Photoplus: Auxiliary Information for Printed Images Based on Distributed Source Coding

Ramin Samadani and Debargha Mukherjee
Media Technologies Lab
HP Laboratories Palo Alto
HPL-2008-2
January 11, 2008*

Image Registration, Color Correction, Distributed Source Coding, Printed Image Archival, Print Scan Channel

A printed photograph is difficult to reuse because the digital information that generated the print may no longer be available. This paper describes a mechanism for approximating the original digital image by combining a scan of the printed photograph with small amounts of digital auxiliary information kept together with the print. The auxiliary information consists of a small amount of digital data to enable accurate registration and color-reproduction, followed by a larger amount of digital data to recover residual errors and lost frequencies by distributed Wyner-Ziv coding techniques. Approximating the original digital image enables many uses, including making good quality reprints from the original print, even when they are faded many years later. In essence, the print itself becomes the currency for archiving and repurposing digital images, without requiring computer infrastructure.

Internal Accession Date Only Approved for External Publication
To be published and presented at VCIP 2008 - Visual Communications and Image Processing 2008, San Jose, CA

PHOTOPLUS: AUXILIARY INFORMATION FOR PRINTED IMAGES BASED ON DISTRIBUTED SOURCE CODING

Ramin Samadani and Debargha Mukherjee
Hewlett Packard Laboratories, Palo Alto, California, USA
Email: {Ramin.Samadani, Debargha.Mukherjee}@hp.com

ABSTRACT

A printed photograph is difficult to reuse because the digital information that generated the print may no longer be available. This paper describes a mechanism for approximating the original digital image by combining a scan of the printed photograph with small amounts of digital auxiliary information kept together with the print. The auxiliary information consists of a small amount of digital data to enable accurate registration and color-reproduction, followed by a larger amount of digital data to recover residual errors and lost frequencies by distributed Wyner-Ziv coding techniques. Approximating the original digital image enables many uses, including making good quality reprints from the original print, even when they are faded many years later. In essence, the print itself becomes the currency for archiving and repurposing digital images, without requiring computer infrastructure.

1. INTRODUCTION

This paper addresses the approximate recovery of a digital image from a photographic print, by using both the printed photograph (scanning it) together with additional digital auxiliary information. In particular, we are interested in minimizing the amount of auxiliary information required while allowing adequate reprinting and repurposing of the original print. To be considered useful, the amount of auxiliary information required should be smaller than that required with regular image compression (JPEG for instance) for the same distortion level. This problem is closely related to the well-known problem of source coding with side-information [1][2][3][4][5], where the scanned data acts as side-information. We are not aware of prior work combining both a scan from a print and digital auxiliary information to reconstruct the original digital image. We also believe this is a previously unexplored application of distributed source coding. The closest prior work is on registering halftone prints for watermark recovery [6].

The proposed framework for combining the printed photo and the digital Auxiliary Information (AI) is called *PhotoPlus*. Figure 1 shows the general processing flow for the creation (Encoding) and reconstruction (Decoding) of *PhotoPlus*. During *PhotoPlus* creation by the AI Encoder, the digital image is analyzed in order to extract digital auxiliary information that subsequently, during reconstruction, reduces the distortions caused by printing and scanning. This work is agnostic to the way the auxiliary information is stored, and only assumes that the AI bits are recovered error-free for the purpose of joint reconstruction. Complementary research provides different ways to store the auxiliary information: printing the digital information on the back of the printed photograph using dense barcodes [7], or on the border with visually pleasing barcodes, or using *Memory Spot* technology [8].

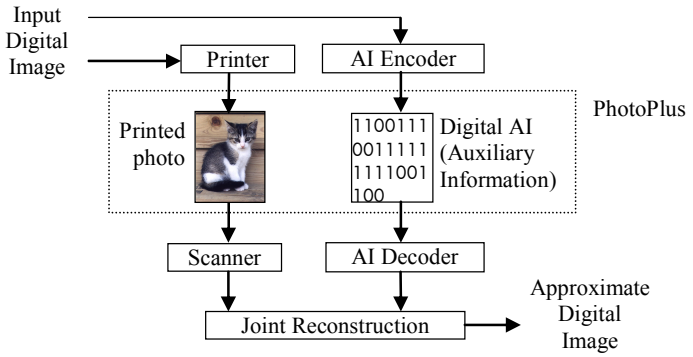


Figure 1. PhotoPlus Framework

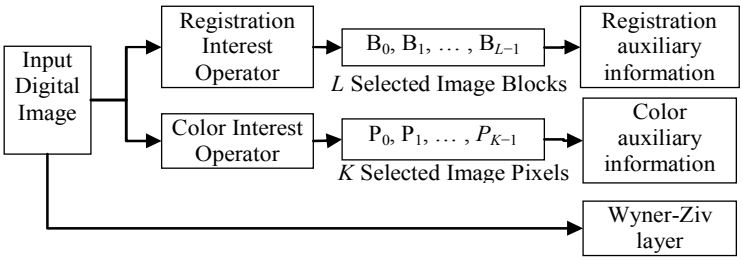


Figure 2. PhotoPlus Auxiliary Information Components

There are three main distortion components introduced by the print-scan channel: 1) registration errors; 2) color distortions; and 3) limitations of the print-scan channel in representing higher frequencies and noise. The auxiliary information that is transmitted comprises three components to correct for these three types of error. It turns out that accurate registration and color-correction can be achieved with very small amounts of auxiliary information. However, there is still degradation due to limitations of the print-scan channel as well as residual noise, which is corrected using a Wyner-Ziv layer comprising the bulk of the bit-stream. During reconstruction, the image is first registered and color-corrected with the first two auxiliary information components, and the resultant image then acts as good *side-information* to decode the Wyner-Ziv layer. Figure 2 shows the processing used to generate the auxiliary information. Details about the information represented and the reconstruction based on it, are described in the following sections.

2. REGISTRATION AND COLOR CORECTION

2.1. Registration

For registration correction, we identify and transmit image regions that may be uniquely matched during recovery. Sampling image regions that are useful for subsequent registration recovery is related to prior work on confidence measures for motion and image registration. One approach is to use confidence measures based on autocorrelation surfaces, such as those found in [9]. Intuitively, picking regions with peaked autocorrelation will result in regions with unique matches. Here, selecting high variance regions has been found to be sufficient for good subsequent registration correction using a global registration model. This is because the global model takes advantage of multiple input sampled regions to provide a robust global registration estimate.

In the implementation, the input image is divided into suitably sized non-overlapping tiles, and for each tile a search is conducted to find the 4×4 luminance block with the highest variance to allow for better subsequent registration with the print-scan image. The spatial locations of the 4×4 blocks, and the actual luminance values comprise the auxiliary information for registration correction. In addition to the local image patches, the number of rows and columns of the original digital image is also transmitted in order to be able to determine the required horizontal and vertical resampling factors for the scanned image.

During reconstruction, the blocks of the input image are matched against the suitably warped printed and scanned photograph. An optimization finds the parameters of the transformation, constrained to be a two dimensional, eight-parameter projective mapping [11], that maps the scanned image to the input image dimensions. In this work we assume that the print includes a border, but extension to prints without a border is trivial. Also, the scan is assumed to be conducted at an appropriately high-resolution. Specifically, if the original digital image size is $M \times N$, and the four corners of the image in the scan at sub-pixel precision are (x_0, x_1), (x_2, x_3), (x_4, x_5) and (x_6, x_7) respectively for top-left, top-right, bottom-left and bottom-right corners, then the original digital pixel (0,0) maps to (x_0, x_1) in the scan, $(0, N-1)$ maps to (x_2, x_3), $(M-1, 0)$ maps to (x_4, x_5), and $(M-1, N-1)$ maps to (x_6, x_7). These conditions yield a unique projective transformation $\mathbf{T_x}\{\}$, with $\mathbf{x}=(x_0, x_1, x_2, x_3, x_4, x_5, x_6, x_7)$, that maps an original pixel (i, j) to corresponding scanned image pixel $(i_s, j_s) = \mathbf{T_x}\{(i, j)\}$. Then the error metric in registration for a given $\mathbf{x}$ is given by:

$$E(\mathbf{x}) = \sum_{(i,j)\in R} \left| y(i,j) - y_s(i_s, j_s) \right|^2, \quad (i_s, j_s) = \mathbf{T_x}\{(i,j)\}$$

where R is the set of all pixels (i, j) included as part of the registration auxiliary information with luminance $y(i, j)$, and $y_s(i_s, j_s)$ is the luminance value of a pixel at sub-pixel location (i_s, j_s) in the scanned image obtained by interpolation. An optimization procedure minimizes $E(\mathbf{x})$ over $\mathbf{x}$. From the optimum value of $\mathbf{x}$, denoted $\mathbf{x}^*$, the

optimum color reconstruction of the image is obtained by interpolating the color pixel values from the color scan at locations $\mathbf{T}_{\mathbf{x}^*}\{(i,j)\}$ for each pixel (i,j) in the image.

2.2. Color Correction

The goal of the encoding of the color auxiliary information is to obtain a sampling of the full range of colors in the original digital image to allow subsequent color correction during decoding. To allow for color recovery that is robust to slight registration errors, we identify and transmit locations and colors of pixels distributed throughout the image and that span the color space well, while also ensuring they are picked from smooth image regions so that registration errors do not cause large errors in the colors retrieved from the scanned image.

During encoding of the input RGB image, to obtain a representative set of colors, the image is first divided into C×C pixel non-overlapping blocks and several colors are sampled from each block. For each block, the similar colors are first clustered or palletized. In our implementation, each block is palletized to obtain up to 8 quantized colors, $\boldsymbol{q}_i$. The region within the block, where each palletized color $\boldsymbol{q}_i$ occurs is defined by a C×C binary indicator map $\boldsymbol{m}_i$ which has value 1 at locations where the color $\boldsymbol{q}_i$ occurs, and has value 0 elsewhere. The next step is to extract one particular sample value of an original, non-quantized color from the region where the indicator map $\boldsymbol{m}_i$ is 1. The sample location selection is guided by the desire to select locations where the colors are changing slowly, so that small residual registration errors translate into small color errors. Two techniques are applied to ensure the colors are sampled from smooth regions. First, very small color regions that occur in highly textured areas are removed by applying a binary morphological erosion filter to each indicator map $\boldsymbol{m}_i$. For each region $\boldsymbol{m}_i$ that *survives* the morphological filter (i.e. contains non-zero values), the original color value is found for the location that has smallest local color variance. This process results in color samples that are less sensitive to registration errors. The row and column locations, as well as the actual color sample values are included in the auxiliary information.

During PhotoPlus decoding, the color reconstruction must correct for the printing and scanning color distortions, transforming all of the scanned image colors to approximations of the original input image colors. A two stage process is used to recover the color information. The first stage applies a 1D polynomial lookup tables followed by a multivariate polynomial approximation [12] to smoothly fit the colors. For PhotoPlus, however, the exact input image color values are known for the colors extracted during PhotoPlus encoding. The result from the first color correction stage does not necessarily interpolate these known values. To correct for this, the color residuals from the first stage pass through the second stage which applies a scattered data interpolation [13] to convert the approximating fit into an interpolating function that passes through the known colors in the input image.

Let $\mathbf{c} = (r,g,b)^T$ be the original color for one input image pixel sampled during PhotoPlus encoding. After printing, scanning and registration but before color correction, the distorted scanned color is given by $\mathbf{c}_s = (r_s, g_s, b_s)^T$. There will be a total of N_c sampled colors that can be used to aid the color correction. During color correction, first a monotonically increasing third order polynomial (1D table) is applied independently to each color component of the registered scan colors $\mathbf{c}_s$. The polynomial is denoted:

$$p(x) = p_0 + p_1 x + p_2 x^2 + p_3 x^3$$

where x represents a color component of $\mathbf{c}_s$. Applying these polynomial results in the modified color $\mathbf{c}' = (r',g',b')^T$, where:

$$\begin{pmatrix} r' \\ g' \\ b' \end{pmatrix} = \begin{pmatrix} p(r_s) \\ p(g_s) \\ p(b_s) \end{pmatrix}$$

Next, the nine cross terms $r',g',b',r'g',r'b',g'b',r'^2,b'^2,g'^2$ are computed, generating a 9-component vector $\mathbf{v} = \{r',g',b',r'g',r'b',g'b',r'^2,b'^2,g'^2\}^T$. To this vector, a 3×9 matrix M is applied to form estimate $\mathbf{c}''$, given by

$$\mathbf{c}'' = \mathbf{M}\mathbf{v}$$

The value $\mathbf{c}''$ is the estimate of the original color $\mathbf{c} = (r,g,b)^T$ after the first approximating step of color correction. A non-linear minimization procedure finds the four coefficients of the polynomial $p(x)$ and the 27 elements of Matrix M that minimize the expected error between $\mathbf{c}$ and $\mathbf{c}''$ for the sampled colors. The quantity minimized is the average CIELAB delta E value for all the samples.

The first fitting stage described above imposes a smooth transformation that only determines an approximation instead of interpolating through the known color values. For well registered sampled scanned colors $\mathbf{c}_s$ however, the

exact color values **c** are known. To take advantage of this precise information, a second stage of color correction applies a 3D inverse weighted [13] scattered interpolation operating on the sampled color residuals, $\boldsymbol{\delta}_i = \mathbf{c}_i - \mathbf{c}_i''$, obtained from the first color correction stage. For the known sampled colors $\mathbf{c}_i$ adding the correction factor $\boldsymbol{\delta}_i$ to the output of the first color correction stage results in $\mathbf{c}_i'' + \boldsymbol{\delta}_i = \mathbf{c}_i$, the original color. The scattered data interpolation provides improved residual corrections for the majority of other colors in the image, even those not sampled. In practice, this second stage substantially improves the PSNR, sometimes improving the results by over 1dB.

The interpolation used for the second stage proceeds in two steps. First, all of the sampled colors $\mathbf{c}_i$ are used in an inverse weighted scattered data interpolation given by:

$$\boldsymbol{\delta}(\mathbf{c}'') = \sum_{i=1}^{N_s} \mathbf{w}_i(\mathbf{c}_i - \mathbf{c}_i'') = \sum_{i=1}^{N_s} \mathbf{w}_i \boldsymbol{\delta}_i$$

In this equation, the normalized weights $\mathbf{w}_i$ used are proportional to the inverse cube of the distance: $\|\mathbf{c}'' - \mathbf{c}_i''\|$. The DSGrid 3D interpolation function from the NGMath package of the National Center for Atmospheric Research (NCAR) Graphics package [14] was used to implement the scattered interpolation. This scattered data interpolation routine was first used to generate the correction residual values on a regular RGB grid of 64×64×64 points, for each of three component residuals for R, G and B. Thereafter, tri-linear interpolation was used to obtain the residual color error estimates for each pixel. The error estimate was subsequently added to the corresponding pixel color values to obtain the final estimate.

2.3. Iterative Registration and Color Correction

Iterating the registration and color correction steps a few times improves results over applying one-step registration and color-correction. It is observed that each step improves the accuracy of the other complementary step. In the experiments, four iterations were used.

3. WYNER-ZIV LAYER

As previously noted, the registered and color corrected image yields a good side-information image to use to decode the third distributed coding layer. Our distributed coding layer operates in the block-DCT domain in the *Y*, C_b, C_r color space with 4:2:0 sub-sampling, as in the case of JPEG. However, the availability of the highly correlated registered and color-corrected image at the decoder before decoding this layer, enables us to save bit-rate over traditional DCT + quantization + entropy-coding compression approaches. In this regard, there are two different approaches to consider. In the first case, the objective is to obtain exactly the same quantization as one would obtain from a regular image coder, without consideration for the reconstruction function that may be employed at the decoder. In this case, the quantization step sizes must be maintained the same as that in the regular codec, and coding of the quantization indices essentially boil down to Slepian-Wolfe coding. In the second case, the objective is to target the same expected distortion as in a regular codec, while considering an optimal reconstruction function within a quantization bin. In this case, the quantization step sizes may be larger, leading to bigger bit-rate savings. In this work, we adopt the second approach, and because this is essentially Wyner-Ziv coding, the Photoplus distributed coding layer is termed Wyner-Ziv layer.

We use a simple form of memoryless coding where the DCT coefficients *X* are first quantized using a uniform deadzone quantizer with step-size *QP* using function $\phi()$ below, and then circular zero-centered cosets are computed with modulus *M* to obtain the coset index *c* using function $\psi()$ below.

$$Q = \phi(X, QP) = sign(X) \times \lfloor |X| / QP \rfloor \quad C = \psi(Q, M) = \begin{cases} Q - M\lfloor Q/M \rfloor & , \; Q - M\lfloor Q/M \rfloor < M/2 \\ Q - M\lfloor Q/M \rfloor - M & , \; Q - M\lfloor Q/M \rfloor \geq M/2 \end{cases}$$

Only the symbols *C* are entropy coded and transmitted. The question that needs to be addressed is how to obtain the right parameters *QP* and *M* for a given target quality. In [4][5], we have provided a detailed analysis of parameter choice, encoding and decoding mechanisms for Wyner-Ziv coding of Laplacian sources under the assumption of an additive noise model $Y = X + Z$, where *X* is a Laplacian source to be transmitted with known variance σ_X^2, *Z* is additive i. i.d. noise (Gaussian or Laplacian) independent of *X* with known variance σ_Z^2, and *Y* is the side-information available at the decoder. For the *PhotoPlus* problem, we found that the block DCT AC coefficients *X* and the corresponding coefficients *Y* from the registered color-corrected image, are related by: $Y = \rho X + Z$, where $0 < \rho < 1$ is an attenuation factor that decays fast for higher frequencies, and *Z* is the scanning noise which is closer to Laplacian in distribution. This model is quite intuitive since the print-scan channel not only loses high-frequency information, but also adds noise. Since we can rewrite the model as $Y/\rho = X + Z/\rho$, the same procedure from [4][5] can be used to obtain the optimal *QP* and *M* parameters, with $(\sigma_Z/\rho)^2$ replacing σ_Z^2 and Y/ρ replacing *Y* by during decoding.

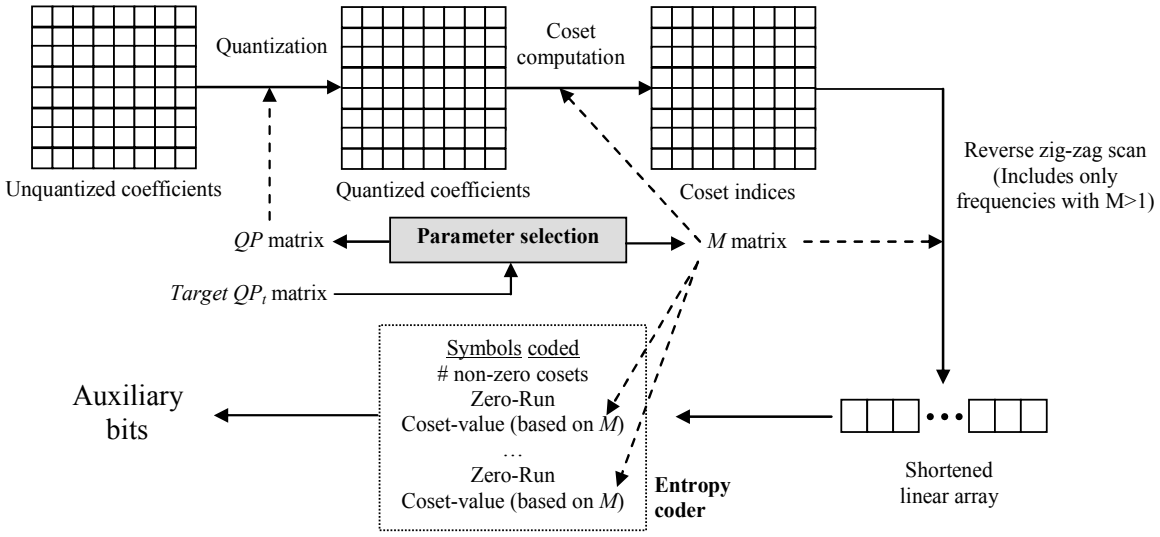


Figure 3. Parameter choice and encoding with memoryless coset codes

The only other issue to consider is to know what values of ρ, σ_X^2 and σ_Z^2 to use. Fortunately, the variance σ_X^2 for each frequency can be computed for each image to be coded and transmitted as part of the auxiliary information. For the attenuation factor ρ and the noise variance σ_Z^2, we make the best estimate using $\{X, Y\}$ pairs for each frequency and color component taken from a set of training images post registration and color-correction. For the training set, ρ for a given frequency and color is estimated as:

$$\rho = \arg\min_{\rho'} \sum_{\{x,y\}\in T} \left\| y - \rho' x \right\|^2 = \frac{\sum_{\{x,y\}\in T} xy}{\sum_{\{x,y\}\in T} x^2},\ T \text{ is the training set for given frequency and color}$$

We found that the ρ values for each frequency and color to be amazingly image-independent for a given printer-scanner combination. Based on this finding, we keep ρ fixed at the values obtained from the training set:

$$\rho_{\mathrm{y}} = \begin{pmatrix} 1.00 & 0.87 & 0.73 & 0.55 & 0.41 & 0.28 & 0.17 & 0.09 \\ 0.87 & 0.76 & 0.64 & 0.50 & 0.37 & 0.25 & 0.15 & 0.08 \\ 0.73 & 0.64 & 0.55 & 0.44 & 0.32 & 0.22 & 0.14 & 0.07 \\ 0.55 & 0.50 & 0.44 & 0.36 & 0.27 & 0.18 & 0.11 & 0.06 \\ 0.41 & 0.37 & 0.32 & 0.27 & 0.21 & 0.14 & 0.09 & 0.05 \\ 0.28 & 0.25 & 0.22 & 0.18 & 0.14 & 0.09 & 0.05 & 0.03 \\ 0.17 & 0.15 & 0.14 & 0.11 & 0.09 & 0.05 & 0.03 & 0.02 \\ 0.09 & 0.08 & 0.07 & 0.06 & 0.05 & 0.03 & 0.02 & 0.01 \end{pmatrix},$$

$$\rho_{\mathrm{u}} = \begin{pmatrix} 1.00 & 0.94 & 0.86 & 0.75 & 0.63 & 0.50 & 0.38 & 0.25 \\ 0.94 & 0.87 & 0.80 & 0.70 & 0.58 & 0.46 & 0.34 & 0.24 \\ 0.86 & 0.80 & 0.74 & 0.64 & 0.53 & 0.41 & 0.31 & 0.23 \\ 0.75 & 0.70 & 0.64 & 0.55 & 0.45 & 0.35 & 0.26 & 0.19 \\ 0.63 & 0.58 & 0.53 & 0.45 & 0.37 & 0.28 & 0.21 & 0.15 \\ 0.50 & 0.46 & 0.41 & 0.35 & 0.28 & 0.22 & 0.16 & 0.12 \\ 0.38 & 0.34 & 0.31 & 0.26 & 0.21 & 0.16 & 0.13 & 0.10 \\ 0.25 & 0.24 & 0.23 & 0.19 & 0.15 & 0.12 & 0.10 & 0.07 \end{pmatrix}, \rho_{\mathrm{v}} = \begin{pmatrix} 1.00 & 0.94 & 0.88 & 0.78 & 0.67 & 0.54 & 0.39 & 0.30 \\ 0.94 & 0.89 & 0.83 & 0.73 & 0.61 & 0.48 & 0.34 & 0.25 \\ 0.88 & 0.83 & 0.76 & 0.67 & 0.55 & 0.42 & 0.30 & 0.22 \\ 0.78 & 0.73 & 0.67 & 0.57 & 0.46 & 0.35 & 0.25 & 0.18 \\ 0.67 & 0.61 & 0.55 & 0.46 & 0.36 & 0.26 & 0.19 & 0.14 \\ 0.54 & 0.48 & 0.42 & 0.35 & 0.26 & 0.18 & 0.12 & 0.09 \\ 0.39 & 0.34 & 0.30 & 0.25 & 0.19 & 0.12 & 0.09 & 0.06 \\ 0.30 & 0.25 & 0.22 & 0.18 & 0.14 & 0.09 & 0.06 & 0.04 \end{pmatrix}$$

where ρ_y, ρ_u, ρ_v, are the values for the Y, U and V components taken over the 8×8 DCT coefficients.

The noise variance σ_Z is next modeled as scaled version $\sigma_Z = \gamma\sigma_X$ of σ_X for each frequency, based on the aforementioned training data and the ρ values. Thus, given the set of transform coefficient variances σ_X^2 available at both the encoder and the decoder for a given image, the variances σ_Z^2 are estimated as $(\gamma\sigma_X)^2$, and both are then used

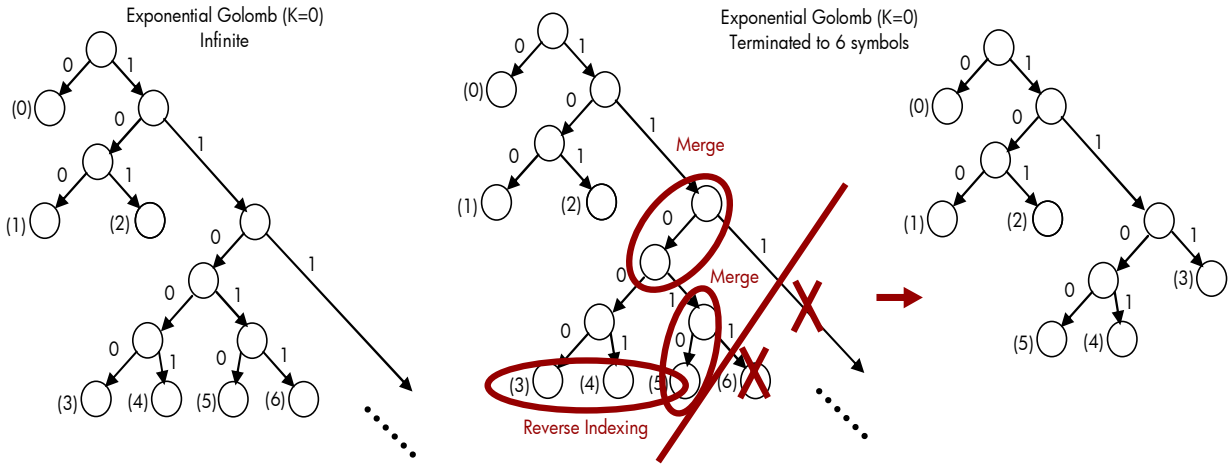


Figure 4. Termination strategy for infinte tree codes. Shown is an example depicting termination to 6 symbols for Exponential Golomb codes with $k = 0$.

in the parameter choice procedure [4][5] to obtain the optimal *QP* and *M* parameters for each frequency for a given target quality. The target quality is represented by quantization parameter QP_t for regular non-distributed coding, which provides a mechanism to adapt the bit-rate over a wide range as in the case of a regular coding. We only consider memoryless codes with no channel-coded planes, because as shown in [5], when the σ_Z values cannot be estimated accurately enough, use of channel coding can substantially degrade the performance under channel mismatch conditions.

The flow of parameter choice followed by encoding is depicted in Figure 3. The parameter choice mechanism conducted in both the encoder and decoder yields the QP and *M*-matrices to be used for quantization and coset computation for each block. In a typical *M*-matrix several frequencies are associated with *M*=1 implying that these coefficients are not transmitted. After quantization and coset computation, the coefficients are reverse zigzag scanned with the scan only including coefficients with *M*>1. This shortened array is coded using the symbols shown with the code for the Coset-values adapted based on the value of *M*. We use a non-conventional entropy coder, where each AC coset is associated with a known *M* value that restricts the possible values of *C* to lie between $-\lfloor M/2 \rfloor$ to $\lfloor (M-1)/2 \rfloor$. Our entropy coder uses zero-run and coset-value symbols in reverse zigzag scan, similar to the CAVLC entropy coder in H.264/AVC [15], but the coset-value symbols are coded using a family of *terminated* index-remapped exponential-Golomb or similar codes, that are especially convenient since they can be adapted on the fly when a coset-value is known to be restricted by the modulus *M* known at both the encoder and the decoder. The mechanism used for terminating an infinite tree code is depicted in Figure 4 for Exponential Golomb codes for *k*=0. Some braches can be merged up from the truncated tree without violating the prefix condition thereby reducing the code-length for the symbols attached to that branch. Note the code-lengths still provide a non-decreasing sequence over symbol indices in decreasing order of probabilities. This mechanism is actually quite generic and can be applied to any infinite tree code (exp-Golomb, sub-exponential, etc.).

Before coding with the terminated code, index-remapping must also be conducted since the symbols need to be in descending order of probability,. If *c* is the coset index, the remapped index $r(c)$ and its inverse $r^{-1}(c)$ required at the decoder, are given by:

$$r(c) = \begin{cases} 2c, & c \geq 0 \\ -1-2c, & c < 0 \end{cases} \qquad r^{-1}(c) = \begin{cases} c/2, & c \text{ even} \\ -(c+1)/2, & c \text{ odd} \end{cases}$$

To provide an example, for $M = 6$, the coset symbols {–3, –2, –1, 0, 1, 2} are remapped to {5, 3, 1, 0, 2, 4}. The remapped indices can then coded with a 6-symbol terminated code as depicted in Figure 4 for the case *k*=0.

Once the decoder decodes a coset index, it invokes an optimal reconstruction procedure to obtain the best reconstruction value $\hat{x} = E(X/Y=y)$ for the coefficient based on the assumed model. The optimum reconstruction function is provided in [4][5].

In order to allow finer adaptation of the coding process from block to block, in practice, each 16×16 coding block is classified into one of several types based on the sum of energies of the AC coefficients in it. All classes have the

(a) Scanned Image (b) Original Image (c) Registered Color-corrected Image

Figure 5. Registration and color-correction results for *Woman* Image

same value of ρ for each frequency, but for each class a different set of std. deviation ratios γ are estimated from the training images. For every image, a set of source variances σ_X^2 is transmitted for each class, along with the class index per block. This amounts to choosing a different set of parameters matched to the statistics of each class.

4. RESULTS

Figure 5 shows the results for registration and color-correction applied to the *Woman* image. The left of the figure shows the unprocessed scan, the center shows the original image, and the right shows the decoded image using the process described in Section 2 for registration and color-correction. The results show a subjectively good reconstruction using less than five kilobytes of auxiliary information for registration and color correction only. The reconstructed peak signal-to-noise ratio (PSNR) is 30.4 dB for luminance, and 43.0 dB and 42.3 dB for the chrominance channels, to make a weighted average of 32.0 dB.

Figure 6 shows the overall PSNR vs. bit-rate curves for the *Woman* and *Bike* images including the reconstruction from the Wyner-Ziv layer, compared against a standard DCT based encoder. In order to make a fair comparison both the Wyner-Ziv and standard versions use the same aforementioned entropy coder, the only difference being that the coset modulus matrix M is assumed to be all ∞ for the standard case, while it is determined based on parameter choice for the Wyner-Ziv case. In both the curves, we see that with only registration and color-correction information, a reasonably good quality is achieved with a very small bit-rate. This constitutes the lowest bit-rate point in the Wyner-Ziv Rate-Distortion curve. Thereafter as more and more bit-rate is spent on the Wyner-Ziv layer, the quality improves gradually, but remains superior to the standard encoder at all bit-rates. At higher bit-rates however, the gap tends to close.

5. CONCLUSIONS

We have presented a novel application of distributed image coding called *PhotoPlus*, and have provided a comprehensive solution that allows both generation of good quality side-information, as well as recovery of lost frequencies and scanning noise. Applying our technology together with the complementary technologies for 2D barcode printing has the potential to make the print itself a medium for repurposing the content. Our tests show that carefully selected auxiliary information based on distributed source coding concepts along with the scan allow reconstruction of images that are vastly superior to traditional source (image) coding.

6. REFERENCES

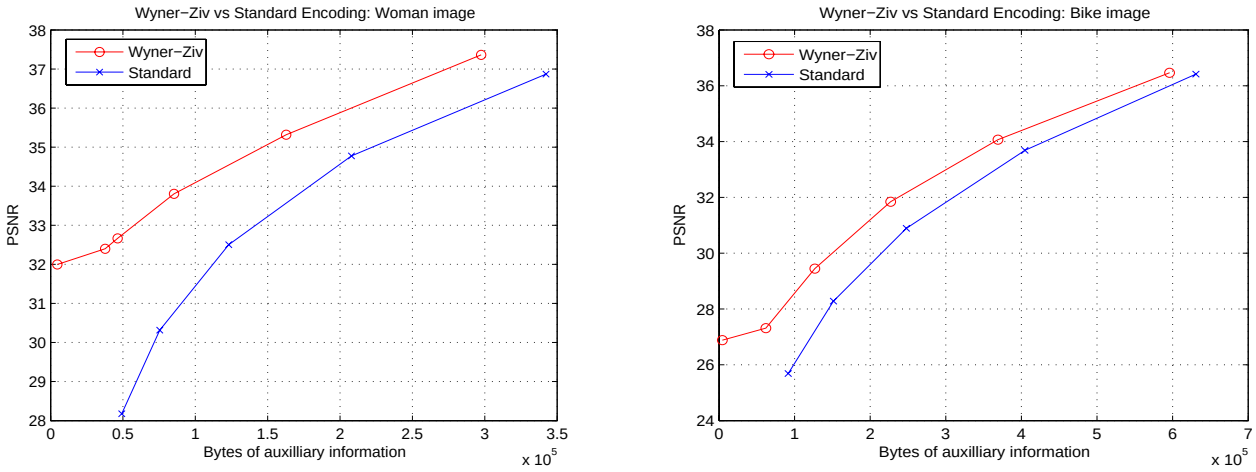
[1] J. D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Theory*, vol. 19, pp. 471–480, July 1973.

[2] A. D. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inf. Theory*, vol. 22, no. 1, pp. 1–10, Jan. 1976.

[3] B. Girod, A. Aaron, S. Rane and D. Rebollo-Monedero**,** "Distributed video coding," *Proc. of the IEEE*, *Special Issue on Video Coding and Delivery,* vol. 93, no. 1, pp. 71-83, January 2005.

[4] D. Mukherjee, "Optimal Parameter choice for Wyner-Ziv coding of Laplacian Sources with Decoder Side-Infomration," *HP Labs Technical Report*, HPL-2007-34.

[5] D. Mukherjee, "Parameter selection for Wyner-Ziv coding of Laplacian sources," *Proc. SPIE, Visual*



(a) Woman Image (b) Bike Image

Figure 6. Rate-distortion curve for *PhotoPlus*

Communications and Image Processing, San Jose, Jan 2008.

[6] K. Solanki, U. Madhow, B. S. Manjunath, "Estimating and undoing rotation for print-scan resilient data hiding," *Proc. IEEE Int. Conf. Image Processing*, vol. 1, 2004.

[7] D. L. Hecht, "Printed embedded data graphical user interfaces," *Computer*, vol. 34, no. 3, pp. 47-55, 2001.

[8] *HP Memory Spot*, http://www.tfot.info/content/view/79/59/, http://www.engadget.com/2006/07/17/hps-very-tiny-wireless-chip-dubbed-memory-spot/, http://www.popsci.com/popsci/flat/bown/2006/product_62.html

[9] P Anandan, "A computational framework and an algorithm for the measurement of visual motion", *International Journal of Computer Vision,* Vol. 2, No. 3, 283-310, Jan. 1989.

[10] S. Gaurav and H.J. Trussell, "Digital Color Imaging"*, IEEE Transactions on Image Processing,* Vol. 6, No. 7, July 1997, pp. 901-932.

[11] I. Faux, M. Pratt, *Computational Geometry for Design and Manufacture*, Ellis Horwood Ltd., 1979.

[12] C. E. Mancill, *Digital color image restoration*, Ph.D. thesis, Image Processing Institute, University of Southern California, 1975.

[13] I. Amidror, "Scattered data interpolation methods for electronic imaging: a survey," *Journal of electronic Imaging*, vol. 11, pp. 157-76, 2002.

[14] NCAR Graphics NGMath library. http://ngwww.ucar.edu/ngmath/

[15] T. Wiegand, G. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technology*, vol. 13, no. 7, pp. 560–576, Jul. 2003.