Multilevel Vector Error Diffusion with Solvent Control

Serge Cattarinussi and Ana Dimitrijevic ; Olivetti Engineering SA; CH-1401 Yverdon-Les-Bains, Switzerland

Abstract

We propose a new multilevel vector error diffusion method which introduces a control parameter S related to the physical amount of solvent (liquid) allowed for producing a given input color by combining dark (saturated) and light (unsaturated) ink separations. The solvent S appears as a fourth input variable to the error diffusion process where the quantizer is designed to choose the closest primitive in the CMYS space.

Introduction

Traditional ink-jet printing systems use three standard color inks *CMY* in a bi-level output mode, where a dot is either printed or not. The whole printing process is typically composed of the calibration, including ink-limitation and linearization, and of an image halftoning procedure like, e.g., error-diffusion (ED) [7]. Such binary printing devices cannot respond to highly demanding customer requests for photo quality reproductions. The main concern is an apparent graininess due to visible dots in the highlights. The most common method to overcome this problem is to replace bi-level by multi-level printing, using inks having the same chromaticity but different densities [2]. Typically, diluted versions of cyan (light cyan *c*) and of magenta (light magenta *m*) are used. The increased number of drops as well as their smaller contrast with the background produce a smoother and more pleasing appearance in weakly saturated areas. On the contrary, the shadows should be preferably printed using optically dense inks which allow to reach saturation with smaller ink quantities. In the following we will focus on "CcMmY" photo printing and use the adjectives *dark* and *light* when referring to high and low density ink separations, respectively.

The introduction of such light inks brings new challenges to color characterization and has become an active research topic. Actually, the straightforward application of a multilevel ED [6] is not well suited to multi-density ink printing because it does not allow to control the total ink amount. It is important indeed not to exceed the maximum allowed ink amount to be deposited on the substrate since ink running through the substrate can produce highly undesirable image artifacts as bleeding, cockling or banding. In bi-level systems, this limit is usually controlled by the input values. However, if there is not an univocal correspondence between the produced color and the amount of ink (liquid) deposited on the paper, as in multi-level printing, it is not possible to control the paper wetting. Therefore, in all proposed approaches, the "ink mixing" (i.e. the determination of ink percentage mapped to light and dark inks) is integrated into the calibration step rather than into the halftoning.

Usually, a set of empirically optimized ink separation curves are applied to perform smooth mixing of inks of similar hue. Several criteria to construct the separation curves must be used, such that: avoiding the creation of dot patterns, getting smooth transitions along color gradients and preventing the mixing from over inking [5]. There are however difficulties to separate light and dark inks when they have significant differences in their hue angle. Shaw et al. [8] proposed an algorithm that overcomes this problem by building an ink interaction model to predict the colorimetric response of any ink combination, allowing to obtain a smooth light to dark ink transition. Apart from an increased algorithm complexity, the important disadvantage of this method is that ink limitations are applied independently to each ink, and not globally. Moreover, the separation curves are fixed, meaning that for every change in the calibration due to the colorant or to the paper substrate, the whole process should be repeated.

In the previous approach, light inks are considered as an extension of the dark ones, thus preserving the conventional three-color degrees of freedom (CMY). As an alternative, additional inks could be considered as separate channels, thus increasing the number of degrees of freedom and resulting in a mathematically under-constrained optimization problem [9]. To solve this problem, additional constraints have to be defined such as upper bounds on the total colorant amount and preferences for colorants. The methods proposed in literature that consider ink channels independently are usually based on a parametric spectral model characterizing the *CcMmY* printer. A color separation mapping, from the *CMY* to the *CcMmY* space, is then derived from the model [1]. These latter approaches result in a computationally highly demanding image processing chain because both the color interpolation and the halftoning procedure have to be performed in a five-dimension space. Moreover, a lot of memory is required to store the $5D$ color mapping lookup tables. This can be an important practical obstacle for embedded imaging systems such as standalone ink-jet printers.

In the present paper, we propose an alternative approach to the multiple ink problem that incorporates the solution into a modified halftoning method. To that purpose, a new variable S, referred to as *solvent*, is introduced into an error diffusion process and controls the quantity of ink used for producing the required colors. The variable S is considered as a function $\mathscr{F}_S$ of the color variables C, M, Y which, all together, enter into a quantizer Q (see figure 1) having more output than input channels [3], as typical for multilevel error diffusion. The method has the clear advantage to be completely vectorial (a global ink saturation limit can be applied) and, at the same time, is neither computationally expensive nor memory demanding. Also, the choice of the solvent function is quite flexible, giving the possibility to obtain a smooth transition between light and dark inks even, possibly, if they have significant differences in hue angle.

In the following, we will first introduce our multi-level error diffusion scheme before to focus our attention on the properties that the solvent function $\mathscr{F}_S$ must satisfy. We will then continue with five simulated applications of the method, varying the number of colors, the number of output levels, and the properties of the solvent function.

Error diffusion with solvent control

The diagram in figure 1 shows a general error diffusion scheme with a quantizer having more output than input channels [3], which is typical for a multilevel error diffusion [6]. In that case, the quantizer primitives [4] are not independent vectors in the C, M, Y vectorial space. For a given constant input I, the quantizer output can be considered as a linear combination of the primitive vectors. If the latter are not independent, different linear combinations can produce the same output. This may have a number of different consequences but we consider only one: Among the set of all the linear combinations of primitives, which produce the same output, some may exceed the substrate water absorption capabilities.

This is a practical problem because the maximum amount of ink to be deposited on the substrate is usually controlled by limiting the input values either on a scalar ($C < C_{max}, M < M_{max}, Y < Y_{max}$) or on a vectorial ($C+M+Y < T_{max}$) base[10]. However, the lack of a one-to-one correspondence between the input values and the amount of ink deposited on the paper prevents the paper wetting from being kept below the maximum paper absorption.

To solve the above wetting problem, a new *solvent* variable S is added to the three color input values C, M, Y. It is meant to control the quantizer Q in order to avoid a possible excess of ink. This variable $S = \mathscr{F}_S(C, M, Y)$ is a function of the color variables C, M, Y and must be physically related to the amount of liquid or solvent that can be allowed for producing the required color. The function $\mathscr{F}_S$ will be referred to as the *solvent model*.

Before to actually construct a solvent model $\mathscr{F}_S$, we must address the question of *how* it can be used control the quantizer. We know that the ED takes the decision of whether or not to print a dot according to what has happened to previously processed pixels, in such a way that the local average color matches the requested input color. As a dithering process, it allows to locally match the average color input, on a small neighbourhood, not on one single pixel.

The central contribution of the present work is to recognize that such a behaviour is suitable, not only for the color densities which form an image, but also for the amount of solvent used.

Since we need to control *on average* and *locally* the quantity of solvent that is deposited on the printing media, it becomes natural to include the variable S into the error diffusion process. Once the variable S is properly calculated, all we need is a quantizer Q which takes the modified input

$$I_m = (C_m, M_m, Y_m, S_m) \tag{1}$$

and outputs one of the 2^5 primitives

$$O_p = (D_C, D_M, D_Y, D_c, D_m) \tag{2}$$

as shown in the figure 1 . If the output primitive O_p comprises a drop of ink X, then, in equation 2, $D_X = 1$, otherwise $D_X = 0$.

In order to complete the classical ED cycle depicted in figure 1, a value $V_p(O_p)$ in the (C, M, Y, S) vector space must be associated to each output primitive O_p.

$$V_p(O_p) = (C_p, M_p, Y_p, S_p) \tag{3}$$

In equation 3, (C_p, M_p, Y_p, S_p) are the cyan, magenta, yellow and solvent values associated with the primitive O_p . The error E in figure 1 is then calculated as $E = I_m - V_p(O_p)$.

As a general rule, the quantizer Q should favor the output of primitives comprising light inks when the input coordinate S_m is large, while it should favor the output of primitives comprising only dark inks whenever S_m is small.

Accordingly, the quantizer Q can simply be designed to chose, as an output, the primitive O_p whose color and solvent values V_p is "closest" to the modified input I_m . In order to determine the closest output primitive, euclidiean distances must be calculated within the 4D space (C, M, Y, S). The error calculation and the error diffusion follow the usual path except that an error is calculated and diffused for the solvent variable as well.

Although being fully standard in its form, the above error diffusion algorithm has one special feature which worth to be emphasized: the forth input variable (solvent S) is a *function* of the other three (color) variables. This feature, along with the fact that each output primitive has a fixed contribution to the solvent *and* to the color variables, put important constraints to the range of possible output mean values. It is therefore of fundamental importance to choose the solvent function $\mathscr{F}_S$ such that the input $I = (C, M, Y, S)$ is always within the range of the output possible values.

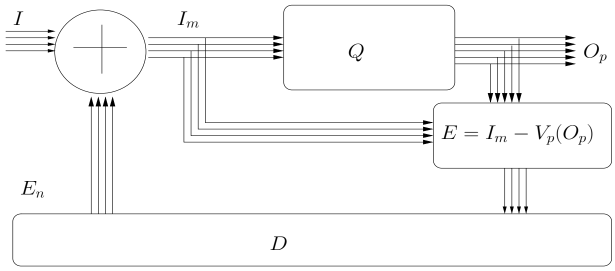

Figure 1. *Generic block diagram for the error diffusion halftoning algorithm with a quantizer having less input channels* (4) *than output channels* (5).

The definition domain and the bounding volume of the solvent model function

Let us consider a system capable of printing a maximum of one drop per ink variety meaning five drops per pixel for a CcMmY printer. The color variables being defined in the range [0, 255], the solvent values are supposed to vary within the range [0, 5×255].

Depending on the pixel size (i.e. resolution), the drop volume, and the substrate absorption capabilities, one may be need to limit the amount of solvent to a value S_{max} considerably lower than the above numerical limit of 5×255. $S_{max} = 500$ can be considered a typical value for printing on porous photo paper at a resolution of 1200 dots per inch with ink drops having a volume of 4 picoliters. Let us define $(\alpha_c, \alpha_m, \alpha_y)$ the color density

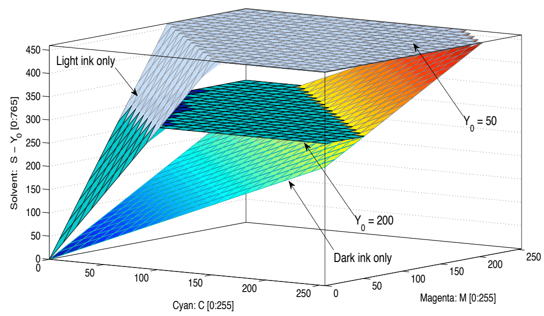

Figure 2. *Bounding volume for the solvent model function* $\mathscr{F}_S$. $S_{max} = 500$ *and* $Y_0 = 50$ *and* 200 .

ratios of light and dark inks. Since there is no yellow light ink, we usually have $\alpha_y = 1$. Typical values for light cyan and light magenta are $\alpha_c = \alpha_m \approx 3$.

Within the hyper-cube defined by $0 \le X \le 255$, $X \in \{C, M, Y\}$ and by $0 \le S \le 5 \times 255$, the solvent model function $\mathscr{F}_S$ is bounded by the hyperplanes P_{lim}, P_{dark} and P_{light} such that:

$$\begin{array}{lcl} P_{lim}: & S = & S_{max} \\ P_{dark}: & S = & C + M + Y \\ P_{light}: & S = & \alpha_c C + \alpha_m M + \alpha_y Y \end{array} \tag{4}$$

$$
\begin{array}{ll}
\mathscr{F}_S \leq P_{lim} & \mathscr{F}_S \leq P_{light} \\
\mathscr{F}_S \geq P_{dark} &
\end{array}
\tag{5}
$$

Since $\alpha_y = 1$, and in order to visualize the bounding volume formed by the planes in equation 4, we can fix the amount of yellow ($Y = Y_0$) and form the variable $S^* = S - Y_0$. The figure 2 shows such a bounding volume for two different values of Y_0 in the space (C, M, S^*).

By combining the equations 4 and the inequalities 5, we get the relation $C + M + Y \leq S_{max}$ which limits the definition domain of $\mathscr{F}_S$. Therefore, it is important, when setting up a solvent model, to consider carefully both the bounding volume *and* the definition domain of the solvent function. Violating any of these constraints will result in numerical divergences within the error diffusion process.

Monochromatic printing with three output levels

As a first simple test application we consider a monochromatic printer using gray (k) and black (K) inks, which can produce three different output levels on every single pixel. Assuming that the volume of gray and black drops is identical, the table 1 shows the output values $V_p(O_p)$ assigned to the three output primitives (W, k, K) .

The application of our ED scheme requires, as a first step, the construction of a quantizer $Q^{(3)}_{mono}$ which will be of dimension 2 in the present monochromatic case. The figure 3 shows, in the (G, S) space, the primitives (W, k, K) and the space partitioning which associates any input $I_m = (G_m, S_m)$ to its closest primitive. As a second step, a solvent model $S = \mathscr{F}_S(G)$ must

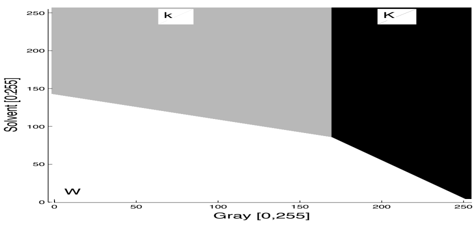

Figure 3. *The monochromatic quantizer $Q^{(3)}_{mono}$ which has three output levels corresponding to the (W, k, K) primitives of table 1 .*

be constructed, which fulfills the conditions in equations 4 and 5 and allows to calculate the desired solvent value for every gray input tone G. We will analyse the two models (M_1 and M_2) presented in figure 4. Both models assume a solvent absorption limit $S_{max} < 255$. The model M_1 maximizes the usage of the light (k) ink and can be expressed as:

$$
\begin{array}{rcl}
M_1: \quad S & = & min(\alpha_g\, G,\ S_{max}) \\
\alpha_g & = & V_G(K)/V_G(k)
\end{array}
\tag{6}
$$

where, according to table 1, $\alpha_g = 255/85 = 3$ is the color density ratio of light and dark inks. A gray ramp whose values vary linearly from $G = 0$ to $G = S_{max} = 134$ has been halftoned with our new ED scheme, using the solvent model M_1 and the quantizer $Q^{(3)}_{mono}$ described above.

By counting, on the output image, the number of light (k) and dark (K) drops, we have estimated the total amount of solvent as well as the *resulting* inks separation curves. The figure 5 shows that the prescribed maximum amount of solvent is never exceeded. Moreover these ink separation curves reproduce fairly well the "standard ink separation curves" reported in the literature [8].

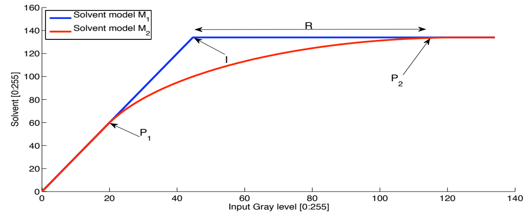

Figure 4. *Two solvent models for monochromatic system. Model M_1 maximizes the usage of light inks while M_2 ensures a smooth progression of the solvent amount (dS/dG is continuous).*

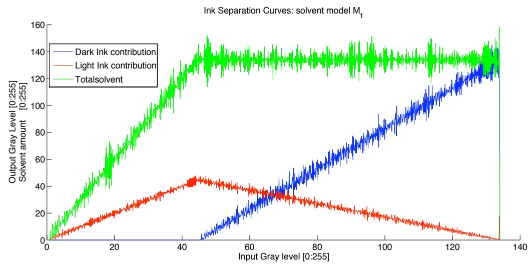

Figure 5. *Ink separation curves obtained for monochromatic system with three output levels, using the solvent model M_1.*

While the results in figure 5 demonstrate the effectiveness of the method, the solvent model M_1 suffers from the same limitations as the standard ink separation methods. In particular, the discontinuous progression of the solvent amount can prevent a precise output tone linearization. In order to overcome these difficulties, we introduced a second monochromatic model (M_2) which has a continuous derivative dS/dG. Such a property is ensured by construction: M_2, as depicted in figure 4, is formed by two straight segments D_1 and D_2 connected at $P_1 = (P_{1,g}, P_{1,s})$ and $P_2 = (P_{2,g}, P_{2,s})$ by a tangent circle $\mathscr{C}$, centered at $C = (C_g, C_s)$. For $\mathscr{C}$ to be tangent to D_1 and D_2, the points P_1 and P_2 must be located at the same distance R from I, the intersection point of D_1 and D_2. More precisely, the model M_2 is defined by

$$
\begin{array}{llll}
D_1: & S = \alpha_g\, G & & G \in \left[0,\ P_{1,g}\right] \\
D_2: & S = S_{max} & & G \in \left[P_{2,g},\ S_{max}\right] \\
\mathscr{C}: & (G - c_g)^2 + (S - c_s)^2 = R & & G \in \left[P_{1,g},\ P_{2,g}\right]
\end{array}
\tag{7}
$$

The above M_2 model is fully determined by the three parameters $\left(\alpha_g, S_{max} \text{ and } R\right)$ which are needed to calculate the center of the circle $\mathscr{C}$ as well as to uniquely define the straight lines D_1 and D_2. In the case of the model M_2 presented in figure 4, we have repeated the calculation performed with model M_1. The ink separation curves and the total amount of solvent are presented in figure 6, showing that a smooth solvent distribution can be reached with a simple improvement of the solvent model.

Table 1. Primitives output values used for monochromatic system simulations

O_p	G (gray)	S (solvent)
W	0	0
k	85	255
K	255	255
k^*	170	255
k^{**}	170	510

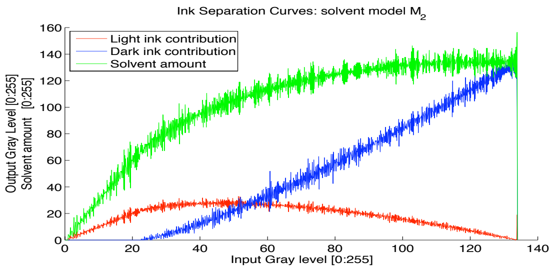

Figure 6. *Ink separation curves obtained for a monochromatic system with three output levels, using the solvent model M_2 which ensures a smooth progression of the amount of solvent.*

Monochromatic printing with four output levels

Providing a printer with the capability of producing more output levels is an effective way for improving the overall printing quality. We show in this section that our ED method is appropriate for handling monochromatic multilevel printers producing at least four output levels per pixel.

Producing one more output level can be obtained either by using a different ink variety of an intermediate density, or by printing two dots of light ink on the same pixel. The corresponding primitives, labelled k^* and k^{**}, respectively, are defined in table 1.

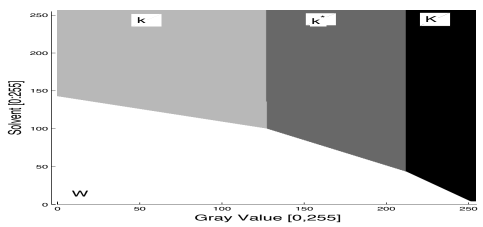

Figure 7. *The monochromatic quantizer $Q^{(4)}_{mono}$ which has four output levels, corresponding to the primitives (W,k,k^*,K) of table 1.*

Adding the primitive k^* of table 1 and using a different quantizer $Q^{(4)}_{mono}$ as shown in figure 7 are the only changes required to adapt our ED method to a four levels printing system. The solvent model can be either M_1 or M_2 or any function $\mathscr{F}_S$ fulfilling the conditions in equations 4 and 5.

Using the model M_2 (with $S_{max} = 200$) and the quantizer $Q^{(4)}_{mono}$, we halftoned a gray ramp as in the two previous sections, but with $G_{max} = S_{max} = 200$. Again, the amount of solvent and the ink separation curves have been estimated by counting the dots on the halftoned image. The results are presented in figure 8 which shows that the maximum amount of solvent is never exceeded while the three overlapping ink separation curves provide a smooth transition from light, to medium, to dark inks.

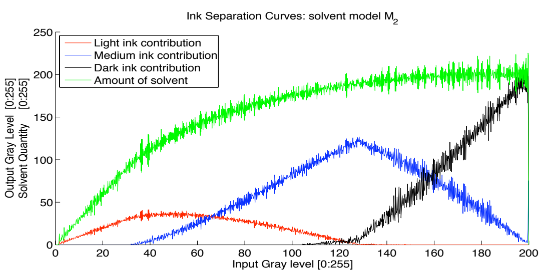

Figure 8. *Ink separation curves obtained for a monochromatic system with four output levels, using the solvent model M_2.*

Table 2. Primitives output values used for color system simulations

O_p	C	M	Y	S
W	0	0	0	0
c	85	0	0	255
C	255	0	0	255
m	0	85	0	255
M	0	255	0	255
cm	85	85	0	510
cM	85	255	0	510
Cm	255	85	0	510
CM	255	255	0	510
Y	0	0	255	255
cY	85	0	255	510
CY	255	0	255	510
mY	0	85	255	510
MY	0	255	255	510
cmY	85	85	255	765
cMY	85	255	255	765
CmY	255	85	255	765
CMY	255	255	255	765

Replacing the primitive k^* by k^{**} requires some more work because the solvent function bounding volume must be redefined (equations 4 and 5 are not valid anymore). Nevertheless, similar results are obtained which demonstrate the simplicity and flexibility of the method.

Color printing with three output levels per color channel

In the two previous sections we have limited the simulated applications of the proposed "multilevel error diffusion" scheme to monochromatic systems. The purpose was to illustrate the main features of the method on simple systems. Nevertheless, this algorithm presents some of its most important advantages when used to drive a color printing system.

First of all, a set of primitive O_p and the corresponding output value $V(O_p)$ have to be defined as, for instance, in table 2.

Then, a quantizer $Q^{(3)}_{col}$ must be constructed as a partitioning of the 4D space (C,M,Y,S), such that, for instance, any given input (C_m,M_m,Y_m,S_m) (see equation 1) is associated to the closest primitive.

Concerning the choice of the solvent model, we consider two different ways of extending the monochromatic models M_1 or M_2 in order to deal with color systems. The first method, herein referred to as the *scalar solvent* method M_3, consists in applying equation 6 or 7 to each color component. The total solvent is then obtained as the sum of the contribution of each color component. For a $CcMmY$ system we would write :

$$\begin{array}{llcl} M_3: & S_i & = & \min\left(\alpha_i X_i,\ S_{i,max}\right) \\ & \alpha_i & = & V\left(O_p\right)/V\left(O'_p\right) \\ \text{with} & & & \\ & i & \in & \{C,M,Y\} \\ & X_i & \in & \mathscr{D}_3 \\ \text{and} & & & \\ & S & = & S_C+S_M+S_Y \end{array} \tag{8}$$

where, O_p and O'_p represent the primitives consisting in one dot of dark or light ink, respectively. Moreover, $S_Y = Y$ and $\alpha_Y = 1$ since there is not, usually, any light yellow component.

With this scalar solvent model M_3, the bounding volume of the solvent function is still defined by the equations 4 and 5, but its domain $\mathscr{D}_3$ is defined by $X_i \leq X_{i,max}$ for each color component.

In the numerical simulations below we used a simple truncation procedure $X_i = T(X_i)$ in order to confine the input within the definition domain boundaries:

$$T(X_i) = \begin{cases} X_{i,max} & \text{if } X_i > X_{i,max} \\ X_i & \text{otherwise} \end{cases} \tag{9}$$

As an application of this model M_3, we halftoned a color image having the cyan component varying linearly from 0 to 255 along one direction, and the magenta component varying linearly from 0 to 255 in the other direction. The yellow component was kept fixed at $Y_0 = 76$. The ink-limitation procedure of equation 9 was applied with $X_{i,max} = 134$ and, accordingly, the maximum allowed solvent quantity per component i was set to $S_{i,max} = 134$ as well. We used the quantizer $Q^{(3)}_{col}$ described above.

As for the monochromatic models, the analysis of the output halftoned image allowed us to produce, in figure 9, the resulting ink-separation surfaces for the cyan component. Similar, not shown, symmetric surfaces can be obtained for the magenta component. As a consistency check, the total amount of solvent is also presented, showing that our multilevel color ED gives a full control over the amount of solvent. However, beside the effectiveness of the method, the numerical analysis of the surfaces on figure 9 shows the main weaknesses of the solvent model M_3 :

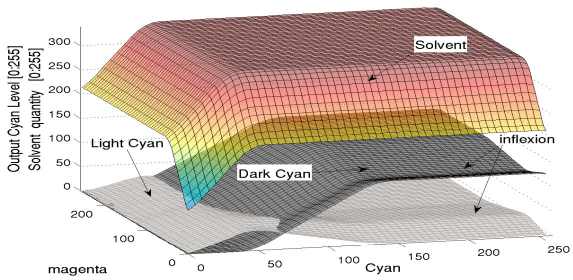

Figure 9. *Cyan ink separation surfaces obtained, for a color system with three output levels per component, using the solvent model M_3.*

- the per component ink limitation, required by model M_3, precludes the color gamut volume of being maximized since single component colors (e.g. pure cyan or pure magenta) are limited to $S_{i,max} = 134$, far below the substrate absorption limit which would be

$$S_{max} = \sum_i S_{i,max} = 402 \tag{10}$$

- The model M_3 induces the usage of dark ink components within relatively low density areas. Actually, the dark cyan ink-separation surface does not vanish for small values of C, as soon as the magenta $M \gtrsim 100$.
- Two, somewhat unexpected, inflexions (see arrows on figure 9) appears for high cyan values. These are inherent to the model which allows for more solvent as soon as the magenta component start growing. The consequence is that part of the solvent excess is used by light cyan instead of being fully consumed by light magenta. The net result is the appearance of dark, isolated, magenta dots on a cyan bed.

Applying both the ink-limitation and the solvent limit on a vectorial base, can readily solve these problems. On that purpose, we constructed a *vector solvent* model called M_4 and defined by :

$$\begin{aligned} M_4: \quad S &= \min\left(\sum_i \min(\alpha_i X_i,\ 255),\ S_{max}\right) \\ \alpha_i &= V(O_p)/V(O'_p) \\ \text{with} \quad i &\in \{C,M,Y\} \\ X_i &\in \mathscr{D}_4 \end{aligned} \tag{11}$$

With model M_4, the bounding volume of $\mathscr{F}_S$ is given by equations 4 and 5 while the definition domain $\mathscr{D}_4$ is such that $0 \leq C+M+Y \leq S_{max}$, where S_{max} is the substrate absorption limit of equation 10 .

In the following application of this model, the input values are confined into the definition domain by crudely multiplying each component C, M, Y by a factor

$$\beta = \begin{cases} 1 & \text{if } C+M+Y < S_{max} \\ S_{max}/(C+M+Y) & \text{otherwise} \end{cases} \tag{12}$$

The simulation performed with model M_3 has been repeated using model M_4. The results are summarized on the figures 10 and 11.

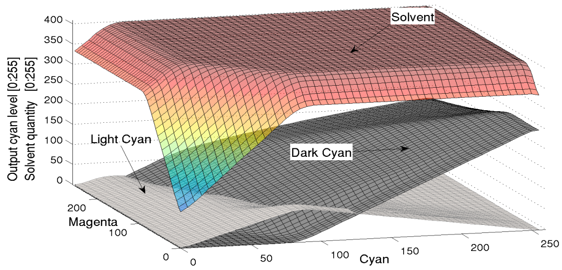

Figure 10. *Cyan ink separation surfaces obtained, for color system with three output levels per component, using the solvent model M_4.*

The figure 10 shows the ink-separation surfaces for light and dark cyan, along with the total amount of solvent. Similar surfaces can be obtained for the magenta component. As expected, single component colors (pure cyan and pure magenta) reach the full coverage with dark ink only (on output $C = 255$ and $c = 0$) . The substrate absorption limit ($S_{max} = 402$) is reached quickly and never exceeded, insuring a safe maximisation of the light ink usage. To illustrate this feature, the figure 11 allows comparing the ratio of light ink used with model M_3 and M_4.

Clearly, model M_4 uses only light inks on a much larger portion of the image. Moreover, the light ratio is much larger everywhere, except in two neighboring regions of pure cyan and pure magenta. This occurs because, in these regions, the vector model allows for much more saturated color, which can only be produced using mainly dark inks.

Similarly to model M_1, the model M_4 does not produce a smoothly growing solvent distribution. However, it should not be difficult to obtain such a desirable feature by replacing the min functions in equation 11, possibly taking inspiration from model M_2. The quite crude ink-limitation procedure of equation 12 should also be replaced by some smoother function in order to obtain a production-ready printing system.

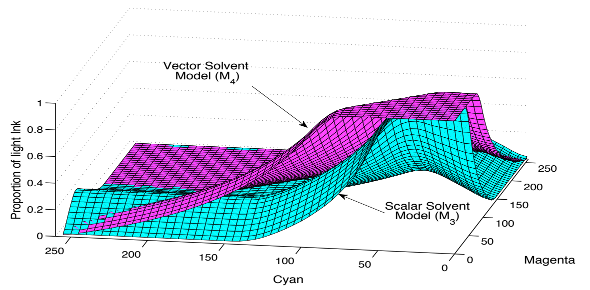

Figure 11. *The two surfaces represent, for model M_3 and M_4, the ratio of the light ink amount (both cyan and magenta) to the total ink amount, disregarding the constant Y_0 component. A value of 1 means that only light ink is used while a value of 0 corresponds to dark ink only.*

Conclusions

We have presented a novel color image error diffusion scheme targeted at those printing systems which make use of multiple inks having the same chromaticity but different densities. The main feature of these method consists in the introduction of a solvent variable which enters the error diffusion process and is expressed as a function $\mathscr{F}_S$ of the color variables $\{C,M,Y\}$.

While offering a precise control over the total amount of solvent (or liquid) deposited on the substrate, the method allows to consider all the inks having the same hue as one single color component, all along the image processing chain. The ink separation takes place at the very last step, that is during the quantization process.

Several important performance advantages of the method come from this last feature: (a) The printer color characterization still consists in determining a fully constrained mapping from a $3D$ to another $3D$ color space; (b) The color interpolations take place in a $3D$ space as well; (c) The number of error diffusion channels is limited to 4 (three for color and one for solvent), whatever the number of inks having the same hue; (d) The system is fully vectorized since the quantization is performed based on the input vector I_m (see equation 1). Moreover, the maximum amount of solvent and the maximum amount of color can be both set as a whole (not component by component).

The global conditions that must be fulfilled by the arbitrary solvent function $\mathscr{F}_S$ have been expressed in terms of the function *definition domain* and *bounding volume*.

The effectiveness of the method have been demonstrated through five simulated applications, varying the number of colors, the number of light inks with the same hue, and the method for evaluating the solvent component. It has been shown that the method hits the mark by faithfully producing the requested quantity of solvent. The analysis of the halftoned images shows that the resulting ink separation curves (or surfaces) are closely related to the standard ones.

Since we have not yet applied the method to a fully controlled practical case, we do not know how effective the method is in controlling a possible hue shift between light and dark ink [8]. Nevertheless, if inks vary only in their dilution properties, we expect the root cause of the observed shift to be precisely the amount of solvent deposited on the substrate. Therefore the proposed method should be of a great help in controlling such output color discontinuities.

We expect the method to be easily extendible to printing systems providing an additional black ink. The usual *under color removal* procedure could be advantageously replaced by a solvent function which reduces the amount of liquid allowed in the dark gray regions, thus forcing the system to use the K component instead of the compound gray.

Future work will be mainly directed towards the experimental application and characterization of the presented method. Furthermore, its applicability and usefulness with systems providing a larger set of inks have to be thoroughly considered. Our "multilevel error diffusion with solvent control" could possibly open the door to low cost implementations of HiFi printing systems.

Acknowledgments

This work was supported by Olivetti Engineering SA.

References

[1] U. A. Agar. Model based color separation for cmykcm printing. In *Color Imaging Conference*, pages 298–302, 2001.

[2] G. Braverman, S. Harush, G. Karvitsky, and S. Costis. Hue preservation aspects of cmyk - cmykcm color transformations. In *ICIS '06*, pages 637–640, Rochester, New York, May 7 2006.

[3] R. W. Floyd and L. Steinberg. An adaptive algorithm for spatial grayscale. In *Proc. SID*, volume 17(2), pages 75–77, 1976.

[4] H.E.J. Neugebauer. Die theoretischen grundlagen des mehrfarbenbuchdrucks. *Zeitschrift für wissenschaftliche Photographie Photophysik und Photochemie*, 36:4:73–89, 1937.

[5] Ying X. Noyes, Jon Y. Hardeberg, and Moskalev M. Anatoly. Linearization curve generation for ccmmyk printing. In *IS&T/SID Eighth Color Imaging Conference*, pages 247–251, November 2001.

[6] V. Ostromoukhov and R.D. Hersch. Structure artifact free multilevel error diffusion algorithm. In *Proceedings Europto Conf. Series*, volume 3409 of *Proceedings Europto Conf. Series*, Zürich, May 1998. SPIE. International Symposium on Electronic Image Capture and Publishing,.

[7] G. Sharma and H.J. Trussell. Digital color imaging. *IEEE Transactions on Image Processing*, 6(7):901–932, July 1997.

[8] M.Q. Shaw, R. Bala, and G. Sharma. Smooth blending of two inks of similar hue to simulate one ink. In Reiner Eschbach and Gabriel G. Marcu, editors, *Color Imaging: Processing, Hardcopy, and Applications*, volume 5667, pages 409–416. SPIE–The International Society for Optical Engineering, January 2005.

[9] S. Sugiura and T. Makita. An improved multilevel error diffusion method. *Journal of Imaging Science and Technology*, 38(6):545–551, 1994.

[10] Huanzhao Zeng. Ink limit control for ink-jet printer color calibration. In R. Eschbach and G. G. Marcu, editors, *Proc. SPIE Vol. 4300: Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts VI*, Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference, pages 93–96, January 2001.

Author Biography

Serge Cattarinussi *studied physics at the University of Neuchâtel (Switzerland) where he received his Dipl. de Physicien in 1986. In 1992 he received the PhD degrees in biophysics from the International School of Advanced Studies in Trieste (Italy). After having studied membrane protein dynamics at the Swiss Federal Institute of Technology he joined the Olivetti company in 2000, where he is in charge of developing the image processing algorithms required in the printing technologies.*

Ana Dimitrijevic *received her BS in Electrical Engineering from the University of Belgrade, Serbia (1999) and her PhD in Signal Processing from EPFL, Switzerland (2005). Since then she is with Olivetti Engineering where she is in charge of the image printing methods and color characterization of ink-jet printing devices.*