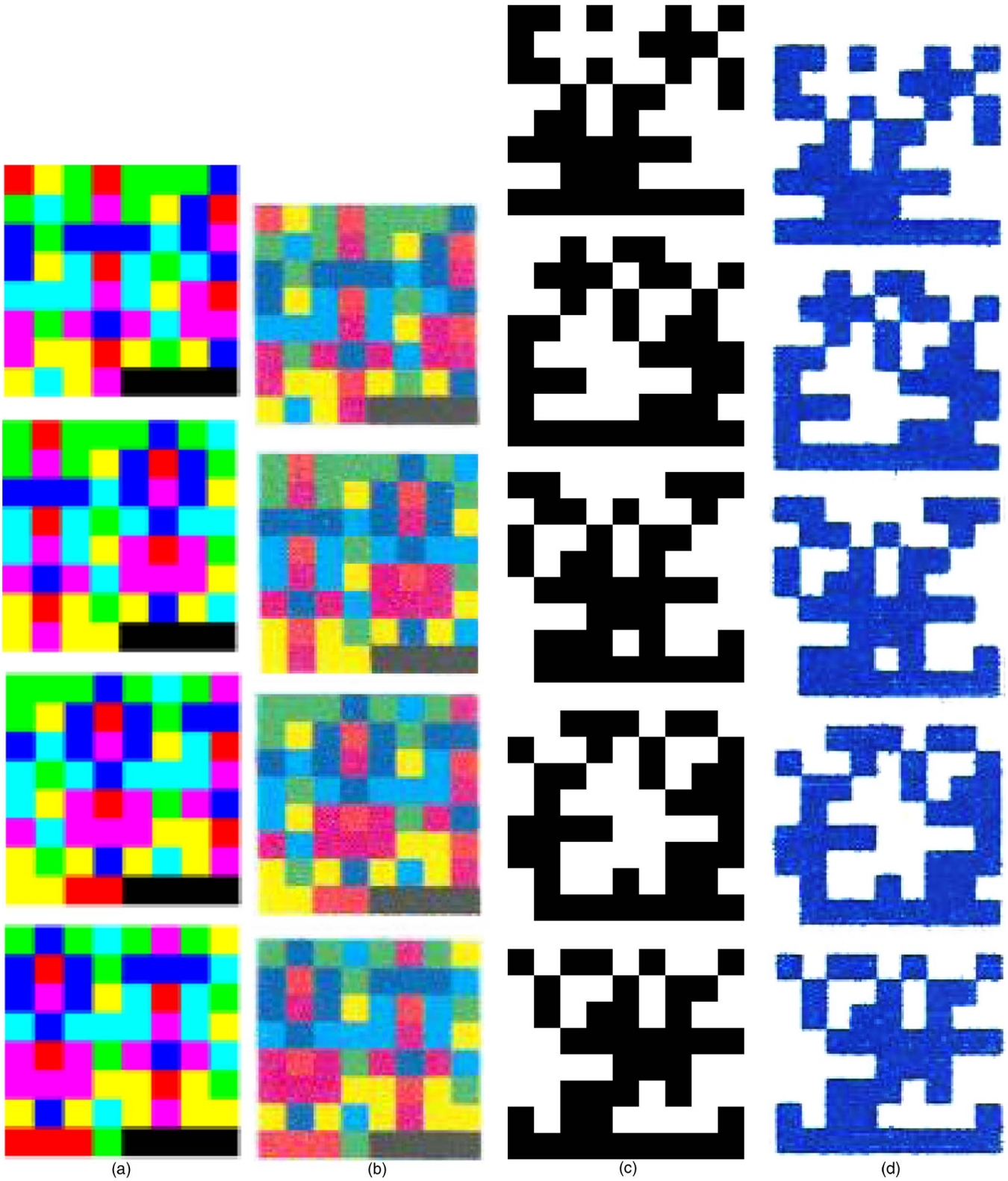

Figure 3. Sample pages printed for qualification: color tiles [pair on left side, columns (a) and (b)] and binary tiles [pair on right side, columns (c) and (d)]. For each pair, the raster files to be printed are on the left, and the scanned pages are on the right. For the binary tiles, the print raster is binary (black and white), and the C6170A spot color blue ink was printed and scanned using the "black ink" cartridge in the thermal ink jet printer. Both sets represent the largest dimension tested (1.25×1.25 mm tiles).

$$\text{Saturation} = [255 * (1 - \min(R,G,B) / \text{sum}(R,G,B))]. \quad (1)$$

The threshold value is determined from the moving average-smoothed saturation histogram and is the minimum point of the saturation histogram above the peaks for black and white (which usually overlap) and the next peak (typically for blue).

b. Segmentation. The resulting thresholded image is then prepared for segmentation with a sequence of thinning (to eliminate speckle noise), fattening (to return nonerased regions to their original size), and run-length smearing (to prevent gaps in features). These preparatory steps are well known for 2-D segmentation, extending back 25 years.[14] Because we are looking for nontext regions, default segmentation preparation as described in Ref. 14 is used: we then filter out the regions formed based on size and aspect ratio (and later histograms) to locate the tile features. Next, regions are formed, and the set matching the expected size of the security printing features is identified and outlined. The processing to this point requires less than 0.5 s on a mid-range laptop computer for a 10×15 cm^2 image, suitable for authentication of a single package or document. For a full page (e.g., 20×25 cm^2 cropped image), the same mid-range (2 GHz processor clock, 512 MB RAM) laptop requires approximately 3 s of processing time.

c. Subsegmentation. These regions are extracted to individual files and corrected for skew, if present. The features are then sliced into eight columns and eight rows (per the specification of the features as 8×8 tiles in size) and these 64 regions assigned in reading order. The four black tiles at the end are used to make sure they are oriented properly, and then discarded to leave a 60 tile sequence. Because these images are now the size of the deterrent itself, the subsequent steps are performed very rapidly (a much smaller image is processed much more quickly), generally in less than 10 ms, for example, on a mid-range laptop.

d. Find color peaks. The (CMY) color peaks are found first. Separate *C*, *M*, and *Y* maps the same size as the feature are created and the values for *C*, *M*, and *Y* calculated as

$$C = B + G - R,$$

$$M = B + R - G,$$

$$Y = G + R - B.$$

Each of these maps is histogrammed, and the largest peak above the midpoint (255 for an 8–bit/channel or 24–bit image) of the range (0–511 for 24-bit image) of the histogram is defined as the *C*, *M*, or *Y* peak in each of these maps. The median value in the peak is taken as the representative value for each of these three colors. The pixels assigned to any of these three peaks are then ignored (not added to the histograms) when the (*RGB*) color peaks are defined.

The values for *R*, *G*, and *B* are calculated as

$$R = 255 + R - B - G,$$

$$G = 255 + G - B - R,$$

$$B = 255 + B - R - G.$$

The pixels not assigned to *C*, *M*, or *Y* peaks in the previous step are now histogrammed. Here, the largest peak above the midpoint of the range of the histogram is defined as the *R*, *G*, or *B* peak. The median value in the peak is taken as the representative value for each of these three colors.

e. Assign color value to every pixel in the feature. Next, the distance from the defined (median) value of each peak is computed for every pixel, and each pixel is assigned a color value corresponding to the minimum distance (Fig. 4, middle image).

f. Assign color value to every tile in the feature. For each tile region, the number of pixels assigned to each color is summed, and the color with the maximum value is assigned to the tile (Fig. 4, right image). Ambiguous tiles (wherein the color with the maximum value is assigned less than half the pixels) are reported.

g. Report tile sequence. The 60 tile sequence is organized into 30 consecutive pairs. These 30 pairs of tiles are decoded into a 30 character string which is then compared to the intended sequence. Errors are listed as single or dual tile errors (the latter counts as two "errors").

Figure 4 shows the effects of these steps on a scanned color tile feature. The output of the authentication is the sequence as follows, which is directly compared to the printed sequence (in this case, there is no error): "FGHIJKLMNOPQRSTUVWXYZ012345678"

The steps for authenticating binary tiles are similar to that for color tiles, though in general simpler:

a. Thresholding. Thresholding is again performed on the saturation values of the scanned pixels. We chose a blue ink to provide the "most challenging" thresholding test of the set {*RGBCMY*} (blue ink had the lowest saturation peak of these six peaks). The threshold value is again determined from the moving average-smoothed saturation histogram.

b. Segmentation. Segmentation is performed as for color tiles.

c. Subsegmentation. These regions are extracted to

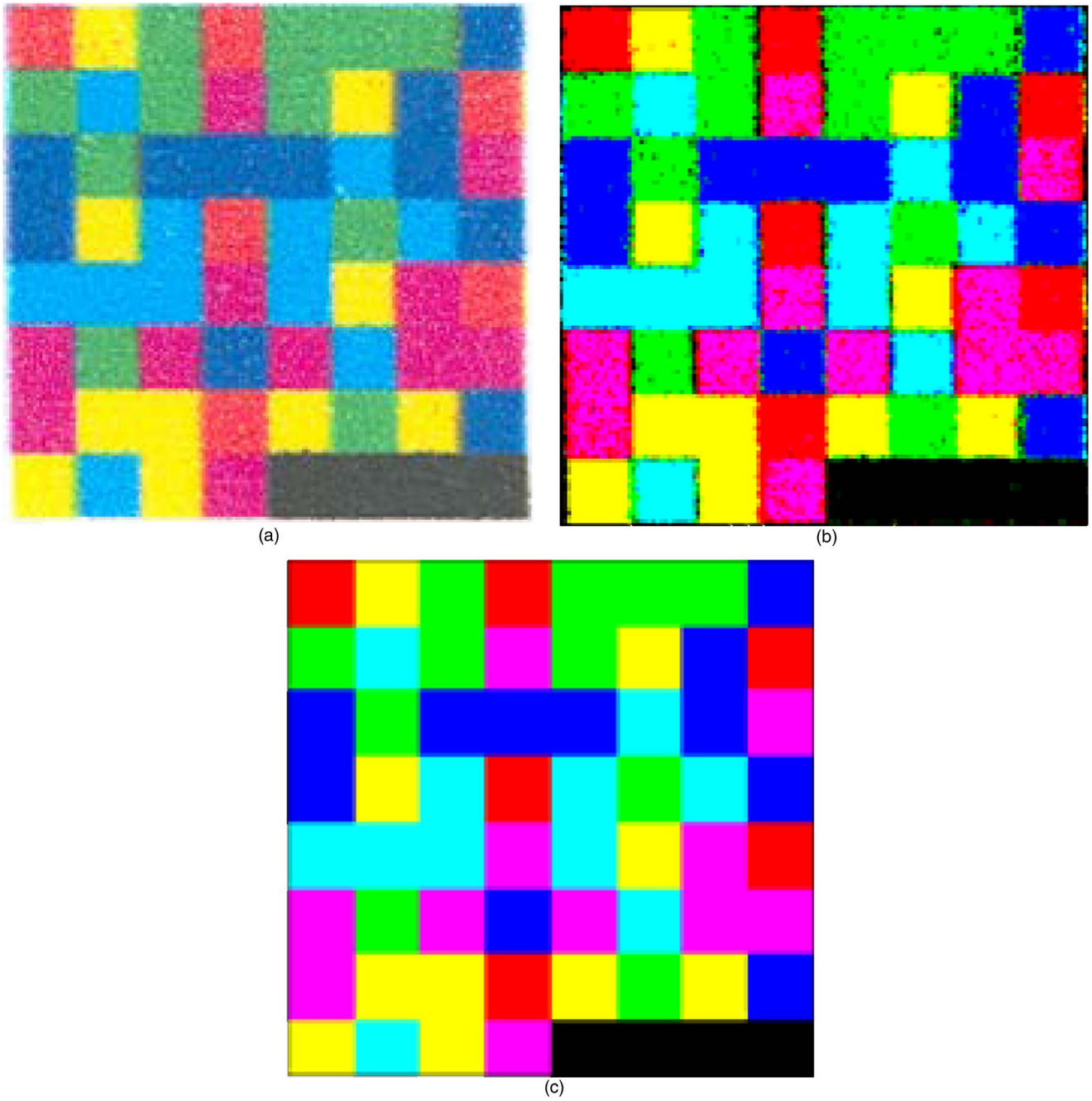

Figure 4. (a) Sample color tile feature after being segmented and extracted from the top of column (b) of Fig. 3. (b) White and black pixels assigned to black and individual pixels assigned to one of the color set {*RGBCMY*}. (c) Subsegmentation of the color tile feature and the color assignment of each tile.

individual files and corrected for skew, if present. The features are then sliced into nine columns and eight rows (per the specification of the features as 9×8 tiles in size) and these 72 regions assigned in reading order. The eight consecutive black tiles at the end are used to make sure they are oriented properly, and then discarded to leave a 64-tile sequence.

d. Assign foreground/background value to every pixel in the feature. For each tile region, the number of pixels assigned to foreground (blue) is summed, and if this number is greater than the number assigned to background (white), then the tile is assigned to "foreground" ("1" in the sequence). Otherwise the tile is assigned to "background" ("0").

e. Report tile sequence. The 64 tile sequence is recorded, which can then be compared to the intended sequence.

Table I. Results for color tile qualification. Read failures correspond to features with insufficient color saturation. The number of correct reads is out of a possible 2160 total color tiles read at each tile size.

Tile dimension (mm)	Read failures (%)	Errorless reads (%)	Tile error rate (%)	Correct reads (no.)
0.13	100.0	0.0	100.0	0
0.25	69.4	0.0	93.64	42
0.38	80.6	0.0	43.33	238
0.50	52.8	33.3	6.18	957
0.63	69.4	11.1	7.58	610
0.75	50.0	2.8	11.02	961
0.88	8.3	16.7	5.05	1880
1.00	2.8	30.6	2.48	2048
1.13	2.8	61.1	1.90	2060
1.25	0.0	75.0	0.74	2144

QUALIFICATION

There were several types of errors in reading the color tiles (Table I). The first was due to features having insufficient color saturation (Table I, second column from left), in which the scanned feature had insufficient consistency of saturation of the colors to segment as a single region (due to low saturation pixels being assigned to the "black" and "white" pixel category). Figure 5 illustrates several examples of these features (these are 0.25×0.25 mm^2 tiles). Halftoning likely contributed to this phenomenon, since the "additive" colors (*RGB*) fared more poorly than the subtractive colors (*CMY*). The latter correspond more exactly with the ink pigment colors, and so are less affected by halftoning. Features that segmented incorrectly were simply registered as "read failures," and these occurred for tile dimensions up to 1.125×1.125 mm^2.

The second type of failure was an incorrect color assignment for a (properly) segmented tile. This is reported as the "tile error rate" (Table I, fourth column from left). This value dropped to 6.2% at a tile size of 0.50×0.50 mm^2, then increased again, dropping to 5.0% at 0.88×0.88 mm^2. This nonlinear behavior for tiles from 0.50 to 0.88 mm in dimension may simply be an artifact of the small number of pages scanned. If not, it is likely a consequence of the automatic subsegmentation approach of the simple authentication algorithm deployed for the qualification work presented here. Regardless, by the time the tiles were 1.25 mm on a side, read failures had dropped to zero, 75% of the features were read without a single tile error, and the overall tile error rate was less than 1%. Thus, individual tile reading accuracy surpassed 99% at this size (Fig. 6).

The graph for binary tile errors (Fig. 7) was relatively well behaved. The smallest two sizes (3 and 4 pixels, or 0.125 and 0.167 mm, on a side for each tile) were essentially unreadable, with error rates of ~50%. By 0.208 mm on a side (Fig. 7), however, the tiles were readily readable, with an error rate just over 10%. The error rate dropped below 1% by the time the binary tiles reached 0.63×0.63 mm^2 in size.

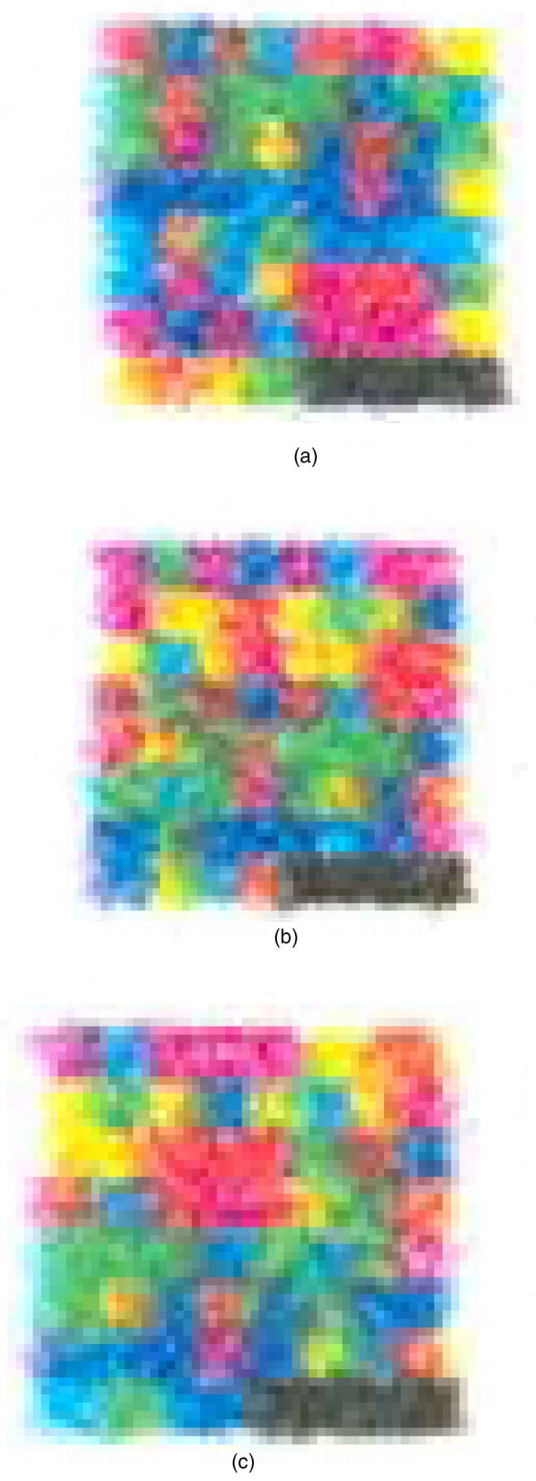

Figure 5. Color tile patterns (0.25×0.25 mm in size) with low print quality. Many of the pixels in the colored (*RGBCMY*) areas of these features are closer in saturation terms to the black peak than to the color peaks. Even when these lower-resolution features are segmented correctly, there is a high tile reading error rate (Table I, Fig. 6).

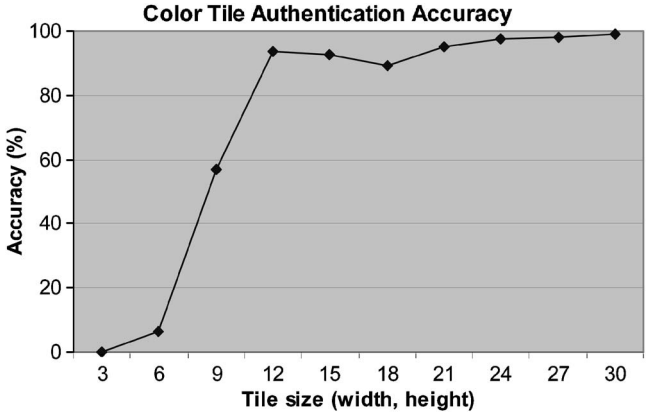

Figure 6. Color tile authentication accuracy as a function of tile size. 99% accuracy is achieved by 30 pixels (at 600 ppi, or 240 dots/cm), or 1.25×1.25 mm in width×height. Tiles are squares ranging from 0.125 (3 pixels) to 1.25 (30 pixels) mm in size.

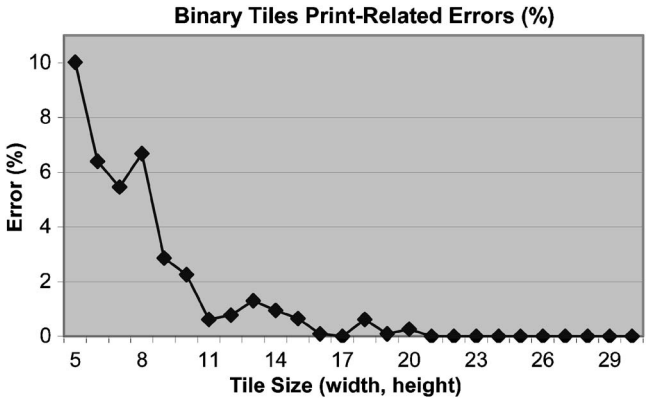

Figure 7. Binary tile authentication error rate (100%-accuracy) as a function of tile size. 99% accuracy is achieved by 15 pixels (at 600 ppi, or 240 dots/cm), or 0.625 mm in width/height.

DISCUSSION

Performing the qualification of a security printing feature is important to ensure that customers retailers, and/or field investigators will willingly and consistently perform authentication. Of course, this is not simply a technical issue. An important means for encouraging compliance is to put in place convenient systems for gracefully handling exceptions (read failures, periodic authentication, etc.). Another means of improving compliance is to largely eliminate "read failures," which, for example, argues for a 1.25×1.25 mm^2 color tile for the hardware used in this qualification study. The output of qualification is a recommendation for the deployment of the feature: its size and density (e.g., how many tiles to use and how large the tiles are), the printing and reading/scanner hardware to be used, and the purpose of the feature. The latter point was not addressed directly in this paper, but is directly related to an accuracy curve such as that shown in Fig. 6. If the color tile size is selected to be "just beyond" the knee of the curve (e.g., a 12×12 pixel, or 0.5×0.5 mm^2, color tile is chosen), then the feature can provide an anticopying deterrence in addition to the security of the sequence itself. If, on the other hand, the size is made as large as possible to prevent any "read failures," then a counterfeiter may be able to more readily copy a batch of features. (Since copying degrades the features, it will effectively move the feature further toward the "knee" of the authentication accuracy curve, but the greater reliability of a large tile will prevent a large increase in read failures.) Thus, smaller tiles perform a function more like that of a copy detection pattern[15] (that is, covert), while larger tiles perform a function more like that of a bar code (that is, overt). It is important to note that even if a counterfeiter can successfully copy an overt feature, the presence of a secure (database) registry for polling with the for-authentication sequences will always discourage wholesale counterfeiting (so long as the codes are actually routinely verified by the end user—customer, retailer, and/or field inspector).

Performing the qualification is also an excellent means of evaluating the effectiveness of the authentication system one is planning to use with a product. In performing the experiments above, for instance, it was observed that for tile-based deterrents, there are at least two distinct, broad classes of errors made during authentication. The first class of errors, which are highly dependent on the size of the tiles, and thus follows a classic "S curve" such as that shown in Figs. 6 and 7, are broadly termed "printing errors." These errors, which are manifest at sizes larger than the individual printing dots, are addressed through improving the printing technique (e.g., by changing the hardware, such as using a device with more precise ink placement) or approach (e.g., by eliminating halftoning through the use of six spot color inks for the color tiles), or by changing the ink itself (this is not an easy prospect, since ink chemistry is constrained by the physics of the printing), with varying improvements. It should be noted that these print errors (smearing, blotching, etc.), if uncorrected, prevent any increased deterrent density through magnification.

The second type of error is the error associated with the authentication algorithm itself. For the experiments described, relatively simple authentication approaches were adopted. Because of this, we were able to make on-the-fly changes to these algorithms to reduce the overall error rate. For example, during the performance of the binary tile authentication, we noted that occasionally the authentication algorithm would crop the 9×8 feature to effectively an 8×8 feature. This resulted in infrequent occurrences of a significant misread of a feature because the algorithm was attempting to impose a 9×8 structure on an 8×8 matrix. Increasing the size of the gap smeared by the run-length smearing eliminated this algorithm error. As a second example, during the performance of the color tile feature authentication, we noticed that finding the subtractive (*CMY*) color peaks first reduced the overall error rate considerably in comparison to finding the additive (*RGB*) color peaks first.

Feature qualification focuses on the different aspects of

the security printing feature to which the overall authentication process is sensitive. The size of the feature, as shown here, is clearly an important (perhaps the most important) factor. However, many other factors are important to consider, including the device independence of the authentication. Any off-the-shelf version of the scanning hardware used for qualification work should perform as well as the one used during qualification. Other factors include control over the printing process (for example, being able to reduce the effects of halftoning significantly improve color tile authentication accuracy), the ability to match the printing and scanning resolutions (or at least have them be integral multiples of each other), and the processing available for authentication. For example, if processing power is unlimited, then it is advantageous to put much more intelligence into the authentication algorithm, including the ability to respond adaptively to ink- and other print-related problems that might otherwise contribute to tile read errors. One of the principal purposes of qualification is to determine where to focus one's energies—on the printing, the scanning, or the authentication.

Based on the results, the color tile feature can be deployed using relatively inexpensive thermal ink jet printers and desktop scanners for production and authentication, respectively, with a bit density of $\sim$160 bits/cm^2. The binary tile feature can be deployed at $\sim$250 bits/cm^2. These densities assume that a tile read accuracy of $\geqslant$99% is acceptable. More generally, however, these bits will be incorporated into an overall deterrent, which includes positioning outline (akin to those on a 2-D DataMatrix barcode, for example[16]) and error code checking such as the Reed-Solomon algorithm.[17] The final density of these tile-based deterrents, then, will be on the order of 100 bits/cm^2 using the authentication equipment described herein.

The qualification work is used to recommend a deployment size and parameter definition. It is also used to define how many check bits, redundant bits, etc. must be added to prevent read errors. For example, at a bit density of 160 bits/cm^2, 25% of the color tiles will suffer at least one tile classification error. This means that the true "as-deployed" density of the tile feature will be reduced to incorporate error checking tiles. There is a trade-off between reducing the size of the tiles (which increases the tile error rate) and needing to incorporate more color tiles to provide error checking. An ideal tile-based security printing feature reaches a consistent low error rate above a certain size, allowing the error-checking approach to be reliably deployed. In addition, magnification can be used to increase the density, though with exacerbation of any print defects (see Fig. 5).

Additionally, the overall "ecosystem" in which the tile-based security printing features are to be deployed affects the selection of parameters in the features. For example, if the raw sequences encoded in the color tiles are stored in a (sparse) registry such that the odds of a random sequence being in the registry are quite low,[5] then low error rates in the color tiles can be overcome by using a pattern matching approach (such as that employed in bioinformatics) to find the best fit in the registry to the (mis-)reported sequence. Sequences with too high a number of errors can either be rejected as counterfeit, or trigger an event asking the user to rescan the feature. Alternatively, if a large volume of deterrents are being scanned simultaneously, the packages with "read failures" can be manually authenticated, authenticated with a more sophisticated scanner, or simply ignored (due to the rest of the deterrents successfully authenticating), depending on the needs and governance rules for the product and its authentication.

In-house and externally developed security printing features are fully qualified using the processes described herein. Printing and scanning are performed with the exact hardware to be used by consumers, retailers, and field inspectors. In most cases, this will require a plurality of scanning hardware; for example, a camera phone or PDA-like device for consumers, handheld scanners for retailers, desktop scanners for field inspectors, and a vision system for forensic investigators. Additional authentication hardware may be qualified for use on the production line (where the features may be read and registered in a secure database).

While more advanced authentication algorithms are being developed, it should be noted that this was not the purpose of this paper. The purpose was to use extremely simple authentication algorithms and inexpensive hardware for authentication, and demonstrate how high density security deterrents can be created through layering. The deployment recommendations are to use 1.25×1.25 mm^2 color tiles with an appropriate error-code checking (ECC) algorithm (e.g., Ref. 17), and to use 0.63×0.63 mm^2 binary tiles, also with an appropriate ECC algorithm. However, before deploying these security printing features, we would also perform a large set of qualification tests at and near the deployment size. This is necessary to predict more tightly the actual deployment error rate. Typically, one will perform many (hundreds or thousands) of tests at this more restricted range (e.g., at 1.1, 1.2, 1.3, and 1.4 mm dimensions for the color tiles), using multiple pieces of printing and scanning hardware.

It is worth noting that feature density is not the only consideration in choosing between color and binary tiles. Color tiles provide a more difficult to reproduce look and feel, and may also "degrade" more quickly near the deployment tile size than binary tiles (as evidenced by a better "S"-shape in Fig. 6 when compared to Fig. 7). Moreover, color tiles can be "pretreated" for color space shifts that occur between the printing and scanning processes. For example, if the red and magenta tiles are found to be difficult to distinguish during authentication, additional blue may be added to the magenta and/or additional yellow may be added to the red. Additional color combinations can also be tested with the qualification protocol described here. In this way, color can be used to optimize the density of information encoded.

The color tile features, in addition, are a means of fulfilling FDA recommendations for overt, covert, and forensic

anticounterfeit technologies.[18] Clearly, the visible color patterns are an overt feature and can be used for branding in addition to product track and trace and authentication. The text encoded in the sequence of tiles provides a covert deterrent (visible, but not generally intelligible). The microtext superimposed on the color tiles, if deployed, can offer a forensic-level feature because the microtext fonts themselves can be varied with an astronomical number of combinations[8] that must be hand-authenticated.

In addition to the color tile and binary tile qualification,[19] the use of multiple layers was considered. Because this "sandwich printing" feature is commercially available, it does not need qualification. On the HP Indigo digital press, sandwich printing is used for a variety of applications, one of which is a peel-off label.[10] Sandwich printing is possible due to this press' ability to print as many as 16 layers of ink on a substrate in a single pass (or "shot") with perfect registration. The "sandwich" refers to the "front" design, the "back" design, and the opaque layer (the "cheese" of the sandwich, usually white ink) between them. When a transparent substrate is used for this layered design, there are two images created, each one visible from one side of the substrate. The opaque layer separates these two images.

The layers of (usually white) ink between the ink layers for the two images serve two purposes: they provide the side which is currently viewed with a white underground and they hide the layer (against the substrate) that is behind. While the LEP ink (ElectroInk) is not opaque, it has roughly the transparency of an intentionally transparent screen printing ink. Thus, for it to block light between the two images in the layers of the sandwich, it must be applied in multiple layers. This is achieved through providing a separation in the print job for the opaque ink (usually white ink). Using sandwich printing, two layers of tiles, one for overt protection and the underlying second set for covert protection, can be layered, or "sandwiched," over the same area on a package or document. This doubles the byte density possible for the layered deterrent. With sandwich printing, the layered deterrent described here can provide more than 3600 bits/in^2, or $\sim$560 bits/cm^2, of information. Thus, 1.8 cm^2 is required to provide 1024 bit security identifiers, which can be authenticated with inexpensive, commercially available scanners (without magnification).

ACKNOWLEDGMENTS

The authors gratefully acknowledge Jordi Arnabat, David Auter, Dan Briley, Maureen Brock, Carlos Martinez, Philippe Mücher, Andrew Page, Henry Sang, Eddie Torres, Juan Carlos Villa, and other colleagues for their assistance with aspects of this work.

REFERENCES

[1] International Chamber of Commerce, Counterfeiting Intelligence Bureau, *Countering Counterfeiting* (ICC Publishing SA, Paris, France, 1997).
[2] D. M. Hopkins, L. T. Kontnik, and M. T. Turnage, *Counterfeiting Exposed* (Wiley, Hoboken, NJ, 2003).
[3] K. Eban, *Dangerous Doses* (Harcourt, Orlando, FL, 2005).
[4] U.S. Food and Drug Administration, Medwatch, the FDA Safety Information and Adverse Event Reporting Program, website, http://www.fda.gov/medwatch/.
[5] R. G. Johnston, "An anti-counterfeiting strategy using numeric tokens,"Int. J. Pharmaceutical Medicine (in press), also posted at: http://verifybrand.com/pdf/Drug_Anti-Counterfeiting_2004.pdf.
[6] S. J. Simske and R. Falcon, "Variable data security printing and the layered deterrent", *DigiFab 2005* (IS&T, Springfield, VA, 2005) pp. 124–127.
[7] S. J. Simske and D. Auter, "A secure printing method to thwart counterfeiting", HP Docket No. 200407401, filed with the US Patent and Trademark Office March 9, 2005.
[8] S. J. Simske, D. Auter, A. Page, and E. Torres, "A secure printing feature for document authentication", HP Docket No. 200500190, filed with the US Patent and Trademark Office August 1, 2005.
[9] S. J. Simske, L. Ortiz, M. Mesarina, V. Deolalikar, C. Brignone, and G. Oget, "Ink coatings for identifying objects", HP Docket No. 200405356, filed with the US Patent and Trademark Office October 12, 2004.
[10] S. J. Simske, P. Mücher, and C. Martinez, "Using variable data security printing to provide customized package protection", *Proc. IS&T's DPP2005* (IS&T, Springfield, VA, 2005) pp. 112–113.
[11] Anoto substitute black ink, SunChemical AB, P.O. Box 70, Bromstensvagen 152, SE-163 91 SPANGA Sweden.
[12] S. J. Simske, "Low resolution photo/drawing classification: Metrics, method and archiving optimization", *Proc. ICIP 05* (IEEE, Piscataway, NJ, 2005).
[13] S. J. Simske, D. Li, and J. Aronoff, "A statistical method for binary classification of images", *DocEng 2005* (ACM, New York, NY, 2005) pp. 127–129.
[14] F. M. Wahl, K. Y. Wong, and R. G. Casey, "Block segmentation and text extraction in mixed/image documents", Comput. Vis. Graph. Image Process. **20**, 375–390 (1982).
[15] J. Picard, C. Vielhauer, and N. Thorwirth, "Towards fraud-proof ID documents using multiple data hiding technologies and biometrics, "Proc. SPIE /ISSN 0-8194-5209-2, 416–427 (2004).
[16] Data Matrix, http://en.wikipedia.org/wiki/Data_Matrix.
[17] Reed-Solomon error correction, http://en.wikipedia.org/wiki/Reed-Solomon_error_correction.
[18] FDA Counterfeit Drug Task Force Interim Report, U.S. Department of Health and Human Services, FDA, also posted at: http://www.fda.gov/oc/initiatives/counterfeit/report/interim_report.pdf, 46 pp., 2003.
[19] S. J. Simske, J. S. Aronoff, and J. Arnabat, "Qualification of security printing features", Proc. SPIE in press.